GIAC | 全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

基于BPMN流程引擎驱动的前端研发平台

姜天意
阿里巴巴-前端技术专家

Topic

- 前端工程的变迁及对研发流程的诉求
- 基于BPMN流程引擎 - DSL化 / 可编排的前端研发平台技术介绍
- 基础服务BAAS化与前端工程架构分层
- 研发流程管理带来的价值

没有CDN的时代，开发者通过直接开发代码上传到ftp进行部署

工作流：

要有工程

刀耕火种

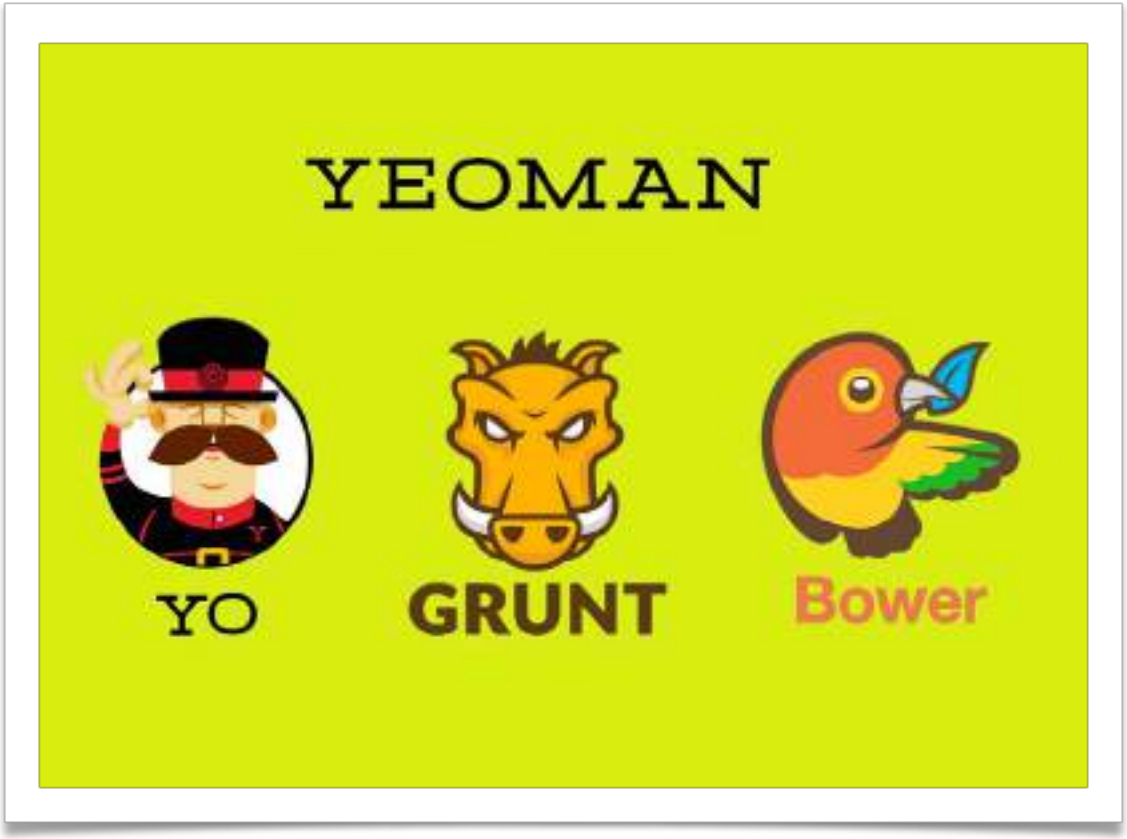
Apache Ant

是一个将软件编译、测试、部署等步骤联系在一起进行自动化的工具，大多用于Java环境中的软件开发。

```
<?xml version="1.0" standalone="yes"?>
<project default="build" name="refund">
    <target name="build">
        <echo>Hello world!</echo>
        <concat destfile="a_b.js">
            <path path="a.js"></path>
            <path path="b.js"></path>
        </concat>
    </target>
</project>
```

工作流：

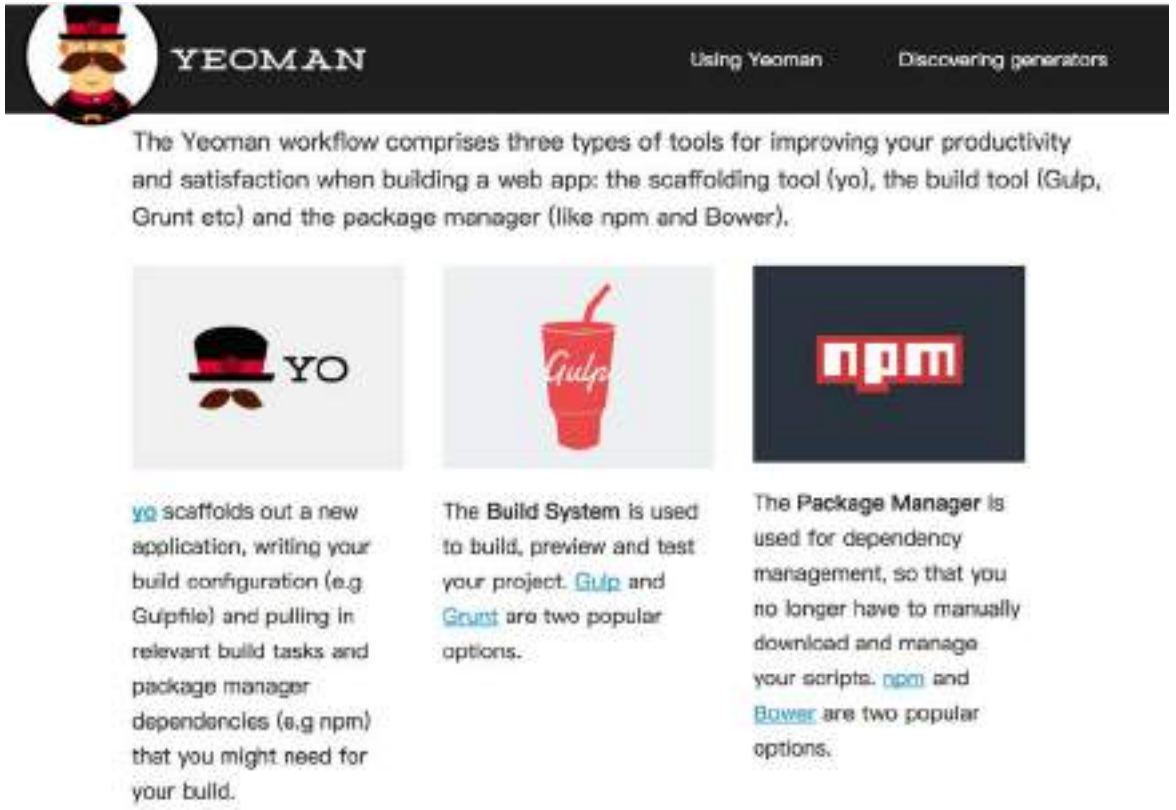
新时代



工作流：

包管理	脚手架	开发	合并	压缩	打包	部署

yeoman的进化



工作流：

包管理	脚手架	开发	合并	压缩	构建	部署

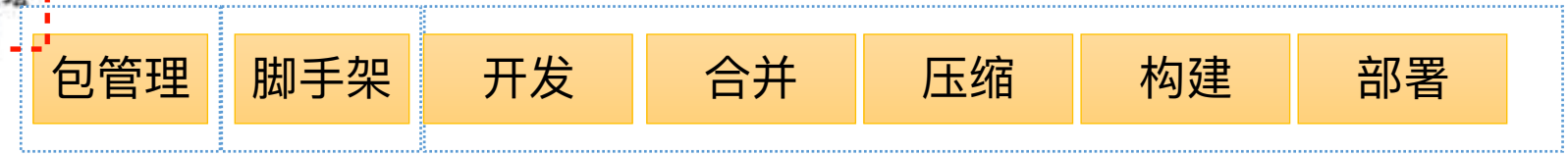
Baidu 百度

让前端开发更高效

berg

D2前端技术论坛
2012.7.6, 杭州

工作流：



线下工作流的百花齐放

PARCEL
Blazing fast, zero configuration web application bundler

F.I.S

webpack
MODULE BUNDLER

BABEL

browserify

jspm.io

rollup.js v0.25.8
the next-generation JavaScript module bundler

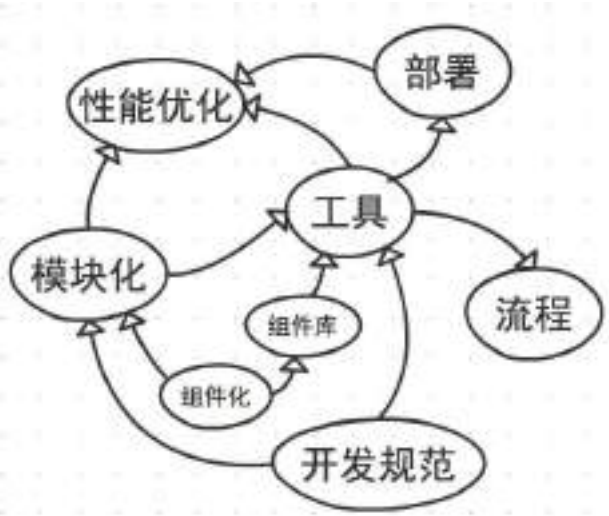
工作流：

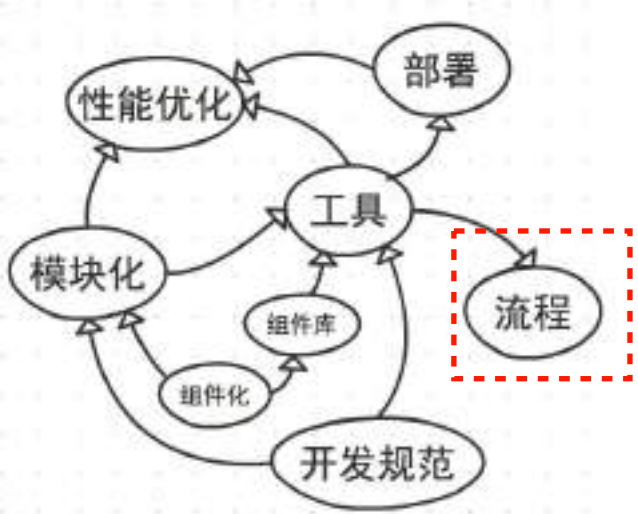
包管理	脚手架	开发	编译	后处理	合并	压缩	打包	部署

包管理	脚手架	开发	合并	压缩	构建	部署

够了吗?

- 静态代码检查
- 线上编译与构建
- 自动化测试
- 灰度发布
- 自动雪碧
- 前端性能和错误监控
- 源站版本管理与组件管理





准备阶段

线下开发阶段

需求 | 包管理 | 脚手架 | 开发 | 合并 | 压缩 | 构建 | 打包

线上监控 | 灰度 | 部署 | 预发环境 | 自动化测试 | 测试环境 | 代码检查

上线阶段

发布阶段

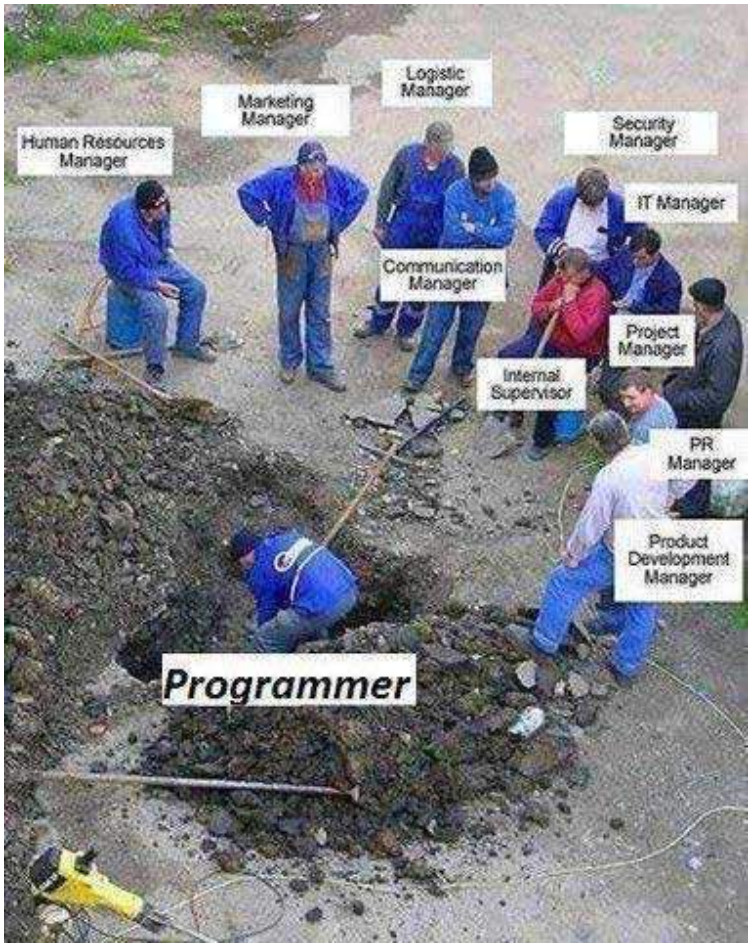
测试阶段

case1

- 拉代码
- ~~开发完啦~~
- ~~静态代码检查~~
- ~~线上编译与构建~~
- ~~自动化测试~~
- ~~灰度发布~~
- ~~自动雪碧~~
- ~~前端性能和错误监控~~
- ~~源站版本管理与组件管理~~
- 上线啦！
- bug

case2

- 需求A，B拉代码
- A开发完了，提测
- B拉代码，开发，提交
- 测试部署：tm什么鬼，A这功能有bug吧?
- A把B干掉，重新revert commit
- 测试好啦，上预发！
- 线上有个bug，C赶紧拉代码改一下！
- 用户投诉太多了！C赶紧上线！
- A预发验证好了，合并主干失败？？
- D改了个基础组件，Z位升级发到npm
- A解（删）决（掉）冲突，预发重新构建
- D你这什么破组件，这个场景有bug？？
- A写死版本号，封网了。。。



case3

- 新人，记得先去~~XXX平台~~创建一个新需求，然后去~~XXX~~代码仓库克隆下代码，拉代码的时候使用~~XXX平台~~的脚手架，然后去~~XXX站~~下载一个开发工具，开发的时候去~~XXX文档中心~~看下我们的开发规范，去~~XXX组件库~~找你要的组件，开发完毕后去~~XXX平台~~创建一个测试任务，并在~~XXX平台~~申请一台测试机，等测试过了去~~XXX平台~~提交代码部署，然后去~~XXX平台~~通过一下自动化测试，然后在~~XXX平台~~提一个发布单，去~~XXX平台~~要个预发绑定，上线时别忘了去~~XXX平台~~申请一个灰度发布单，对了，记得去~~XXX平台~~查看业务数据，去~~XXX平台~~查看监控数据，去~~XXX平台~~查看错误数据，之后别忘了去~~XXX平台~~写篇文章总结下！！

我唱了八十八个差不多 都差不多

差不多先生不会在乎这么多

流程缺失

效率低下

无法追溯

难以量化

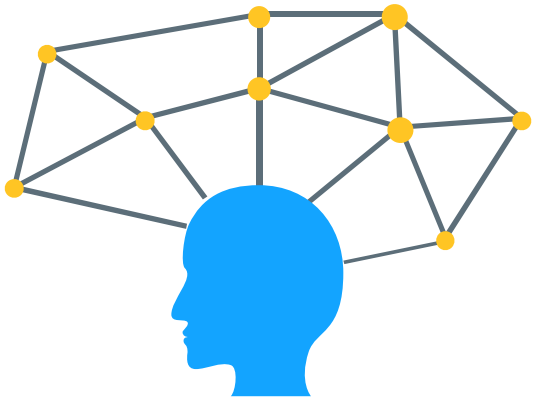
流程管控

流程管控的诉求

- 研发流程应该是DSL化的，可描述的
- 研发流程应该是收敛的
- 研发流程必须可追溯
- 研发流程必须方便接入其他服务，只专注于流程

演示

流程管控平台？怎么特么又有一个平台？？？？？？？



如何实现

BPMN流程引擎

+

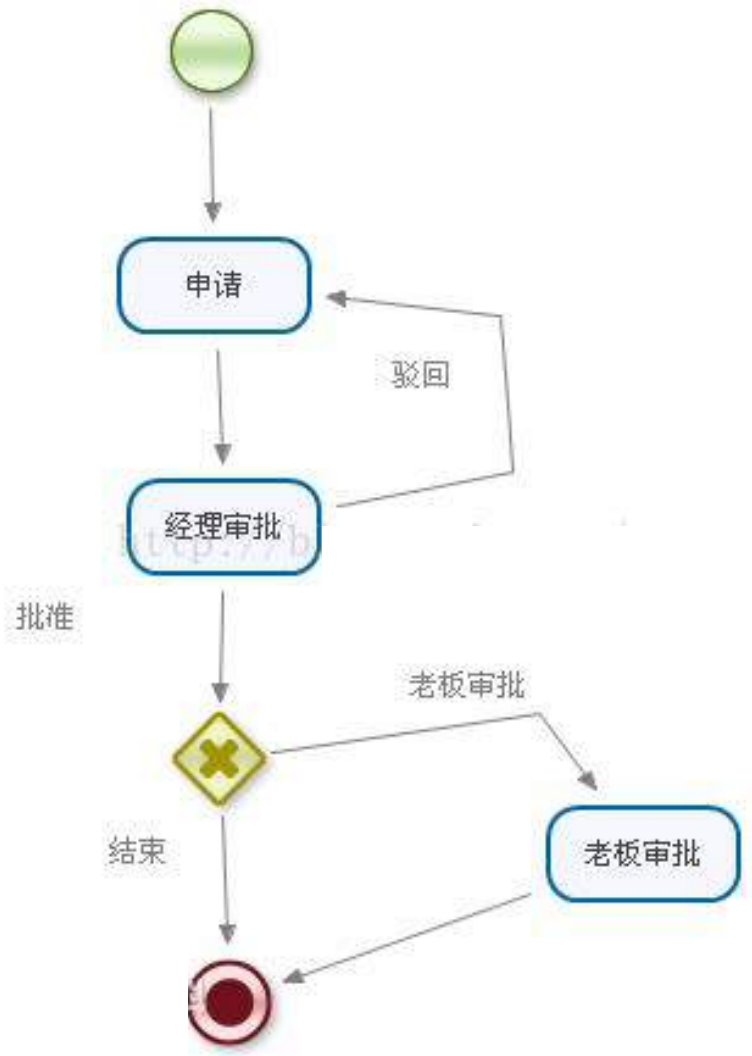
基础服务BAAS化

流程引擎-概念

BPMN：业务流程建模与标注

何为工作流

工作流（Workflow）指“业务过程的部分或整体在计算机应用环境下的自动化”。是对工作流程及其各操作步骤之间业务规则的抽象、概括描述。



流程引擎介绍

- Shark - XPDL - Process Activity
- osworkflow - fsm - State-action
- Jbpm, Activiti5 - bpmn2.0

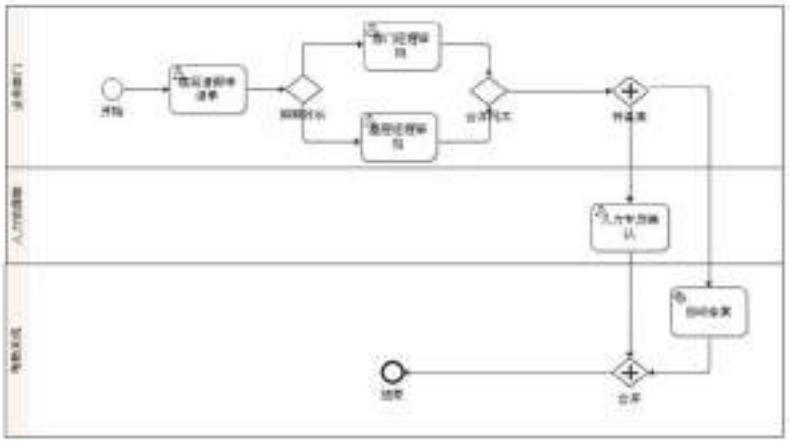
BPMN：业务流程建模与标注

- Activities（活动）——在工作流中所有具备生命周期状态的都可以称之为“活动”，如原子级的任务（Task）、流向（Sequence Flow），以及子流程（Sub-Process）等
- Gateways（网关）——顾名思义，所谓“网关”就是用来决定流程流转指向的，可能会被用作条件分支或聚合，也可以被用作并行执行或基于事件的排它性条件判断
- Events（事件）——在BPMN2.0执行语义中也是一个非常重要的概念，像启动、结束、边界条件以及每个活动的创建、开始、流转等都是流程事件，利用事件机制，可以通过事件控制器为系统增加辅助功能，如其它业务系统集成、活动预警等

工作流为什么适合驱动研发流程

- 可编排，工作流引擎的描述有标准，驱动也有标准引擎。
- 强大的驱动能力，可以完成节点的自动运行，重试等复杂场景。
- 服务与流程解耦，前端基础服务不再揉杂到平台中，而是下沉，流程只需关注研发过程本身。
- 改进和优化业务流程，提高流程工作效率。

BPMN 实例



序号	图元	类型	功能描述
1	开始	Start Event	标志一个流程的起始
2	填写请假申请单	User Task	填写请假表单，注明请假人、天数、原因
	部门经理审批	User Task	如果假期时长大于 3 天，由部门经理审批
	基层经理审批	User Task	如果假期时长小于 3 天，由基层经理审批
	人力专员确认	UserTask	人力专员确认
3	假期时长	ExclusiveGateway（Diverging）	互斥分叉网关，判断假期时长，用来决定下步流程分支
4	合并网关	ExclusiveGateway（Converging）	互斥合并网关，合并条件分支，重回流程主干
5	转备案	ParallelGateway（Diverging）	并行分叉网关，并行执行"人力专员确认"和"自动备案"任务
6	合并	ParallelGateway（Converging）	并行合并分支，待"人力专员确认"和"自动备案"任务都完成后，合并流程分支，重回流程主干
7	自动备案	ServiceTask	系统自动记录请假信息及审批结果，备案
8	结束	EndEvent	标志流程结束

回顾，如何解决这个case

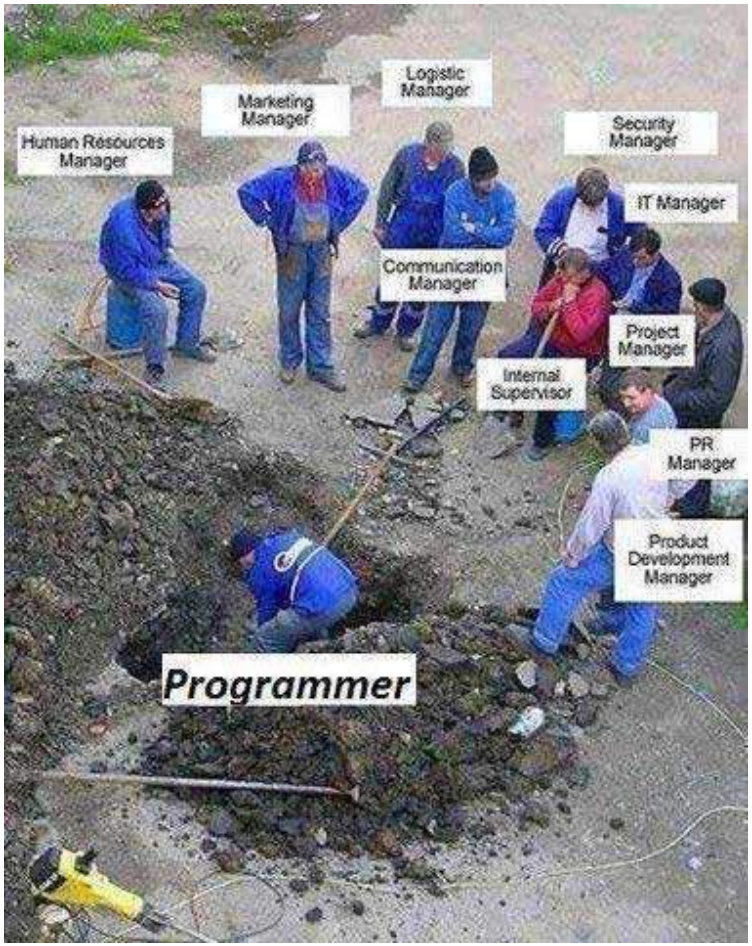
- 拉代码
- ~~开发完啦~~
- ~~静态代码检查~~
- ~~线上编译与构建~~
- ~~自动化测试~~
- ~~灰度发布~~
- ~~自动雪碧~~
- ~~前端性能和错误监控~~
- ~~源站版本管理与组件管理~~
- 上线啦！
- bug

一根筋类型开发

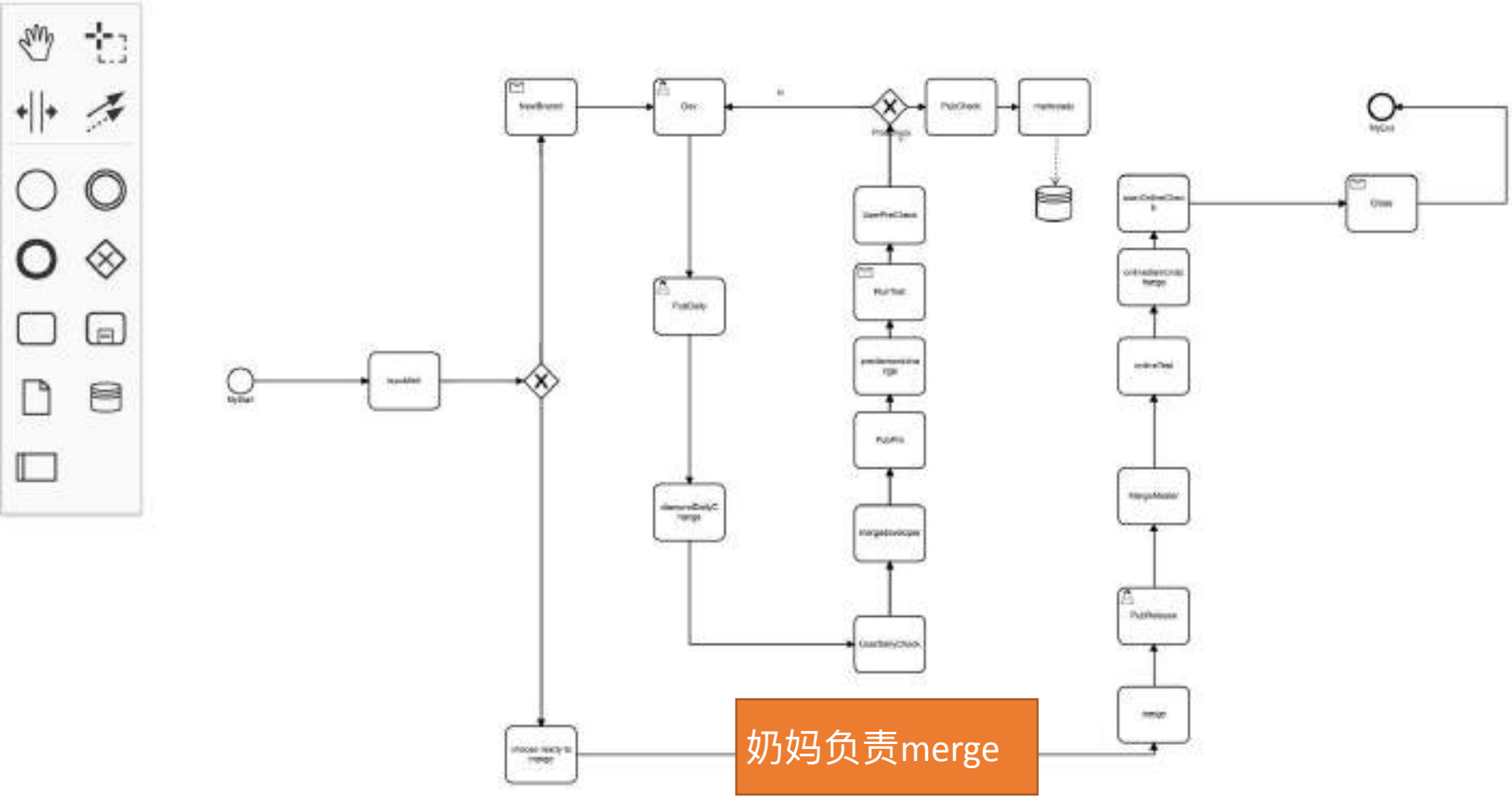

初始化迭代
新建分支
开发
发布预发
预发测试
预发测试结果验证
预发写基线
手动验证
线上发布准入
分支合并检查
发布线上
合并分支到主干
关闭迭代
正式写基线
迭代结束
迭代详情
基础信息
group: just-space
dataId: com.alibaba.just-space.common.test-just-space.testassests1
content: 0.0.12
预发环境
http://diamond.alibaba-inc.com/diamond-ops/static/pages/config-detail/index
space.common.test-just-space.testassests1&group=just-space&serverId=pre
预发测试
预发基线
当前流程
验证通过，并发布线上
验证不通过，退回开发中

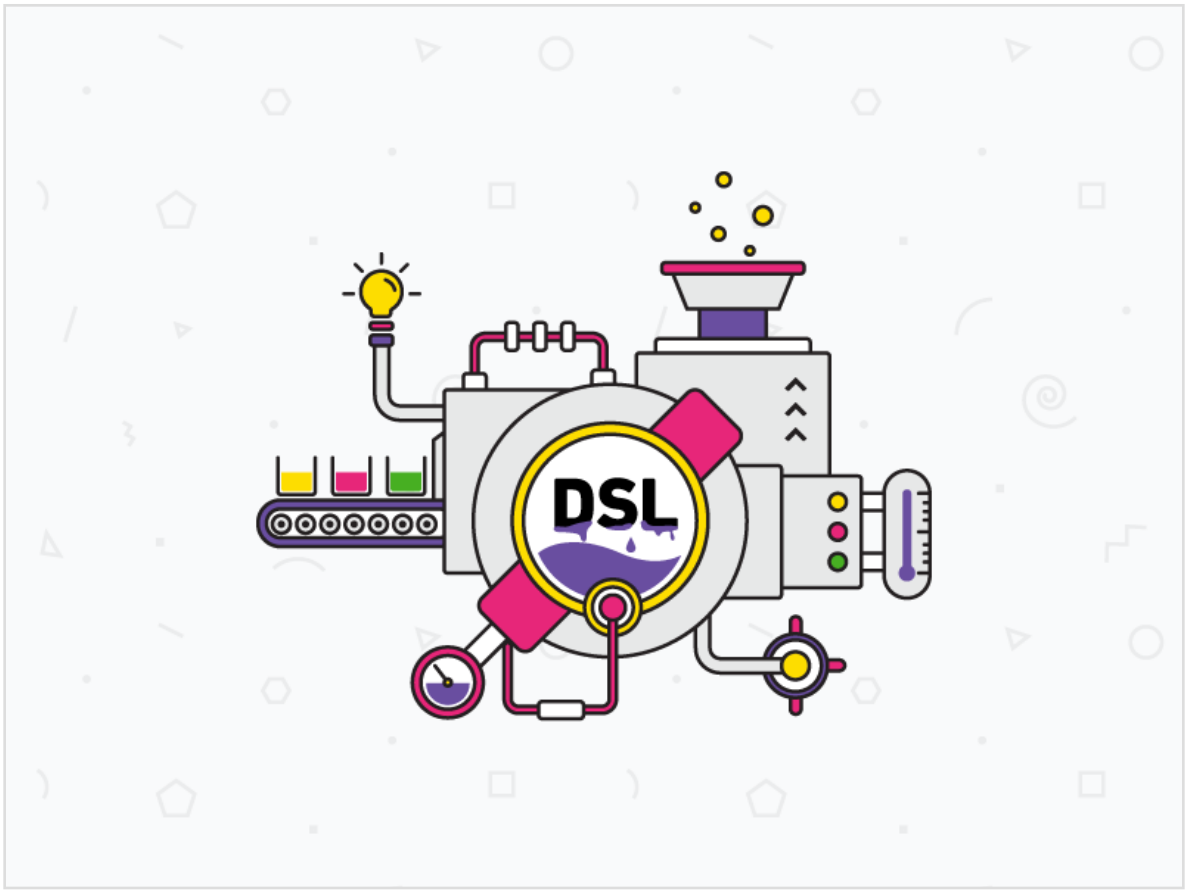
回顾，如何解决这个case

- 需求A，B拉代码
- A开发完了，提测
- B拉代码，开发，提交
- 测试部署：tm什么鬼，A这功能有bug吧?
- A把B干掉，重新revert commit
- 测试好啦，上预发！
- 线上有个bug，C赶紧拉代码改一下！
- 用户投诉太多了！C赶紧上线！
- A预发验证好了，合并主干失败？？
- D改了个基础组件，Z位升级发到npm
- A解（删）决（掉）冲突，预发重新构建
- D你这什么破组件，这个场景有bug？？
- A写死版本号，封网了。。。



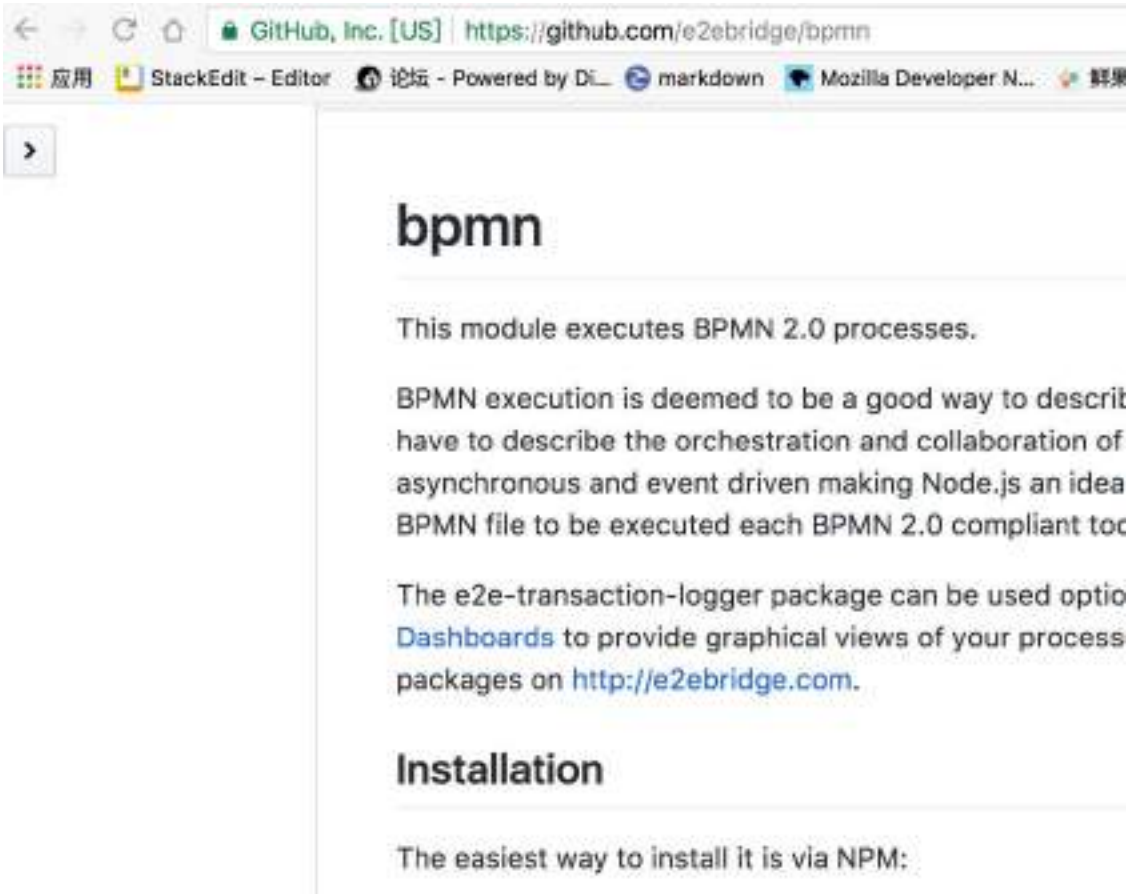
多人，单仓库，奶妈型开发流程





实现：工作流引擎如何驱动研发流程

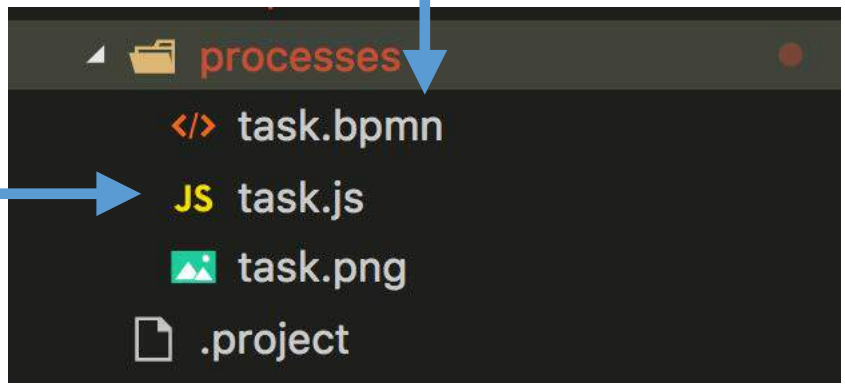
开源框架



- 添加mysql持久化引擎
- 适配async/await
- 缓存服务支持接入redis
- 子流程能力支持

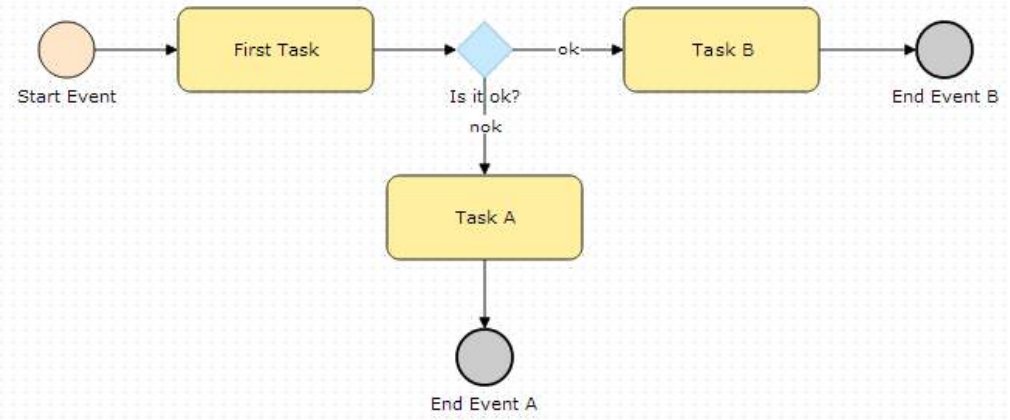
流程引擎-最简单实现


```
exports.MyStart = function(data, done) {
  console.log('process start');
  done(data);
};

exports.MyTask = function(data, done) {
  console.log('task run');
  done(data);
};

exports.MyTaskDone = function(data, done) {
  console.log('task done');
  done(data);
};

exports.MyEnd = function(data, done) {
  console.log('process end');
  done(data);
};
```

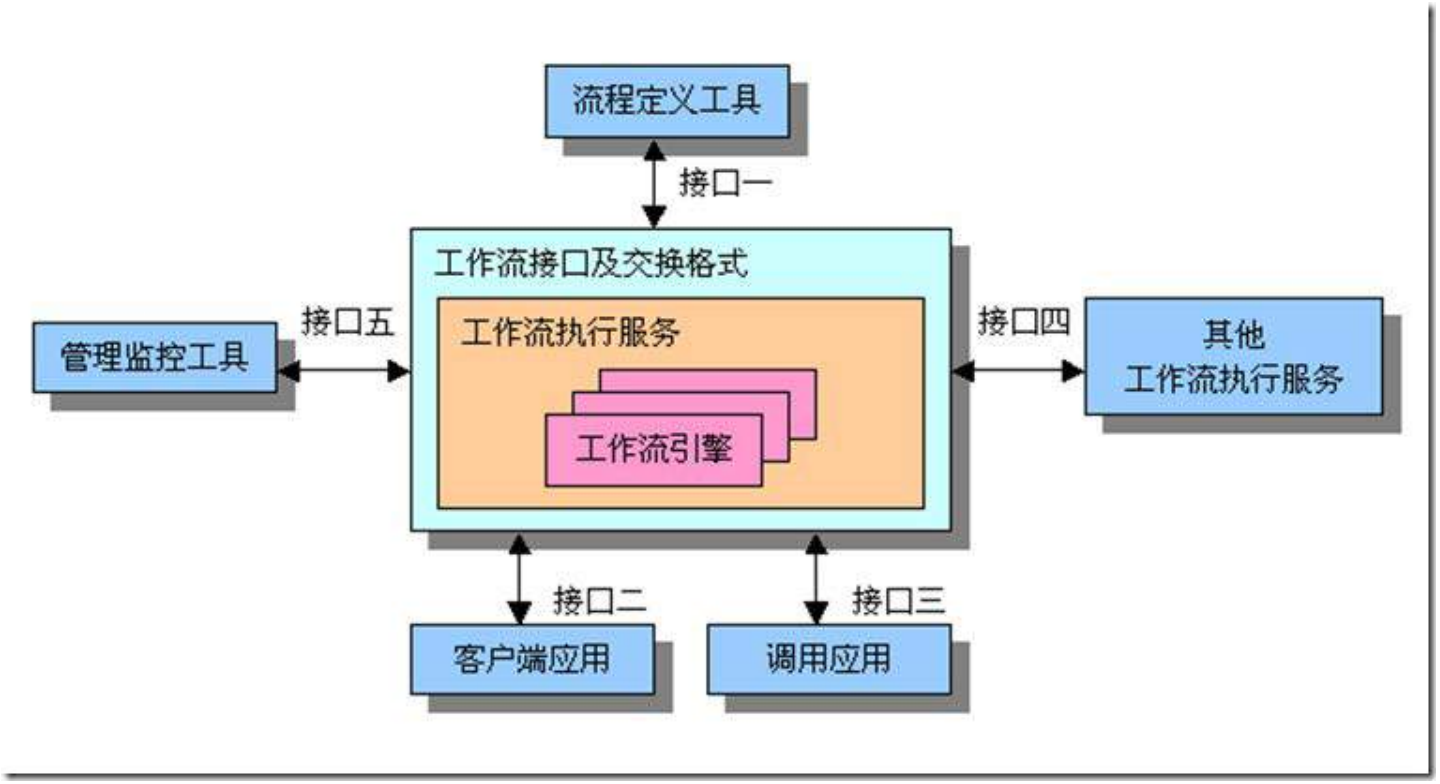



```
var bpmn = require("bpmn");
bpmn.createUnmanagedProcess("./task.bpmn", function(err, myProcess){
    myProcess.triggerEvent("MyStart");
});
```

流程引擎-分支控制

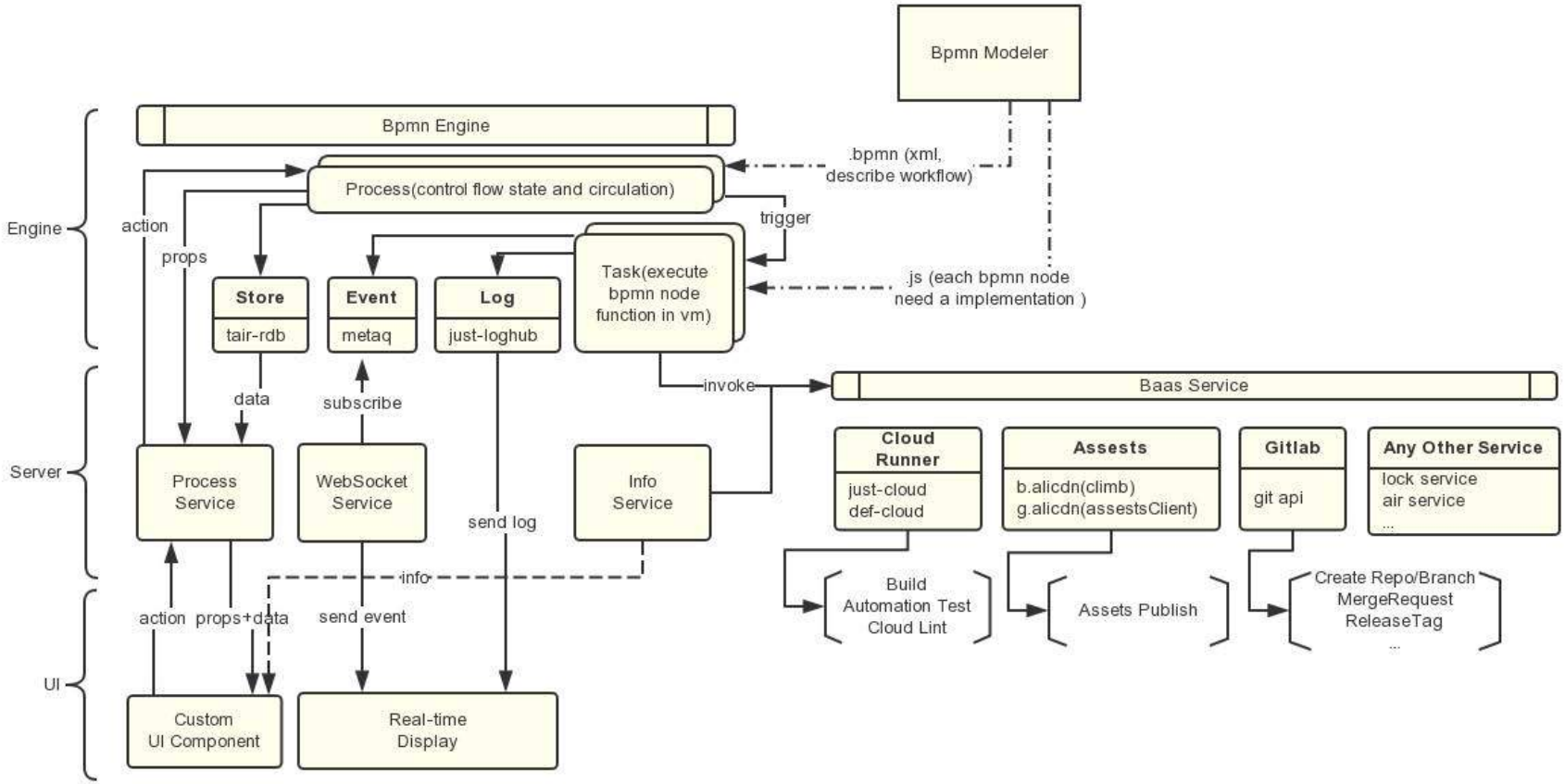



```
exports.Is_it_ok_$ok = function(data) {
  //处理分支ok
  return true;
};

exports.Is_it_ok_$nok = function(data) {
  //处理分支nok
  return false;
};
```

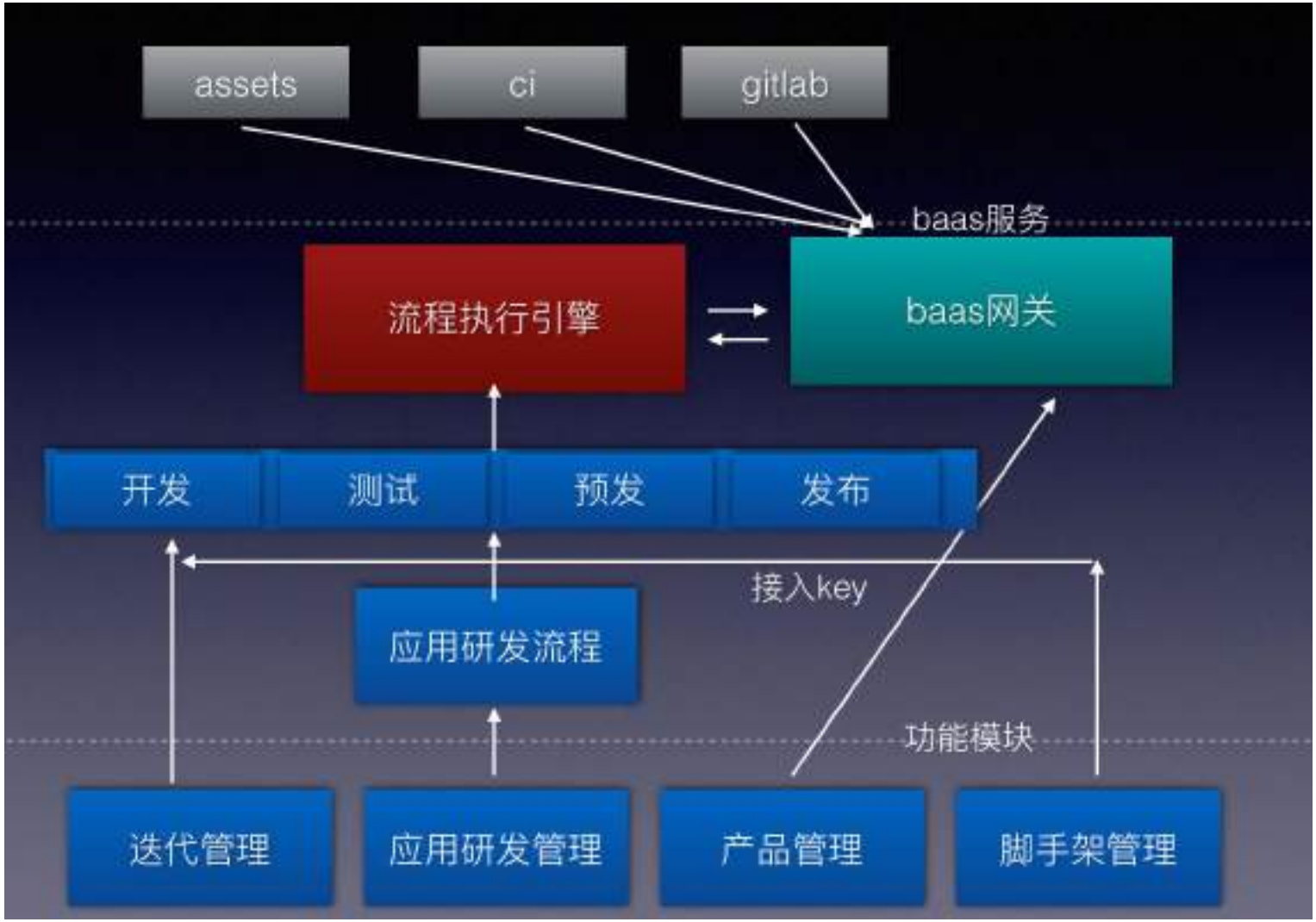

流程定义工具
接口一
工作流接口及交换格式
工作流执行服务
工作流引擎
管理监控工具
接口五
接口四
其他
工作流执行服务
接口二
接口三
客户端应用
调用应用

整体架构-WFMC模型

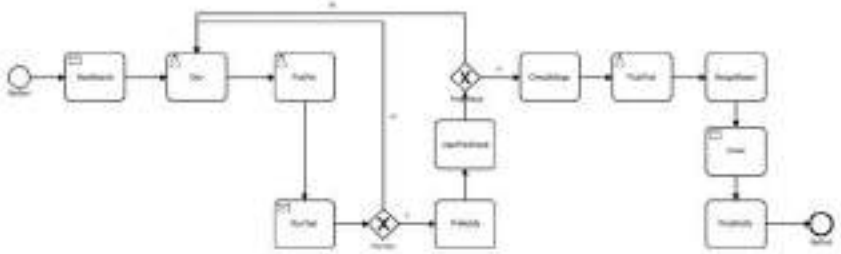
系统整体架构


Bpmn Modeler
Bpmn Engine
.bpmn (xml, describe workflow)
Process(control flow state and circulation)
trigger
Engine
action
props
Store
tair-rdb
Event
metaq
Log
just-loghub
Task(execute bpmn node function in vm)
.js (each bpmn node need a implementation)
invoke
Baas Service
data
subscribe
Server
Process Service
WebSocket Service
send log
Info Service
Cloud Runner
just-cloud
def-cloud
Assests
b.alicdn(climb)
g.alicdn(assestsClient)
Gitlab
git api
Any Other Service
lock service
air service
...
info
action
props+data
send event
Build
Automation Test
Cloud Lint
Assets Publish
Create Repo/Branch
MergeRequest
ReleaseTag
...
UI
Custom UI Component
Real-time Display

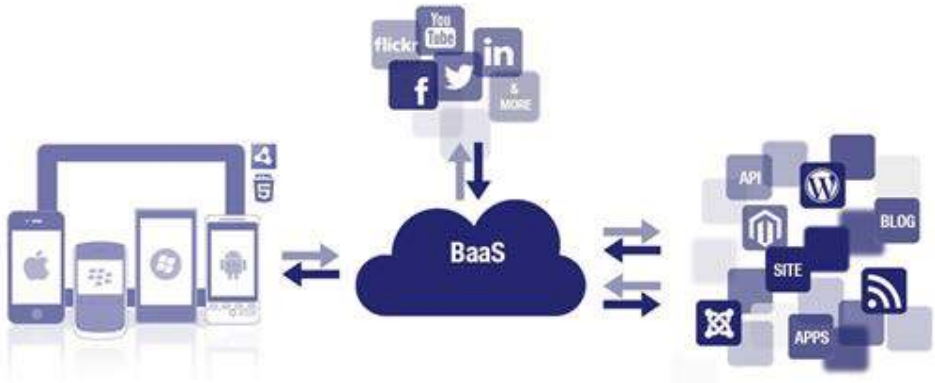
系统分层


assets
ci
gitlab
baas服务
流程执行引擎
baas网关
开发
测试
预发
发布
接入key
应用研发流程
功能模块
迭代管理
应用研发管理
产品管理
脚手架管理

流程定制-可视化搭建



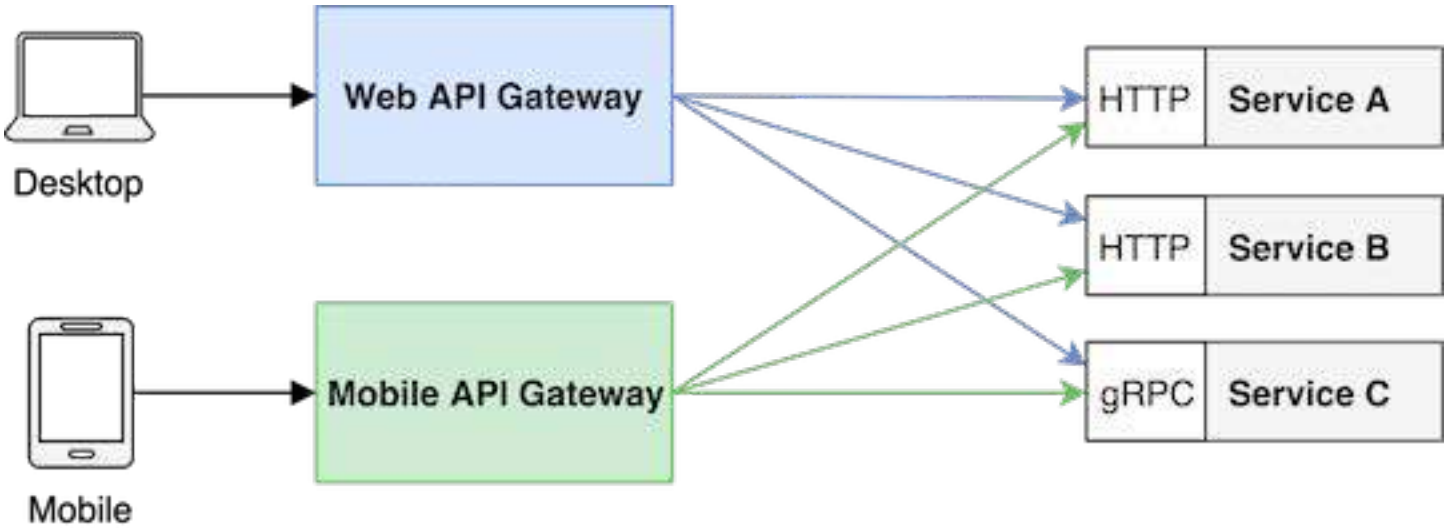
- 初始化迭代
- 新建分支
- 开发
- 发布预发
- 预发测试
- 预发测试结果验证
- 预发写基线
- **手动验证**
- 线上发布准入
- 分支合并检查
- 发布线上
- 合并分支到主干
- 关闭迭代
- 正式写基线
- 迭代结束



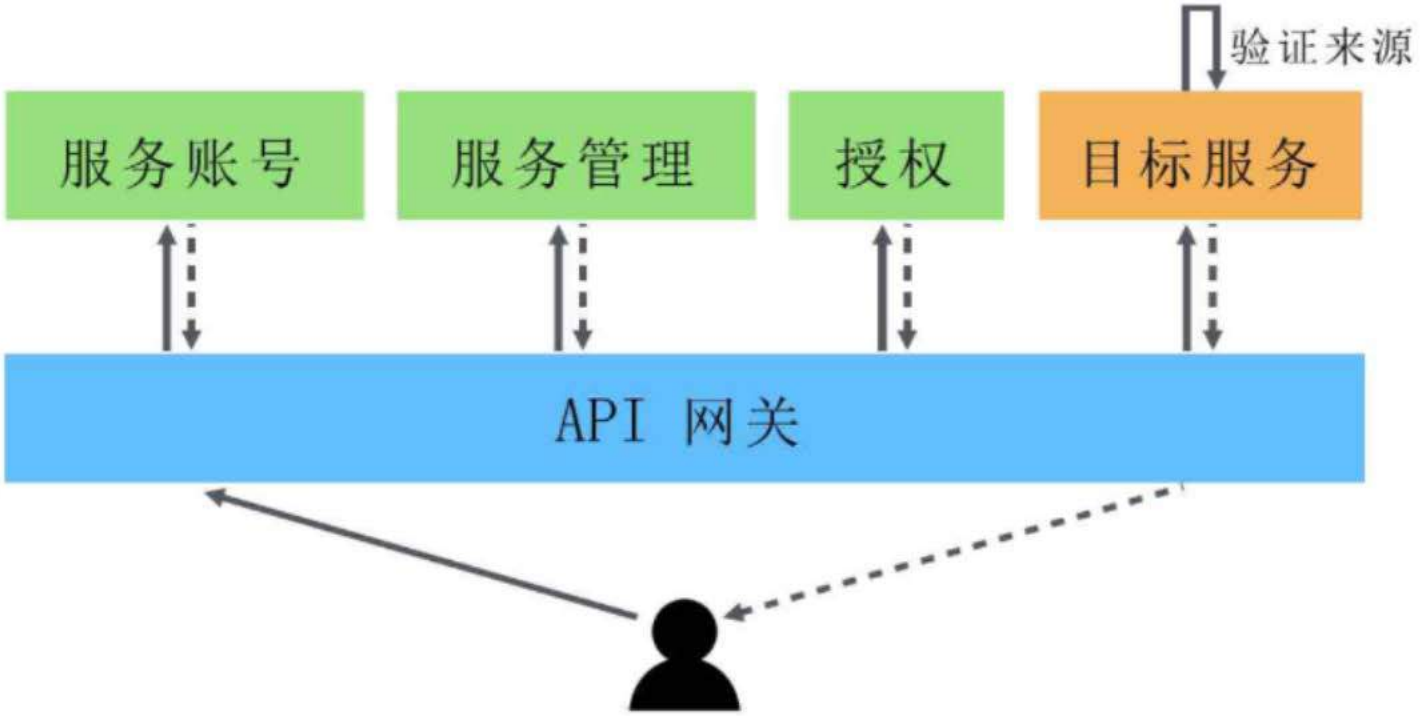
Backend as a Service
后端即服务

基础服务BAAS化

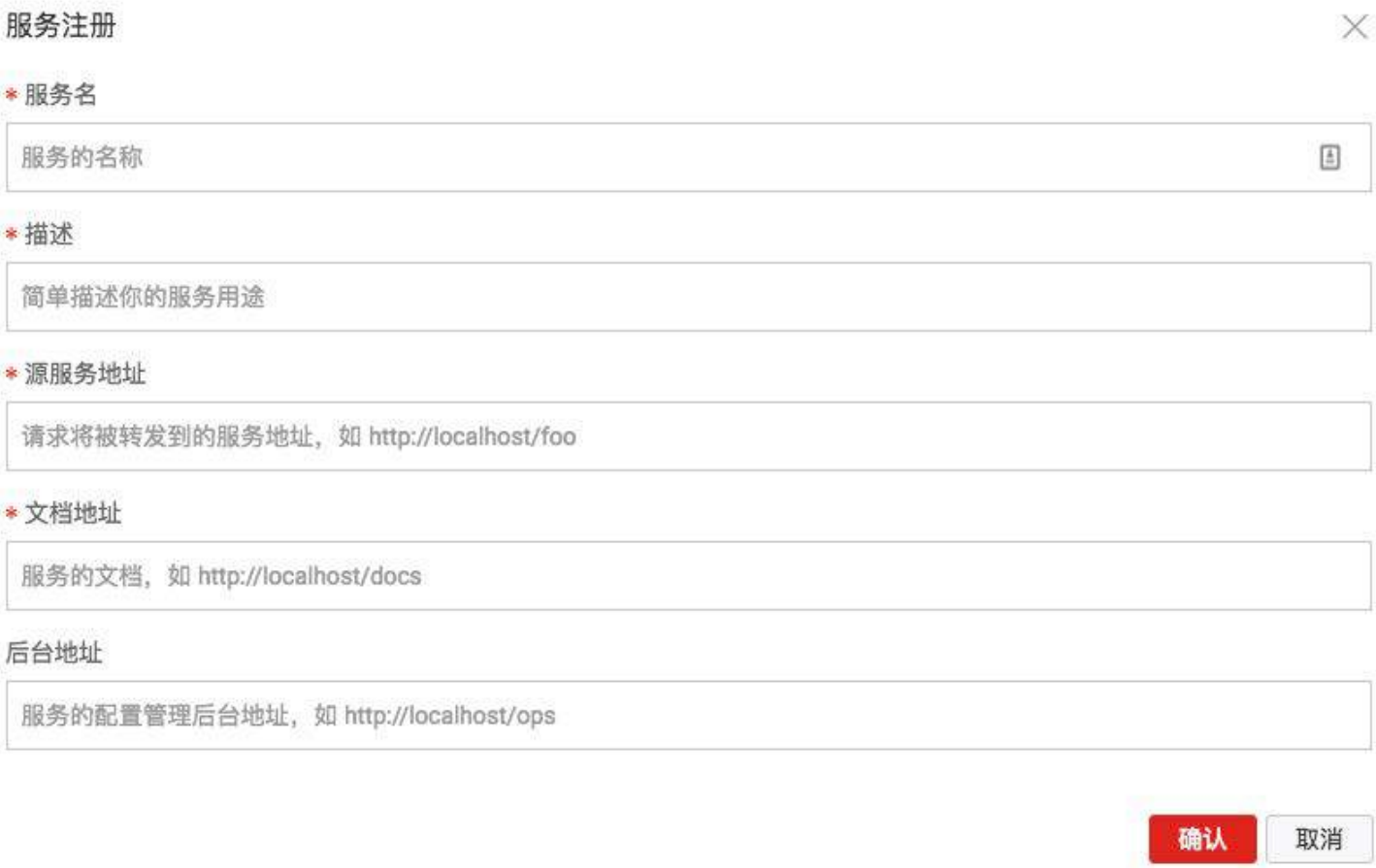
baas化-概念


Desktop
Mobile
Web API Gateway
Mobile API Gateway
HTTP
Service A
HTTP
Service B
gRPC
Service C

baas化-技术架构



baas化-服务注册


服务注册
* 服务名
服务的名称
* 描述
简单描述你的服务用途
* 源服务地址
请求将被转发到的服务地址，如 http://localhost/foo
* 文档地址
服务的文档，如 http://localhost/docs
后台地址
服务的配置管理后台地址，如 http://localhost/ops
确认
取消

baas化-接入注册


应用
成员
服务
设置
ID
1
Secret
3a92fbc4b958780c18302a7407f167930637b80cc1594bc6
服务调用说明
您可以使用产品秘钥（在请求头中携带 x-app-id 和 x-app-secret ）来调用其他授权服务。查看服务调用文档

流程定制-调用服务

获取baas准入

初始化baas client

调用服务

写上下文，持久化

```
RunTest: function* (data) {
  this.logger.info('RunTest')
  const user = this.getRootProperty('user')
  const application = this.getRootProperty('application')
  const change = this.getRootProperty('change')
  const callbackUrl = this.helper.formatCallbackUrl(change.id, 'RunTest')
  const baasKey = this.helper.getBaasKey()
  const cloudRunner = new cloudRunnerClient(baasKey)
  this.logger.info('callbackUrl =>', callbackUrl)
  const res = yield cloudRunner.create({
    bindingId: 1,
    args: JSON.stringify({
      repo: application.repo_http,
      branch: change.branch
    }),
    user: user.workid,
    webhook: callbackUrl
  })
  this.logger.info('LintTask =>', res)
  const logs = this.getRootProperty('logs') || {}
  this.setRootProperty('logs', Object.assign(logs, {
    'RunTest': {
      'TestLog': res.log_web_url
    }
  }))
},
```

流程定制-隔离

沙箱中运行代码，
避免干扰引擎

```
/**
 * 将code编译执行
 * @param  {string} code  [description]
 * @param  {object} extra [description]
 * @return {processHandler}       [description]
 */
const main = function (code, extra) {
  const { libPath, scriptPath } = extra

  // vm 环境:
  // 1. 可以 require 的模块和 app 保持一致
  // 2. FIXME: console 暂时保留方便开发调试（可以考虑替换成 egg 的特定 logger）
  const vm = new NodeVM({
    // console: 'off',
    require: {
      external: true,
      builtin: ['*'],
      root: libPath,
      //context: 'sandbox'
    }
  })
  const script = new VMScript(code)
  const processHandler = vm.run(script, scriptPath)
  return wrapAll(processHandler, extra)
}
```

工程终局

线上平台

- 代码托管
- 研发流程平台
- BAAS网关
- 需求与文档管理

线下工具链


基础服务

- 代码扫描
- 自动化测试
- 源站部署
- 云构建

我们的收获

开放接入

开放接入最小化改造成本，最大化集中收益

研发数据

研发全流程数据集中，分析这些数据更利于针对性优化

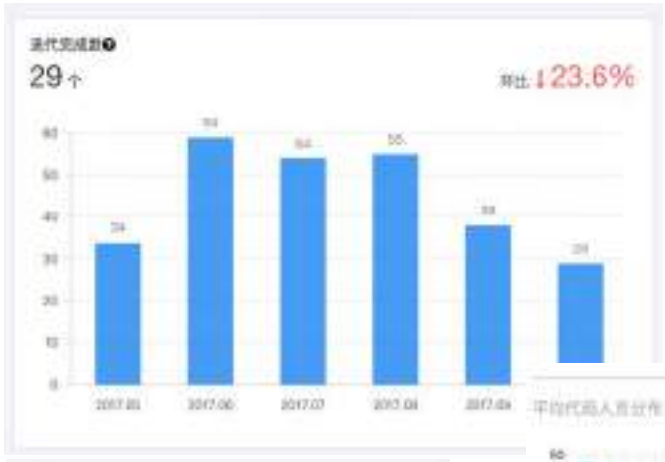
发布全貌

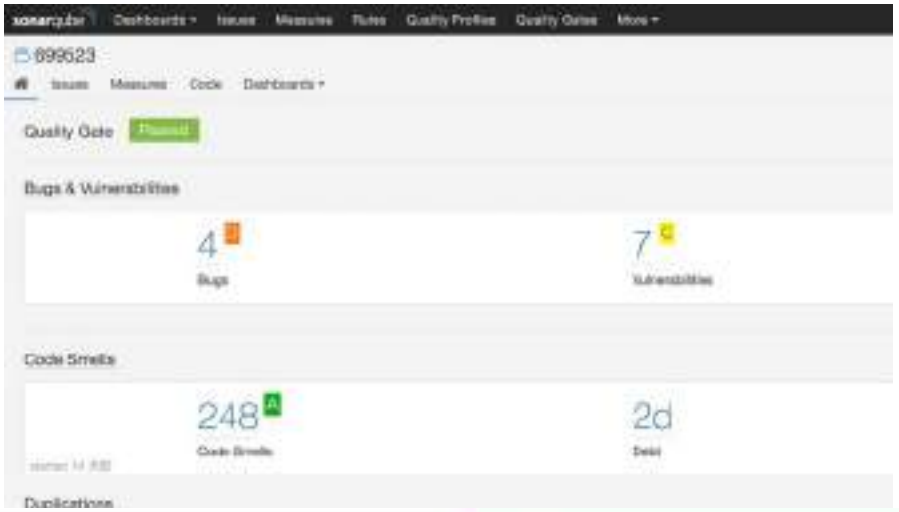
前端开发者可以在平台中看到发布的全貌，方便开发者回溯自查，也方便线上问题及时定位

流程卡点

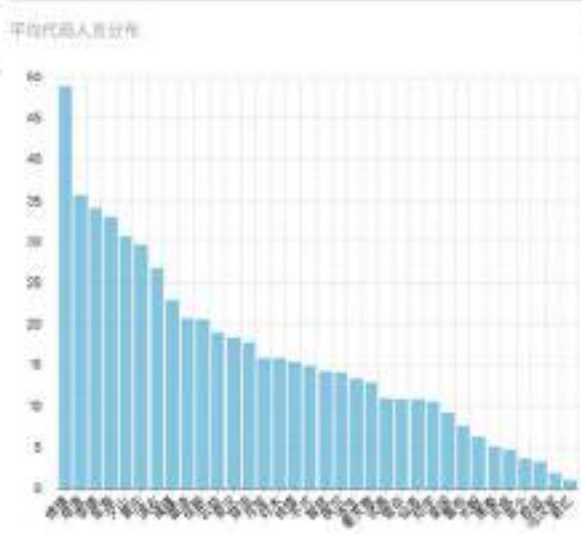
严控研发过程，避免开发者绕过质量检查步骤

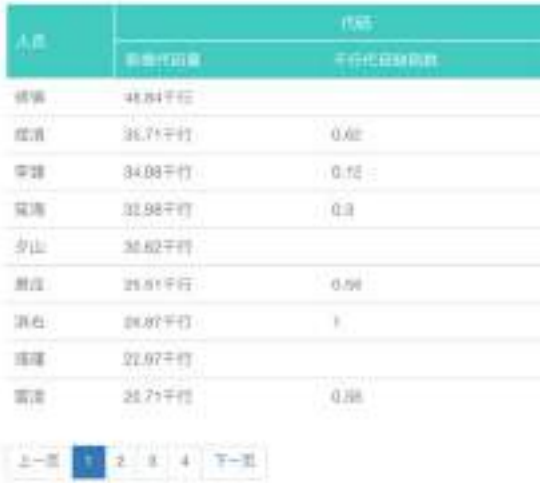
研发能力指标


29个
环比↓23.6%

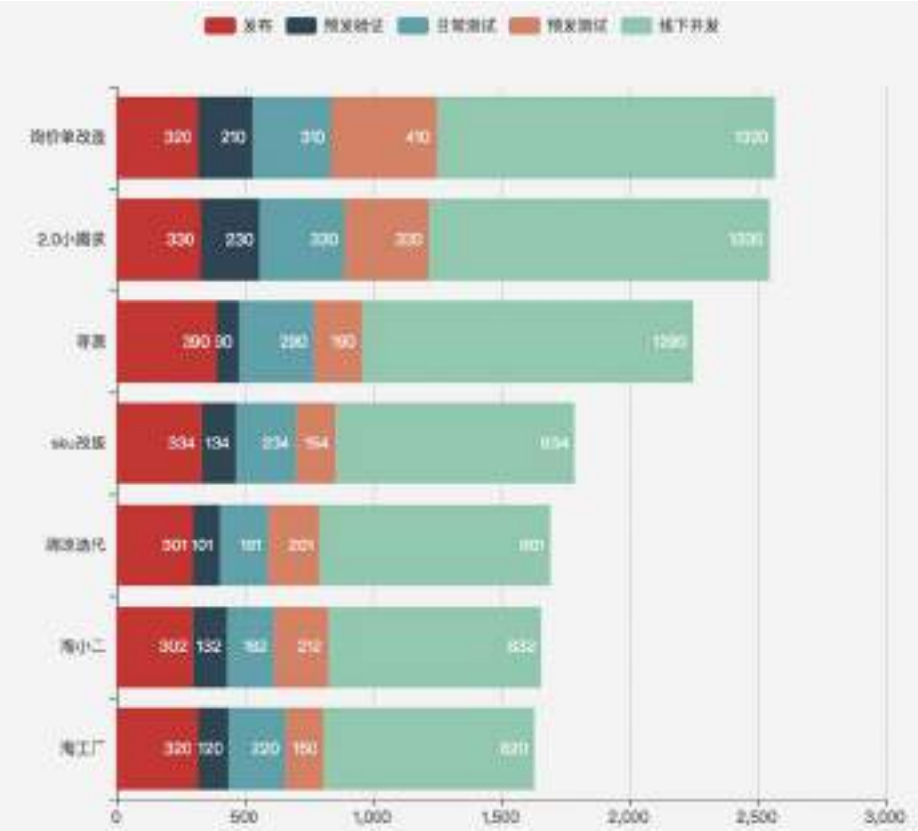

699623
Quality Gate
Bugs & Vulnerabilities
4
Bugs
7
Code Smells
248
2d

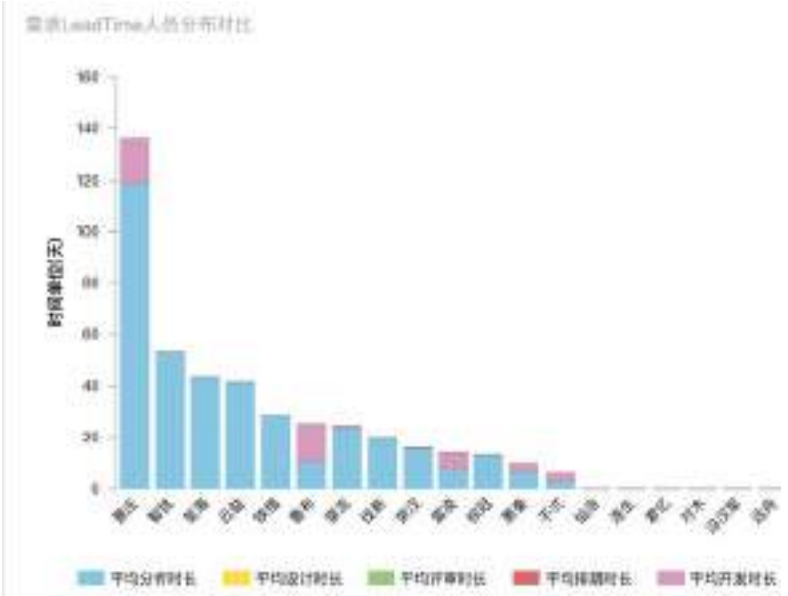

7.2 千行
环比↓8.5%

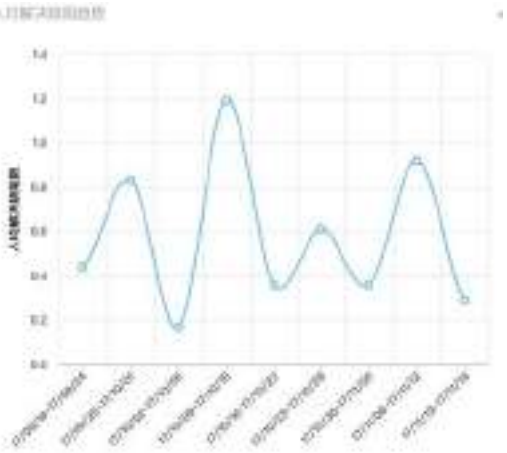




研发效率指标


发布
预发验证
日常测试
预发测试
线下开发
询价单改造
2.0小闹铃
寻源
sku改版
海小二
海工厂
0
500
1,000
1,500
2,000
2,500
3,000


时间单位(天)
平均分析时长
平均设计时长
平均评审时长
平均排期时长
平均开发时长



研发质量指标

主搜-itemlist 关注 复制 运行 编辑 删除

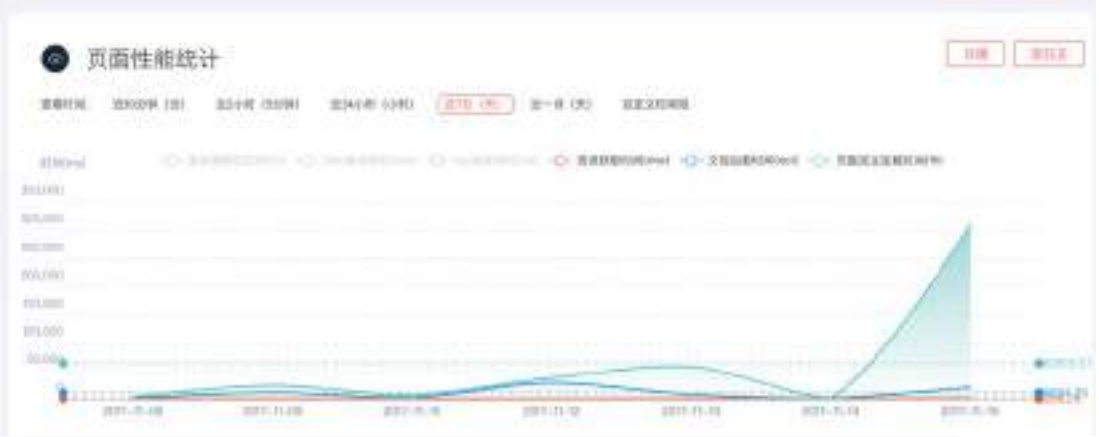
UI 测试 通过

Current History

主搜-itemlist（运行机器：10.189.225.48）

- 阿里站外资源检测
- 检测页面链接是否合法
- 检测页面是否有 console.log 输出
- 检测页面是否有吊顶
- 检测页面图片是否失真
- 静态资源404检测
- 死链检测
- 淘宝天猫域页面站外或内网链接检测
- 图片链接合法性检测
- 页面标签闭合检测
- 页面标签重复属性检测
- 页面加载时是否有弹出框
- htmlhint-id
- JavaScript错误检测
- performance.timing API 检测网页加载时间
- phantomjs 检测 js error（不建议使用）

5 days ago
林谦
无
不回归测试



您的质量得分 88分，效率得分70分，能力得分99分，超过了阿里99.99%的迭代。。

Thank you
Q&A

GIAC | 全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE

扫码关注GIAC公众号

2017.thegiac.com