SEE Conf
蚂蚁金服体验科技大会

蚂蚁金服前端框架探索之路

云谦

ANT FINANCIAL EXPERIENCE

SEE Conf
蚂蚁金服体验科技大会

陈成（云谦、sorrycc）

🏠 杭州

👩 👦 👦 wife, 2 kids

🏢 淘宝 5 年

🏢 支付宝 ~6 年

spm
awesome-javascript
atool-build
dora
babel-plugin-import
dva
roadhog
umi

https://sorrycc.com

大纲

1. 蚂蚁前端框架简史（WHY，10min）
2. 介绍 umi（WHAT，15min）
3. umi 如何支持蚂蚁业务（HOW，10min）
4. 关于前端框架建设的建议（BONUS，5min）

SEE Conf
蚂蚁金服体验科技大会

蚂蚁前端框架简史

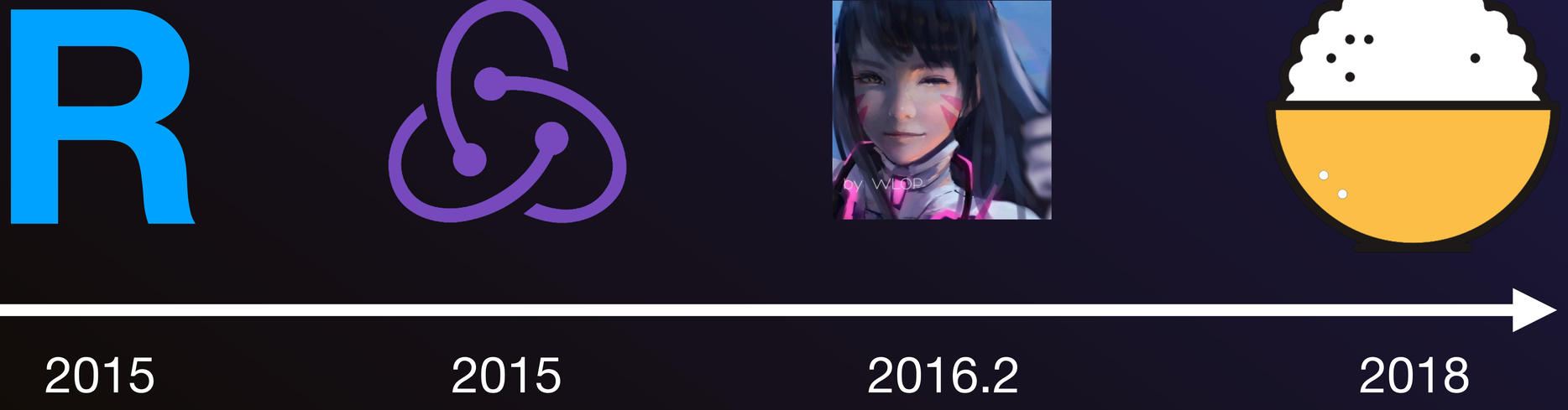
每一代的框架都是为解决“公司当下的业务问题和痛点”而生的

业务的两大痛点：

① 研发效能：研发时缺这缺那，各种不顺手

② 用户体验：上线后产品用户体验不佳

前端框架是为了解决公司的业务痛点


R
2015
2015
2016.2
2018

demo.js — sorrycc-Dt0xp3

JS demo.js

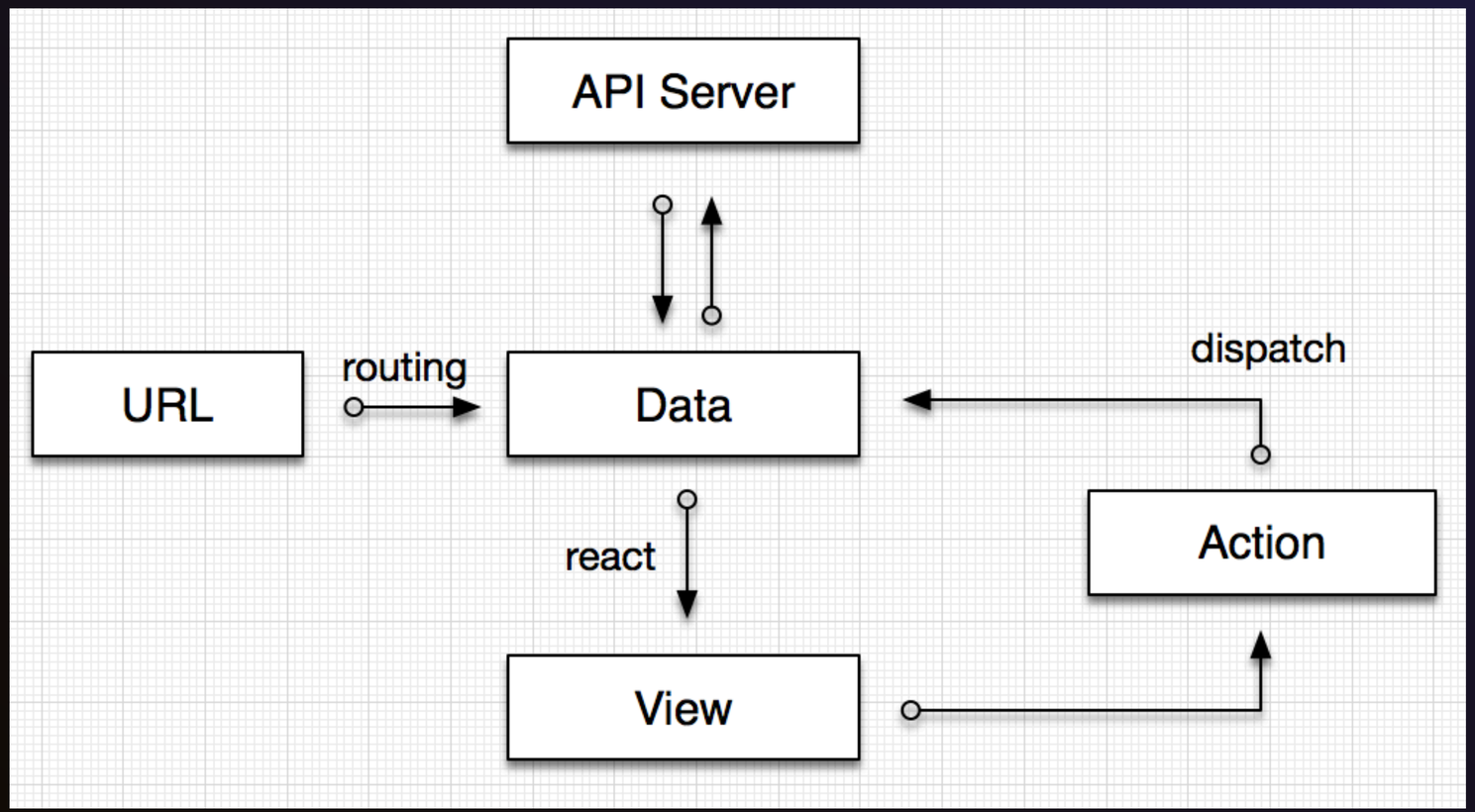
```
1  // 定义 state
2  createRootContainer({
3    user: { name: 'chris', age: 30 }
4  })(App);
5
6  // 绑定 state
7  createContainer({
8    myUser: 'user',
9  })(UserInfo);
```

Roof

拥抱社区很重要，社区才是“大腿”。

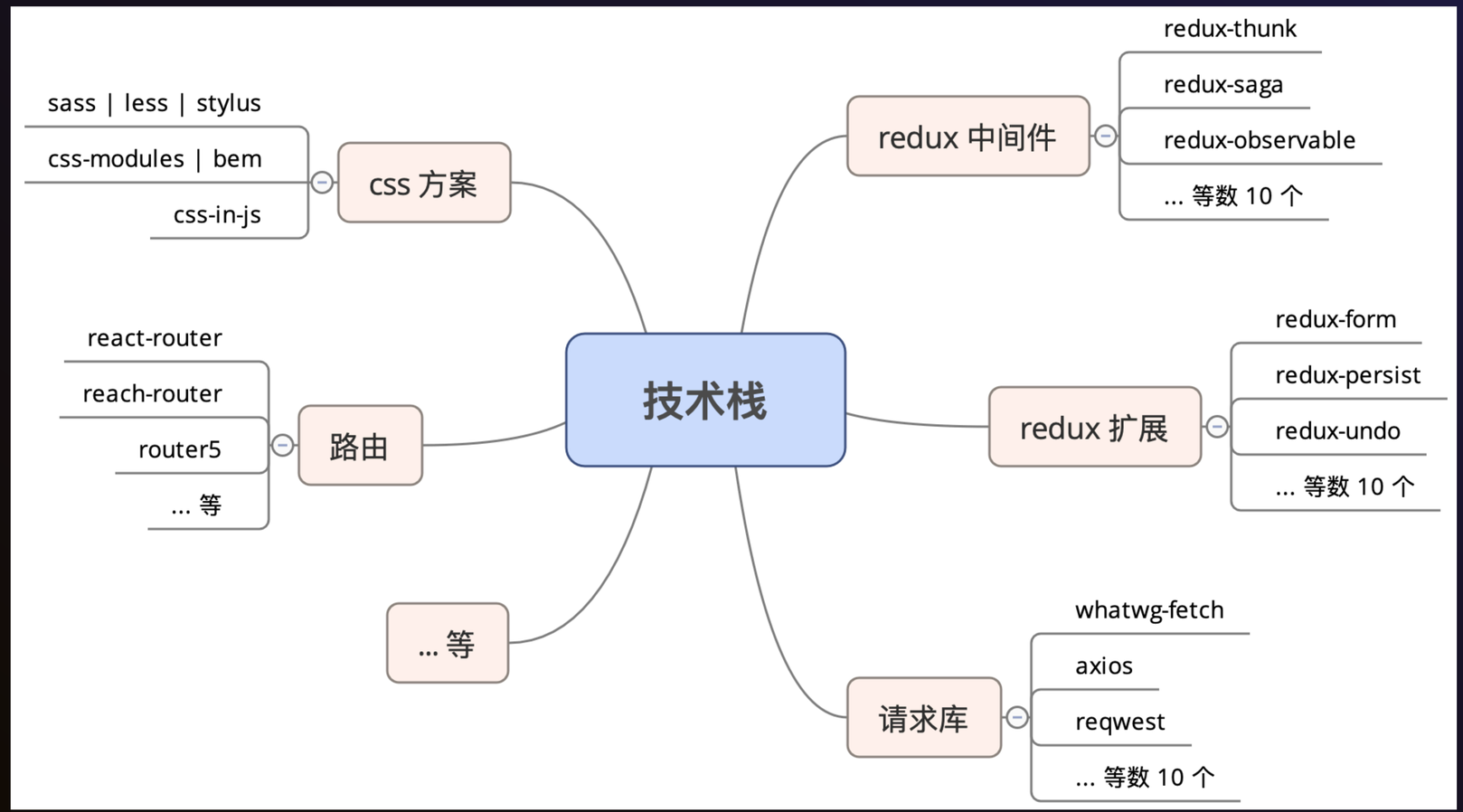
1. 类 baobab，选错阵营
2. 没有跟上社区发展

Roof



redux

大框架不变，变的是每个环节里的小方案。



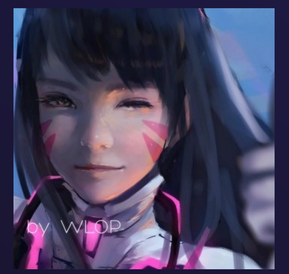
百花齐放对团队来说是个“灾难”，要收放有度。

1. **过于**开放，一个项目一个技术栈
2. **繁琐**，手写 redux 不是每个人都能忍的

dva.js — sorrycc-Dt0xp3

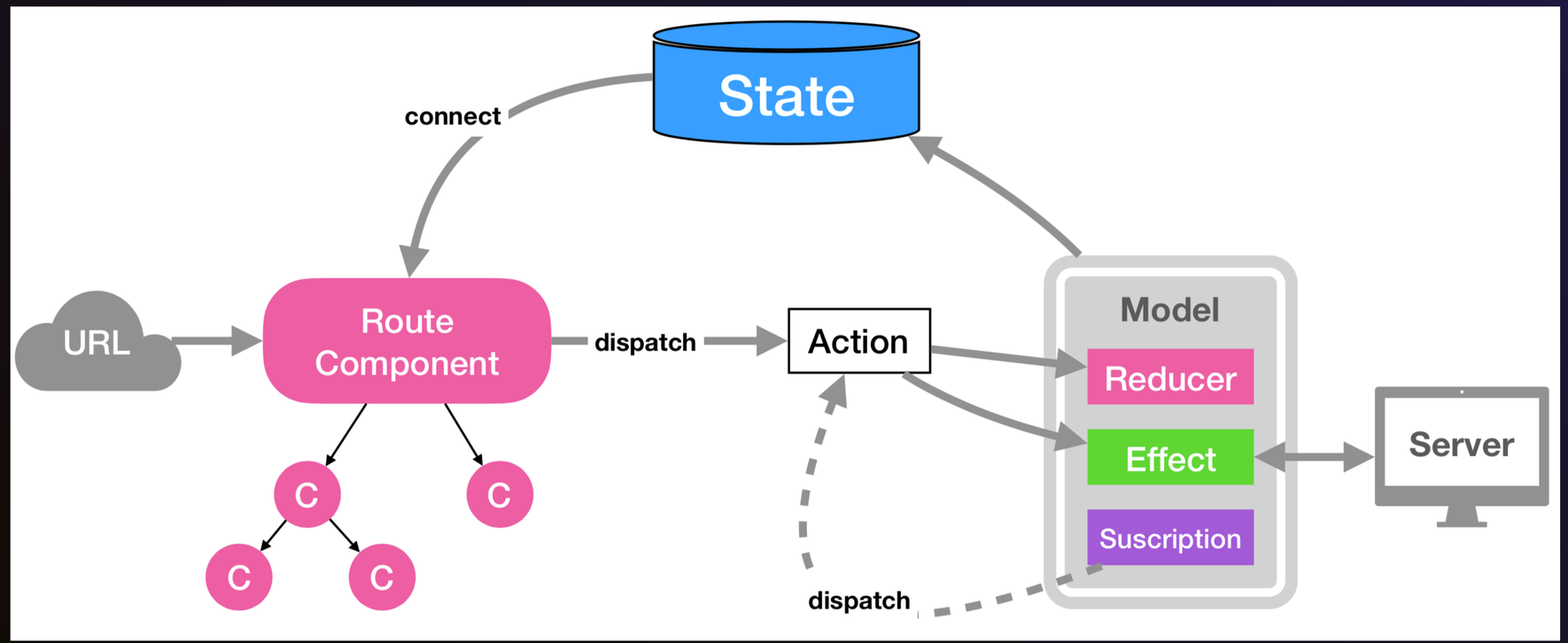
JS dva.js

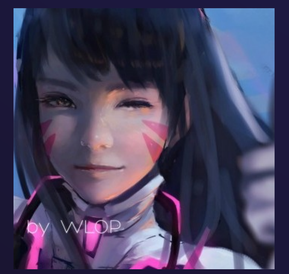
```
1  const app = dva();
2  app.model({
3    namespace: 'count',
4    reducers: {},
5    effects: {},
6    subscriptions() {},
7  });
8  app.router(() => {});
9  app.start();
```



dva

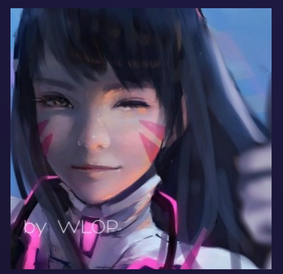
5 个 API，7 个概念。


State
connect
URL
Route
Component
dispatch
Action
Model
Reducer
Effect
Suscription
Server
C
C
C
C
dispatch



dva

1. **定位模糊**，数据流还是框架?
2. **技术收敛只做了一半**，除了数据流，其他都很粗糙
3. **只有社区版，没有针对内部需求做定制化**



dva

umi

SEE Conf
蚂蚁金服体验科技大会

介绍 umi

每个框架都应该思考自己的“核心竞争力是什么”

umi 的核心竞争力是什么？

相比社区的 next.js、after.js、gastby 等有啥优势？

插件机制

插件即社区，你总不可能一个人把所有功能都写了吧。

基于路由

路由即入口，有入口就有很多可能。

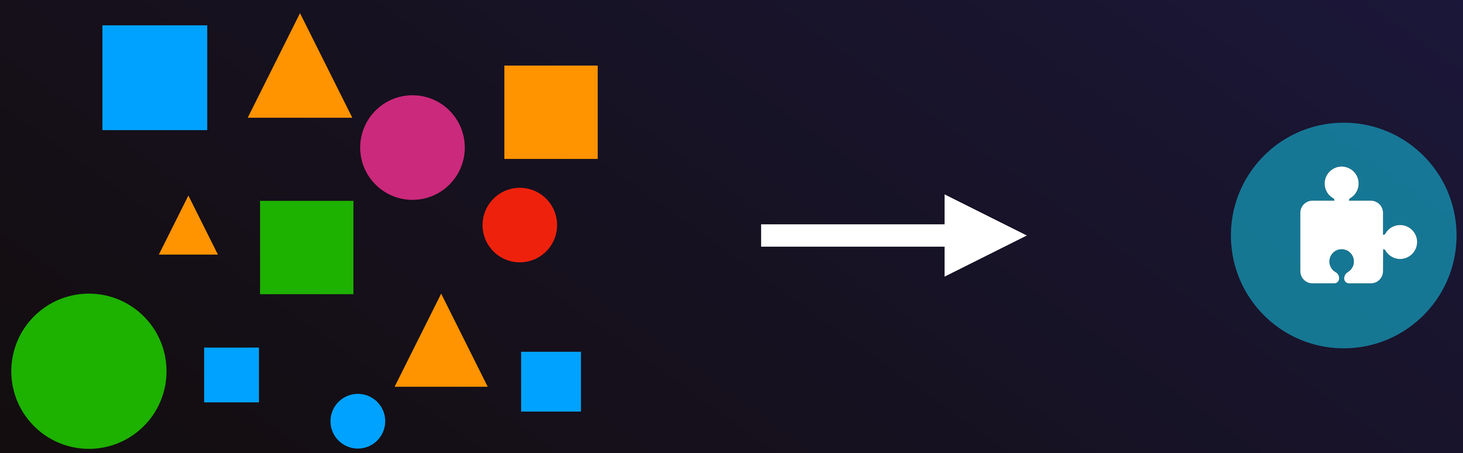
HTML 是一等公民

HTML 是串联流程的粘合剂，可在插件里定制极其重要。

插件机制

插件即社区，你总不可能一个人把所有功能都写了吧。

插件用于满足不同人、场景的不同需求。



插件用于封装复杂逻辑，简化使用。

实现 PWA 需要几步?

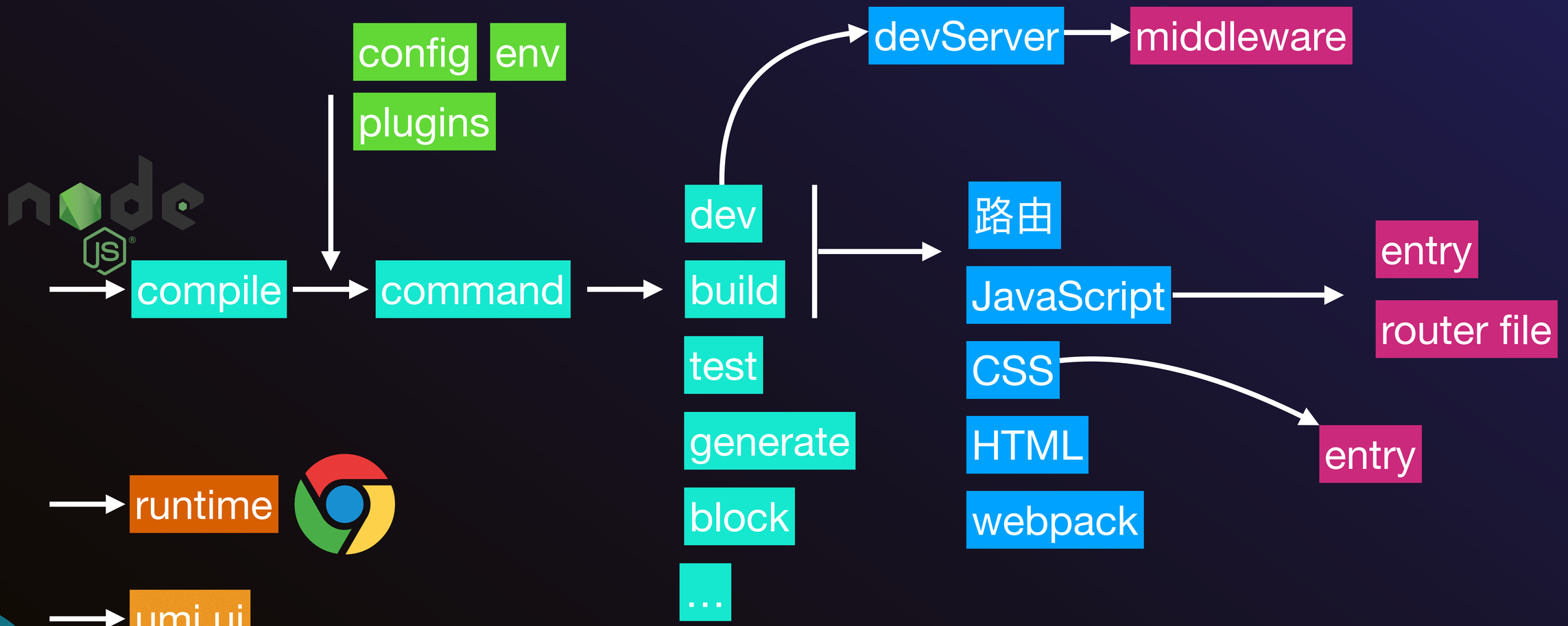
插件机制

config.js

JS config.js ~/Desktop

```
1  export default {
2    plugins: [
3      'pwa'
4    ]
5  }
```

插件机制


config
env
plugins
devServer
middleware
compile
command
dev
build
test
generate
block
...
路由
JavaScript
CSS
HTML
webpack
entry
router file
entry
runtime
umi ui

插件机制

插件不仅作用于“编译时”，还作用于“运行时”。

插件机制

SEE Conf
蚂蚁金服体验科技大会

路由 = 页面

把握入口，真的可以做很多事。

基于路由

按需挂载数据流

■ ■ ■

自动埋点

面包屑

按需加载

按需编译

菜单生成

标题切换

切换动效

代码分割

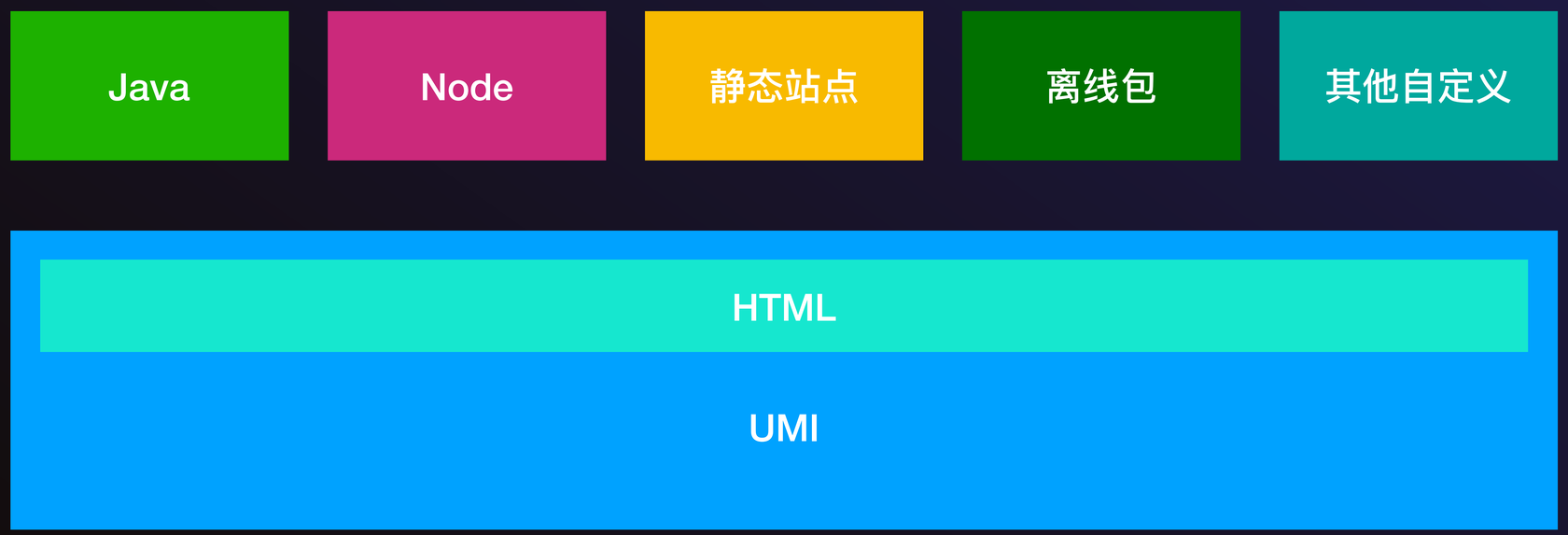
基于路由

HTML 的角色

连接前端和后端，让 umi 可以无缝接入各种后端。

1. 不使用 html-webpack-plugin
2. 可在插件里定制
3. 修改 publicPath、base、cssPublicPath、hash、manifest 等

HTML 是一等公民



打通 HTML 环节后，就可以一“件”接入各种应用场景。

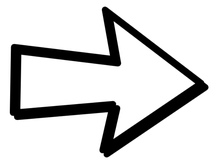
HTML 是一等公民

除此之外，umi 还有什么？

约定式路由、umi-pugin-react、区块市场、umi ui、等等

umi.sh — test

```
+ layouts/index.js
+ pages
  + users
    - _layout.js
    - $userId.js
  - index.js
```



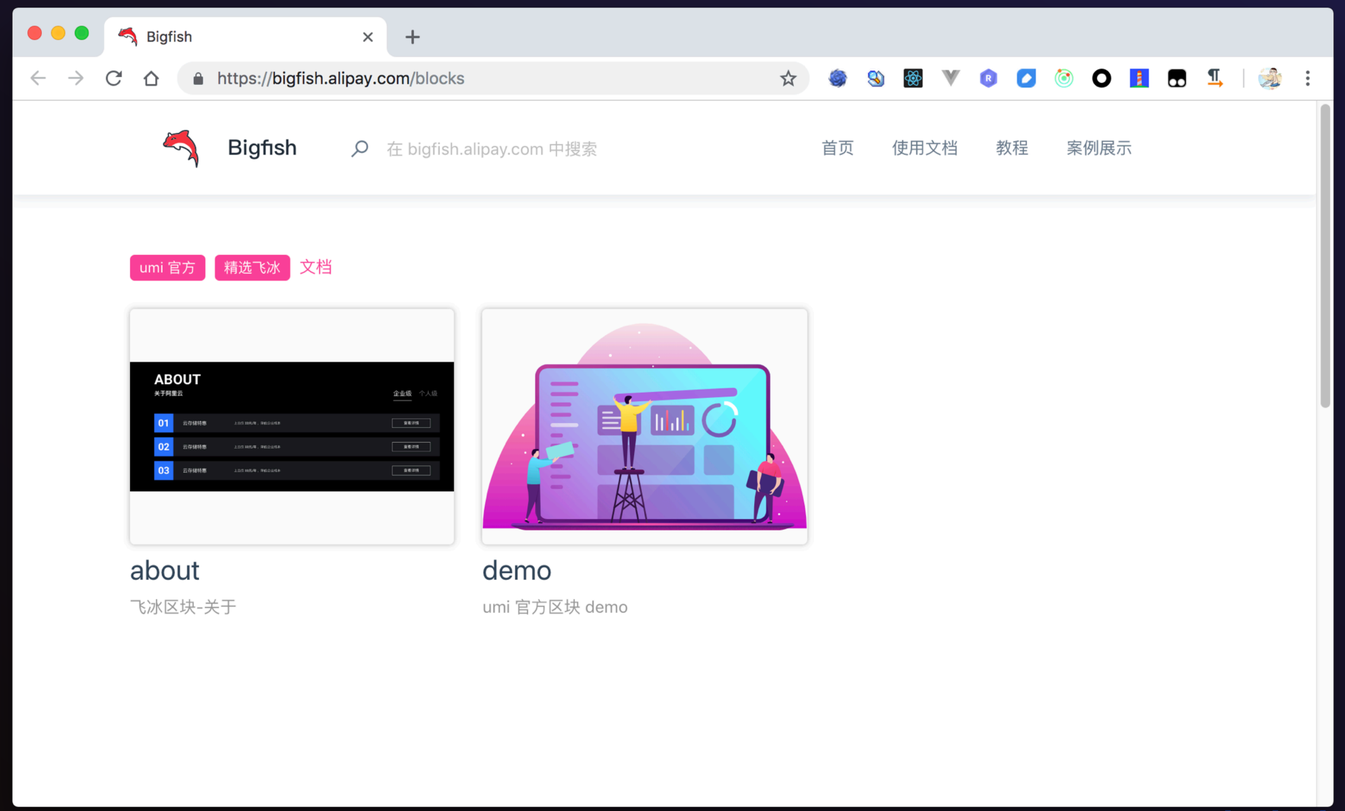
route.sh — test

route.sh

```
1  / (layouts/index.js)
2    /users (pages/users/_layout.js)
3      /users/:userId (pages/users/$u
4    / (pages/index.js)
5
```

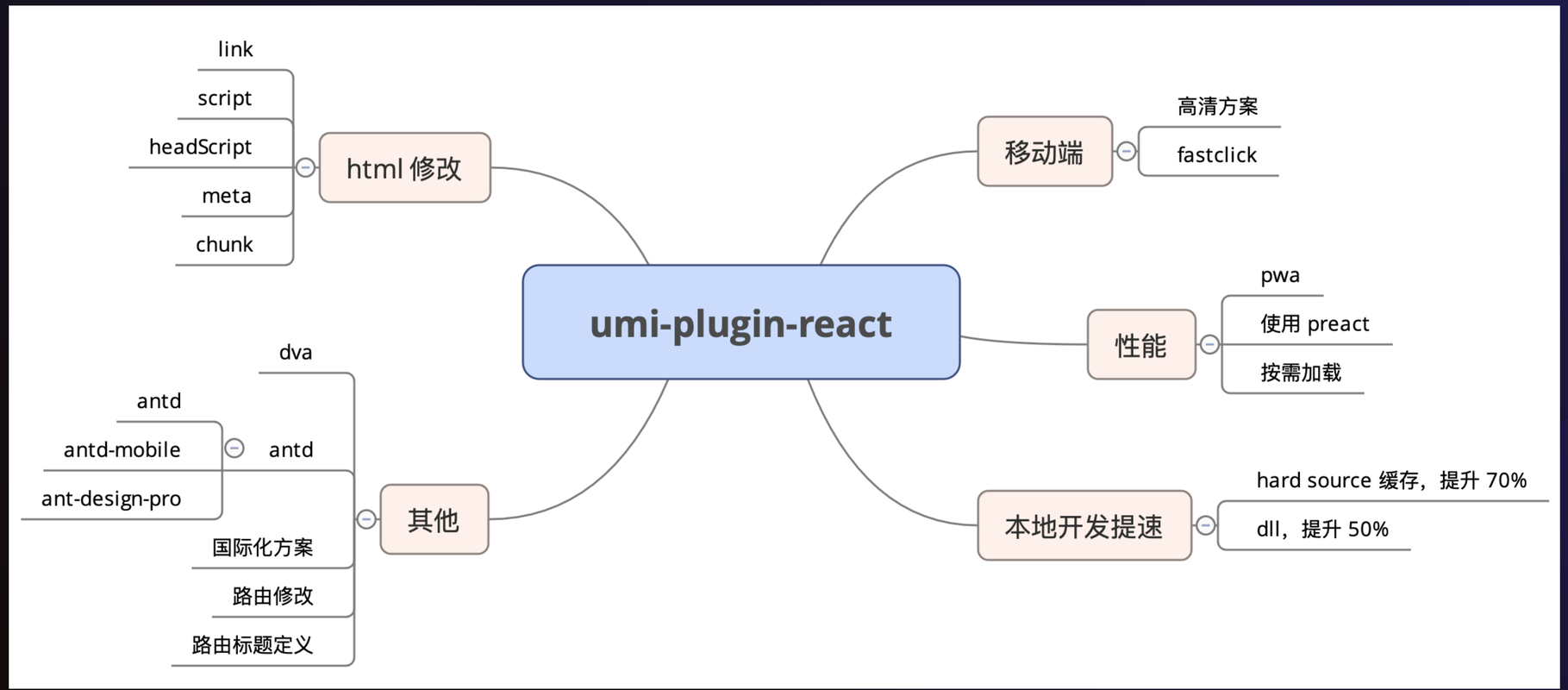
约定式路由

文件系统即路由，减少冗余配置。



区块市场

让你可以从 0 搭建一个 ant-design-pro 。



umi-plugin-react 插件集

一个个插件配太繁琐，给你一个插件集统计配。

更多 umi 特性

umi test

完善的路由支持

umi g

约定大于配置

umi library

抱社区大腿

内置性能优化

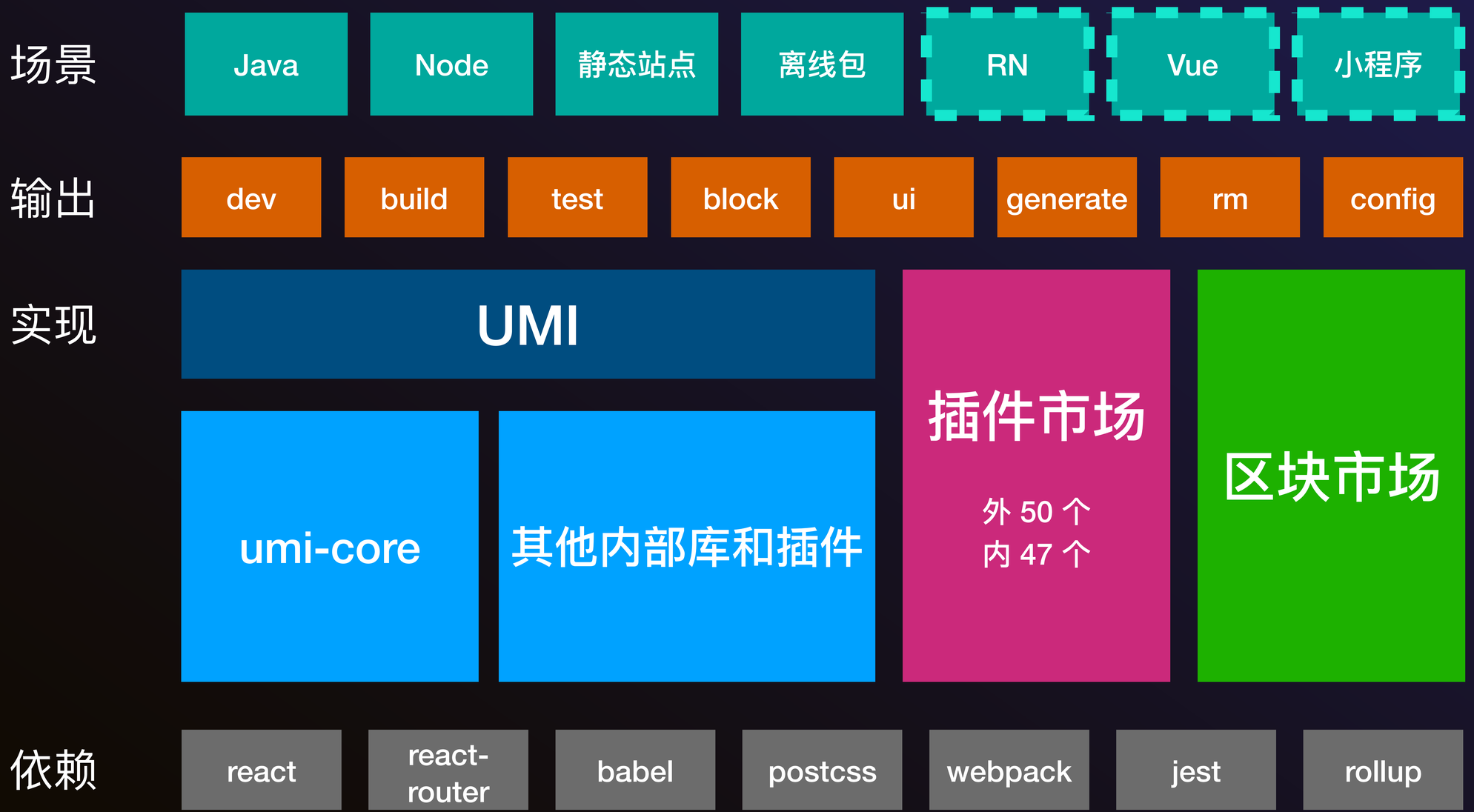
duck directory

umi config

umi ui

静态站点导出

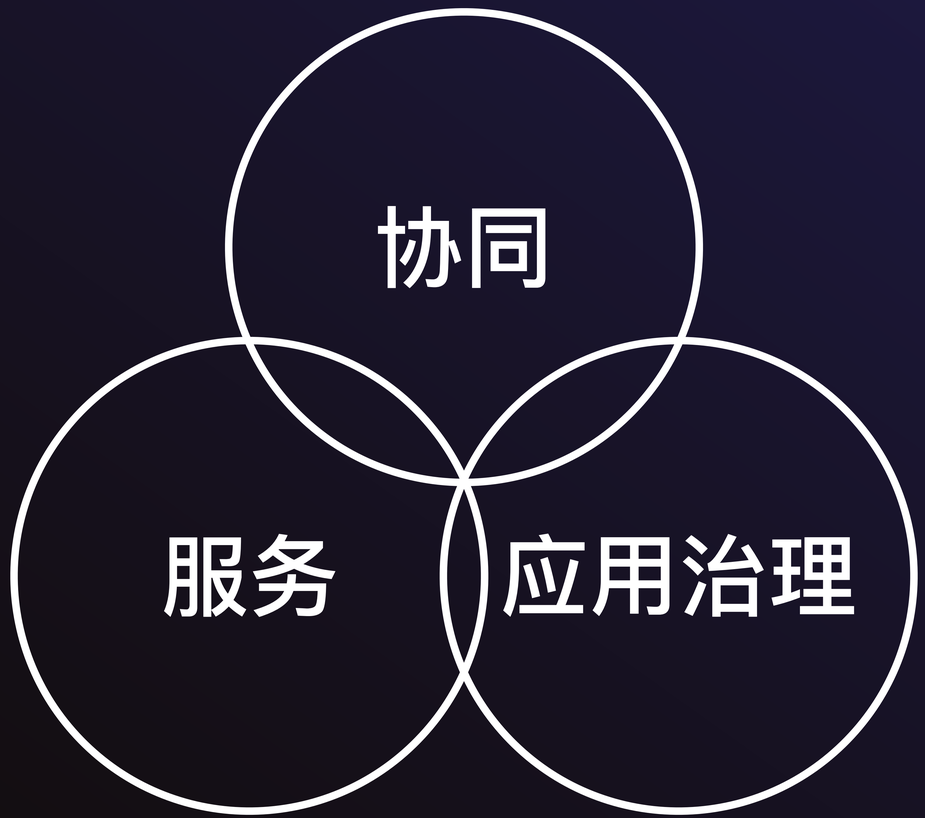
一键兼容 ie9


场景
Java
Node
静态站点
离线包
RN
Vue
小程序
输出
dev
build
test
block
ui
generate
rm
config
实现
UMI
umi-core
其他内部库和插件
插件市场
外 50 个
内 47 个
区块市场
依赖
react
react-router
babel
postcss
webpack
jest
rollup

SEE Conf
蚂蚁金服体验科技大会

umi 如何支持蚂蚁业务

对内，做好 build + dev + test 还远远不够


协同
服务
应用治理

Bigfish

Bigfish = umi + umi-plugin-bigfish


sofa
(java)
chair
(node)
site
(静态站点)
离线包
(无线离线包)
bigfishweb
assets
(静态资源)
h5
(无线 web)
Bigfish
antd
dva
UMI
umi-core
其他内部库
和插件
插件
市场
外 50 个
内 47 个
组件市场
区块
市场
精品
区块
监控
培训
AntV
数据洞察
智能建站
react
react-
router
babel
postcss
webpack
jest
rollup

内部框架不仅是前端的 build、dev、test，

更重要的是和流程、内部服务的打通

appType \ deployMode	bigfishweb	assets	sofa	chair	offline	online	custom
console							
h5							
site							

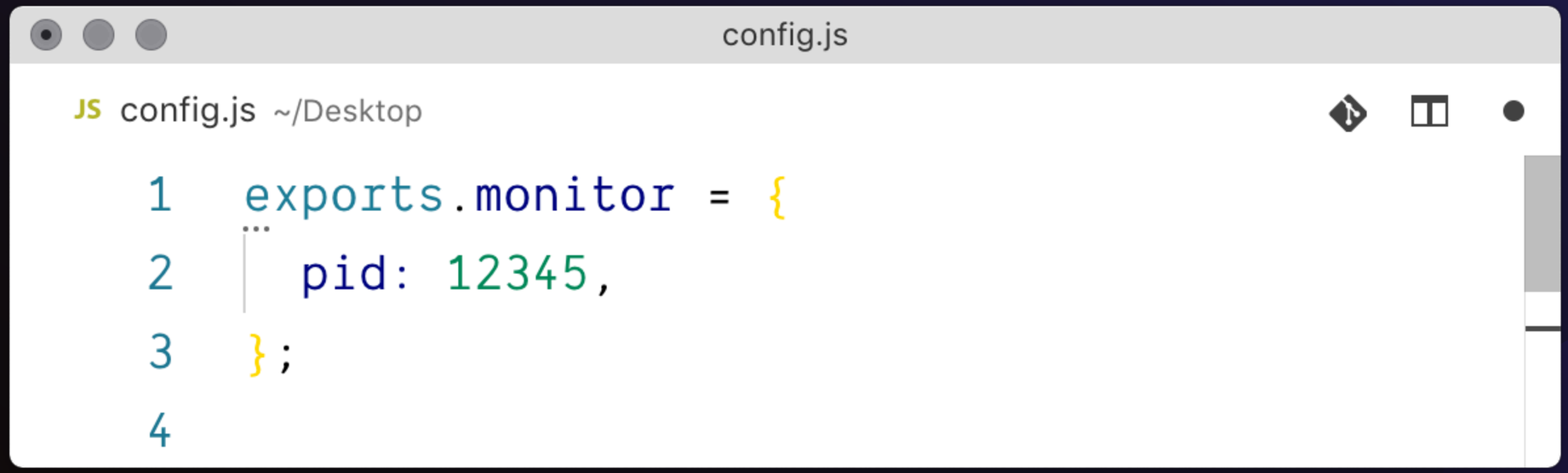
通过 appType 和 deployMode 两个维度对接丰富的应用和部署类型。

route.sh — test

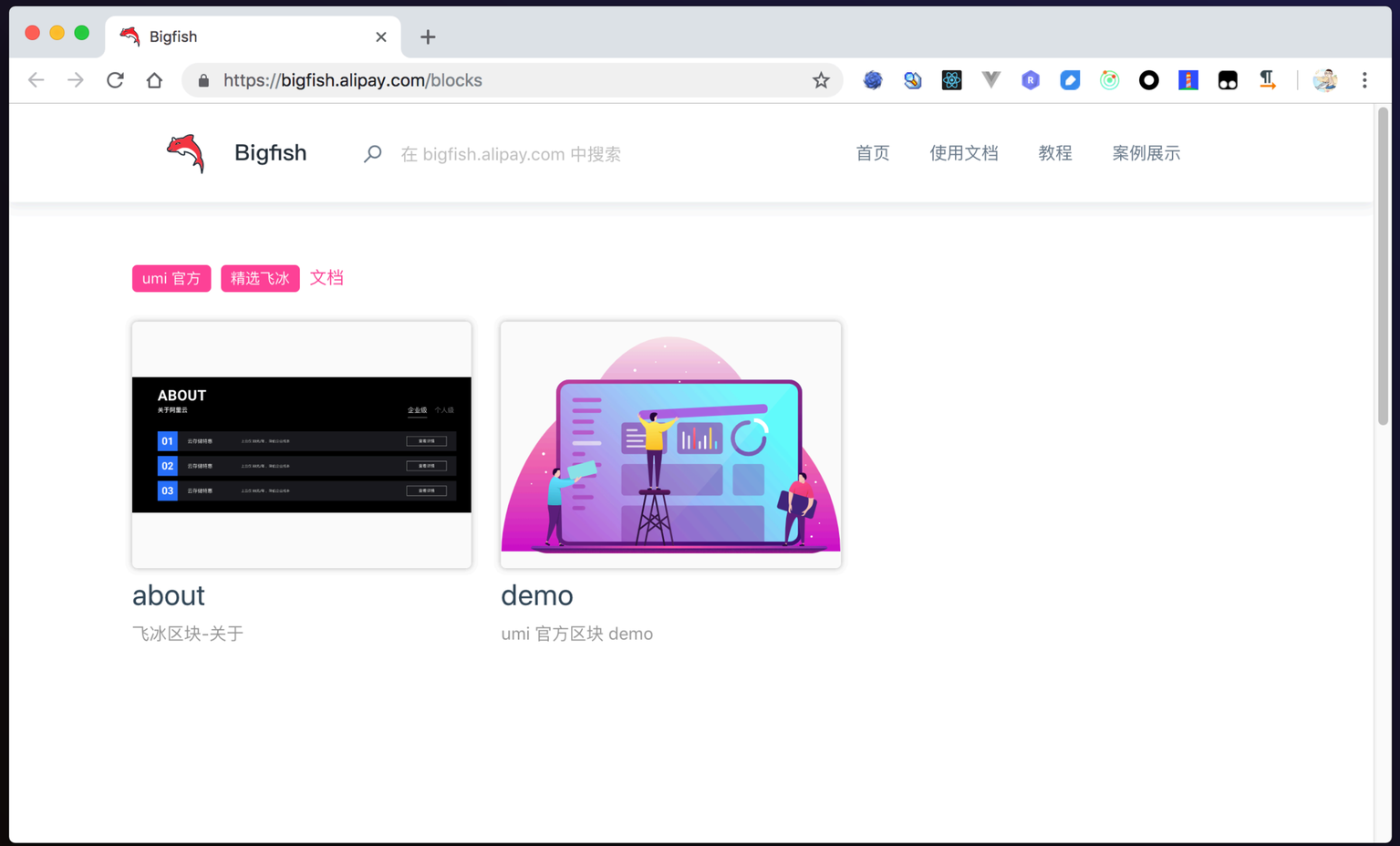
route.sh

```
1  export default {
2    appType: 'h5',
3    deployMode: 'chair',
4  }
```

通过 appType 和 deployMode 两个维度对接丰富的应用和部署类型。


config.js
JS config.js ~/Desktop
1 exports.monitor = {
2 pid: 12345,
3 };
4

一键埋点


Bigfish
https://bigfish.alipay.com/blocks
Bigfish
在 bigfish.alipay.com 中搜索
首页
使用文档
教程
案例展示
umi 官方
精选飞冰
文档
ABOUT
about
飞冰区块-关于
demo
umi 官方区块 demo

精品区块市场

SEE Conf
蚂蚁金服体验科技大会

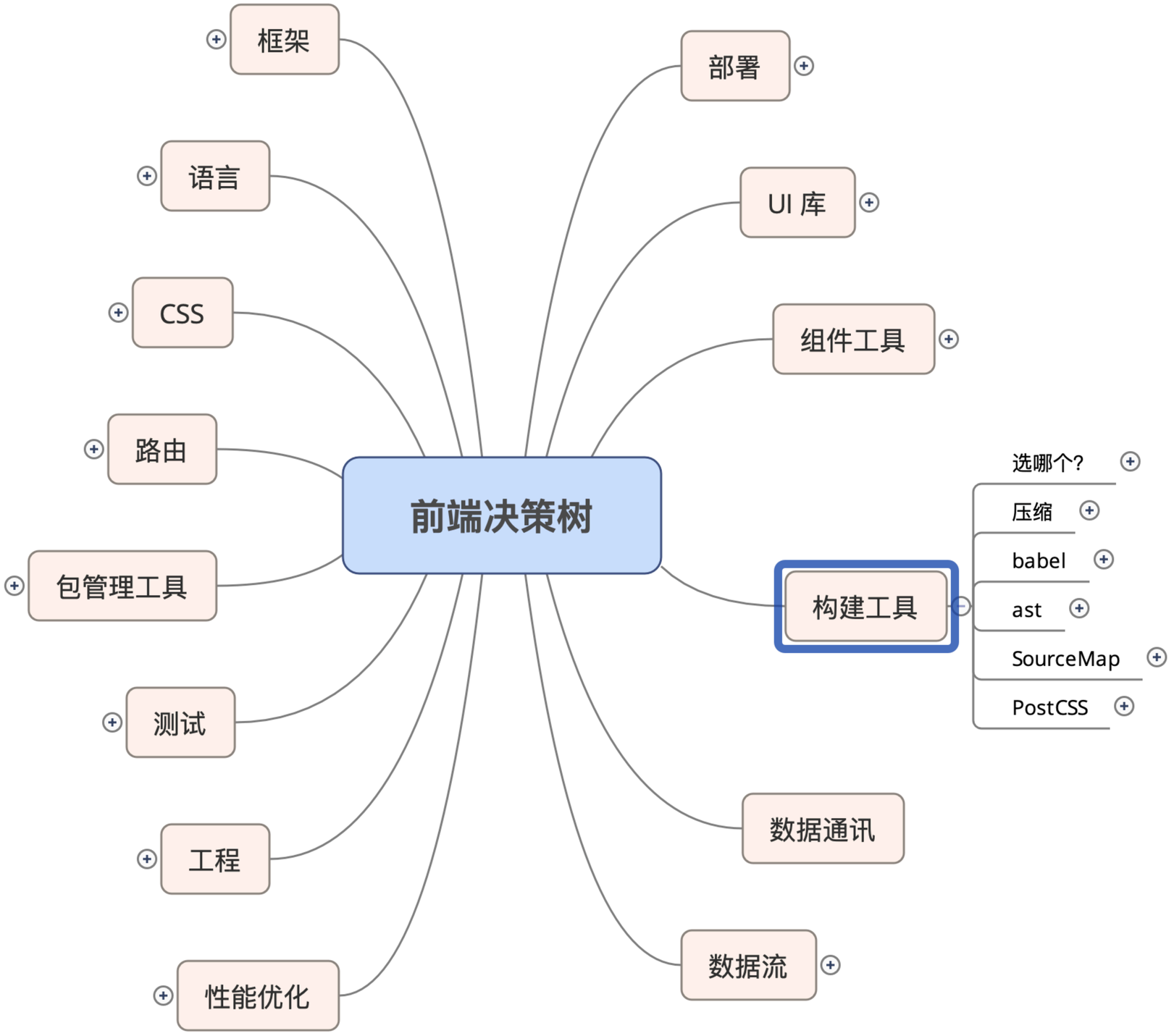
关于前端框架建设的建议

你需要做由数以百计的决策

SEE Conf
蚂蚁金服体验科技大会

前端决策树

https://github.com/sorrycc/f2e-decision-tree



route.sh — test

route.sh

```
1  #!/usr/bin/env node
2  const { fork } = require('child_process');
3
4  process.env.UMI_PLUGINS = __dirname + './plugins';
5  fork(
6    require.resolve('umi/bin/umi'),
7    [...process.argv].slice(2),
8  );
```

基于 umi 实现自己的框架

让我们来回顾一下

1. 从时间的角度看为什么会有 umi 框架
2. umi 是什么，他有哪几个核心竞争力
3. umi 如何服务社区
4. umi 如何服务内部用户
5. 如何基于 umi 搭建框架，解决自己公司的业务问题

SEE Conf
蚂蚁金服体验科技大会

THANKS