

中国平安 PINGAN

保险 · 银行 · 投资

Pafa5

2015年11月21日

平安科技(深圳)有限公司



# 内容概要

- 一、换汤不换药：SOA与微服务
- 二、像搭积木一样构建系统：组件化
- 三、动态运行时：Pafa5中间件
- 四、全局控制台：Pafa5在线管控平台
- 五、PAAS平台梦想
- 六、Dev-Ops：敏捷过程与精细化运维
- 七、Pafa5技术平台
- 八、QA



# 1.1 换汤不换药：SOA与微服务

微服务



轻量级SOA

怎么轻  
???

1. **点对点** 去中心化，避免ESB总线集中方式，避免总线单点，连接瓶颈
2. **轻量级协议** 推荐跨语言跨平台轻量级协议：http，Restful风格的接口，避免使用私有协议和重量级的协议~
3. **弱数据类型** 推荐JSON数据格式，避免WebService、Java、Hessian等强制数据类型的序列化方式
4. **服务动态注册** 推荐动态注册和发现，实时故障隔离，避免预注册模式
5. **强调业务领域** 避免庞大复杂的系统，主张弱化系统概念，强调业务领域的模块化拆分和业务建模，应用小微化，提高业务领域的内聚性，降低模块间耦合。

## 1.2 SOA框架-ESG

### ESG

- 科技自主研发的服务治理平台，推荐使用

### ESG特点

1. 点对点
2. 接口粒度的服务治理
3. Restful风格接口,HTTP协议 ( SPRING)
4. 服务需要预注册
5. 弱数据类型 基于SpringMVC的自动类型转换、校验、绑定

# 1.3 SOA框架-Dubbo

## Dubbo



- 阿里B2B SOA框架 每天为2,000+个服务提供30+亿次访问量支持

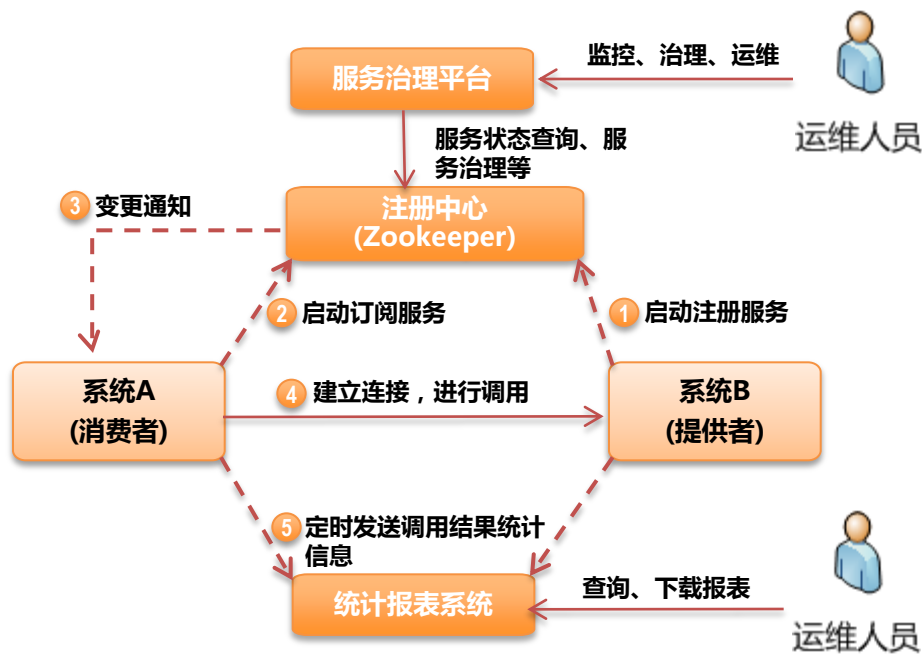
- 阿里B2B 平台技术部产品  
09年启动，已发行70+版本



# 1.3 Dubbo的特点

## Dubbo 特点

1. 点对点
2. 实时故障隔离和动态扩容 长连接心跳+变更通知
3. 接口粒度的服务治理
4. 支持多种调用协议 支持dubbo\http(java)\hessian\http(json)等
5. 弱数据类型 提供者自动协议适配、类型转换、校验、绑定



服务名 | 应用名 | 机器IP

图1: 服务治理平台

SEARCH xyk\_mps.MpsCustomerBindAction

提供者	消费者	应用	路由规则	动态配置
<input type="checkbox"/>	机器IP: <input type="text"/>			
<input type="checkbox"/>	<u>10.31.120.26:20880</u>			
<input type="checkbox"/>	<u>10.31.120.27:20880</u>			

图2: 统计报表系统

20130415 > Summary | +Provider | +Consumer

Statistics (2)				
Method: <input type="text"/>	Success	Failure	Avg Elapsed (ms)	Max Elapsed (ms)
<u>queryUserByEmail</u>	2588 --> 2588	0 --> 0	<u>6 --&gt; 4</u>	<u>1248 --&gt; 841</u>
queryAllInfo	158 --> 158	0 --> 0	10 --> 6	160 --> 101

# 1.4 Dubbo协议特点

## Dubbo 协议特点

- 1. NIO，管道模型，单长连接，无连接池，双向通信
- 2. 超时控制
- 3. 并行(分布式) 计算

NIO



Netty

### Dubbo、rmi、http协议对比：

协议	连接个数	连接方式	传输协议	序列化	超时控制	缺点
Dubbo	单连接	长连接	TCP	hessian java json	Thread.wait	大量数据IO，IO线程压力大会成为瓶颈。虽然可通过增加IO线的数量缓解，但对文件上传下载不建议使用此协议
RMI	多连接	短连接	TCP	java	Socket.soTimeout	调用者远多于服务提供者时，超多连接会压垮提供者，同时会导致网络拥塞
Http	多连接	短连接	TCP	JSON java	Socket.soTimeout	

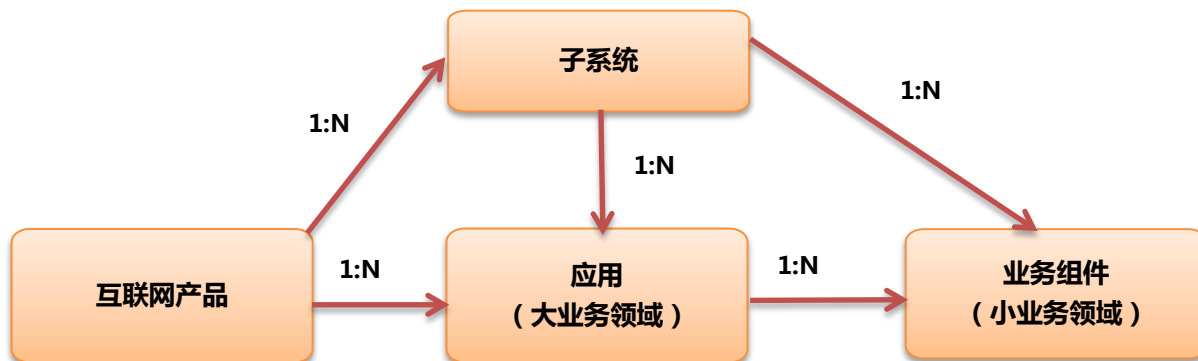
## 2.1 像搭积木一样构建系统：组件化

### 为什么要组件化？

1. 系统庞重，开发、测试、发布粒度大，效率低
2. 业务系统内部模块业务耦合高，内聚性低，冗余代码多
3. 研发过程繁重低效，敏捷流于形式，无法真正做到持续集成和持续交付

### 组件化特点

1. 弱化子系统概念，组件化开发和部署，减小发布粒度，按需发布，避免全局发布和生产抖动，减小测试工作量，降低开发成本。
2. 通过组件化的方式，强调业务领域，提升复用度和业务的内聚性。 ， 降间模块间的藕合。
3. 热部署，热插拔，持续交付和持续集成，真正地做到敏捷过程。



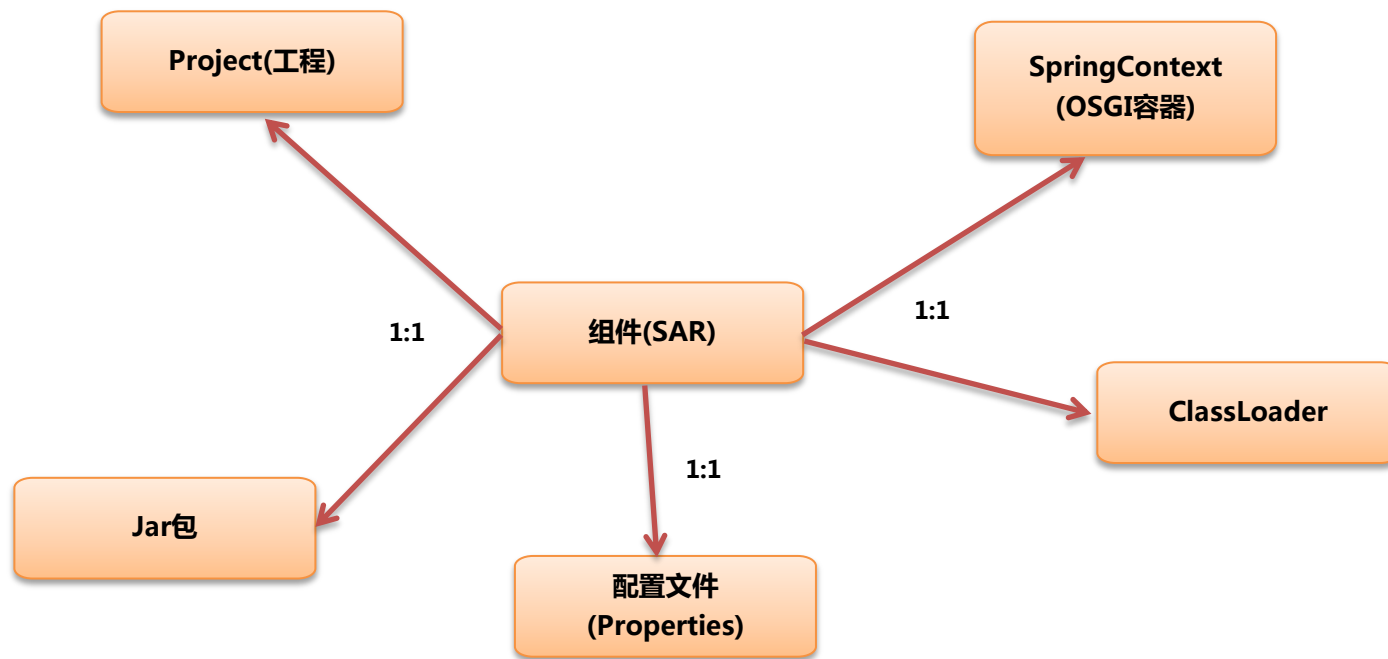



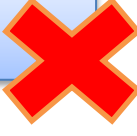
## 2.2 业界技术趋势：组件化

- 支付宝SOFA
- Spring Cloud
- 平安Pafa5

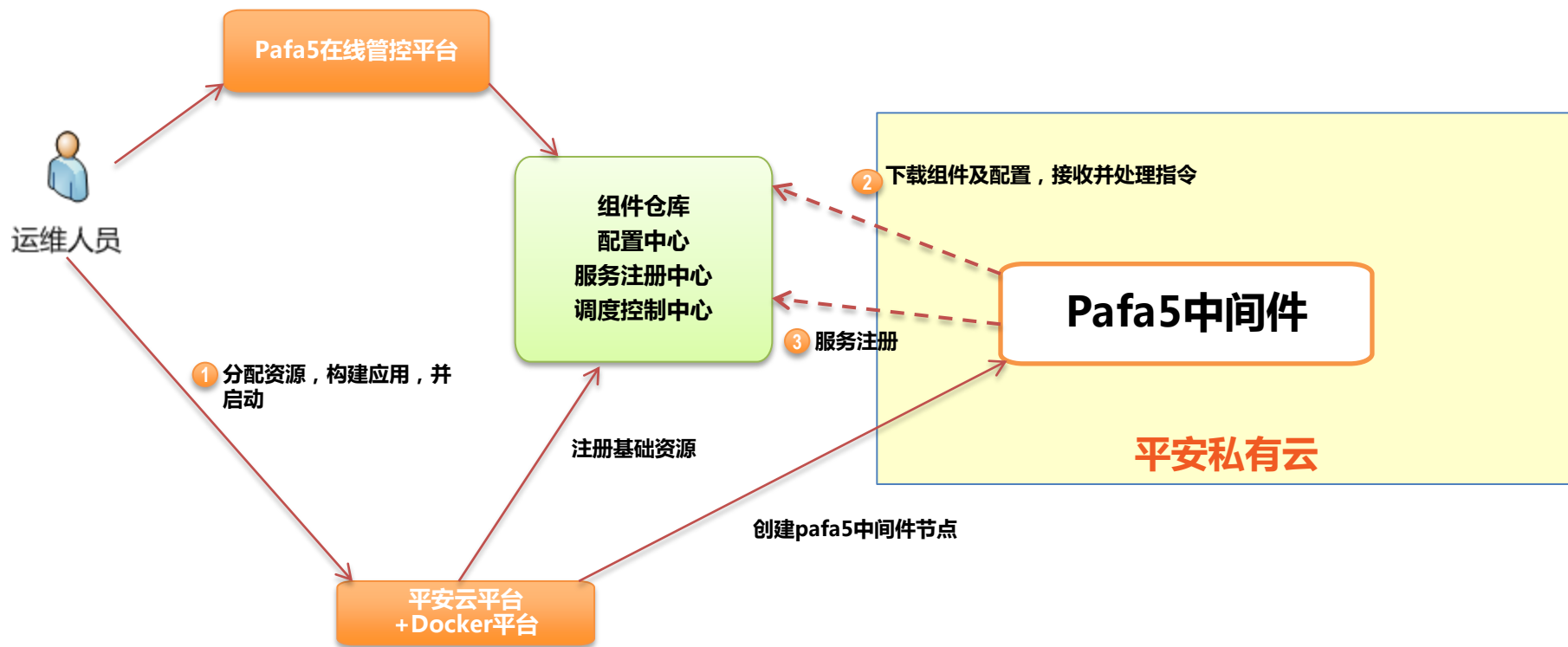


## 2.3 Pafa5组件(SAR)

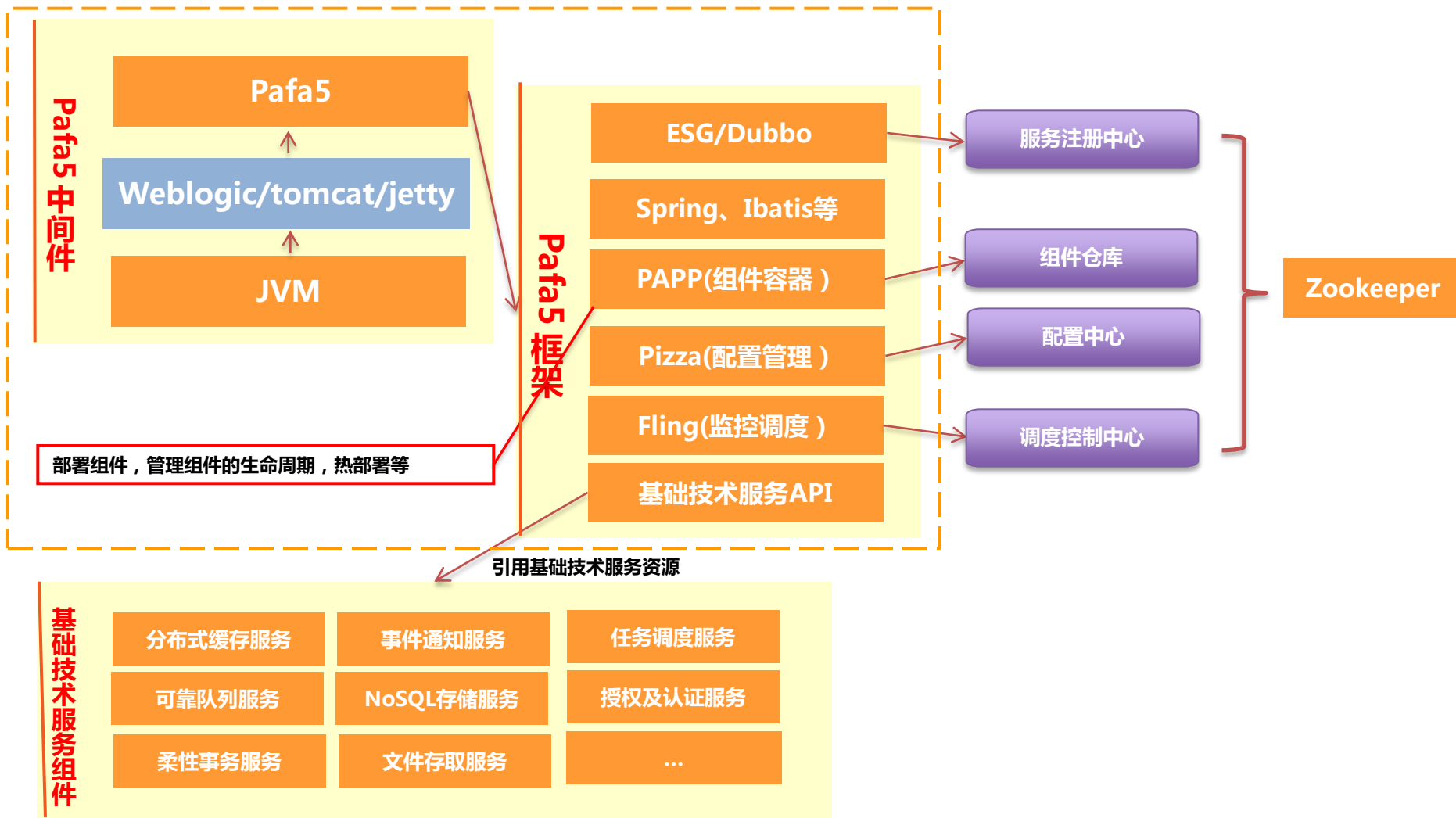


WAR?  EAR? 

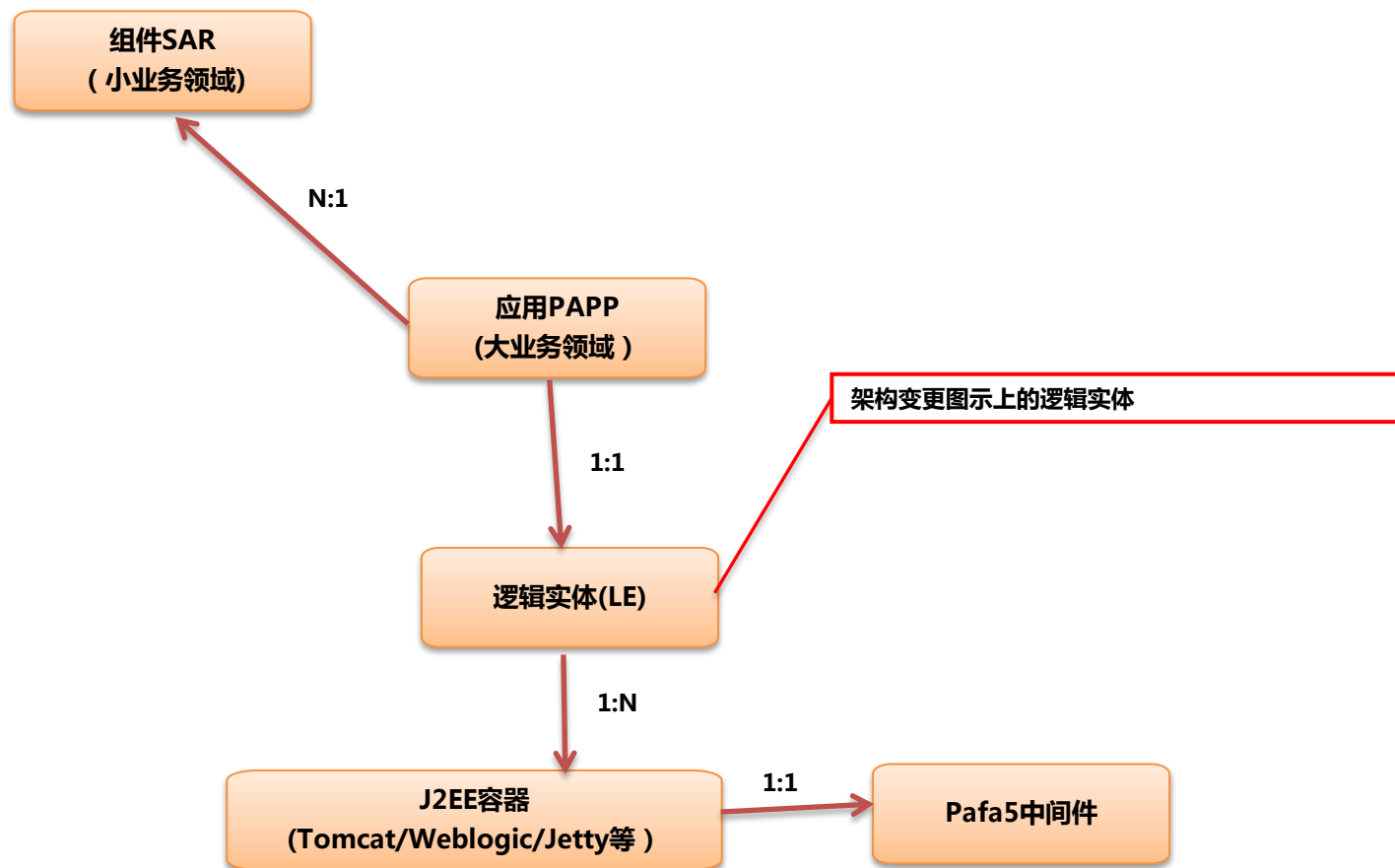
# 3.1 动态运行时：Pafa5中间件



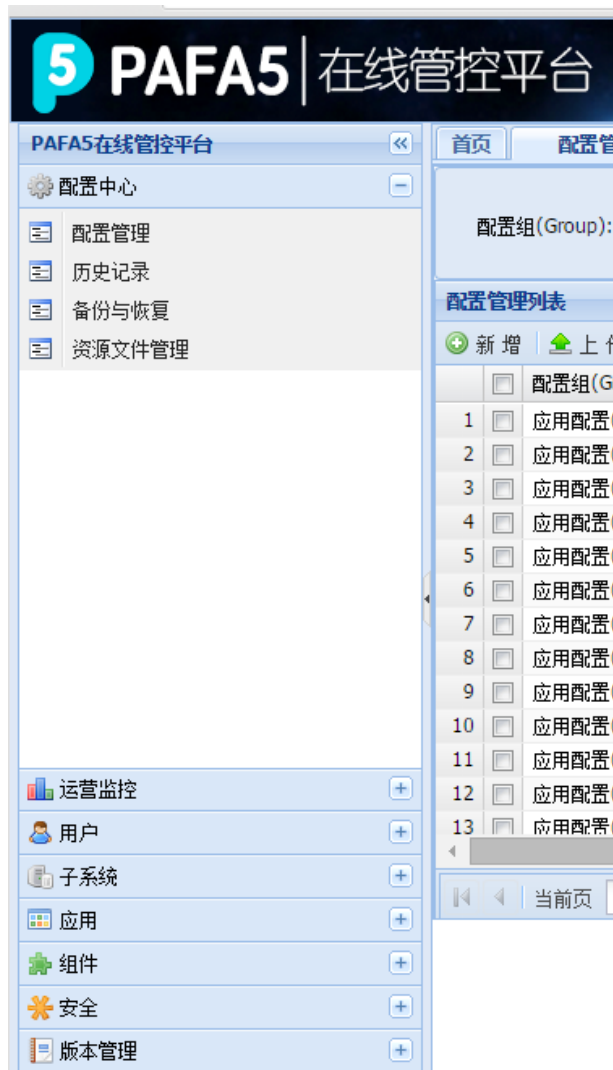
## 3.2 PAFA5中间件概要



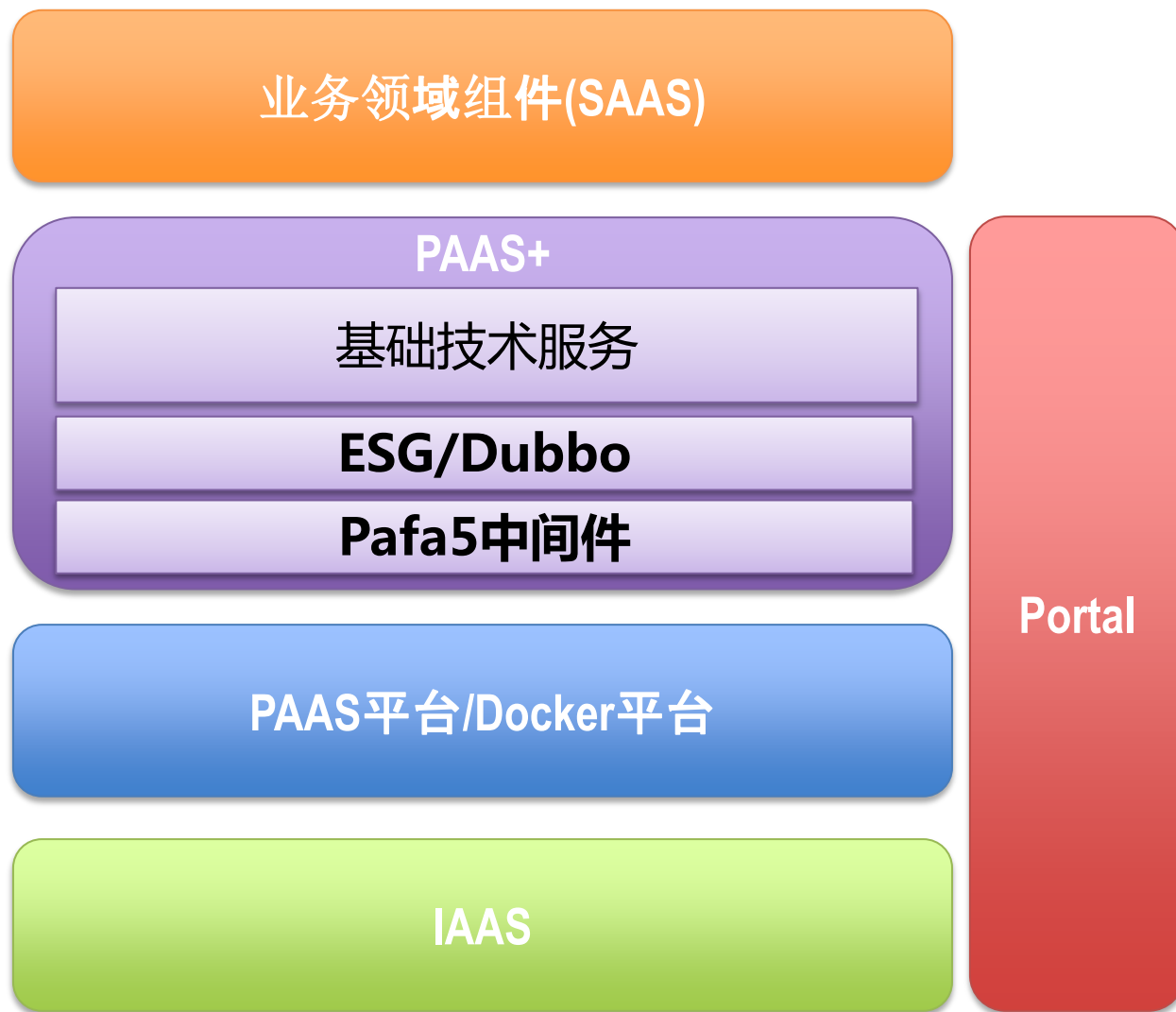
## 3.3 Pafa5中间件的关联关系



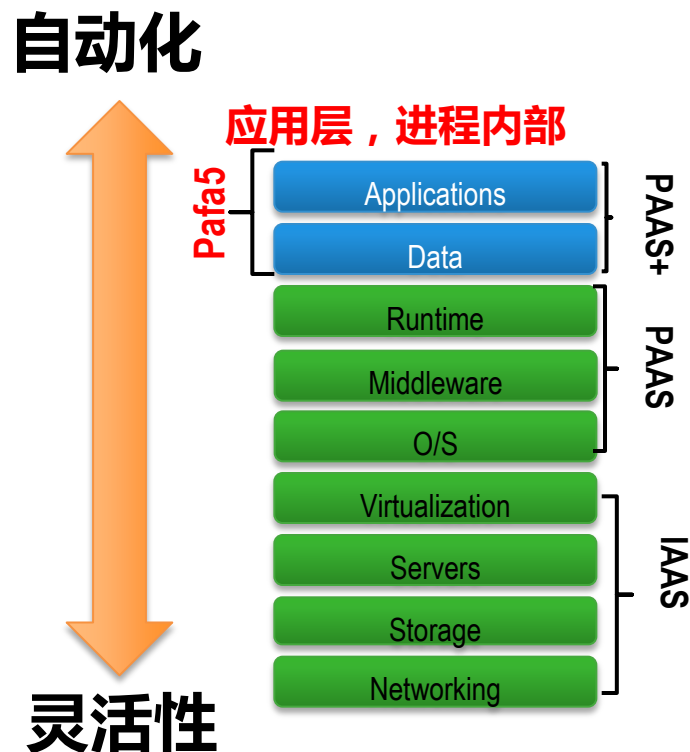
## 4.1 全局控制台：Pafa5在线管控平台



## 5.1 PAAS平台梦想

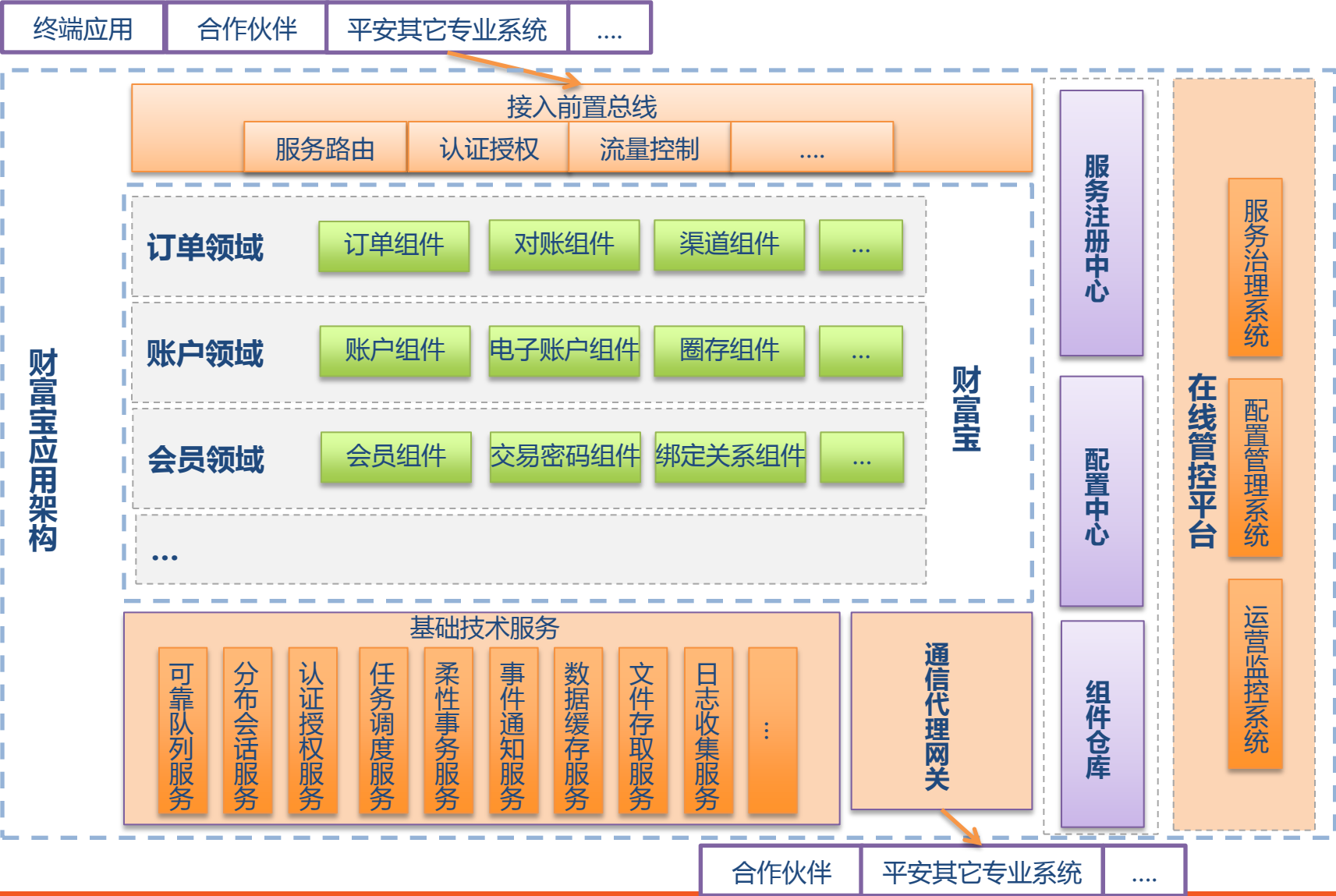


## 5.2 Pafa5与云平台边界





5.3 Pafa5应用架构示例（以信托财富宝为例）



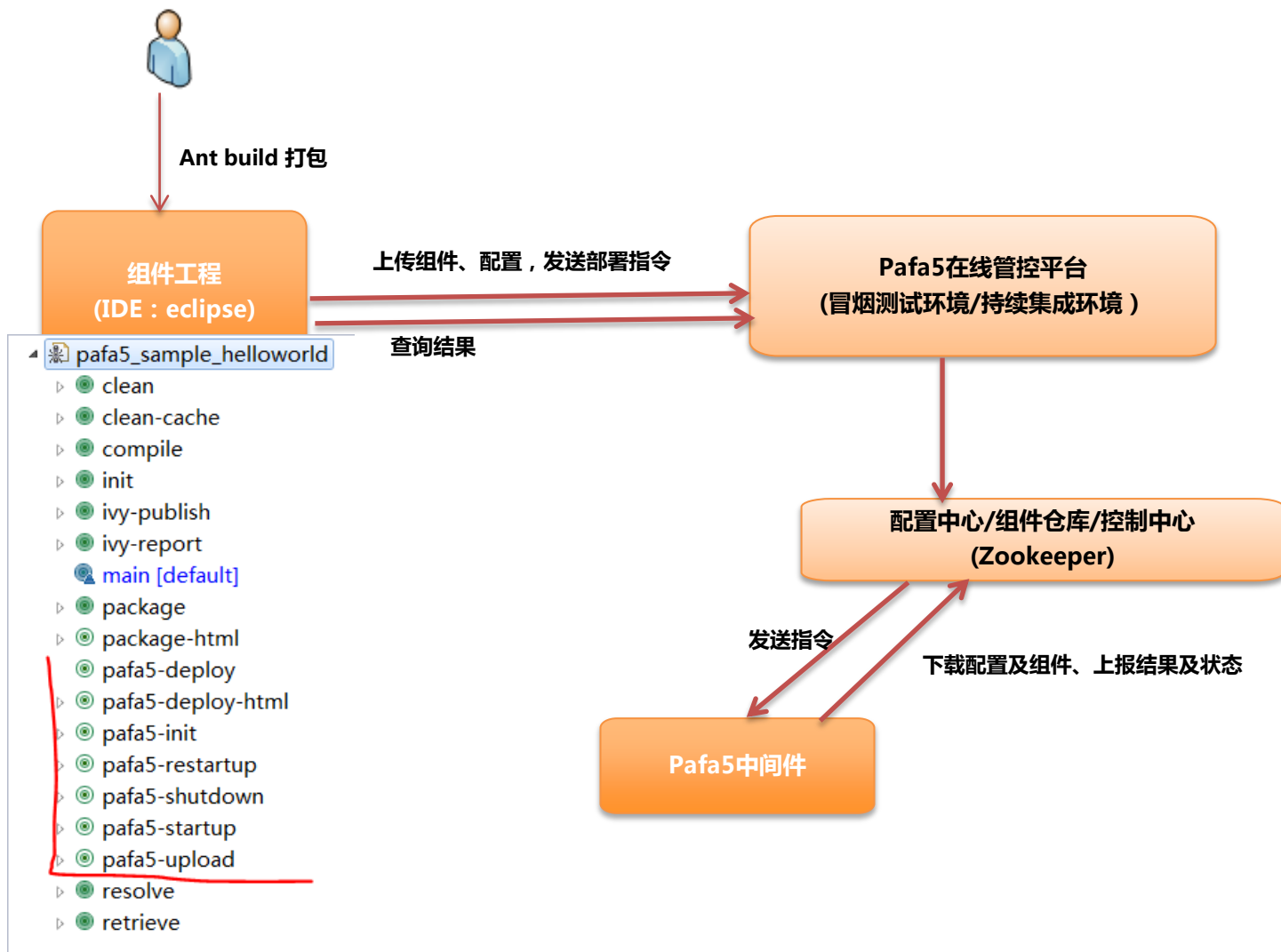
## 5.4 PAAS平台梦想：往远了想

资源池模式，组件热部署  
、热插拔，资源动态分配  
，动态扩容

业务应用跟环境解耦，  
支撑SAAS

异地容灾，异地多活

# 6.1 Dev-Ops : 敏捷过程和精细化运维



## 6.2 ANT一键打包发布

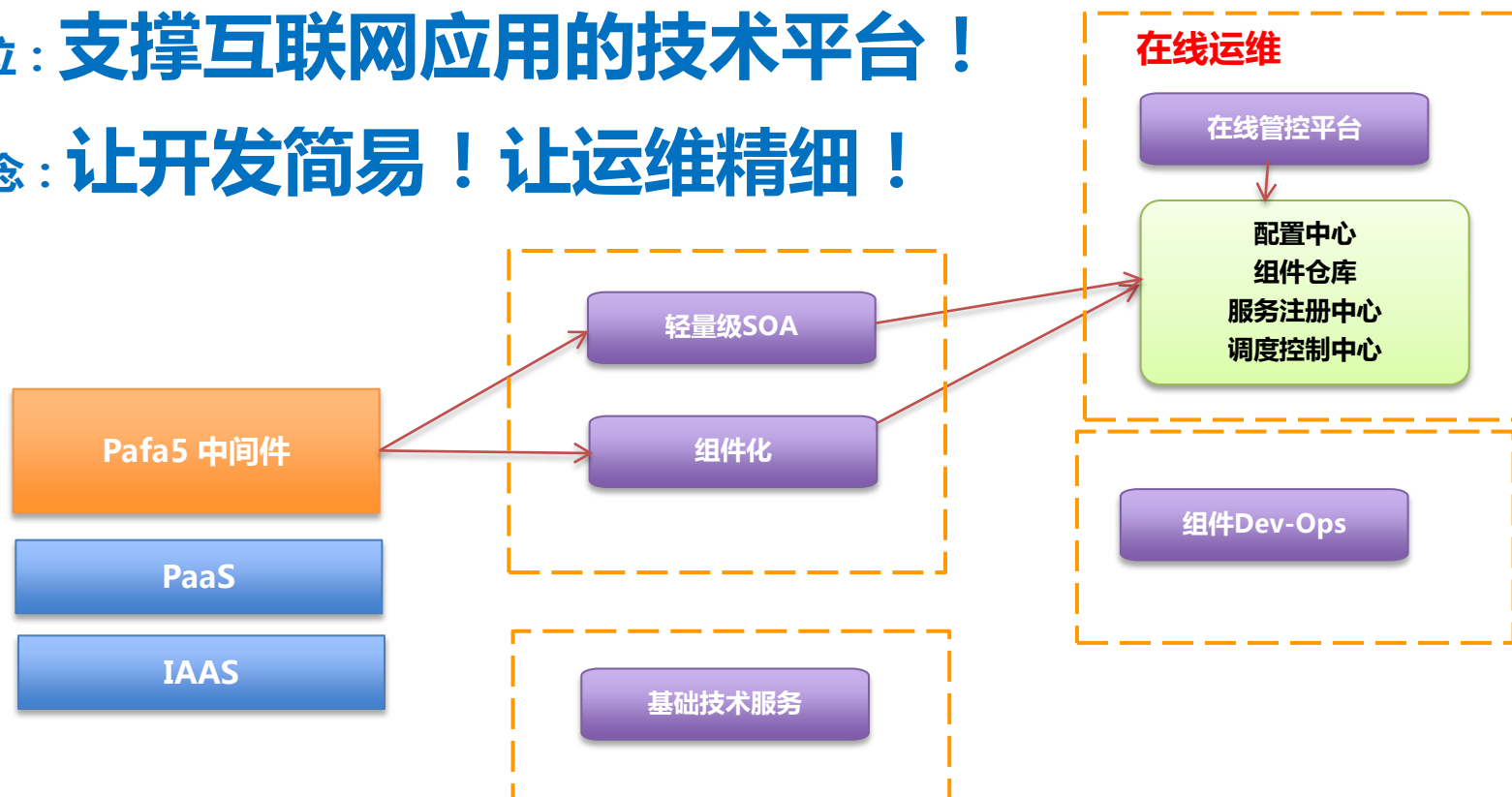
```
[pafa:upload] -----
[pafa:upload] Do login by config=user=root,adminURL=http://nts-stp-monitor-stg.paic.com.cn/admin,passwordFile=D:\Users\lixingnan945/pafa5-deploy/password.txt
[pafa:upload] Do request:http://nts-stp-monitor-stg.paic.com.cn/admin/user/login.do
[pafa:upload] Login Result=true,responseCode=0,responseMsg=success
[pafa:upload] -----
[pafa:upload] Upload file:D:\ws2\pafa5_sample_helloworld\target\paconfigs\sar\pafa5_sample_helloworld.properties, targetGroup:sar started
[pafa:upload] Do request POST http://nts-stp-monitor-stg.paic.com.cn/admin/pizzamgr/uploadConfigItem.do HTTP/1.1
[pafa:upload] Upload file result={"msg":"配置文件未变化。","success":true,"responseCode":"0"}
[pafa:upload] Upload file:pafa5_sample_helloworld.properties ,targetGroup:sar completed.
[pafa:upload] -----
[pafa:upload] Upload file:D:\ws2\pafa5_sample_helloworld\target\pafa5_sample_helloworld.jar, targetGroup:lib started
[pafa:upload] Do request POST http://nts-stp-monitor-stg.paic.com.cn/admin/pizzamgr/uploadConfigItem.do HTTP/1.1
[pafa:upload] Upload file result={"msg":"上传成功!","success":true,"responseCode":"0"}
[pafa:upload] Upload file:pafa5_sample_helloworld.jar ,targetGroup:lib completed.
pafa5-restartup:
[pafa:command] -----
[pafa:command] SendCommand:restartup started, targetSAR:pafa5_sample_helloworld,targetPApp:nts_stp_core.
[pafa:command] Do request http://nts-stp-monitor-stg.paic.com.cn/admin/pizzamgr/sendDictate.do
[pafa:command] Send command completed, result={"dictateId":"569cc323e4b0e675118666f1","success":true,"responseCode":"0","responseMsg":"指令发送成功"}
[pafa:command] QueryCommandResult for 569cc323e4b0e675118666f1, and request count=1.
[pafa:command] CommandResult={"responseCode":"0","instanceSize":3,"completed":false}
[pafa:command] QueryCommandResult for 569cc323e4b0e675118666f1, and request count=2.
[pafa:command] CommandResult={"responseCode":"0","instanceSize":3,"completed":true,"results":[{"status":"2","resultMsg":"failure.","instanceIp":"10.31.171.249"}]}
[pafa:command] Command execute ★★failed★ by Server:10.31.171.249(tm_nts-stp3Server3931),cause:failure..
[pafa:command] Command execute ★★succeeded★ by Server:10.31.171.179(tm-nts-stp1Server10984).
[pafa:command] Command execute ★★succeeded★ by Server:10.31.171.181(tm-nts-stp1Server10985).
[pafa:command] SendCommand execute completed.
pafa5-deploy:
BUILD SUCCESSFUL
```

注：此方式不适用于生产环境！

# 7.1 Pafa5技术平台

产品定位：**支撑互联网应用的技术平台！**

设计理念：**让开发简易！让运维精细！**

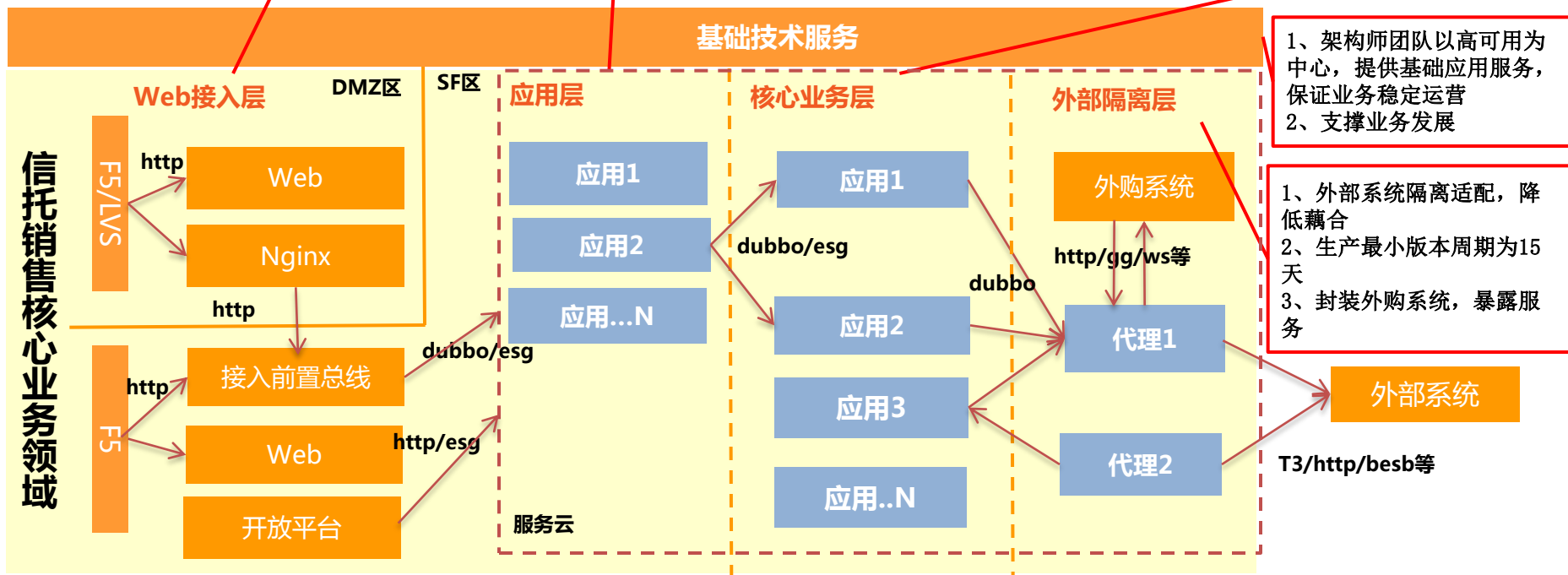


## 7.2 pafa5技术架构示例

- 1、部署html组件，生产最小版本周期为1天
- 2、反向代理及路由业务请求
- 3、暴露指定开放的API接口

- 1、以用户为中心，关注用户体验
- 2、生产最小版本周期为7天
- 3、组合服务，构成用户体验良好的产品

- 1、以业务为中心，重点关注业务本身
- 2、生产最小版本周期为15天
- 3、接口比较稳定，通用，复用程度高



### 7.3 Pafa5技术平台能力视图



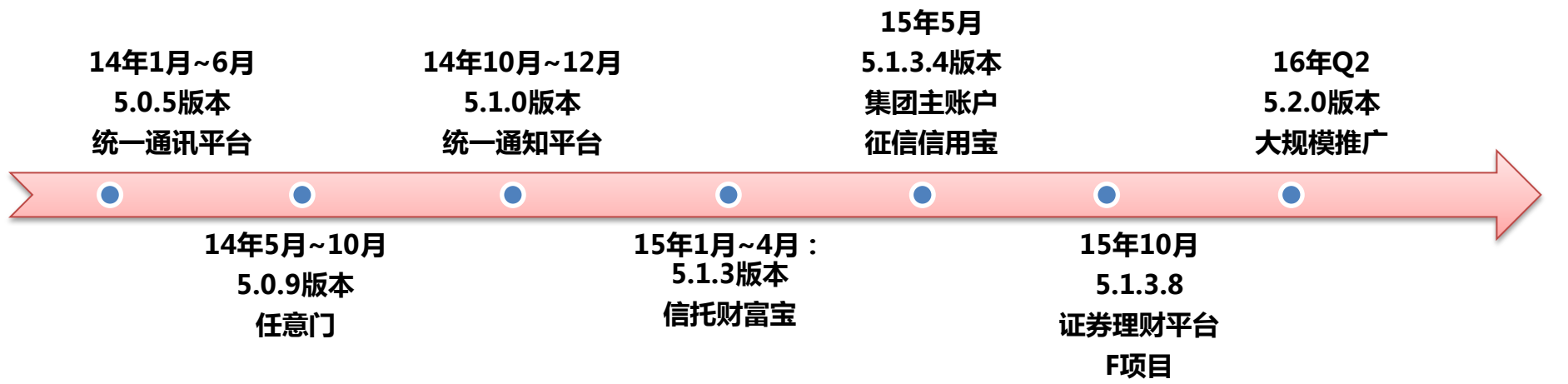
特点	组件化	轻量级SOA	敏捷过程	动态伸缩架构
局限性	技术能力能强的架构团队	Java平台		
目标&意义	快速迭代交付	互联网动态架构	开放及复用能力	

### 目标 意义

1. 微服务：面向服务（SOA），构建“服务云”和“开放平台”
2. 组件化：小微应用，组件化开发和部署，减小部署粒度，避免全局发布，减小测试工作量。提升复用度和业务的内聚性，降低开发成本。
3. Dev-Ops：简化开发过程，让开发更简单，提高开发效率和持续集成能力，以提升团队的需求交付能力
4. 构建基础技术服务组件，提供通用技术解决方案。
5. 云计算架构：资源池化，动态运行时，热部署热插拔，所有资源动态按需分配，无单点瓶颈，构建支撑互联网应用的技术平台



# 7.5 发展历程&规划



Pafa5-下一步

# 精细化运维！ DEV-OPS



# Thank you!

