Vue 探索与实践

—《陪伴计划》开发总结

业务介绍　　　　技术简介　　　　项目实战　　　　技术展望

热闹的前端

纯模版引擎

解决问题的多少

业务介绍

业务介绍

福利中心

用户购满一定金额，享受每月限额福利券及

一项低价换购，还有试用等福利

业务介绍

业务介绍

分类馆

几大品类，品类下按类型，根据宝宝年龄精

准推送推荐商品

业务介绍

业务介绍

陪伴空间

陪伴计划主页，根据宝宝年龄或用户选择，

精准推送特定年龄段段的精选商品和资讯

遇到问题

接口
众多

交互
复杂

快速
上线

近90个数据接口

数据字段不规范、不统一、难理解

接口开发经常延期且频繁变更

各种交互及状态，且一态多用

给用户展示的是多状态共同作用的结果

用户操作异步更新页面

同时规划多版本

多版本并行开发

技术选型

传统移动开发

Zepto等 /

原生js, html, css

技术选型

传统移动开发 ⋯⋯⋯ Zepto等 /
原生js, html, css ⋯⋯⋯▶

代码耦合度高

DOM操作性能

协作、维护困难

技术选型

接口
众多

交互
复杂

快速
上线

统一数据中心

视图逻辑分离

项目工程化

技术选型

纯模版引擎

解决问题的多少

Vue技术简介

Vue技术简介
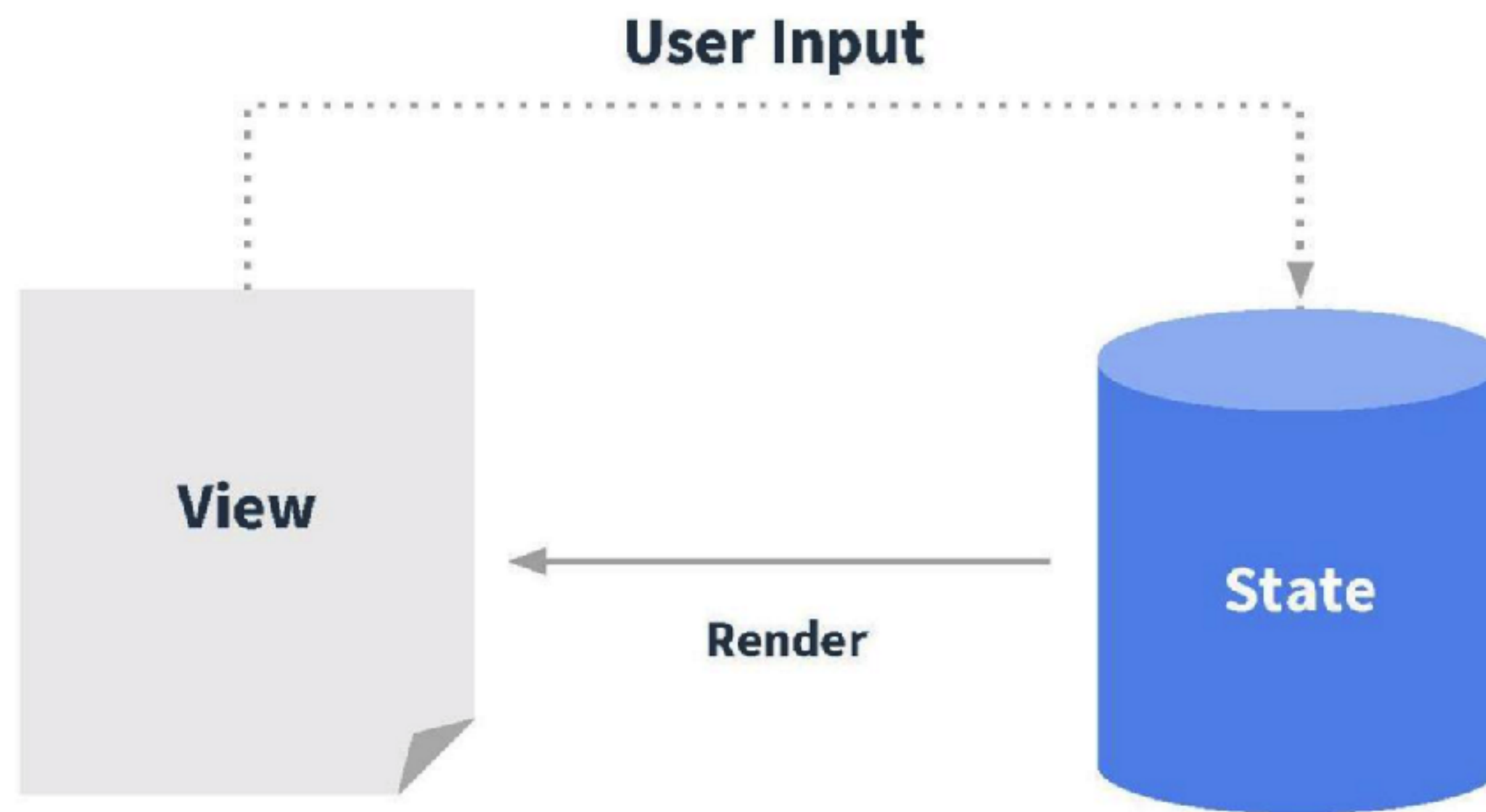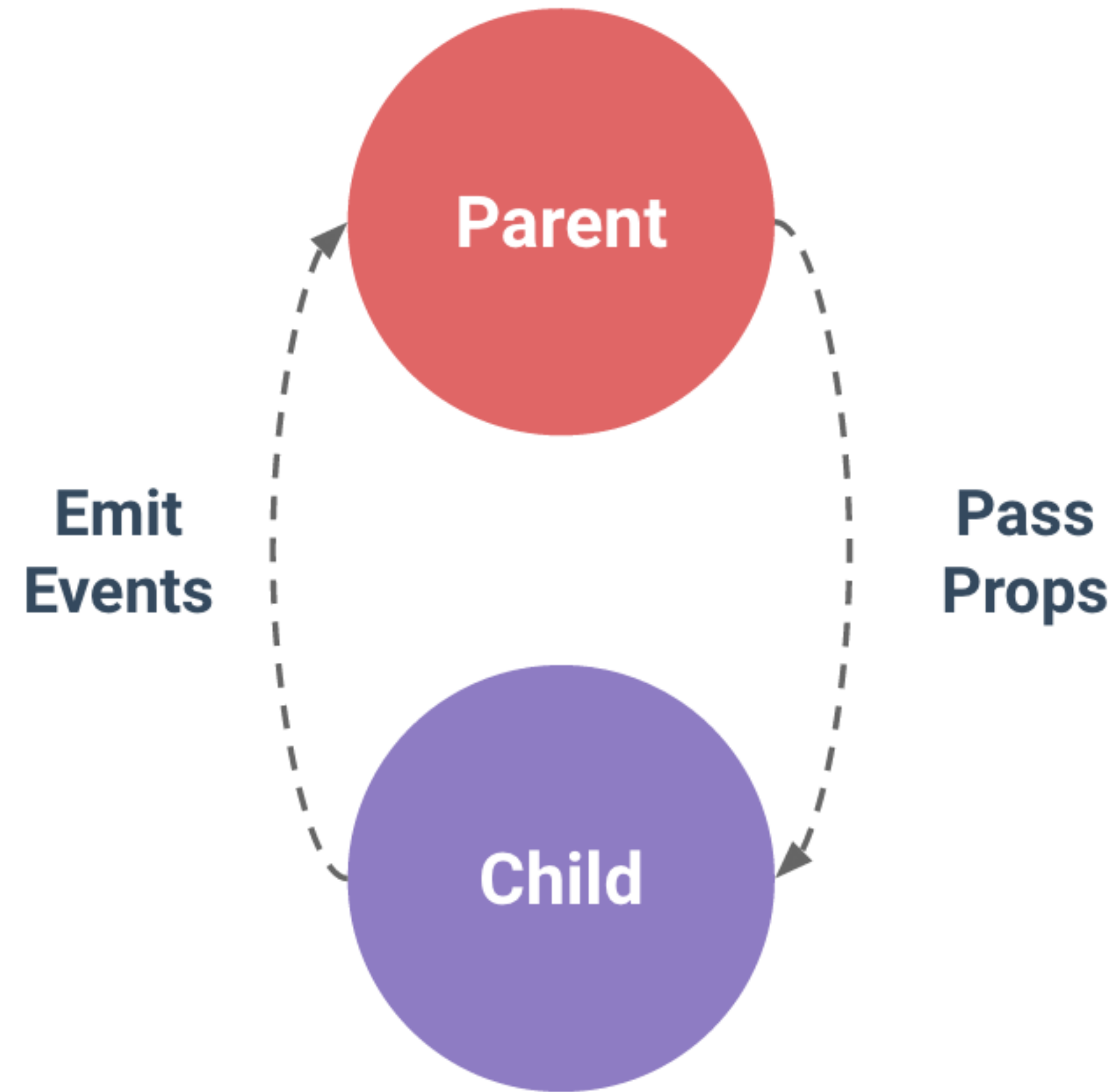
声明式渲染

组件系统

客户端路由

状态管理

Vue技术简介

声明式渲染

客户端路由

客户端路由的各种问题

- 嵌套路由
- 具名路由
- 多个平级路由出口
- 复杂匹配规则
- 当前活跃链接

- 重定向／别名
- 跳转动画
- 异步数据处理
- 跳转规则限制
- 滚动条行为
- 懒加载

Vue技术简介

状态管理

项目实战

项目实战

```
.
├── build ──────────────────→        项目编译相关
├── plugins ────────────────→        自定义Node模块
├── src
│    ├── components ─────────→        组件
│    ├── config
│    ├── directives
│    ├── filters
│    ├── images
│    ├── libs
│    ├── mixins
│    ├── router
│    ├── sass
│    ├── store ──────────────→        数据中心（Vuex）
│    ├── utils
│    ├── views ──────────────→        视图（页面）
│    ├── * App.vue
│    └── * app.js
├── * deploy.conf
├── * index.html
```

项目实战

开发过程
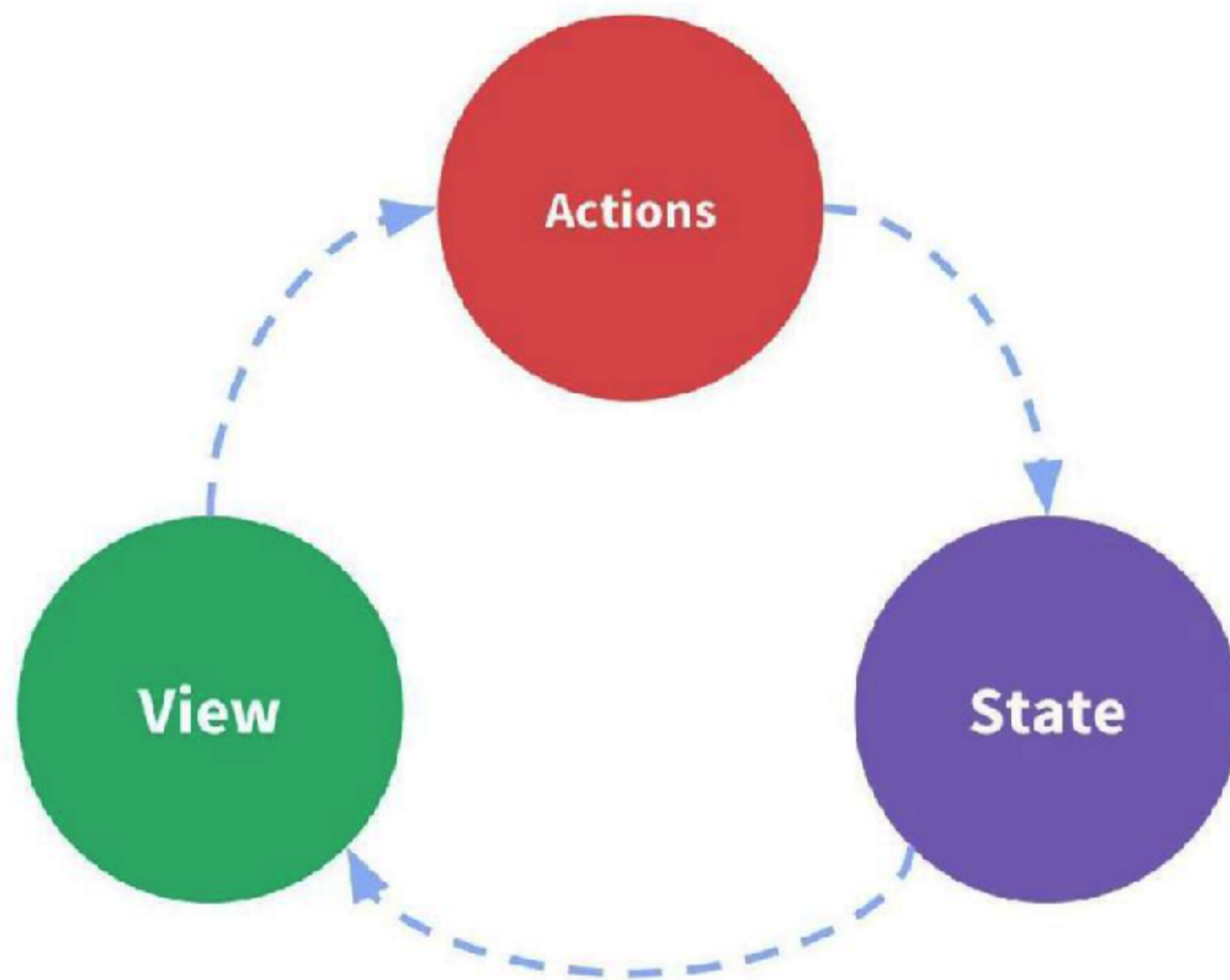
开发辅助

```
├── build
│   ├── config.js
│   ├── dev-server.js
│   ├── loaders.base.conf.js
│   ├── mock.js
│   ├── postcss.conf.js
│   ├── webpack.base.conf.js
│   ├── webpack.dev.conf.js
│   └── webpack.prod.conf.js
```

有开发环境及生产环境的 webpack 配置

项目实战

开发过程

开发辅助

```
resolve: {
  alias: {
    '@': resolve('src'),
    '@img': resolve(['src', 'images'])
  }
},
```

别名，由相对路径变为"绝对路径"

项目实战

开发过程

开发辅助

```
// 关闭
@mixin iconClose {
  mask-border-source: url(~@img/mask/icon_mask_close.png);
  mask-border-slice: 0 0 0 0 fill;
  mask-border-width: 0;
}
```
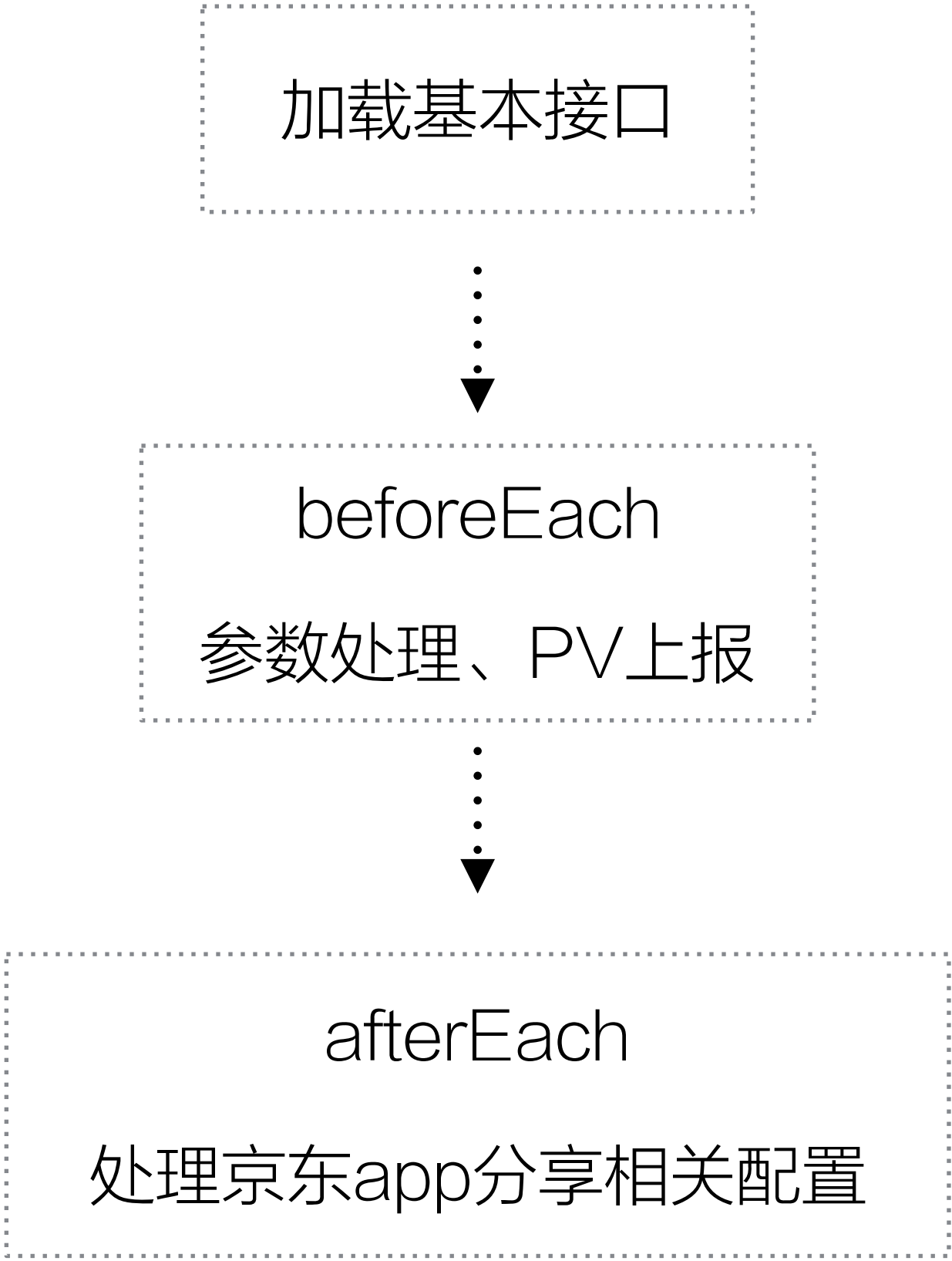
其中图片资源别名（@img）

不仅让路径简写，还让 mixin 可以引用图片资源

项目实战

开发辅助

还有 ESLint 和 Babel 等

项目实战

开发过程

路由

```
{
  name: 'index', path: '/index',
  meta: {title: '陪伴空间', pv: 70},
  components: {default: Index2, navbar: Navbar}
},
```

项目实战

开发过程

路由

加载基本接口

beforeEach

参数处理、PV上报

afterEach

处理京东app分享相关配置

项目实战

开发过程

路由

```
/**
 * fix ios title刷新不了的bug
 */
const iframeLoad = (src) => {
  let iframe = document.createElement('iframe')
  iframe.style.display = 'none'
  iframe.src = src
  document.body.appendChild(iframe)
  iframe.addEventListener('load', function() {
    setTimeout(function() {
      iframe.remove()
    }, 0)
  })
}
```

项目实战

开发过程

组件化

这是一段特别编辑过的代码

```html
<template>
<div v-show="isShow" class="test">
    <!-- slot 的运用 -->
    <slot></slot>
    <slot name="slot2"></slot>
    <template v-if="testProp"></template>
    <template v-else></template>
    <!-- 对于嵌套较深的组件，可以尝试用 Function-type Prop 来代替 emit -->
    <div @click="changeNickname && changeNickname('小镇')"></div>
    <div @click="close()" class="test_btn">{{btnText}}</div>
</div>
</template>

<script>
export default {
  props: {
    testProp: {
      type: [Number, String],
      required: true
    },
    changeNickname: Function
  },
```

slot的运用

项目实战

开发过程

组件化

这是一段特别编辑过的代码

```vue
<template>
<div v-show="isShow" class="test">
  <!-- slot 的运用 -->
  <slot></slot>
  <slot name="slot2"></slot>
  <template v-if="testProp"></template>
  <template v-else></template>
  <!-- 对于嵌套较深的组件，可以尝试用 Function-type Prop 来代替 emit -->
  <div @click="changeNickname && changeNickname('小镇')"></div>
  <div @click="close()" class="test_btn">{{btnText}}</div>
</div>
</template>

<script>
export default {
  props: {
    testProp: {
      type: [Number, String],
      required: true
    },
    changeNickname: Function
  },
  data () {
```

Function-type Prop

项目实战

组件化

这是一段特别编辑过的代码

```
<div v-show="isShow" class="test">
  <!-- slot 的运用 -->
  <slot></slot>
  <slot name="slot2"></slot>
  <template v-if="testProp"></template>
  <template v-else></template>
  <!-- 对于嵌套较深的组件，可以尝试用 Function-type Prop 来代替 emit -->
  <div @click="changeNickname && changeNickname('小镇')"></div>
  <div @click="close()" class="test_btn">{{btnText}}</div>
</div>
</template>

<script>
export default {
  props: {
    testProp: {
      type: [Number, String],
      required: true
    },
    changeNickname: Function
  },
  data () {
    return {
      isShow: false,
      btnText: '',
      closeFn: null
    }
  },
  methods: {
    close () {
```

Props传参

项目实战

开发过程

组件化

这是一段特别编辑过的代码

```
      changeNickname: Function
    },
    data () {
      return {
        isShow: false,
        btnText: '',
        closeFn: null
      }
    },
    methods: {
      close () {
        this.isShow = false
        this.closeFn && this.closeFn()
      },
      // 除了 props 传参，函数传参也是一种方式，有时比 props 更好用
      showPop (btnText = '', closeFn = null) {
        this.isShow = true
        this.btnText = btnText
        this.closeFn = closeFn
      }
    }
  }
</script>
```

Props替代方案

项目实战

开发过程

mixins

几乎每个组件都有埋点

相当多的全局的多页共用的数据

项目实战

开发过程

mixins

```
return {
  // isVip: 是否VIP
  // points: VIP积分（新用户这里即将提供的积分也为0）
  // isVipNew: 是否新用户（新用户的VIP积分为空而不是0）
  // isRealName: 是否实名
  // region: 用户默认收货地址-省市名称
  // regionId: 用户默认收货地址-四级地址
  ...extUserInfo,

  // 是否IOS
  isIOS: /iphone|ipad|ipod/i.test(navigator.userAgent),

  // 宝贝档案
  profiles: profiles,
  // 猜宝贝档案
  guessChild: guess,
  // 档案采源
  profilesSource: children.source || Constants.CHILD_DATA_REAL,
  // 是否大于六岁
  isBeyondSeven,

  // 月龄 = xx,xx
  orMoonsAge: orMoonsAge,
  // 用于 埋点 = x_x，如档案空则为 n_n
  pingMoonsAge: pingMoonsAge,
  // 用于微信手Q埋点的LinkID
  pingLinkID: pingLinkID,
  // 用于埋点，VIP属性_月龄参数
  pingPreParam: `${extUserInfo.isVip ? 1 : 0}_${pingMoonsAge}`
}
```

登陆态

用户信息

档案信息

埋点用的参数

项目实战

开发过程

指令和Filter

```
├── directives
│   ├── vue-backtop.js          ──→  回到顶部
│   ├── vue-lazy-img.js         ──→  图片懒加载
│   ├── vue-lazy.js             ──→  模块懒加载
│   ├── vue-manual-img.js       ──→  运营图加载
│   ├── vue-mfor.js             ──→  多数组交替遍历
│   ├── vue-price.js            ──→  价格获取
│   ├── vue-sticky.js           ──→  吸顶
│   ├── vue-sticky2.js          ──→  吸顶
│   └── vue-timer.js            ──→  倒计时
├── filters
│   └── index.js
```

项目实战

开发过程

常量管理

链接统一管理

```javascript
const DOMAIN = '//mup.jd.com'
const HOST = `${DOMAIN}/dobby`
const NIGHT = `${DOMAIN}/dobby2`
const CHILD = `${DOMAIN}/child`

// 用户扩展信息
export const USER_EXT_INFO = `${HOST}/member_extInfo`
// 福利券信息
export const VIP_COUPON_MAIN = `${HOST}/vip_coupon_main`
// 福利换购信息
export const VIP_PRIVILEGE = `${NIGHT}/vip_privilege`
// 福利换购商品领取 id
export const VIP_PRIVILEGE_GET = `${HOST}/apply_vip_privilege`
// 专享价分类 bizParameterMap[orMoonsAge]
export const EXCLUSIVE_CLASS = `${HOST}/accompany_epa_class`
// 获取专享价列表 bizParameterMap[orMoonsAge] businessType
export const EXCLUSIVE_LIST = `${HOST}/accompany_epa_product`
// VIP优惠券列表
export const VIP_COUPON_INFO = `${HOST}/accompanyCouponInfo`
```

项目实战

开发过程

常量管理

链接统一管理

数据字段统一管理

商品字段常量示例

```
// 基本
export const ID = 'id'
export const SKU = 'sku'
export const LINK = 'link'
export const NAME = 'name'
export const IMAGE = 'image'
// 字段一般为curPrice
export const JD_PRICE = 'jdPrice'
// 字段一般为refPrice
export const PRICE = 'price'
```

项目实战

常量管理

链接统一管理

数据字段统一管理

其他常量管理

```javascript
// 通用三态常量
export const COMMON_STATE_INIT = -1
export const COMMON_STATE_SUCC = 0
export const COMMON_STATE_FAIL = 1


// 购物车状态
export const ADD_CART_INIT = -1
export const ADD_CART_SUCC = 0
export const ADD_CART_FAIL = 1


// 礼包类型
export const GIFTS_BIG = 'giftsBig' // 陪伴大礼包
export const GIFTS_NEW = 'giftsNew' // 新生礼包
export const GIFTS_FULL_MONTH = 'giftsFullMonth' // 满月礼包
export const GIFTS_HUNDRED = 'giftsHundred' // 百日礼包
export const GIFTS_SIX_MONTHS = 'giftsSixMonths' // 半岁礼包
export const GIFTS_AGES = 'giftsAges' // 周岁礼包
export const GIFTS_MONTHS = 'giftsMonths' // 月龄礼包
export const GIFTS_TOPICS = 'giftsTopics' // 主题礼包
export const GIFTS_BOOK = 'giftsBook' // 图书新会员礼包
```

数据中心

Backend API

Vuex

Dispatch

Actions

Commit

Vue Components

Mutations

Devtools

Render
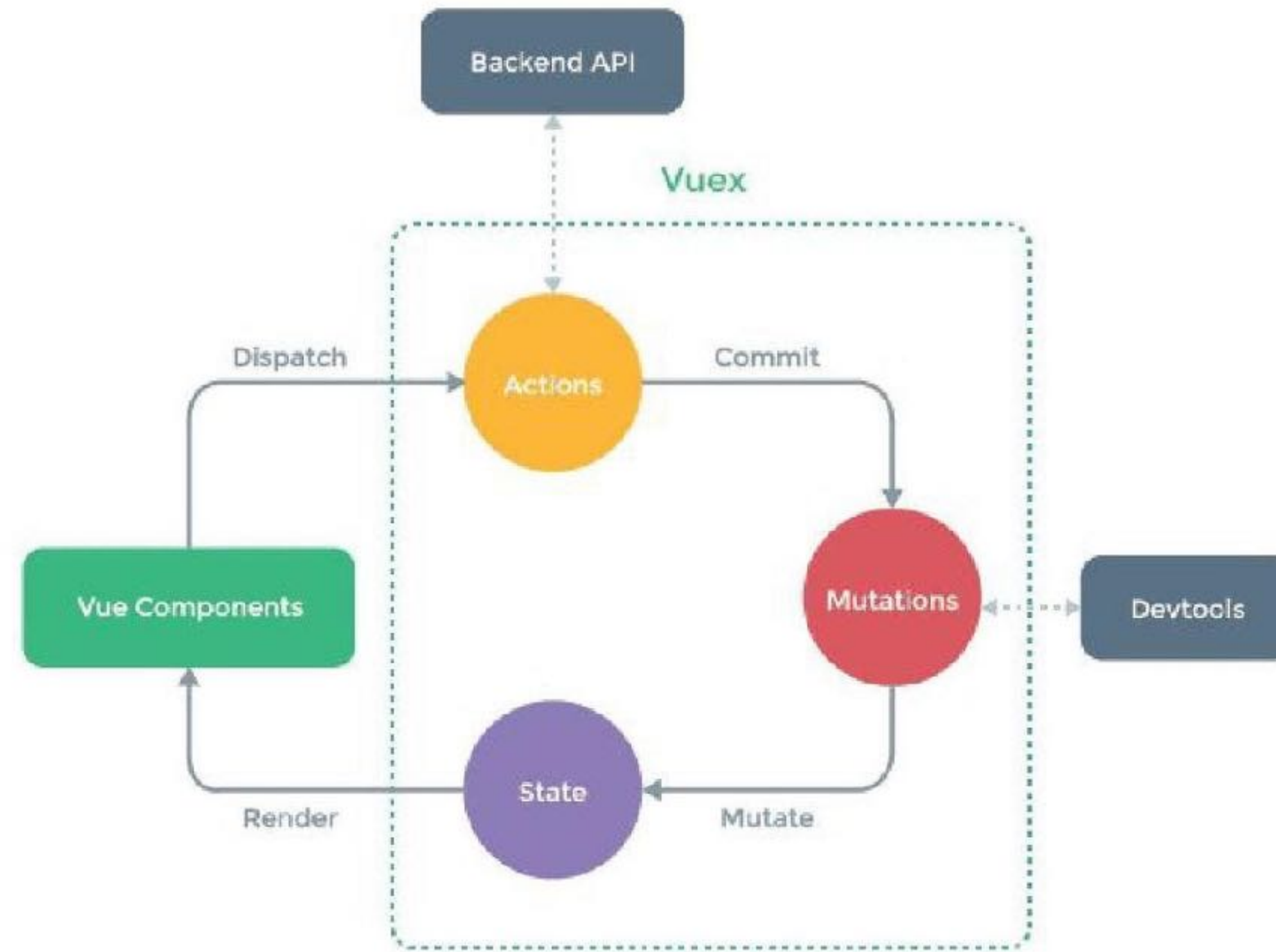
State

Mutate

State

Mutations

Actions

项目实战

```
computed: {
  selectedMoonsAge () {
    let mg = Constants.AGE_GROUPS[this.selectedAgesIdx - 1]
    return mg ? mg.orMoonsAge : this.orMoonsAge
  },
  ...mapGetters({
    pickTop: 'pickTop',
    guideClass: 'guideClass',
    guideList: 'guideList',
    guessIndex: 'guessIndex',
    guessIndexEnd: 'guessIndexEnd'
  })
},
methods: {
  ...mapActions([
    'getPickTop',
    'getGuideClass',
    'getGuideList',
    'getGuessIndex'
  ]),
```

mapGetters

mapActions

项目实战

开发过程

```
import * as URL from '@/config/URL'
import * as Constants from '@/config/Constants'
import * as Commodity from '@/config/Commodity'
import Utils from '@/utils'
import fetch from '@/libs/fetch'
import debug from '@/libs/debug'
import Report from 'o2-analysis'

const PICK_LIST = 'PICK_LIST'                    ——————→ mutation 名称

const state = {
  pickList: [],                                  ——————→ state
  pickListEnd: false
}
const getters = {
  pickList: state => state.pickList              ——————→ getter
}

const mutations = {
  [PICK_LIST] (state, {list, added}) {
    let format = _getPicks(list || [])

    state.pickListEnd = format.length === 0

    if (added) {
      state.pickList = state.pickList.concat(format)
    } else {
```

Actions

Mutations

State

getters

项目实战

actions

Actions

Mutations

State

getters

定义action

```
    if (added) {
      state.pickList = state.pickList.concat(format)
    } else {
      state.pickList = format
    }
  }
}

const actions = {
  getPickList ({commit}, {orMoonsAge = '', page, added}) {
    fetch({
      url: URL.PICK_LIST
    }, {
      params: {
        orMoonsAge,
        common_page_index: page,
        common_page_size: 10
      }
    }).then((ret) => {
      const list = ret && ret.accompany_fine_goods_list || []
      commit(PICK_LIST, {
        list,
        added
      })
    }, (error) => {
      debug(error)
    })
  }
}

function _getPickItem (item) {
  let now = Date.now()
  let start = Utils.getDateTime(item.startTime).getTime()
```
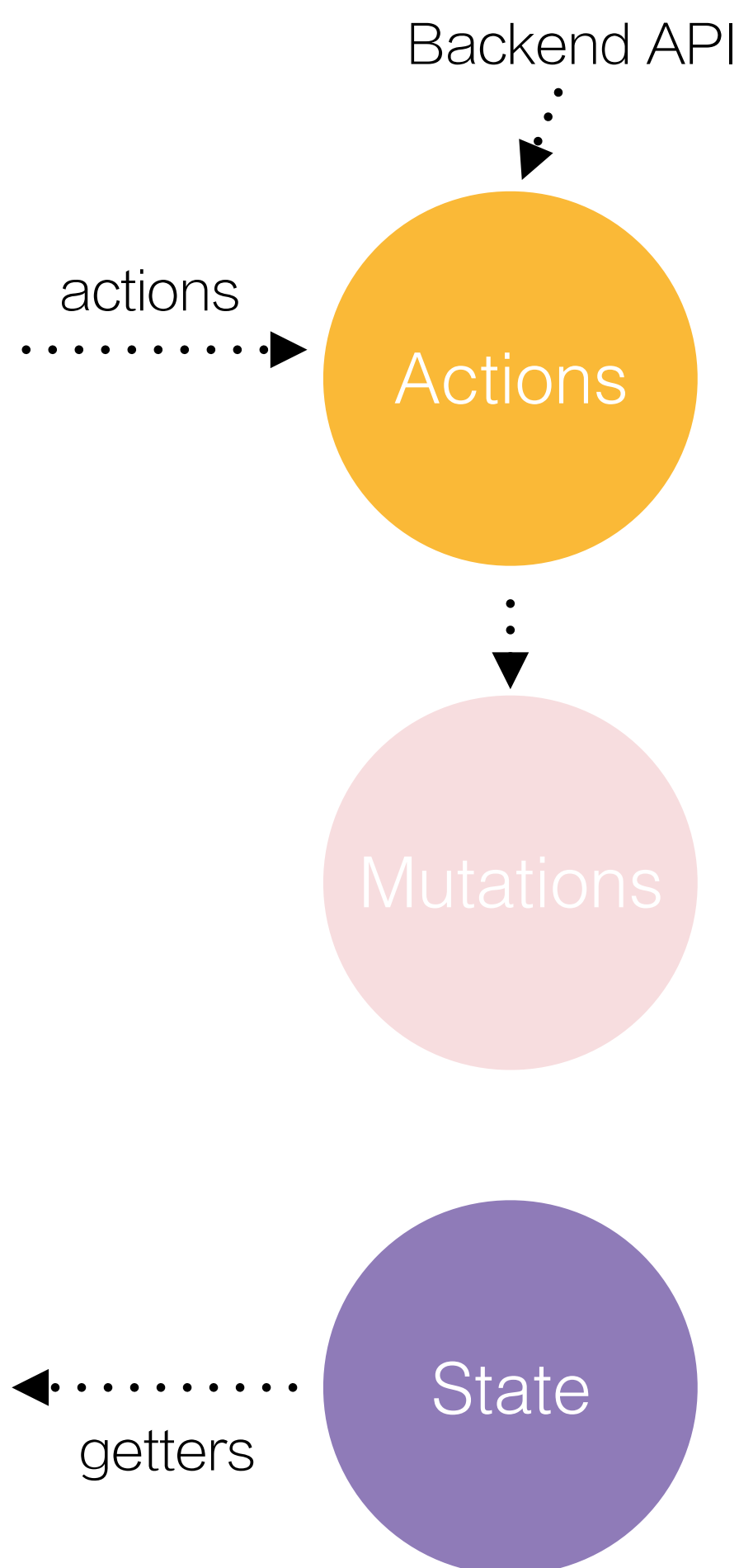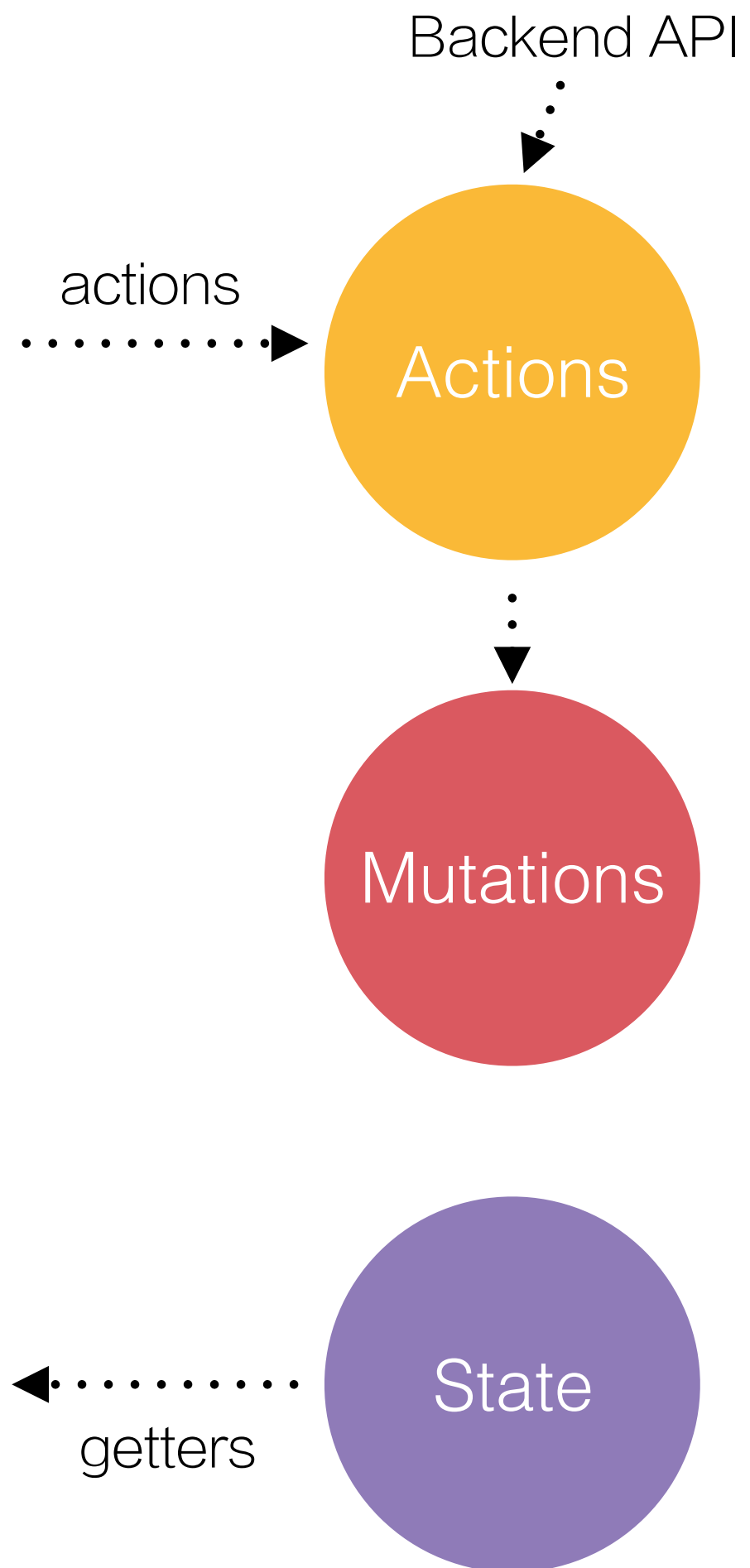
项目实战

actions

Actions

Mutations

State

getters
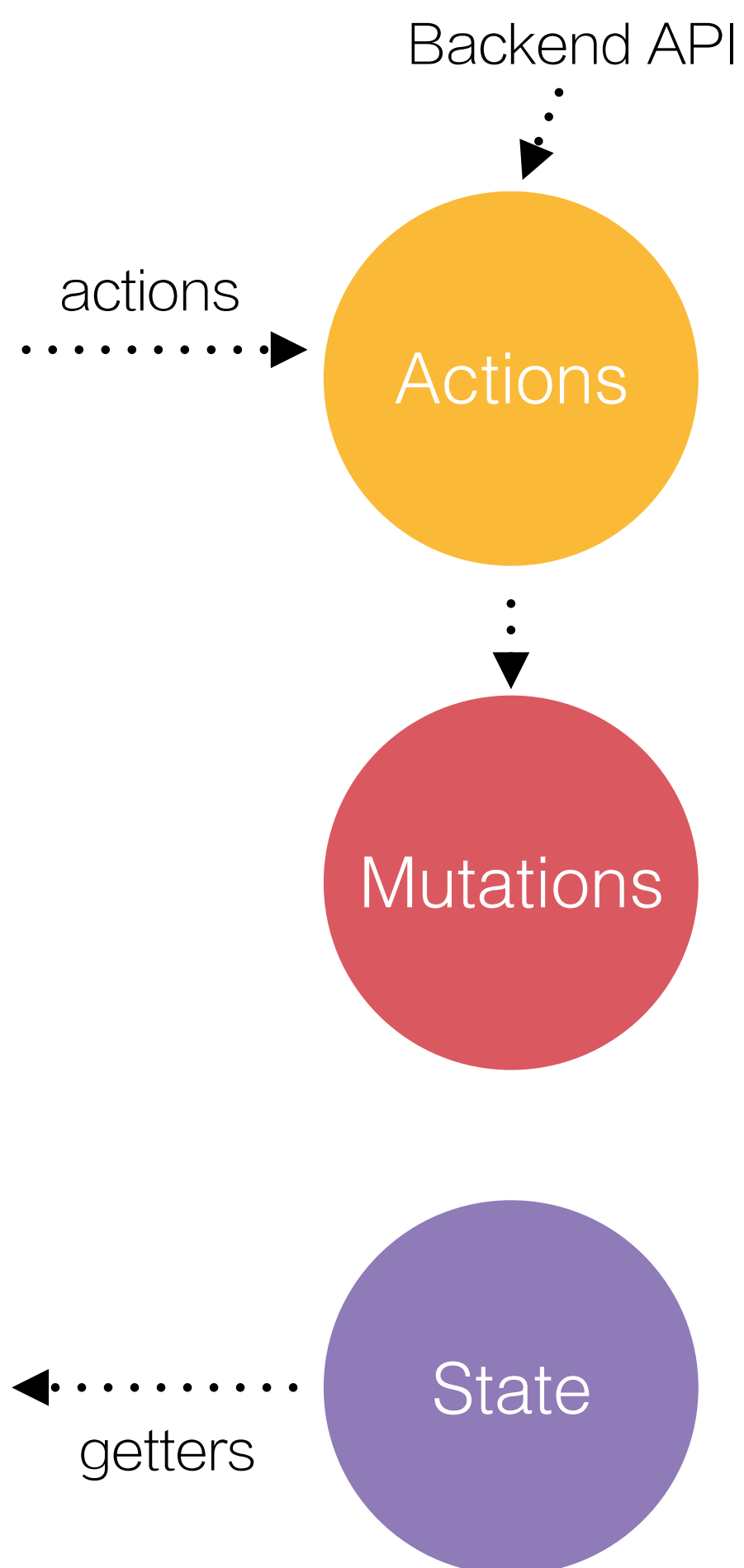
Backend API

commit

```javascript
    if (added) {
      state.pickList = state.pickList.concat(format)
    } else {
      state.pickList = format
    }
  }
}

const actions = {
  getPickList ({commit}, {orMoonsAge = '', page, added}) {
    fetch({
      url: URL.PICK_LIST
    }, {
      params: {
        orMoonsAge,
        common_page_index: page,
        common_page_size: 10
      }
    }).then((ret) => {
      const list = ret && ret.accompany_fine_goods_list || []
      commit(PICK_LIST, {
        list,
        added
      })
    }, (error) => {
      debug(error)
    })
  }
}

function _getPickItem (item) {
  let now = Date.now()
  let start = Utils.getDateTime(item.startTime).getTime()
```

项目实战

Backend API

actions

Actions

Mutations

State

getters

```javascript
import debug from '@/libs/debug
import Report from 'o2-analysis'

const PICK_LIST = 'PICK_LIST'

const state = {
  pickList: [],
  pickListEnd: false
}
const getters = {
  pickList: state => state.pickList
}

const mutations = {
  [PICK_LIST] (state, {list, added}) {
    let format = _getPicks(list || [])


    state.pickListEnd = format.length === 0


    if (added) {
      state.pickList = state.pickList.concat(format)
    } else {
      state.pickList = format
    }
  }
}

const actions = {
  getPickList ({commit}, {orMoonsAge = '', page, added}) {
    fetch({
      url: URL.PICK_LIST
    }, {
      params: {
        orMoonsAge,
```
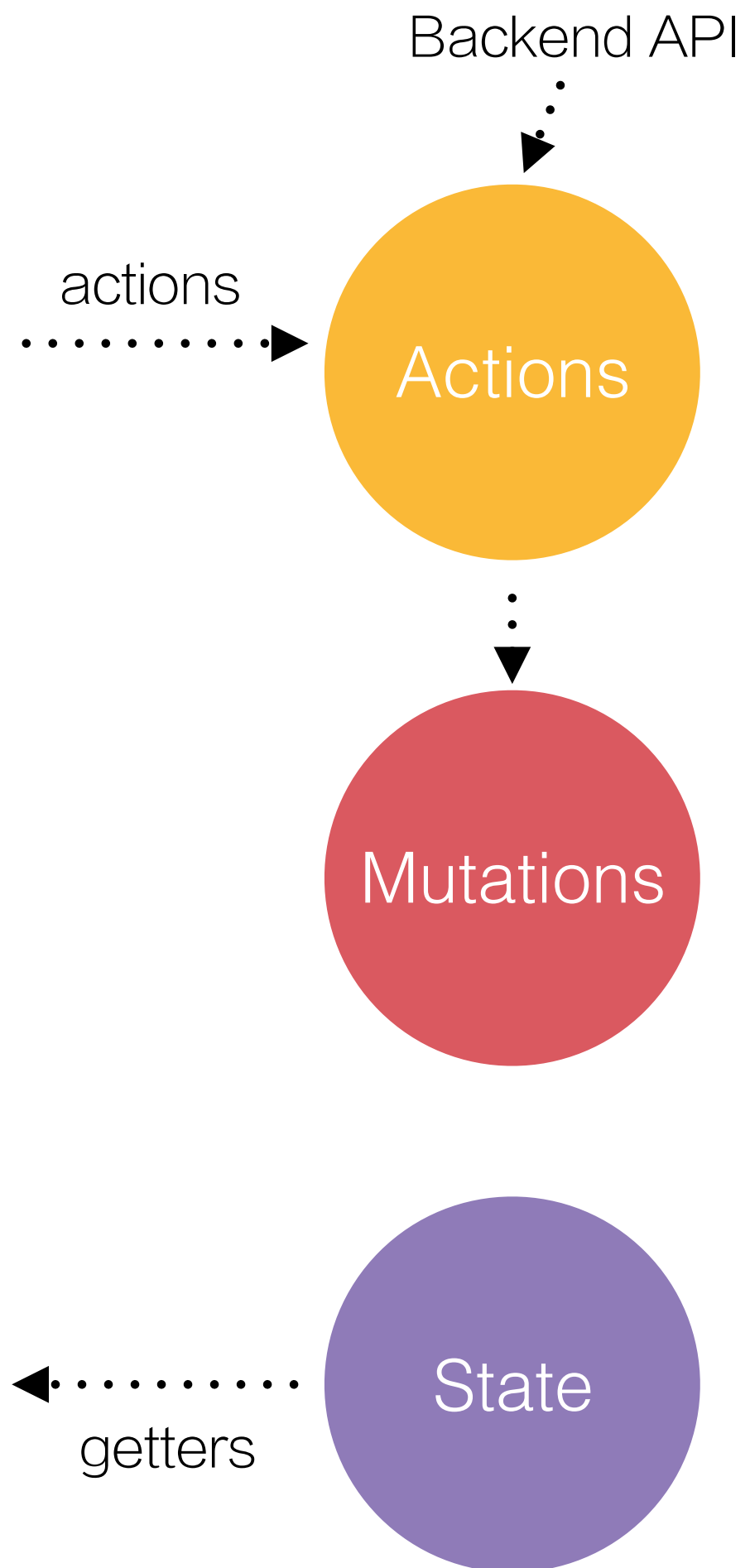
mutation

项目实战

Backend API

actions

Actions

Mutations

State

getters

```javascript
import debug from '@/libs/debug'
import Report from 'o2-analysis'

const PICK_LIST = 'PICK_LIST'

const state = {
  pickList: [],
  pickListEnd: false
}
const getters = {
  pickList: state => state.pickList
}

const mutations = {
  [PICK_LIST] (state, {list, added}) {
    let format = _getPicks(list || [])

    state.pickListEnd = format.length === 0

    if (added) {
      state.pickList = state.pickList.concat(format)
    } else {
      state.pickList = format
    }
  }
}

const actions = {
  getPickList ({commit}, {orMoonsAge = '', page, added}) {
    fetch({
      url: URL.PICK_LIST
    }, {
      params: {
        orMoonsAge,
```

预处理数据

项目实战

Backend API

actions

Actions

Mutations

State

getters

```javascript
        added
      })
  }, (error) => {
      debug(error)
  })
  }
}
function _getPickItem (item) {
  let now = Date.now()
  let start = Utils.getDateTime(item.startTime).getTime()
  let end = Utils.getDateTime(item.endTime).getTime()
  let validity = start < now && end > now
  return {
    [Commodity.ID]: item.id,
    [Commodity.SKU]: item.sku,
    [Commodity.NAME]: item.name,
    [Commodity.IMAGE]: (item.skuImgUrl || '').split(',').filter((item) => item),
    // [Commodity.IMAGE]: _getImageUrl(item.skuImgUrl),
    [Commodity.JD_PRICE]: item.curPrice,
    [Commodity.PRICE]: item.refPrice,
    [Commodity.LINK]: Utils.getDetailUrl(item.sku),
    [Commodity.REMARK]: item.bizParam,
    ['validity']: validity
  }
}
function _getPicks (gds) {
  return gds.map((item) => {
    return _getPickItem(item)
  })
}

export default {
  state,
```
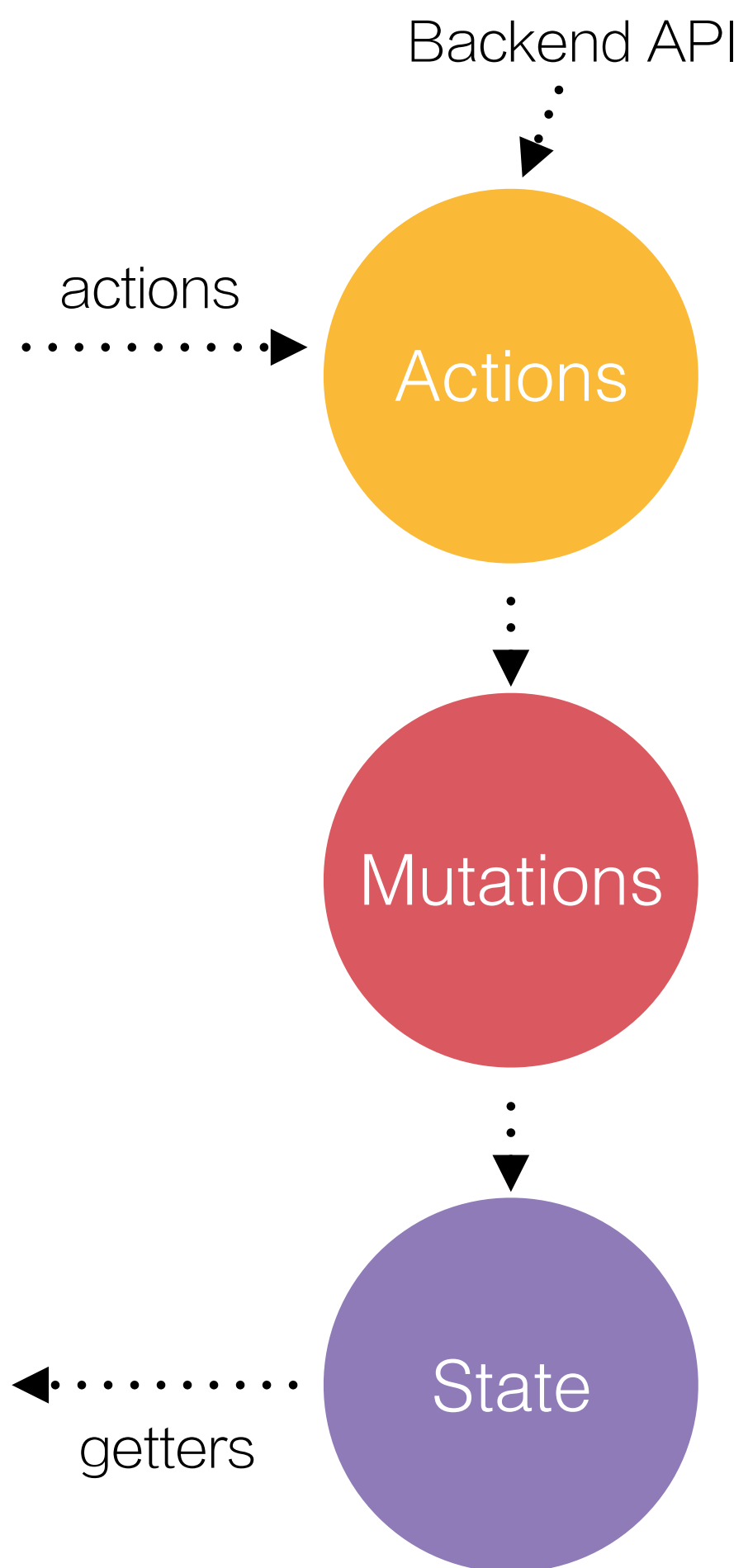
预处理数据

项目实战

Backend API

actions

Actions

Mutations

State

getters

```javascript
import Report from 'o2-analysis'

const PICK_LIST = 'PICK_LIST'

const state = {
  pickList: [],
  pickListEnd: false
}
const getters = {
  pickList: state => state.pickList
}

const mutations = {
  [PICK_LIST] (state, {list, added}) {
    let format = _getPicks(list || [])

    state.pickListEnd = format.length === 0

    if (added) {
      state.pickList = state.pickList.concat(format)
    } else {
      state.pickList = format
    }

  }
}

const actions = {
  getPickList ({commit}, {orMoonsAge = '', page, added}) {
    fetch({
      url: URL.PICK_LIST
    }, {
      params: {
        orMoonsAge,
        common_page_index: page
```

项目实战

开发过程

环境兼容

```
import UA from '@/libs/ua'
import * as uaTypes from '@/config/ua-types'

import App from './app'
import Mobile from './mobile'
import Wechat from './wechat'
import QQ from './qq'

let configs = {
  [uaTypes.APP]: App,
  [uaTypes.WECHAT]: Wechat,
  [uaTypes.QQ]: QQ,
  [uaTypes.MOBILE]: Mobile
}

export default configs[UA.type]
```

兼容App、微信、手Q和浏览器

包括 购物车相关、商详页链接、优惠券链接、搜索链接 等环境差异的方法

项目实战

开发过程

其他

接口延迟 ┈┈┈┈┈┈▶ 封装请求方法，做重试处理

请求的返回顺序 ┈┈┈┈▶ 记录请求顺序 / state变量包含所有id

减少重复请求 ┈┈┈┈▶ 在变量中按参数暂存数据

滚动条行为 ┈┈┈┈▶ 在store中存储每页离开时的滚动条位置

……

项目实战

组件化

模块化

命名
规范

Git工作流

全局组件

业务分块

命名规范

commit 规范

页面组件

逻辑分块

字段统一

分支管理

ATHENA

马赛克

开发

测试

体验

上线

监控

技术展望
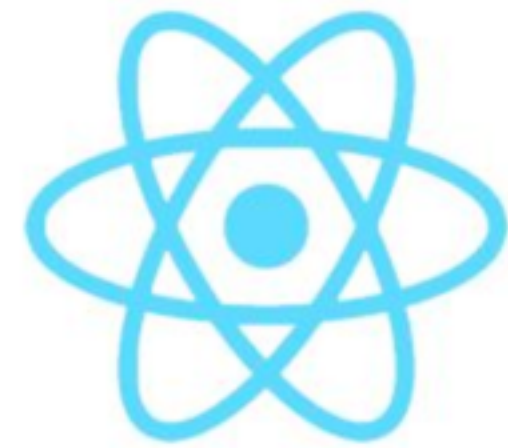
JD React Native
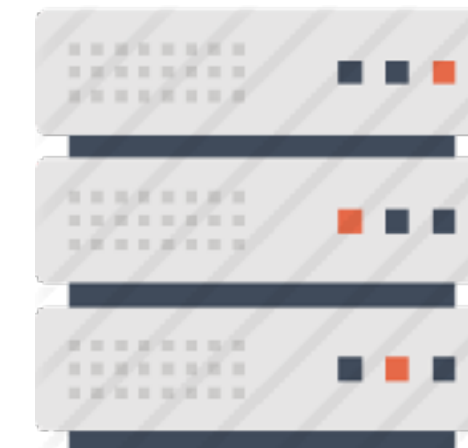
Server Side Render

THANKS

FOR YOUR WATCHING

O2
OPEN ORIENTED
凹 凸 实 验 室