

# 企业级业务开发之Node的价值

网易-严选事业部

金炳



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



# 个人介绍



**Name:** 金炳

**Web developer** @yanxuan

**Github:** stone-jin

**知乎:** 金炳

## Doing

目前致力于Node应用框架开发及生态建设，  
实践Node应用在Serverless、FaaS场景下的迁移和落地，  
探索service mesh在Node应用中的价值。



# 目录

- 一、业务介绍
- 二、Node的作用
- 三、Node的保障
- 四、总结





# 网易严选 网易旗下原创生活类自营电商品牌



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



## 电商前台业务

APP/H5商城

线下店

小程序/PC

渠道平台

....



支撑

## 电商后台业务

商品中心

订单中心

库存中心

会员中心

促销中心

CRM系统

评价中心

采购中心

WMS

物流中心

风控中心

客服系统

财务管理

营销中心

数据中心

支付中心

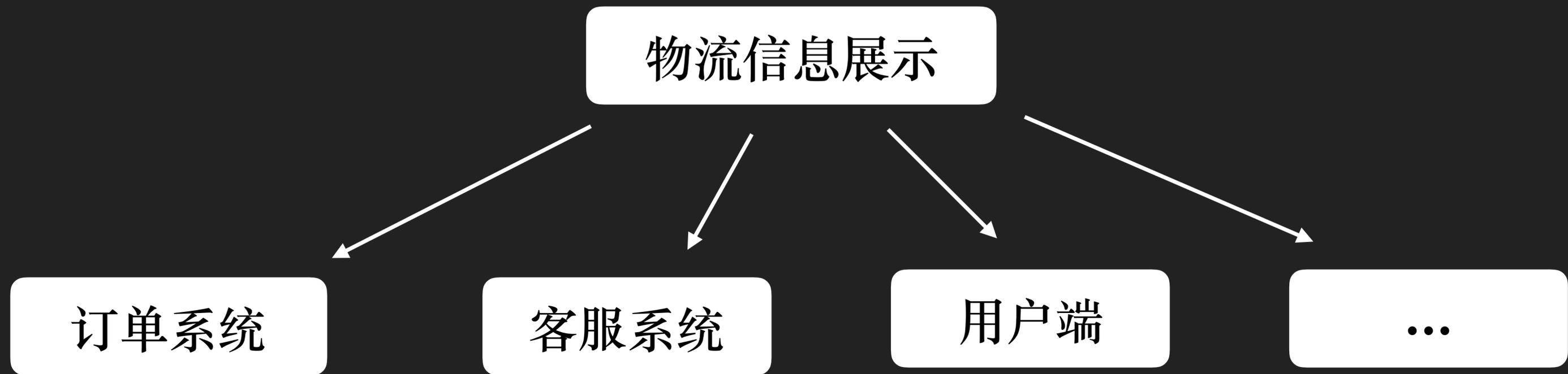
## 产品架构（简化版）



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团





关联性强



权限模块、反馈模块

库存中心

运营中心

评价中心

...

关联性强



如何处理跨系统 关联性强 的业务场景??



前端

基础组件

---

Node



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



前端

基础组件



业务组件

Node



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



# 业务组件-举个例子

## 追踪物流

 物流公司: [redacted]  
快递单号: [redacted]  
4月17日已送达, 比预计提前1天 [查看商品](#)

---

2019-04-17 17:52:34 [redacted]

2019-04-17 17:52:20 [redacted]

2019-04-17 17:51:38 订单已再投 [redacted] 客户  
约定新派送 [redacted]

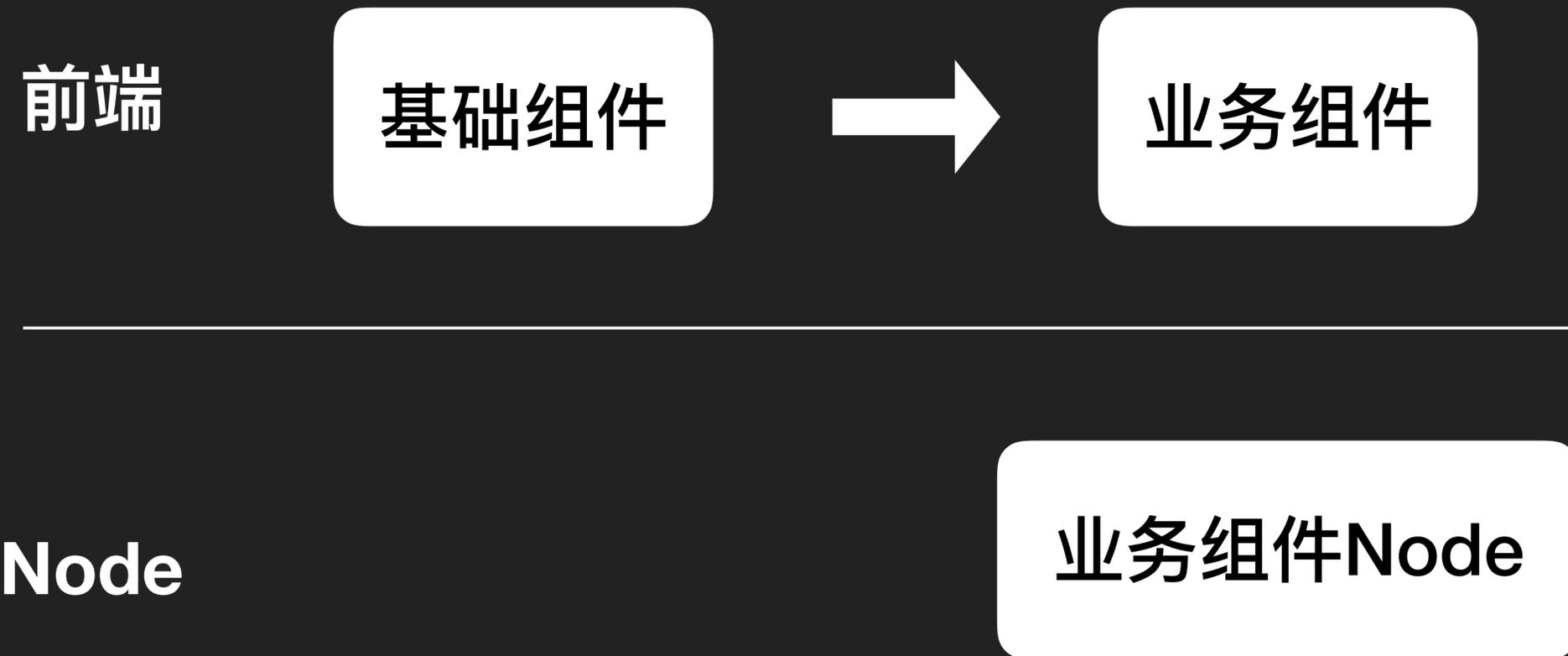
2019-04-17 10:33:10 [redacted] 中 (快递员: 谷秀云, 电话: [redacted])  
请您耐心等待

2019-04-17 00:50:51 您的订单已到 [redacted]

多系统通用, 与业务紧密关联的组件



## 二、Node应用



## 前端业务组件：

```
<yx-logistics-trace [orderId]="orderId"></yx-logistics-trace>
```

## Node承载数据服务：

```
import { LogisticsModule } from '@yx-module/logistics-node';  
  
app.use(LogisticsModule.routes());
```





```
import { BaseModule, TgModule } from '@tiger/boot';
import { LogisticsController } from '../logistics.controller';

@TgModule({
  imports: [],
  controllers: [LogicsController],
  middlewares: [],
  tasks: []
})
export class LogisticsModule extends BaseModule{
}
```

## 数据服务承载层





```
import { RestController, RequestMapping, GetMapping } from '@tiger/boot';
import { Context } from 'koa';
import { LogisticsService } from './logistics.service';
import { valid } from '@tiger/validator';
import { LogisticsVO } from './logistics.vo';

@RestController
@RequestMapping("/logistics")
export class LogisticController{
  constructor(private logisticsService: LogisticsService){
  }

  /**
   * 获取物流信息
   */
  @GetMapping("/getInfo.do", [valid(LogisticsVO)])
  async getInfo(ctx: Context, next: any){
    // 调用service层, 对接后端服务
    ctx.body = await this.logisticsService.getInfo();
  }
}
```

## 接口承载方



# 模块的复用?



推啊前端技术开放日

主办方: **Duiba Group**  
兜吧集团



# 业务模块-举个例子

The image shows a web application interface for user and role management. The main interface is titled "网选 用户角色权限组织演示系统" (Wangxuan User Role Permission Organization Demonstration System). It features a dark sidebar on the left with navigation items: 使用说明, 已接入项目, 权限模块, 用户列表 (highlighted), 角色列表, 权限列表, 严选公共组织架构, 业务系统组织架构, 指令型的演示, 服务类型的演示, and 工单模块. The main content area is titled "用户管理" (User Management) and contains a "新增用户" (Add User) button and a table of users with columns for 用户ID (User ID) and 角色 (Role). The table lists users with IDs 17 through 8. A modal window is open over the "权限管理" (Role Management) section, showing a tree view of permissions. The tree includes categories like "使用说明", "已接入项目", "权限模块", "指令型的演示", "服务类型的演示", and "工单模块". The "权限模块" category is expanded, showing sub-items like "用户列表", "角色列表", "权限列表", "严选公共组织架构", and "业务系统组织架构". The "用户列表" item is selected. To the right of the tree, a "当前选择的权限" (Currently Selected Permission) panel displays details for the selected item: 父权限ID (Parent Permission ID): 300, 权限ID (Permission ID): 3001, 权限名称 (Permission Name): 用户列表, 类型 (Type): 导航页面, 前端路由URL (Frontend Route URL): userCenterManage/userList, 后端接口 (Backend Interface):, and 排列顺序 (Order): 0. At the bottom of this panel are buttons for "修改" (Modify), "添加子权限" (Add Sub-Permission), and "删除" (Delete).



业务模块：包含多个业务功能的集合。



前端

基础组件



业务组件



业务模块

Node

业务组件Node

业务模块Node



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



# 业务模块-使用

## 前端

```
import { UserCenterManageModule } from '@yx-module/user-center-manage';

@NgModule({
  imports: [
    ...
    UserCenterManageModule
    ...
  ]
})
export class AppModule {}
```

## Node数据承载层:

```
import { UserCenterManageModule } from '@yx-module/user-center-manage-node';

app.use(UserCenterManageModule.routes());
```





```
import { TgModule, BaseModule } from '@tiger/boot';
import { UserModule } from './user/user.module';
import { RoleModule } from './role/role.module';
import { PermModule } from './perm/perm.module';
import { OrgModule } from './org/org.module';

@TgModule({
  prefix: '',
  imports: [UserModule, RoleModule, PermModule, OrgModule],
  controllers: [],
  middlewares: [],
  tasks: []
})
export class UserCenterManageModule extends BaseModule{
}
```

## 数据服务承载层





```
import { TgModule, BaseModule } from '@tiger/boot';  
import { UserController } from '../user.controller';
```

```
@TgModule({  
  imports: [],  
  controllers: [UserController],  
  middlewares: [],  
  tasks: []  
})  
export class UserModule extends BaseModule{  
}
```

## 数据服务承载层 子模块





```
import { RestController, RequestMapping, GetMapping, PostMapping } from '@tiger/boot';
import { Context } from 'koa';
import { UserService } from './user.service';

@RestController
@RequestMapping("/user")
export class UserController{
  constructor(private userService: UserService){}

  // 查询用户列表
  @GetMapping("/list.do")
  async list(ctx: Context, next: any){}

  // 更新用户信息
  @PostMapping("/update.do")
  async update(ctx: Context, next: any){}

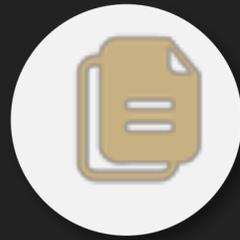
  // 新增用户信息
  @PostMapping("/add.do")
  async add(ctx: Context, next: any){}

  // 锁定用户
  @PostMapping("/lock.do")
  async lock(ctx: Context, next: any){}

  // 导出excel
  @GetMapping("/export.do")
  async export(ctx: Context, next: any){}
}
```

## 接口集

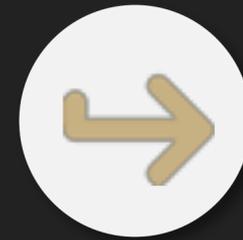
# 业务模块建设



多功能模块的集合



多页面的聚合



路由的集成

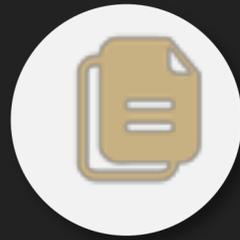


推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



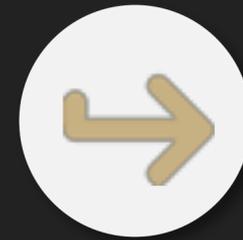
# 业务模块建设



多功能模块的集合



多页面的聚合



路由的集成

业务组件与业务模块的区别

点-面



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



# 工作台?



推啊前端技术开放日

主办方: **Duiba Group**  
兜吧集团





```
import { UserCenterManageModule } from '@yx-module/user-center-manage';
import { FeedBackModule } from '@yx-module/feed-back';
import { GoodsManageModule } from '@yx-module/good-manage';

@NgModule({
  imports: [
    UserCenterManageModule, //权限模块
    FeedBackModule, // 反馈模块
    GoodsManageModule //商品管理模块
    // ... 其他模块
  ]
})
class AppModule{
}
```

前端



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团





```
import { TgModule, BaseModule } from '@tiger/boot';
import { UserCenterManageModule } from '@yx-module/user-center-manage-node';
import { FeedBackModule } from '@yx-module/feed-back-node';
import { GoodsManageModule } from '@yx-module/good-manage-node';

@TgModule({
  imports: [
    UserCenterManageModule, //权限模块
    FeedBackModule, // 反馈模块
    GoodsManageModule //商品管理模块
    // ... 其他模块
  ]
})
class AppModule extends BaseModule{
}

app.use(AppModule.routes());
```

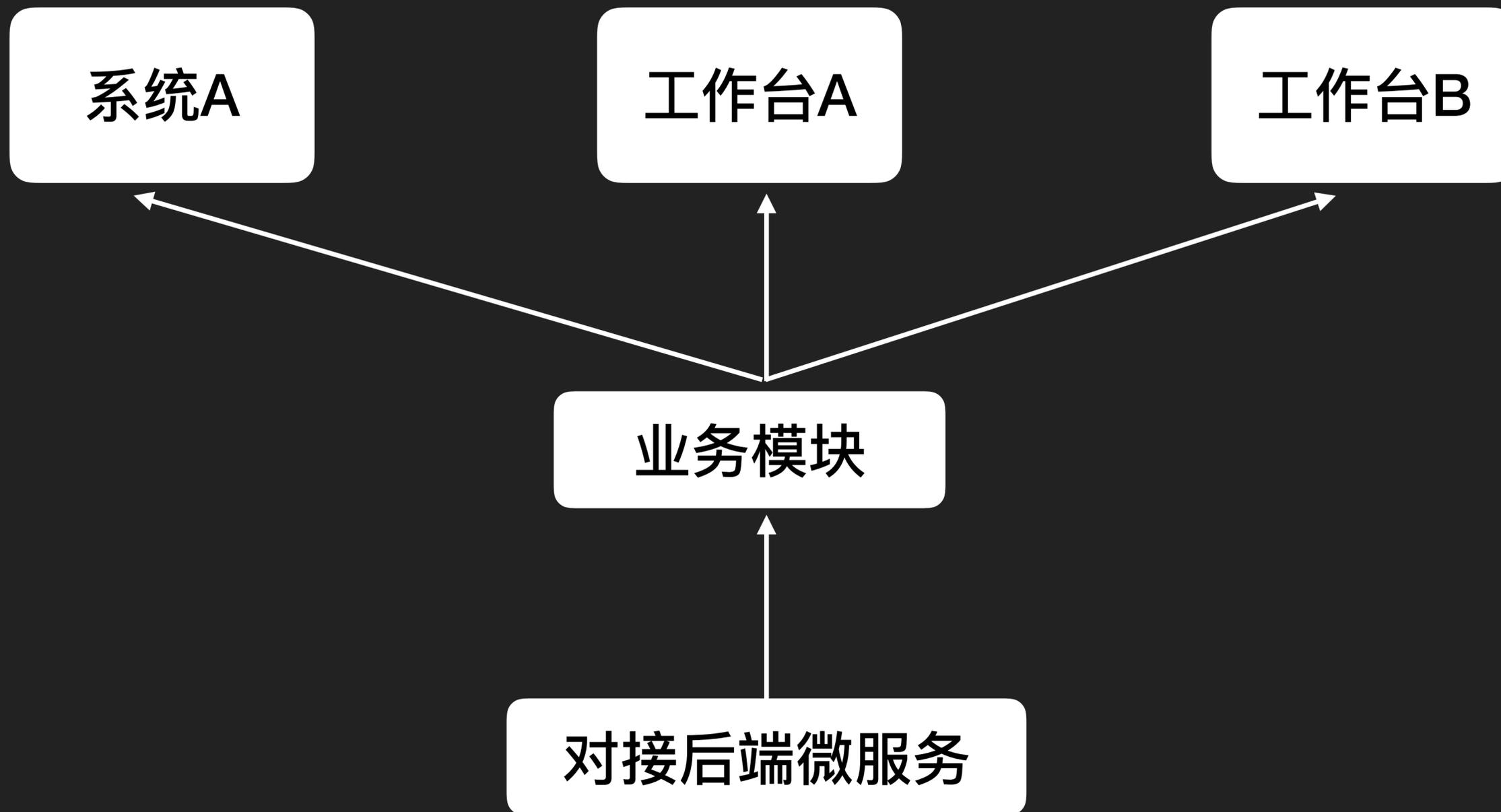
# Node



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团





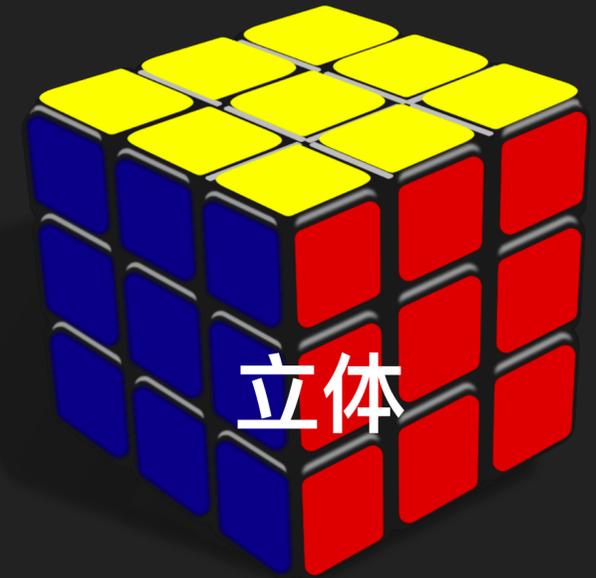
# Node的使用



业务组件



业务模块



工作台



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



# Node使用-角色转变



推啊前端技术开放日

主办方: **Duiba Group**  
兜吧集团



# Node使用-角色转变

业务组件、业务模块

接口的使用方



接口的提供方

如何高效调试我们的接口? curl、postman、swagger?



Authorize

### orgController

- GET** /icac/xhr/org/downOrgUserInfo.xlsx 下载某个orgPosId下的用户信息
- POST** /icac/xhr/org/uploadUserListFile.do 上传批量用户信息的excel
- POST**

Authorize

### userCo

- GET**
- GET**
- GET**
- POST**

### orgController

**GET** /icac/xhr/org/downOrgUserInfo.xlsx 下载某个orgPosId下的用户信息

Parameters Cancel

Name	Description
<b>orgPosId</b> * required number (query)	组织节点id <input type="text" value="orgPosId - 组织节点id"/>
<b>orgName</b> * required string (query)	组织架构的orgName <input type="text" value="orgName - 组织架构的orgName"/>
<b>nodeName</b> * required string (query)	组织节点的名字 <input type="text" value="nodeName - 组织节点的名字"/>

**Execute**

# 引入swagger 高效调试接口



传统swagger方式

swagger中间件开发

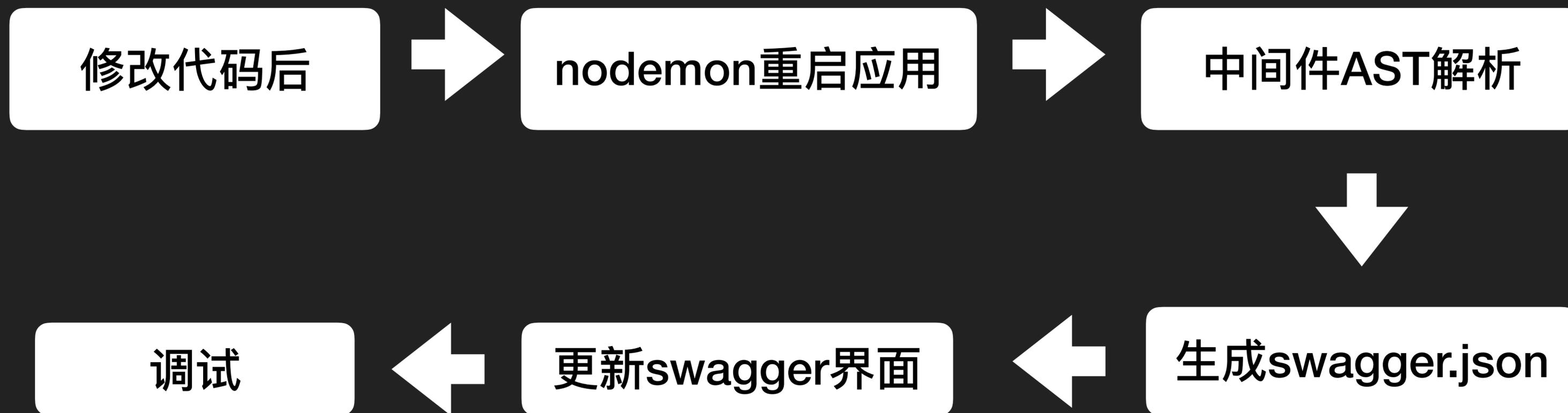
解析ast

生成swagger.json

UI



# Node使用-开发调试流程



# Node使用-反哺前端



加速业务组件、业务模块、业务的前端开发





```
/**  
 * 新增模板  
 */  
export function postTemplateUsingPosing(  
    requestBodyV0: RequestBodyV0,  
    cache: boolean | number = false  
): Promise<ResultV0<String>>{  
}
```

## 前端SDK -typescript



# 如何进行权限拦截、参数校验？



推啊前端技术开放日

主办方: **Duiba Group**  
兜吧集团



# 权限校验



推啊前端技术开放日

主办方: **Duiba Group**  
兜吧集团



# 前端

```
<button type="button" *perm="'user_add'" (click)="addUser()">新增用户</button>
```

# Node

```
// 新增用户  
@PostMapping("/add.do", [perm("user_add")])  
async add(ctx: Context, next: any){}
```



# 参数校验



推啊前端技术开放日

主办方: **Duiba Group**  
兜吧集团





```
@PostMapping("/add.do", [valid(UserV0)])  
async add(ctx: RequestContext<UserV0, ResultV0>, next: any){ }
```



```
export class NameV0{  
  // 用户名称  
  @Validation(Type.string().max(10).min(5))  
  name: string;  
}
```



```
export class NameV0{  
  // 用户名称  
  @IsEmail()  
  @MinLength(16)  
  name: string;  
}
```



## 三、Node保障

稳定性? 性能? 监控?



# 基建保障



推啊前端技术开放日

主办方: **Duiba Group**  
兜吧集团



# 1、监控中心

- 内存、CPU等监控
- 流量等监控
- 服务状态等监控



## 2、报警中心

- 服务状态报警
- 业务出错报警
- 等



## 3、CI/CD

- 加速上线流程
- 回滚



## 4、日志平台

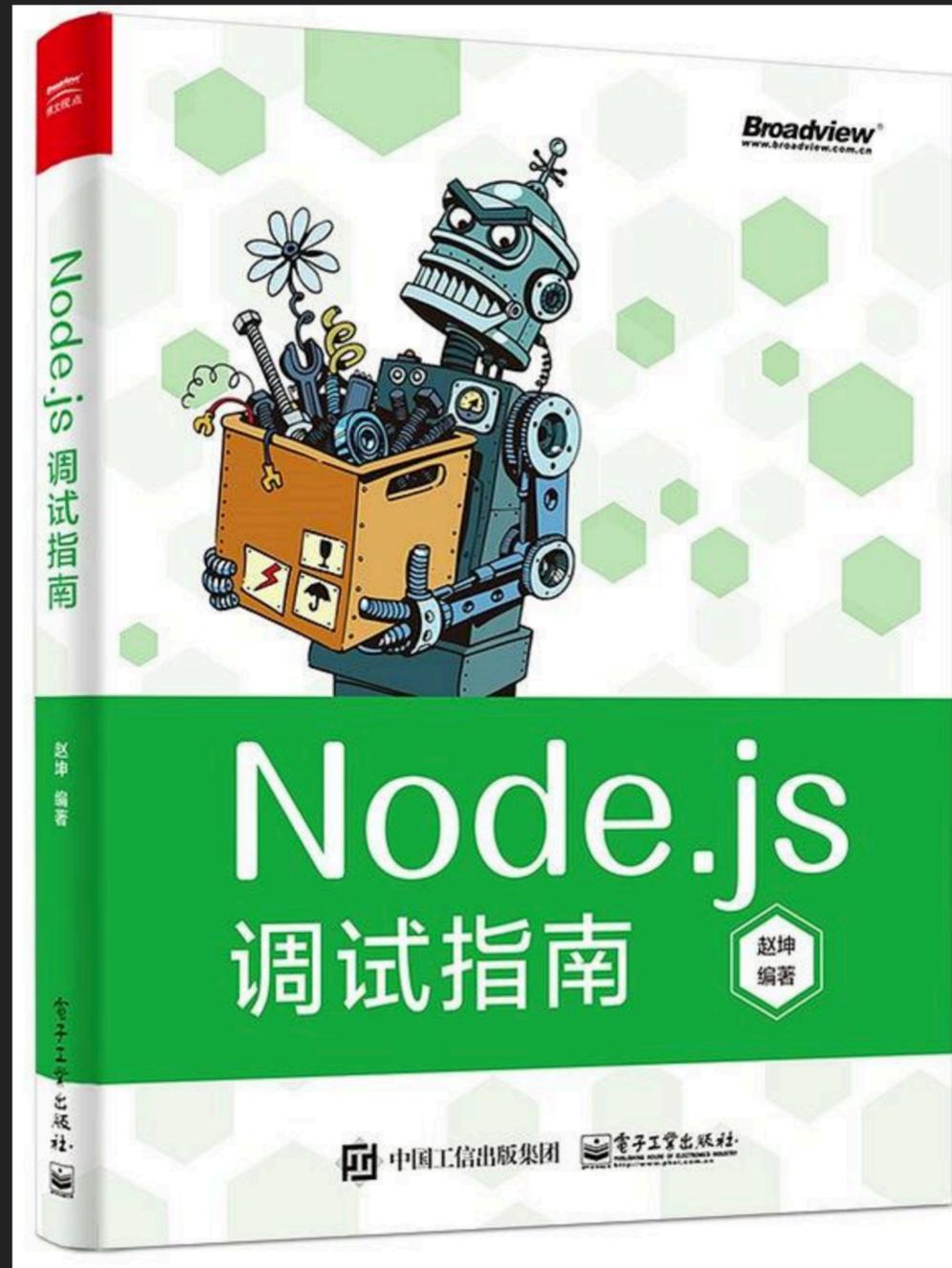
- 日志聚合
- 日志查询



## 5、分布式链路跟踪

- 链路串联
- 错误链路排查
- 慢链路排查





推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



# 四、总结



推啊前端技术开放日

主办方: **Duiba Group**  
兜吧集团



# Node作为前端最友好的后端语言，你值得拥有！



推啊前端技术开放日

主办方: Duiba Group  
兜吧集团



# Q&A

