

The logo for SACC (System Architect Conference China) features the letters 'SACC' in a bold, white, sans-serif font. The letters are outlined with a glowing blue light effect. The 'S' and 'A' have horizontal lines through them, and the 'C's are also outlined with blue light. The logo is set against a dark blue background with a futuristic, wireframe architectural design.

2021 中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2021

数字转型 架构重塑

IT168.com

ChinaUnix.net

ITPUB

云上会议 网络直播 | 2021.5.20-2021.5.22

个人介绍

- 李征，去哪儿网技术总监/业务研发技术委员会委员
- 去哪儿网机票C端前台服务负责人
- 持续专注于基于 API 治理的领域能力标准化、领域服务治理、多租户理念的业务模型落地等工作。
- 致力于通过领域化、模型化、可感知来解决业务复杂问题。期望以 DDD 领域驱动降低系统复杂度，提升团队效能。
- 近期在BFF（服务于前端的后端）的实践思路中进行了大量思考，希望推动业务服务接入简单化。



DDD落地之API文档化

让领域服务治理走向有迹可循

主讲人：李征 / 去哪儿网

Qunar Engineering Director

CONTENTS 目录

DDD启示录

为什么需要API

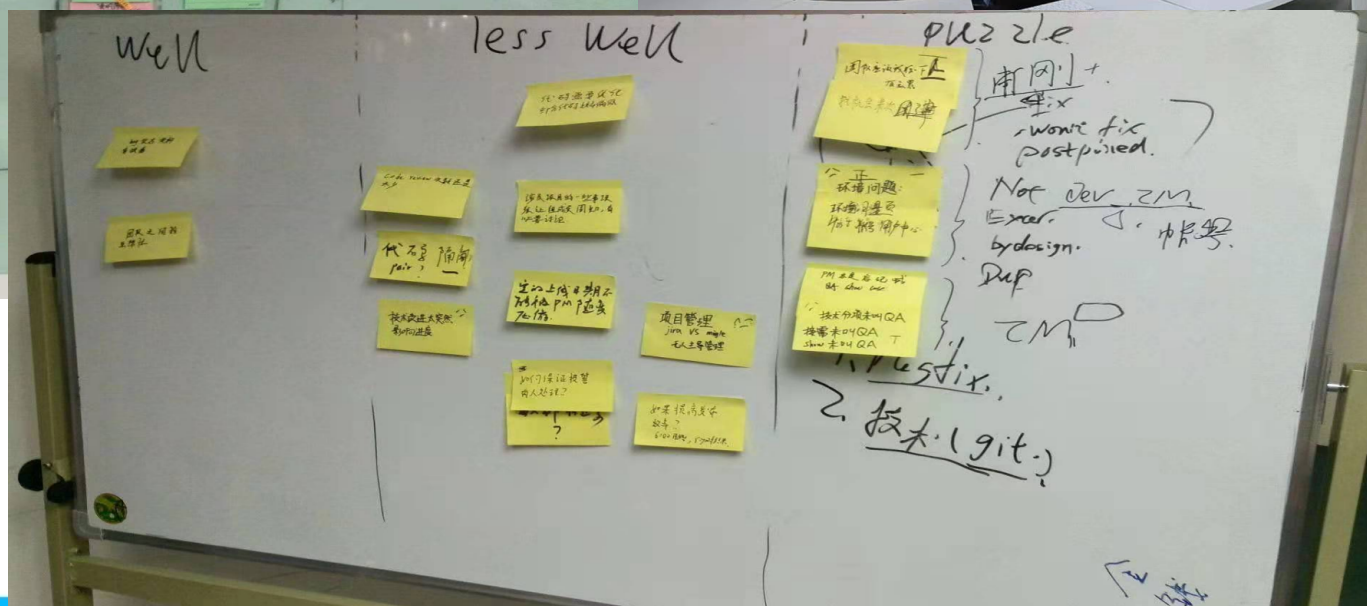
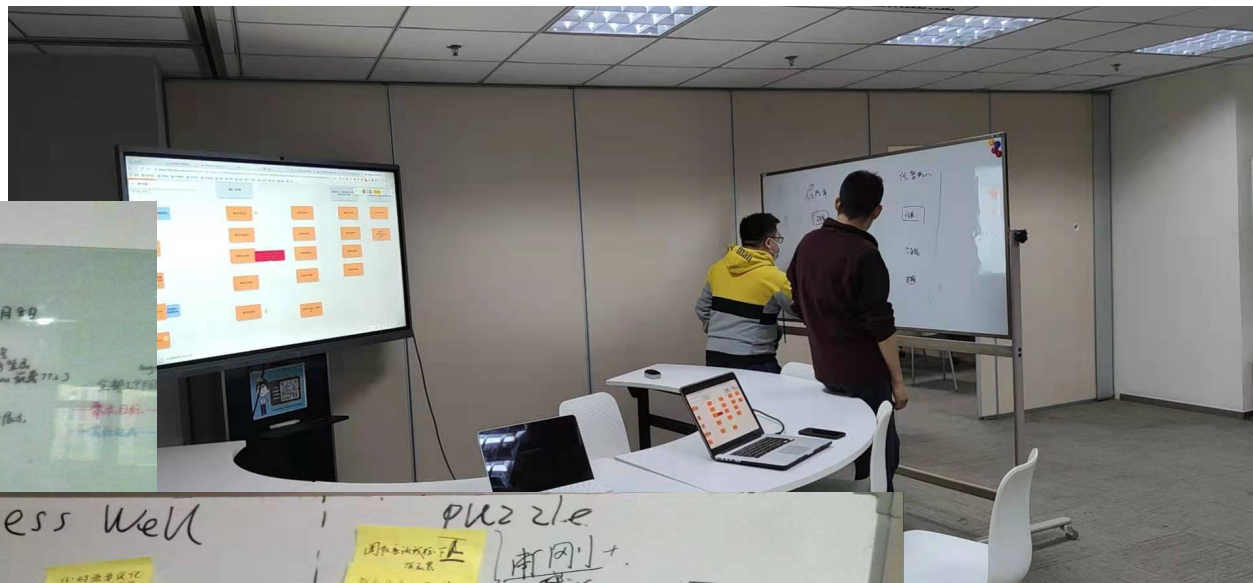
用DDD来设计API

API标准化在Qunar的落地

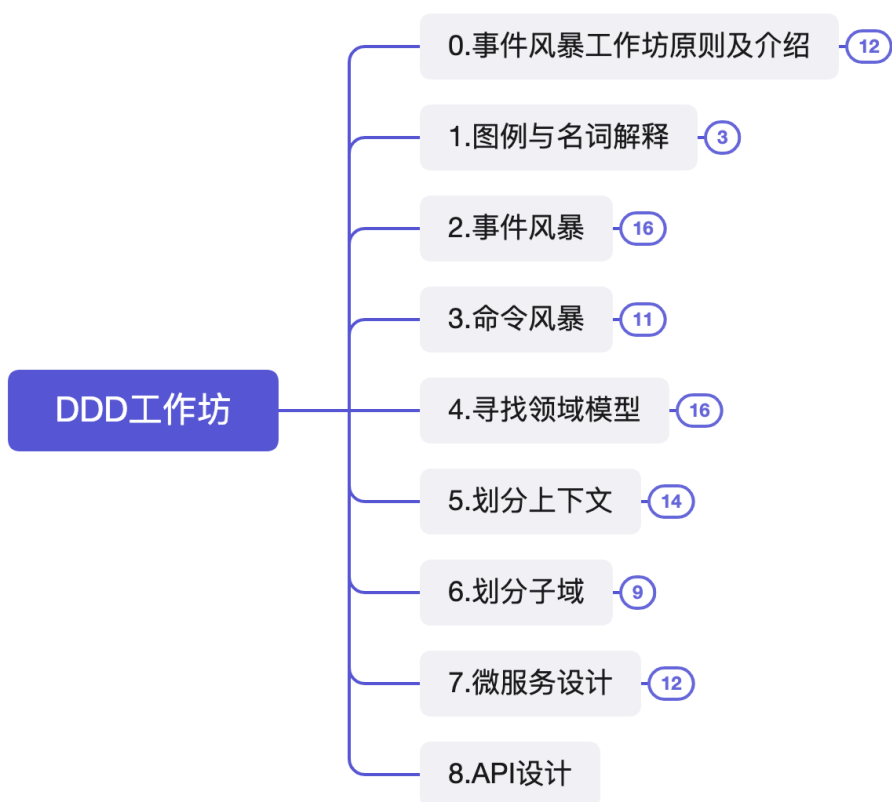
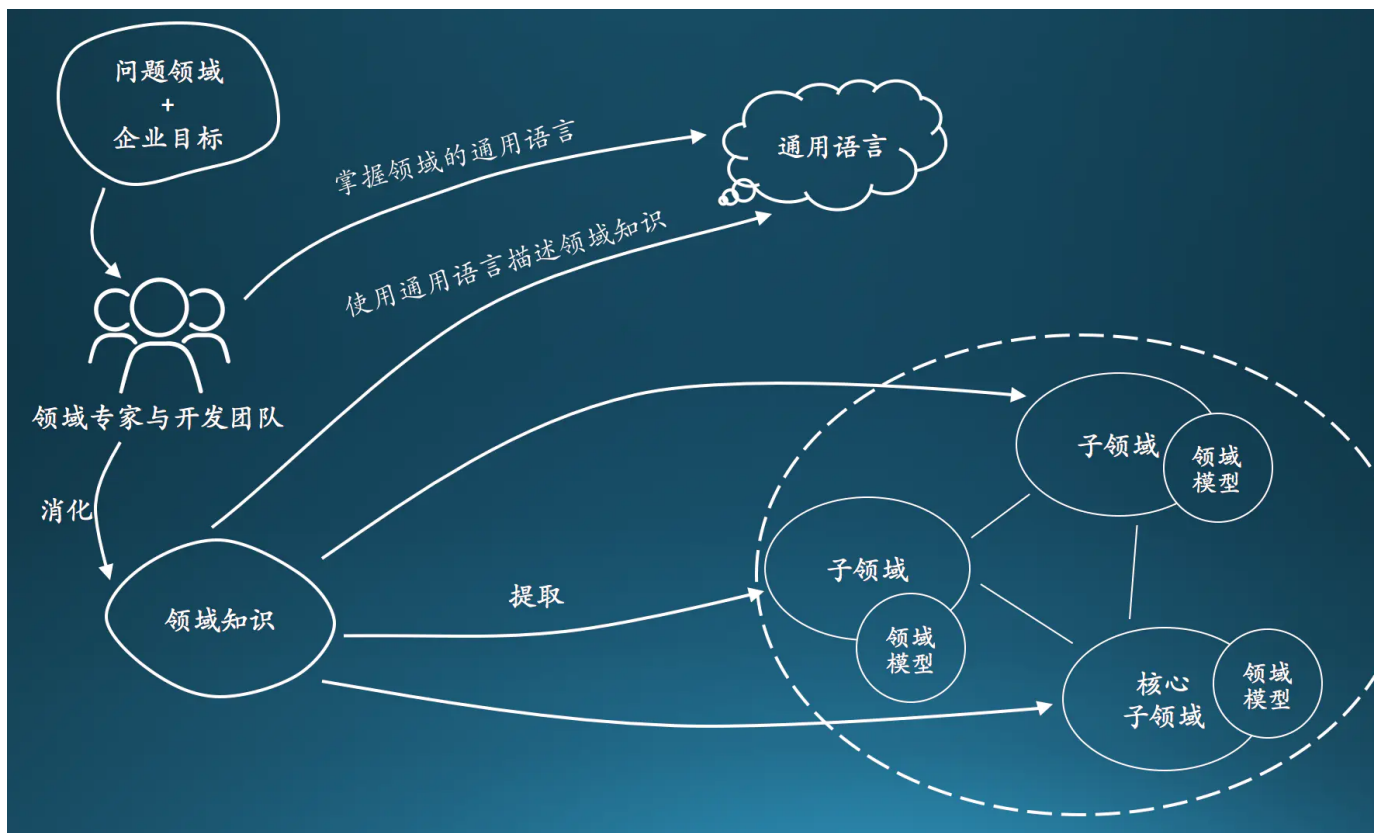


DDD 启示录

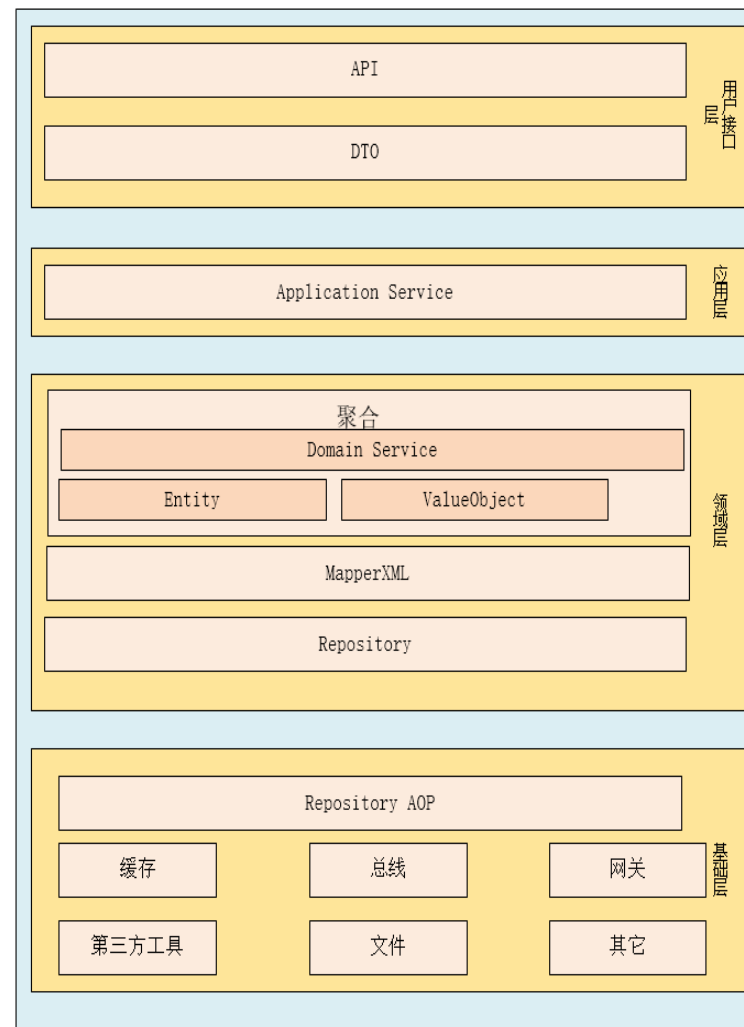
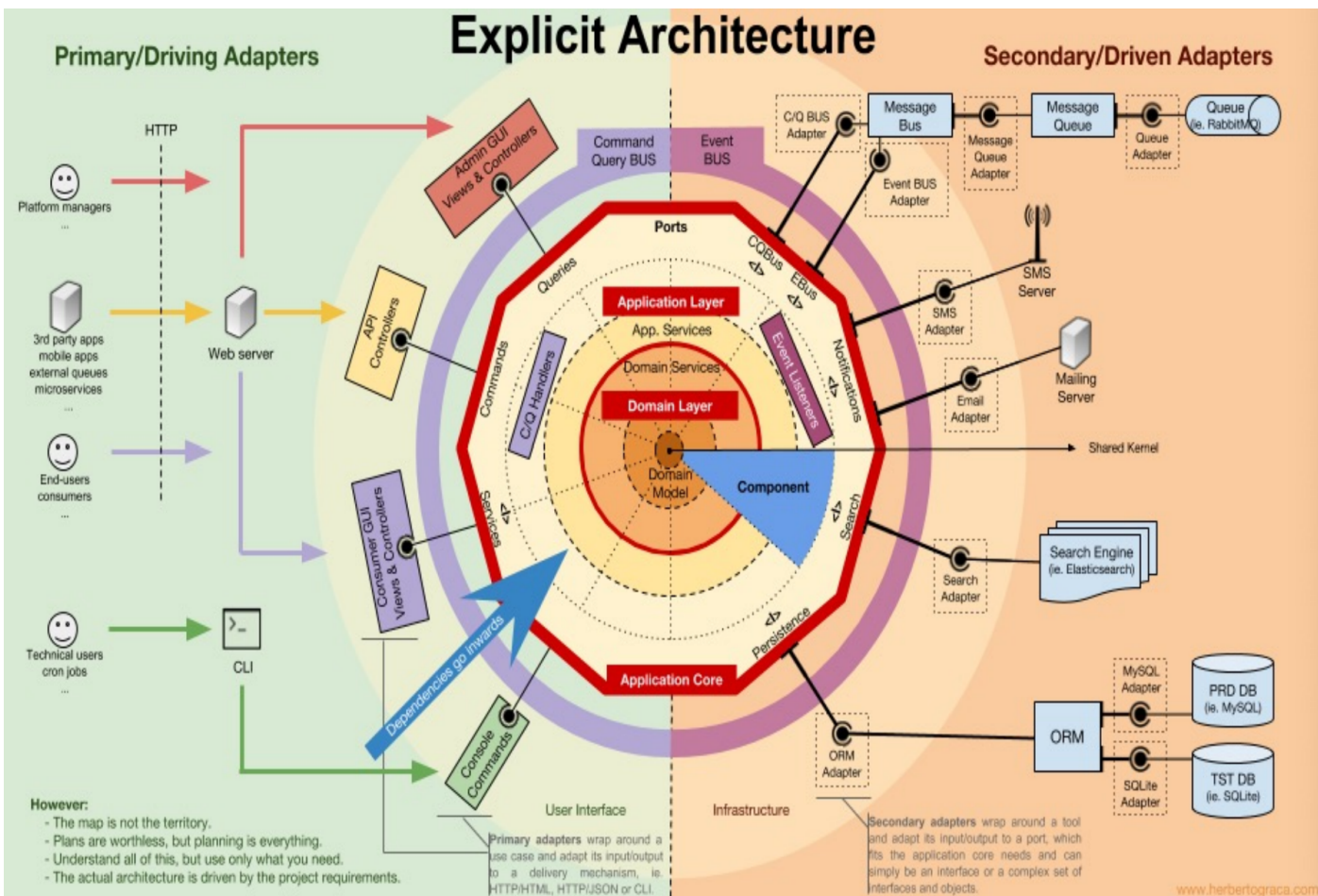
DDD- 启示录



DDD - 启示录



DDD - 启示录



领域模型边界表达 与可延续性

破窗理论

一个房子如果窗户破了，没有人去修补，隔不久，其它的窗户也会莫名其妙地被人打破；一面墙，如果出现一些涂鸦没有被清洗掉，很快的，墙上就布满了乱七八糟、不堪入目的东西；一个很干净的地方，人们不好意思丢垃圾，但是一旦地上有垃圾出现之后，人们就会毫不犹豫地抛撒垃圾。

软件开发~破窗效应

- 脏代码
- 测试
 - 没有测试
 - 混乱的测试代码
 - 难以测试
- 混乱的源代码管理
- 无序的部署方式



亚马逊：“超级API”公司



Jeff Bezos' Big Mandate

(circa 2002 — paraphrased)

1. All teams will henceforth expose their data and functionality through service interfaces.
2. Teams must communicate with each other through these service interfaces.
3. No other communication is allowed other than service interfaces over the network.
4. It doesn't matter what technology they use.
5. All service interfaces must be designed to be externalizable.
6. Anyone who doesn't do this will be fired.

<https://plus.google.com/+RipRawoo/posts/F5vbae5v92>

API标准化的
要求

- 从今天起，所有的团队都要以服务接口的方式，提供数据和各种功能
- 团队之间必须通过接口来通信。
- 不允许任何其他形式的互操作
 - ❑ 不允许直接链接
 - ❑ 不允许直接读其他团队的数据
 - ❑ 不允许共享内存
 - ❑ 不允许任何形式的后门
 - ❑ 唯一许可的通信方式，就是通过网络调用服务
- 具体的实现技术不做规定
- 所有的服务接口，必须从一开始就以可以公开作为设计导向。

- 不遵守上面规定，就开除

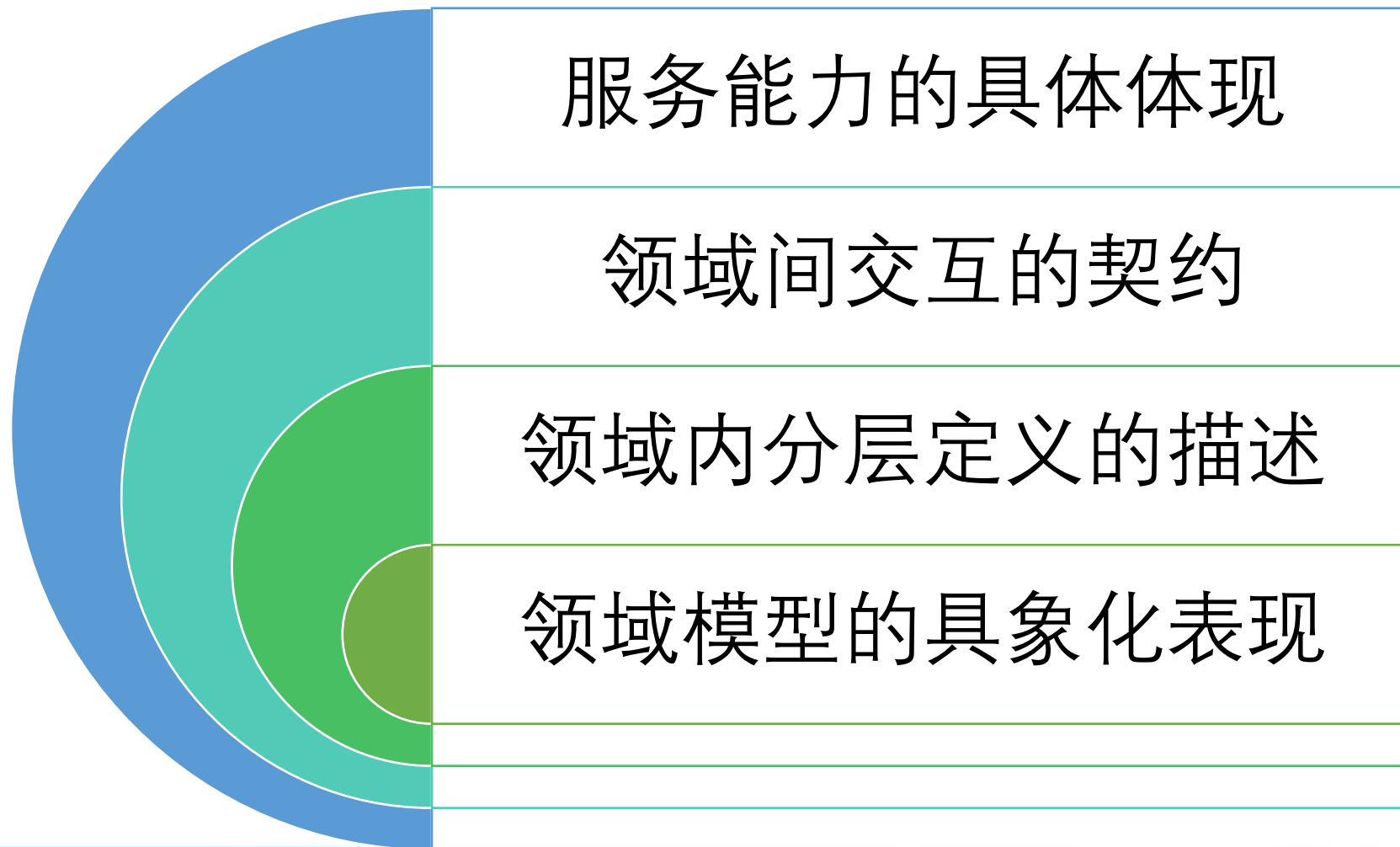
没有例外

API标准化的作用

尤其重要的是，要明确一个团队只在单一的限界上下文中工作，别给其他团队机会去修改你的源代码，从而引发意外，你的团队控制着源代码和数据库并定义了官方API，必须通过这些API才可以调用限界上下文，这是使用DDD所能带来的好处之一”

- <领域驱动设计精粹>

API是什么



软件分析和设计经历了三个阶段的演进

第一阶段

- 单体架构
- 面向过程，数据驱动

第二阶段

- 集中式架构
- 面向对象，分层设计

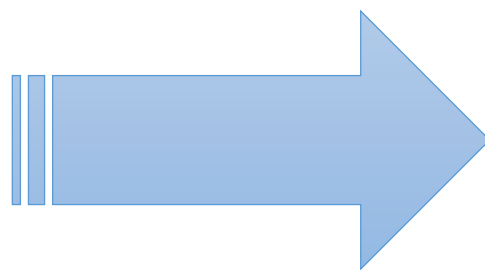
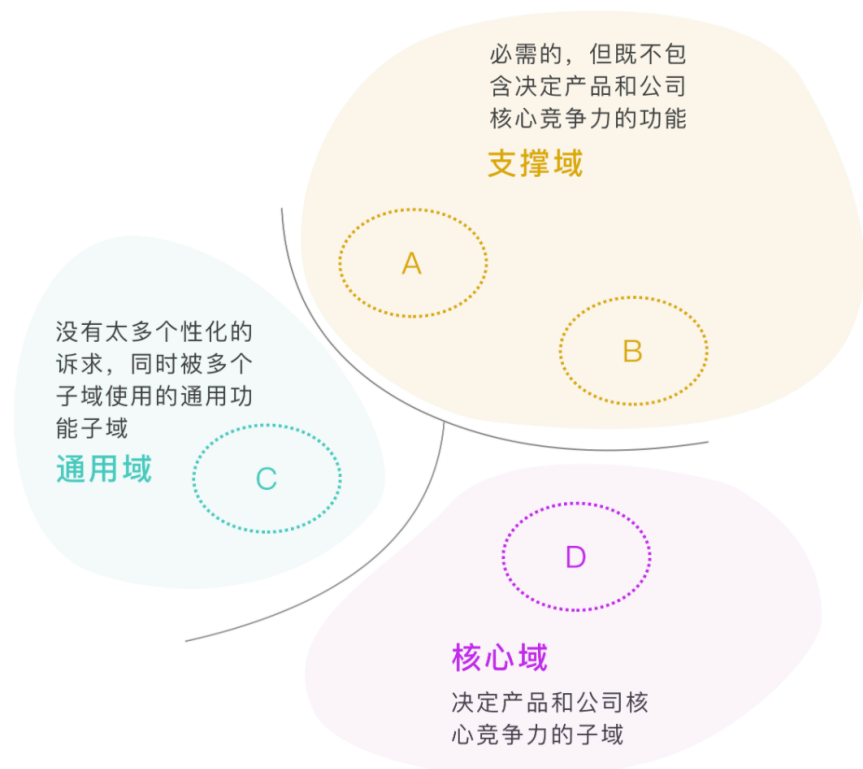
2.5阶段

- SOA与远程调用
- 通过传统SOA或远程调用，实现初级的业务隔离，充分发挥康威定律。（大多在这个阶段）

第三阶段

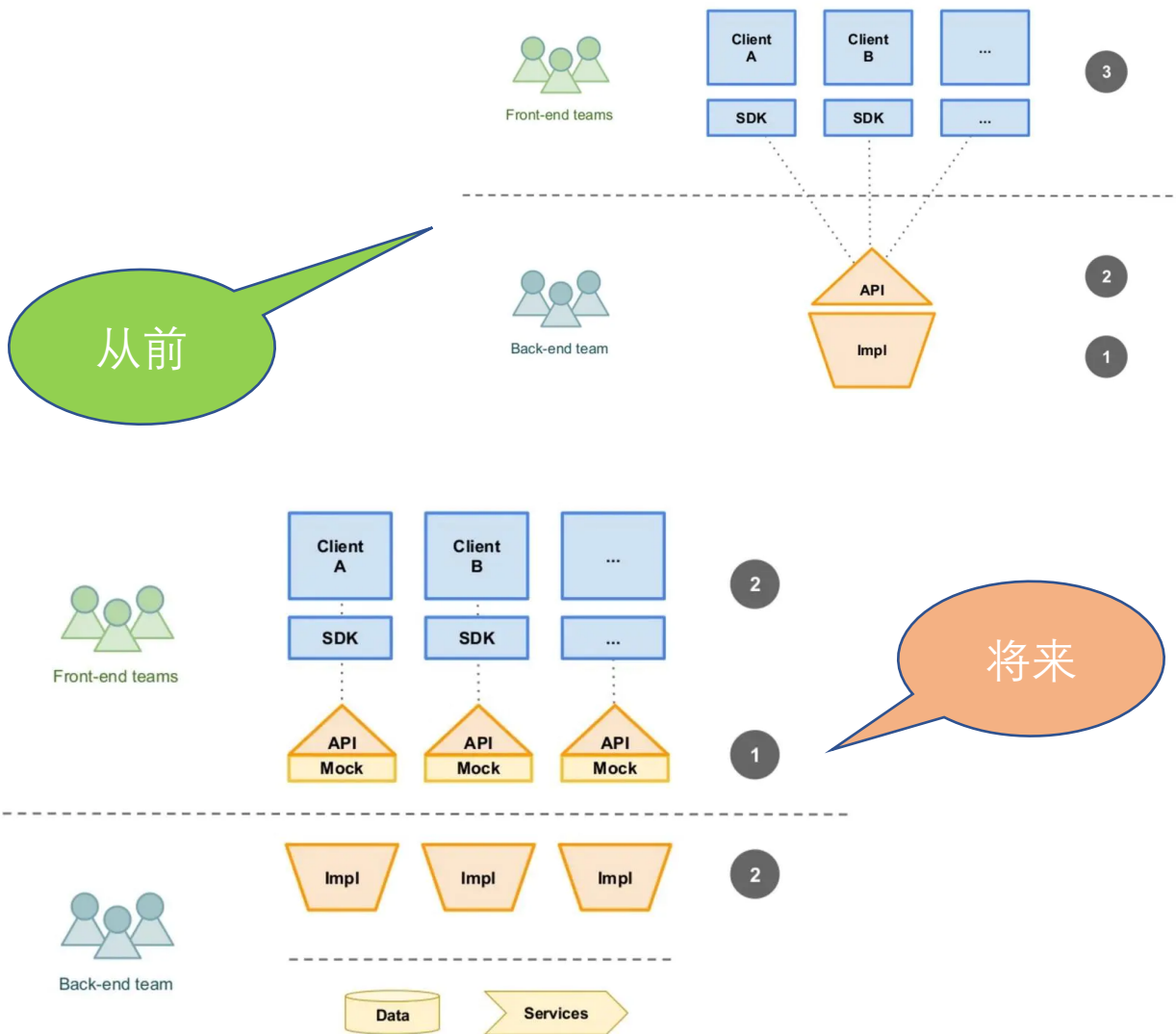
- 分布式架构
- 采用领域驱动设计的方法，使用微服务架构模式

领域模型边界表达与可延续性



用DDD来设计API

思潮：API-first



用DDD设计API – 领域接口化设计

经典案例

对于银行的 API 来说，账户就是一个领域对象（DDD 里的实体）。这次我们不再使用 CRUD 来为账户建模，而是为账户定义一组业务操作。以下是一系列写入操作：

- 开户 (Open) —— 新开一个账户。
- 销户 (Close) —— 注销一个已有的账户。
- 取出 (Debit) —— 从账户里扣掉一些钱。
- 存入 (Credit) —— 往账户里存入一些钱。

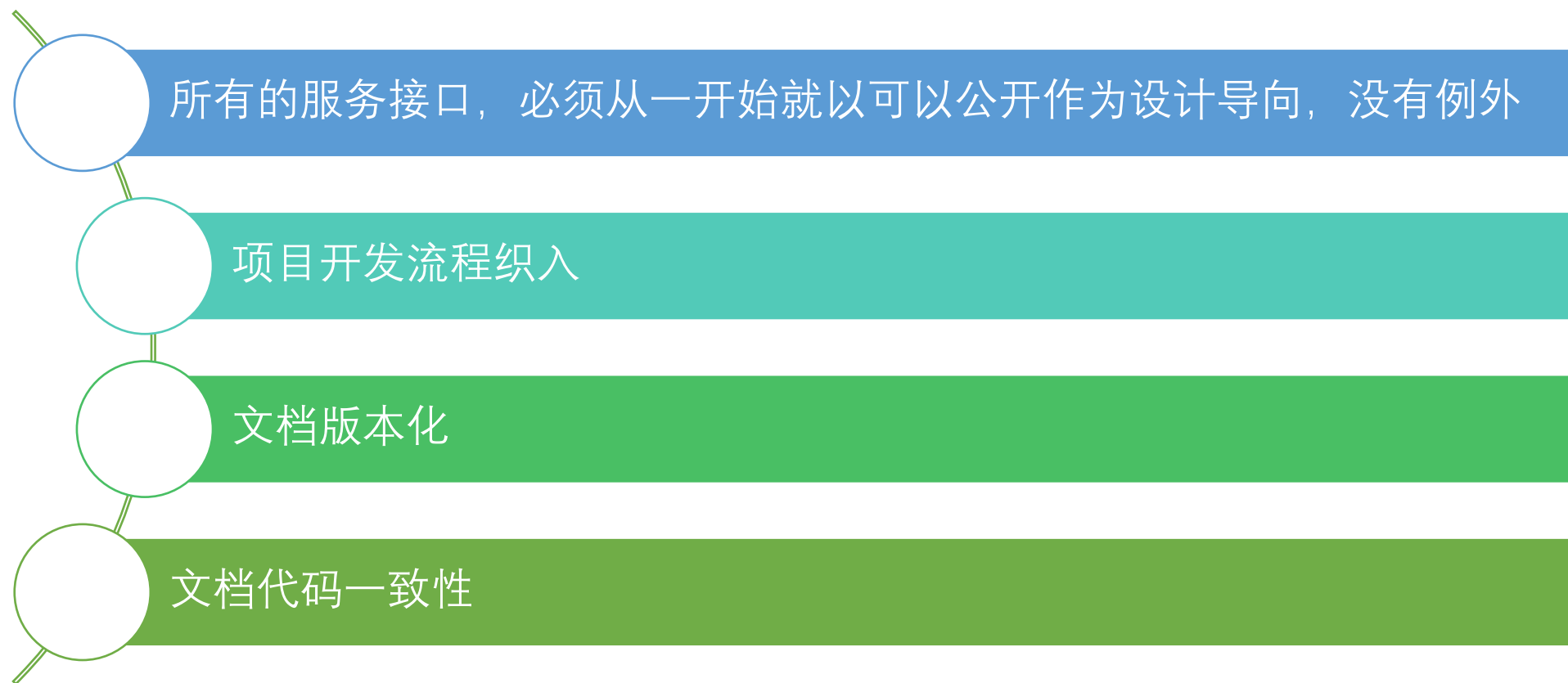
在定义好业务操作之后，就可以将它们与 REST API 映射起来：

- POST /account —— 新开一个账户。
- PUT /account/close —— 注销一个已有的账户。
- PUT /account/debit —— 从账户里扣掉一些钱。
- PUT /account/credit —— 往账户里存入一些钱。
- GET /account/ —— 通过账户 ID 加载相应的账户信息。
- GET /account/transactions —— 列出账户的交易历史。
- GET /accounts/query/customerId/ —— 列出指定客户的所有账户。

API在Qunar的落地

QDoc

落地 - 目标



效果



找不到? 那就搜搜吧~

搜索

领域分类

- 开放平台
- 机票TTS
- 用户中心
- 订单中心
- 会员中心
- 国内酒店

人事认证服务 / C端接口:
认证查询 | 认证服务

人事认证服务 / 服务端接口:
解绑 | 查询认证类型 | 手工认证审核 | 手工认证审核列表分页查询

机票搜索服务:
机票主流程list接口 | 机票主流程ota接口

编辑 备注 分配 更多 开始测试 提测Reject 更多工作流动作 创建BUG

5	后端	qdoc/f_poseidon_pcs	CR 复制	温馨提示: 请操作编辑选择JOB module: --	发prod
6	api文档	f_poseidon_pcs	/PersonnelCertificationServiceImpl.jdStudentReport	FD-36360	向京东上传学生信息
7	api文档	f_poseidon_pcs	/PersonnelCertificationServiceImpl.authorizationQuery	FD-36360	授权查询
8	文本	authorization_mapping.t 文件 school_city_mapping.t 文件 user_register_auth.t 文件 student.properties #向京东上报学生数据服务 get_id_aes_key=b0ea22ba13294274a14529adac22979c get_id_aes_id=1002 get_id_app_key=8679228637f892d088701126DD6688C4 get_id_app_secret=f414ba4afe374a8abffed52c4143050 common.properties #京东来源不发短信 StudentIdAutoAuth.send=false			
9	api文档	f_poseidon_pcs	/AuthServiceImpl.authQuery	FD-36360	认证查询dubbo接口
10	api文档	f_poseidon_pcs	/AuthServiceImpl.authConfim	FD-36360	认证dubbo接口

权益核心权益操作服务

输入文档关键词搜索

权益使用接口

POST /PrivilegeActionServiceImpl.usePrivilege

请求数据类型 application/x-www-form-urlencoded

接口描述
desc:使用权益 (到待核销状态, 需要等待业务线核销之后, 才会扭转到已使用状态) 作者:[fuwei.liu]

请求参数

参数名称	参数说明	请求类型	是否必须	数据类型
bizNo	scene:业务线NO	query	true	string
acquiredProductNo	scene:领取到的商品NO	query	true	string
bizProductNo	scene:业务线商品NO	query	true	string
privilegeCode	scene:权益code	query	true	string
extInfo	scene:拓展数据	query	true	com.qunar.ucenter.privilege.api.model.Ext
extMap			true	object

Diff

POST /PrivilegeActionServiceImpl.usePrivilege

请求数据类型 application/x-www-form-urlencoded 响应数据类型 ["/*"]

接口描述
desc:使用权益 (到待核销状态, 需要等待业务线核销之后, 才会扭转到已使用状态) 作者:[fuwei.liu]

请求参数

参数名称	参数说明	请求类型	是否必须	数据类型
bizNo	scene:业务线NO	query	true	string
acquiredProductNo	scene:领取到的商品NO	query	true	string
bizProductNo	scene:业务线商品NO	query	true	string
bizQualificationId	scene:资格id	query	true	string
privilegeCode	scene:权益code	query	true	string

Apache License 2.0 | Copyright © 2019-Knife4j

文档化 - 路线图



规范定义 - 标准之争

1. 修订历史
2. 接口描述
3. 服务接入

基本信息

服务信息

4. 请求和返回参数

请求参数

返回参数

5. 成功和异常示例

成功示例

请求参数

返回参数

异常示例

请求参数

返回参数

6. 状态码

错误码

业务码

一般的接口定义规范



OPENAPI
INITIATIVE



JavaDoc

- 适用于Java
- 一种文档过注视图方案

Wiki/Word

- 大多数在用
- 合同、协议使用

OAS
(OpenAPI-
Specification)

- 标准的
- 语言无关的
- RESTful API 接口规范

工具审视

Swagger

- 业界公认
- 可集成度高

eolinker

- 商业应用
- 有一定知名度

Postman

- 开发常用工具
- Workbanch方向发展

YAPI

- Qunar开源项目
- Apifox

其他

- smart-doc
- showdoc
- Orion-API-Manager



图览界面

表头自定义
表头自定义
表头自定义

自定义

名称	表头自定义	表头自定义	表头自定义	表头自定义
自定义	自定义	自定义	自定义	自定义
自定义	自定义	自定义	自定义	自定义
自定义	自定义	自定义	自定义	自定义

利用swagger2构建的API文档

用restful风格写接口

book-controller : Book Controller Show/Hide | List Operations | Expand Operations

employee-controller : Employee Controller Show/Hide | List Operations | Expand Operations

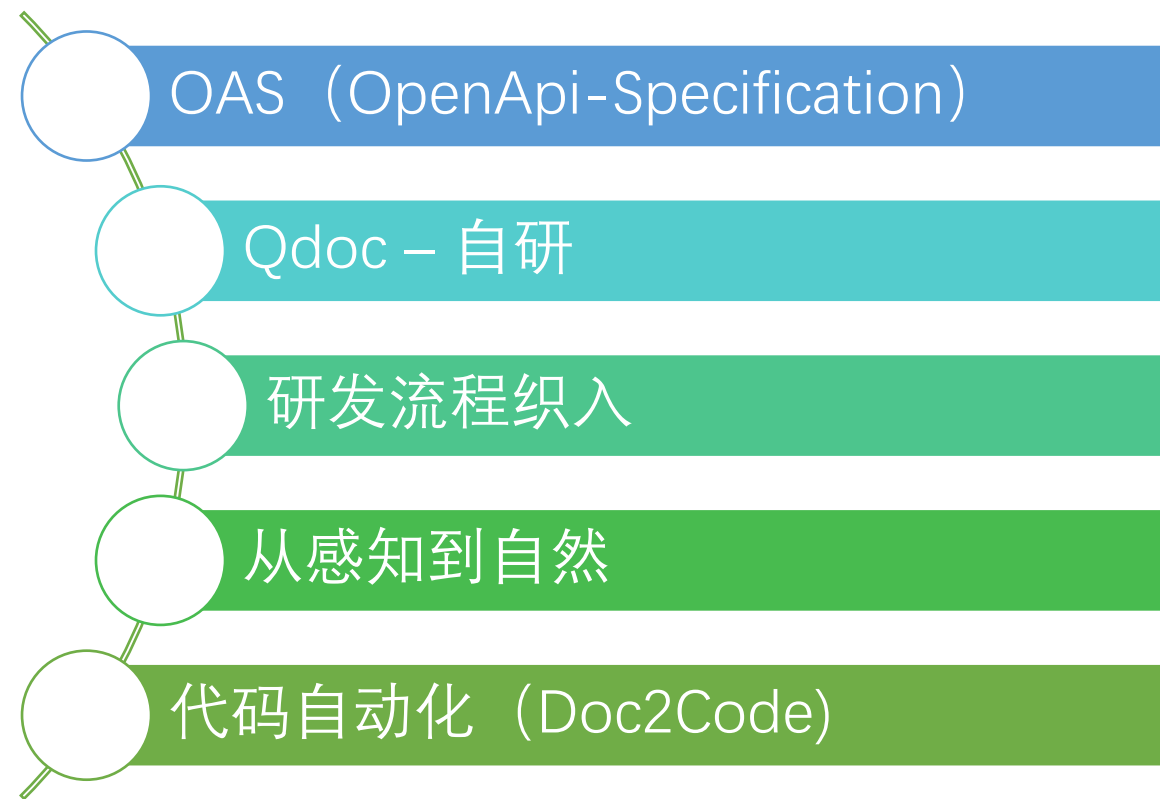
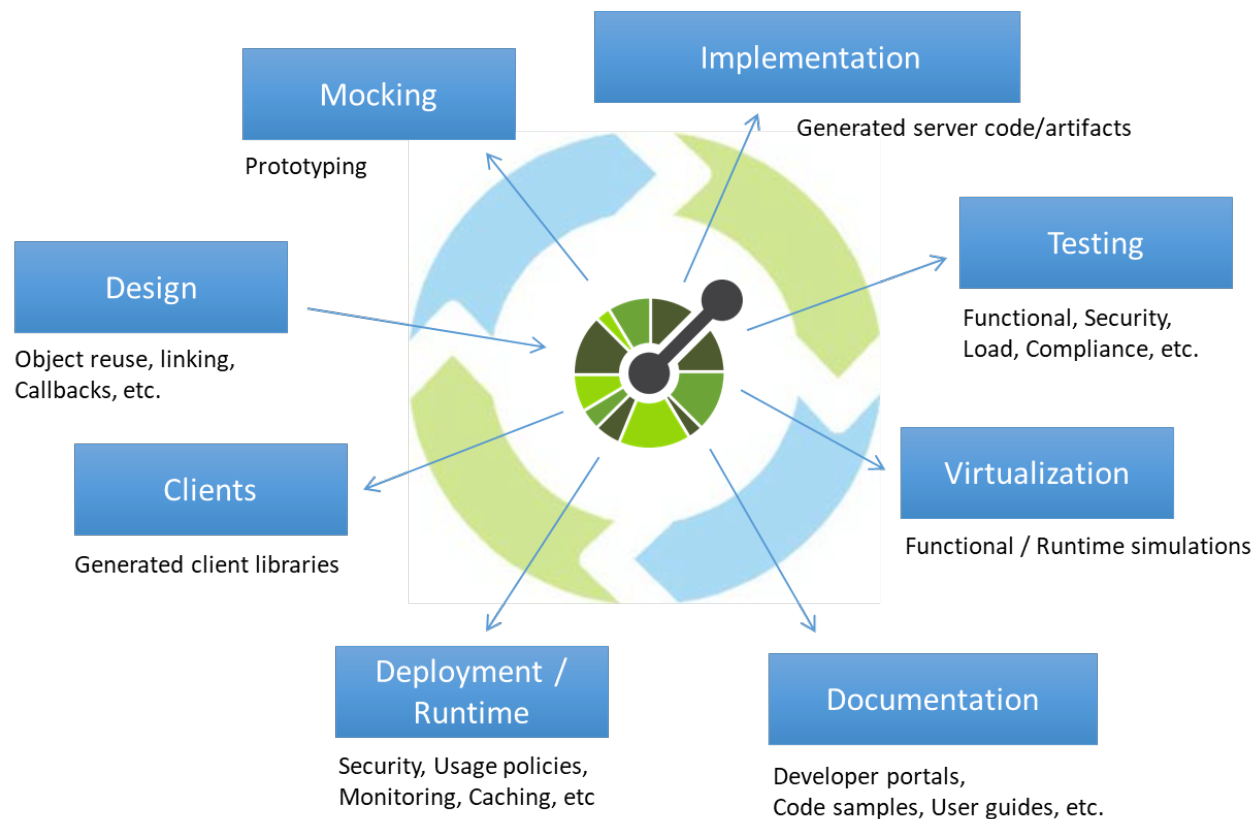
POST	/emp/employee	新增一个用户
GET	/emp/employee/find/{username}	查询用户信息
GET	/emp/employee/grade	findEmployeeAndGrade
GET	/emp/employee/pageSort	查询用户信息
GET	/emp/employee/sort	查询全部用户
DELETE	/emp/employee/{id}	删除用户
PUT	/emp/employee/{id}	修改用户信息

grade-controller : Grade Controller

Show/Hide | List Operations | Expand Operations

[BASE URL: / , API VERSION: 1.0]

Qunar工具选择 - QDoc



流程融入 - 代码即文档

QDoc API文档规范

由 李征 创建, 最终由 乔路 修改于 2021-04-25

- 目标
- 基本原则
- 术语
 - 领域 (domain/group)
 - 应用 (appcode)
 - 服务 (service)
 - 接口 (interface)
 - 参数 (parameter)
 - 类型 (type)
- 文档字段描述
 - 领域 (@QDomainDoc)
 - 应用 (@QAppcodeDoc)
 - 服务 (@QServiceDoc)
 - 接口 (@QInterfaceDoc) 这里面存在参数注解和返回值是否可以删除
 - 参数 (@QParamDocs)
 - 参数 (@QParamDoc)
 - Model参数 (@QParamModel)
 - Model属性的注解 (@QParamModelProperty)
 - 类型 (@QTypeDoc) (待定) 等同于@RequestParam。目前未实现, 后续支持
 - 返回值状态码的描述 (@QResponses)
 - 返回值状态码的描述 (@QResponse)
 - 返回值数据描述说明
 - 返回值数据 (@QResponseDocs)
 - 返回值数据 (@QResponseDoc)
 - 异常 (@QExceptionDoc) (待定) 等同于一个scheme
 - 扩展 (@QExtensionDoc) (待定) 等同于一个scheme
- 文档编写指南
- 工具支持
- 附录

```
@QInterfaceDoc(  
    type = "HTTP",  
    define = "手工认证审核列表分页查询",  
    desc = "后端手工认证列表分页查询",  
    notice = "内网使用",  
    url = "list",  
    authors = "guanqwen.qw"  
)  
  
@QResponses({  
    @QResponse(code = 0, message = "成功"),  
    @QResponse(code = -1, message = "失败")  
})  
  
@QResponseDocs({  
    @QResponseDoc(bindValueName = "authReviewListShowV0", description = "返回主参数", bindValueType = "AuthReviewListShowV0",  
        @QProperty(type = "Boolean", name = "ret", desc = "是否成功标识"),  
        @QProperty(type = "Integer", name = "errorCode", desc = "错误码标识"),  
        @QProperty(type = "com.guang.flight.pcs.vo.auth.AuthReviewListShowV0.DataBody", name = "data")  
    })  
})  
  
@RequestMapping(value = "/list")  
@ResponseBody  
@MonitorAnnotation(name = "show_auth_review_list", monitorType = MonitorType.METER)  
public AuthReviewListShowV0 showAuthReviewList(AuthenticateReviewDTO authenticateReviewDTO,  
        @RequestParam(value = "limit") Integer limit,  
        @RequestParam(value = "pageIndex") Integer pageIndex) {
```

流程融入 - 发布流程集成



【f_poseidon_pcs线上发布成功】

需求: 其他

详情: <http://portal.corp.qunar.com/apptask/console.html?id=5101041>

发布人: zheng.li

请注意:

1. 本次发布有机器因已部署过相同的版本, 跳过部署

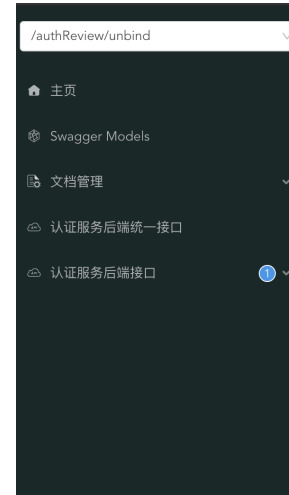


【qdoc执行成功】

工程: <git@gitlab.corp.qunar.com:flightuser/poseidon-pcs.git>

tag: r-210507-171158-zheng.li

job链接: http://qci.corp.qunar.com/job/flightuser_poseidon-pcs_qdoc/247/



认证服务后端统一接口	
简介	认证服务后端统一接口认证服务后端审核列表查询, 审核操作, 枚举查询, 解绑操作解绑操作需要加入ip白名单才能调用
作者	
版本	
host	
basePath	/
服务Url	
分组名称	%2FauthReview%2Funbind
分组Url	
分组location	http://openapi.corp.qunar.com/knife4j/openapi/apidoc?appid=f_poseidon_pcs&path=%252FauthReview%252Funbind&ver=master
接口统计信息	POST 1

```
[INFO] open url : http://openapi.corp.qunar.com/doc/index.html?appid=f_poseidon_pcs&ver=master&path=%2FauthReview%2Freasons#/home
[INFO] import one start : {"openapi":"3.0.1","info":{"title":"认证服务后端统一接口","description":"认证服务后端统一接口认证服务后端审核列表查询, 审核操作, 枚举查询, 解绑操作需要加入ip白名单才能调用"},"servers":[{"description":"beta请求地址"}, {"description":"prod请求地址"}], "tags":[{"name":"认证服务后端统一接口", "description":"认证服务后端统一接口认证服务后端操作需要加入ip白名单才能调用"}], "paths":{"/authReview/authTypeList":{"post":{"tags":["认证服务后端接口"],"summary":"查询认证类型","description":"desc:查询认证类型注[guangwen.qu]","operationId":"f_poseidon_pcsAuthenticateReviewControllergetAuthTypeList_post","parameters":[],"responses":{"default":{"description":"default","application/json;charset=UTF-8":{"schema":{"$ref":"#/components/schemas/object","exampleSetFlag":true}}}}},"components":{"schemas":{"AuthConfigVO":{"title":"AuthConfigVO","type":"object","description":"返回主参数","deprecated":false,"exampleSetFlag":true},"AuthConfigVO":{"$ref":"#/components/schemas/AuthConfigVO","exampleSetFlag":false},"description":"default","deprecated":false,"exampleSetFlag":true}}}}
[INFO] open url : http://openapi.corp.qunar.com/doc/index.html?appid=f_poseidon_pcs&ver=master&path=%2FauthReview%2FauthTypeList#/home
[INFO] import one start : {"openapi":"3.0.1","info":{"title":"认证服务后端统一接口","description":"认证服务后端统一接口认证服务后端审核列表查询, 审核操作, 枚举查询, 解绑操作需要加入ip白名单才能调用"},"servers":[{"description":"beta请求地址"}, {"description":"prod请求地址"}], "tags":[{"name":"认证服务后端统一接口", "description":"认证服务后端统一接口认证服务后端操作需要加入ip白名单才能调用"}], "paths":{"/authReview/unbind":{"post":{"tags":["认证服务后端接口"],"summary":"解绑","description":"desc:解绑注意事项:内网使用作者:[guangwen.qu]","operationId":"f_poseidon_pcsAuthenticateReviewControllerrunbind_post","parameters":[{"name":"userName","in":"query","description":"用户名","required":true,"deprecated":true,"schema":{"type":"string","exampleSetFlag":false},"example":"","name":"authType","in":"query","description":"认证类型","required":true,"deprecated":true,"schema":{"type":"long","exampleSetFlag":false},"example":""}], "responses":{"0":{"description":"成功"},"default":{"description":"default","application/json;charset=UTF-8":{"schema":{"$ref":"#/components/schemas/object","exampleSetFlag":true}}}}},"components":{"schemas":{"PcsResult":{"title":"PcsResult","type":"object","description":"返回主参数","deprecated":false,"exampleSetFlag":true},"object":{"title":"object","required":["PcsResult":{"$ref":"#/components/schemas/PcsResult","exampleSetFlag":false},"description":"default","deprecated":false,"exampleSetFlag":true}}}}
[INFO] open url : http://openapi.corp.qunar.com/doc/index.html?appid=f_poseidon_pcs&ver=master&path=%2FauthReview%2Funbind#/home
[INFO] qdoc build upload to yapi.
```

流程融入 – PMO(JIRA)集成

取消编辑 获取回滚步骤 + 新增 保存 请在此模块编辑

回滚步骤
相关测试结果:

- 1、自
- 2、Q
- 3、cl

api文档

描述
点击添加描述信息

后端
qzz
qzz+ 后端

编辑 备注 分配 更多 开始测试 提测Reject 更多工作流动作 创建BUG

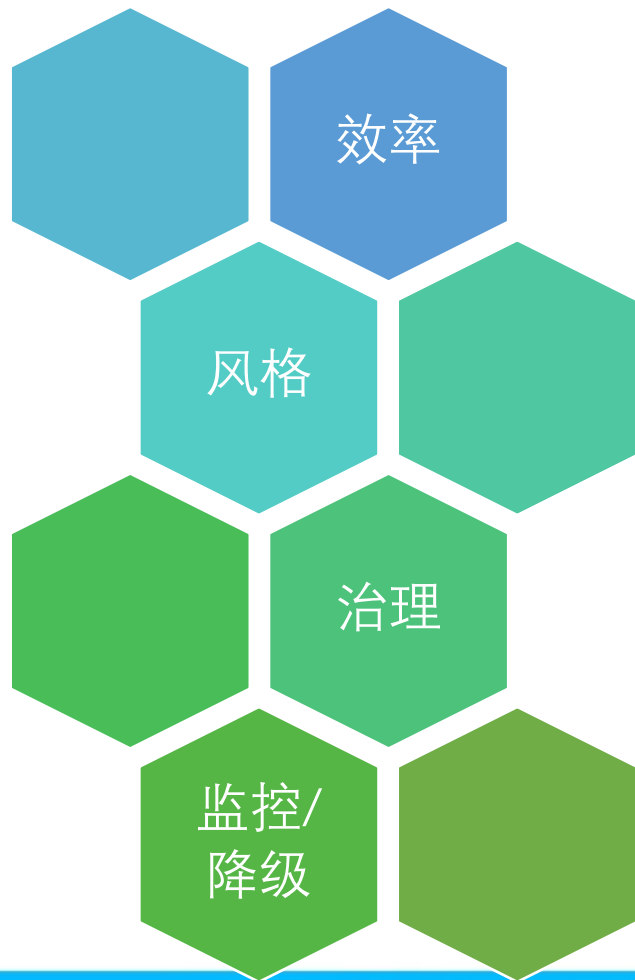
开发计划 项目质量 需求 提测&发布 项目过程核查 项目计划调整

发布步骤

编辑 奥丁桌面 获取模版 Qconfig自动化 TARS自动化 一键群聊 发布周知 Checklist Case Bug

1	后端	flightuser/poseidon-pcs-api	FD-36360 CR 复制	portal_105945 module: --	发beta rebuild i 发prod i	beta
2	后端	flightuser/poseidon-pcs	FD-36360 CR 复制	portal_105853 module: --	发beta rebuild i 发prod i	beta
3	后端	flightuser/poseidon-growth-api	FD-36360 CR 复制	portal_2188 module: --	发beta rebuild i 发prod i	beta
4	后端	flightuser/bear	FD-36360 CR 复制	portal_2191 module: --	发beta rebuild i 发prod i	beta
5	后端	qdoc/f_poseidon_pcs	FD-36360 CR 复制	温馨提示: 请操作编辑选择job module: --	发beta rebuild i 发prod i	
6	api文档	f_poseidon_pcs	/PersonnelCertificationServiceImpl.jdStudentReport	FD-36360	向京东上传学生信息	
7	api文档	f_poseidon_pcs	/PersonnelCertificationServiceImpl.authorizationQuery	FD-36360	授权查询	

代码自动化 – Doc2Code



qdoc / f_poseidon_pcs

Search in this project

- API version:
 - Build date: 2021-03-31T20:27:35.453+08:00[Asia/Shanghai]

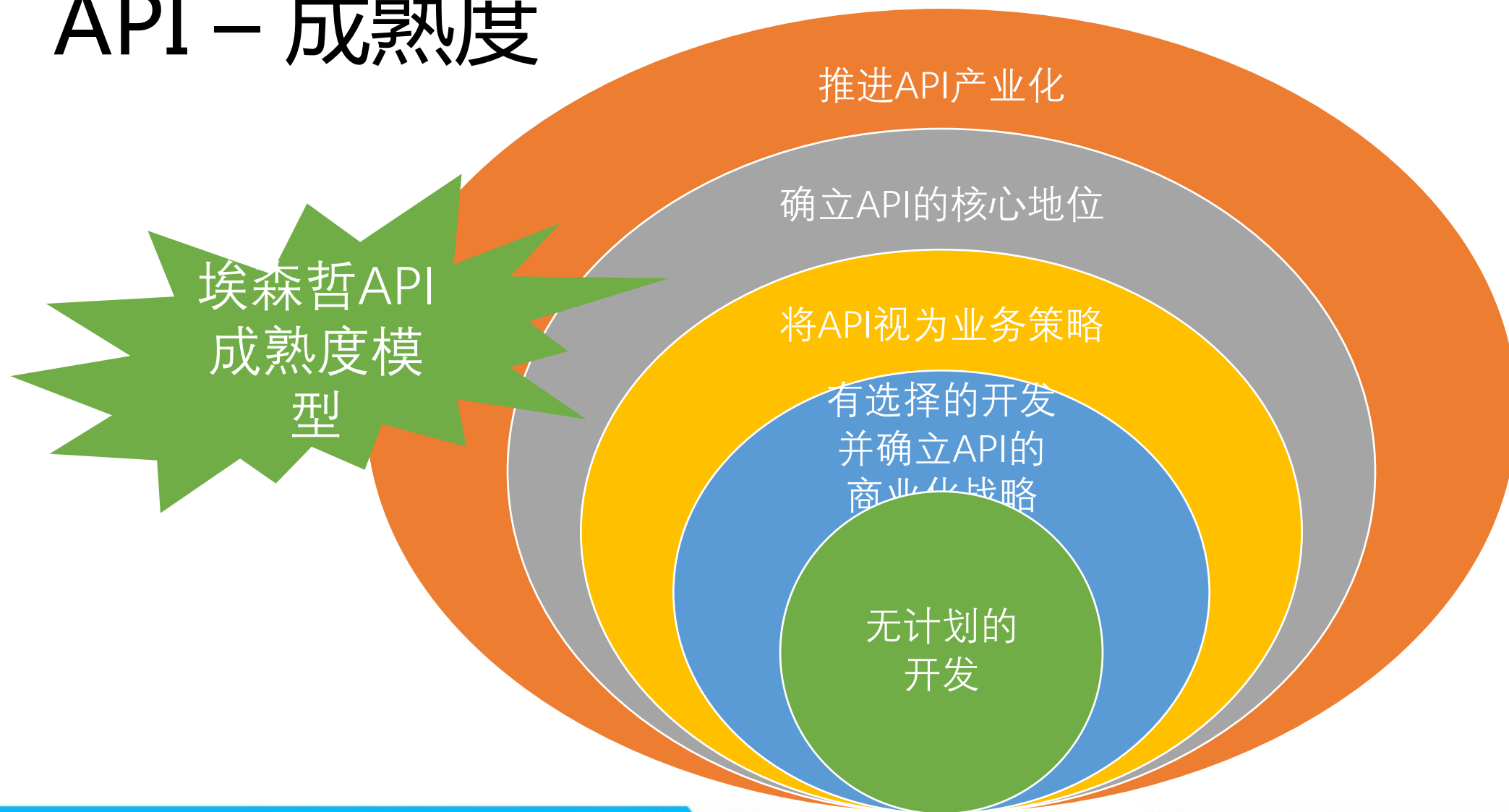
```
<dependency>  
<groupId>com.qunar.qdoc</groupId>  
<artifactId>f_poseidon_pcs</artifactId>  
<version>0.0.0</version>  
<scope>compile</scope>  
</dependency>
```

```
import qunar.qdoc.authReview_unbind.model.*;  
import qunar.qdoc.authReview_unbind.api.DefaultApi;  
  
import java.util.*;  
  
public class DefaultApiExample {  
  
    public static void main(String[] args) {  
        DefaultApi apiInstance = new DefaultApi();  
        String userName = "userName_example"; // String | 用户名  
        Long authType = 56L; // Long | 认证类型  
  
        //beta服务域名  
        final String betaBasePath = "";  
        //prod服务域名  
        final String prodBasePath = "";  
  
        //设置api服务域名  
        apiInstance.getApiClient().setBasePath(prodBasePath);  
  
        try {  
            Object result = apiInstance.fPoseidonPcsAuthenticateReviewControllerunbindPost(userName, authType, 5000L);  
            System.out.println(result);  
        } catch (Exception e) {  
            System.err.println("Exception when calling DefaultApi#fPoseidonPcsAuthenticateReviewControllerunbindPost");  
            e.printStackTrace();  
        }  
    }  
}
```

落地

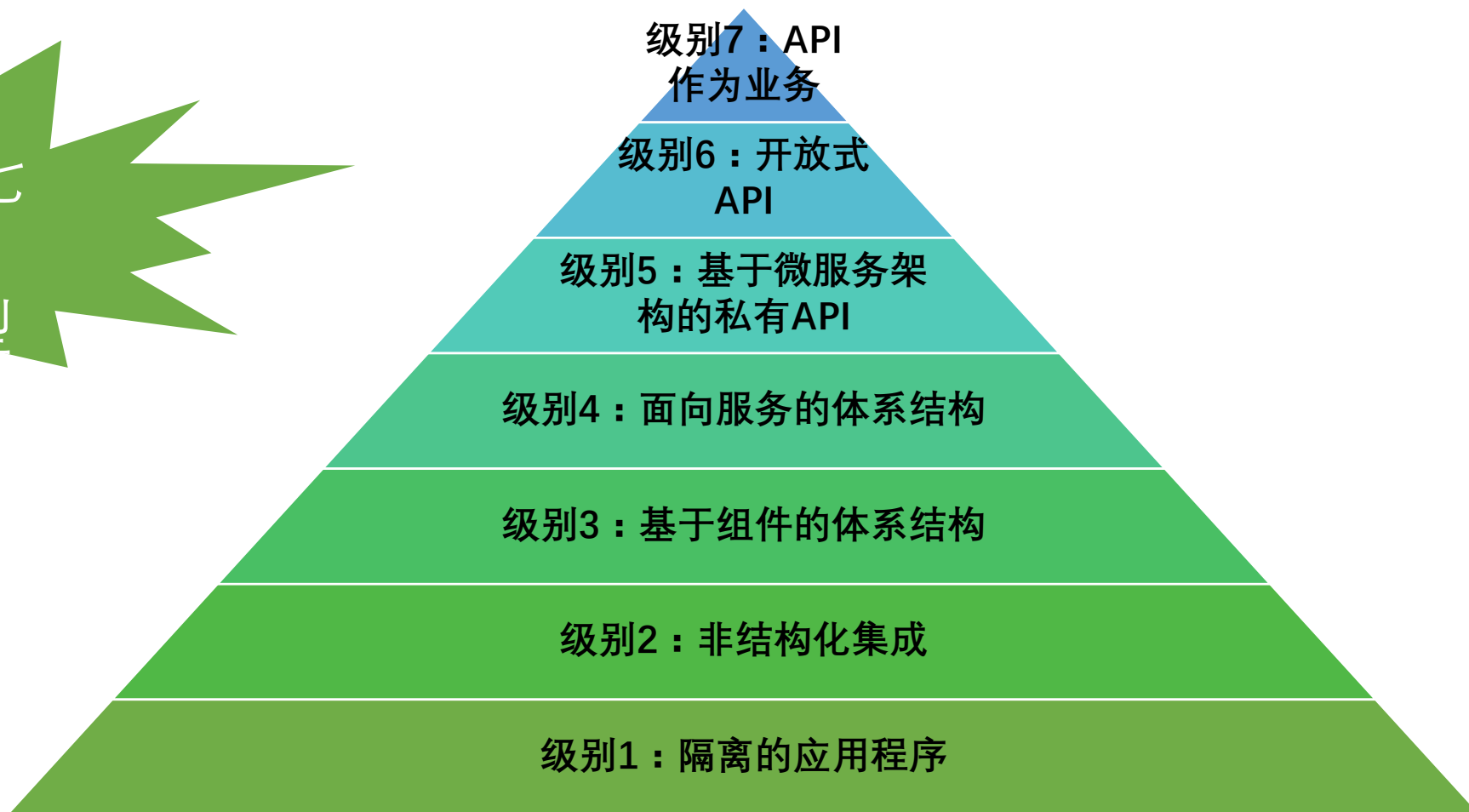


API – 成熟度



API – 成熟度

API架构的七
级
成熟度模型



终话

贝索斯：API宣言



Jeff Bezos' Big Mandate

(circa 2002 — paraphrased)

1. All teams will henceforth expose their data and functionality through service interfaces.
2. Teams must communicate with each other through these service interfaces.
3. No other communication is allowed other than service interfaces over the network.
4. It doesn't matter what technology they use.
5. All service interfaces must be designed to be **externalizable**.
6. Anyone who doesn't do this will be fired.

<https://plus.google.com/+BipRawor/posts/EV9ouesv9f0X>

- 从今天起，所有的团队都要以服务接口的方式，提供数据和各种功能
- 团队之间必须通过接口来通信。
- 不允许任何其他形式的互操作
 - ❑ 不允许直接链接
 - ❑ 不允许直接读其他团队的数据
 - ❑ 不允许共享内存
 - ❑ 不允许任何形式的后门
 - ❑ 唯一许可的通信方式，就是通过网络调用服务
- 具体的实现技术不做规定
- 所有的服务接口，必须从一开始就以可以公开作为设计导向。

没有例外

- 不遵守上面规定，就开除

SACC

THANKS