



GOPS2017  
Shenzhen




# 全球运维大会


2017



深圳站

指导单位： 数据中心联盟  
Data Center Alliance

主办单位： 高效运维社区  
GreatOps Community

 开放运维联盟  
OOPSA Open OPS Alliance



# 如何用调度编排解决企业发布变更之痛

杨文兵 腾讯蓝鲸运营负责人

## 个人简介

- ◆ 腾讯高级工程师，先后担任腾讯游戏DNF、CF、七雄争霸等不同类型的游戏的业务运维，运营规划及运维负责人。
- ◆ 蓝鲸体系首批DevOps，在蓝鲸中开发过多套支撑工具和运营系统。
- ◆ 目前是蓝鲸运营负责人。



杨文兵

# 目录



1

传统发布变更之痛

2

初识调度编排

3

如何构建调度编排的5步曲

4

收益

# 小明的故事



6年前，小明毕业后加入了腾讯，成为了一名**游戏运维**。



```
Terminal
Microsoft PowerShell Core\FileSystem -> /media/john/4282F11882F18965
Microsoft PowerShell Core\Function -> Function
PS C:\> 0x23 * 18
ERROR: Lexical Error: Line 1: token " * " was not expected
PS C:\> 0x23 * 18
51
PS C:\> get-psdrive
Microsoft PowerShell Core\Environment -> Env
Microsoft PowerShell Core\Variable -> Variable
Microsoft PowerShell Core\Alias -> Alias
Microsoft PowerShell Core\FileSystem -> /sys
Microsoft PowerShell Core\FileSystem -> /proc
Microsoft PowerShell Core\FileSystem -> /dev
Microsoft PowerShell Core\FileSystem -> /dev/pts
Microsoft PowerShell Core\FileSystem -> /run
Microsoft PowerShell Core\FileSystem -> /
Microsoft PowerShell Core\FileSystem -> /sys/fs/fuse/connections
Microsoft PowerShell Core\FileSystem -> /sys/kernel/debug
Microsoft PowerShell Core\FileSystem -> /sys/kernel/security
Microsoft PowerShell Core\FileSystem -> /run/lock
Microsoft PowerShell Core\FileSystem -> /run/shm
Microsoft PowerShell Core\FileSystem -> /run/user
Microsoft PowerShell Core\FileSystem -> /mnt/MyMedia
Microsoft PowerShell Core\FileSystem -> /mnt/MoreMedia
Microsoft PowerShell Core\FileSystem -> /proc/sys/fs/binfmt_misc
Microsoft PowerShell Core\FileSystem -> /run/user/john/gvfs
Microsoft PowerShell Core\FileSystem -> /media/john/openSUSE-12.3-DVD-x86_64@110
Microsoft PowerShell Core\FileSystem -> /media/john/4282F11882F18965
Microsoft PowerShell Core\Function -> Function
```



小明眼中的游戏运维



现实之中的游戏运维



- 80%时间都在重复着几件事——发布变更
- 每周3次的例行发布几乎都需要通宵才能搞定
- 业务架构越来越复杂，发布变更的流程也越来越长，为了尽量不漏操作，小明每次发布时准备了几页A4纸，详细记录着每一步的操作。





**作为运维的你，是否也跟小明有着同样的烦恼？**





## 运维 发布变更 痛点

- 操作过程不连贯、无法固化流程
- 人肉在各系统间徘徊，切换等待时间过长
- 操作步骤过多，极易出现人为操作失误
- 大规模、批量化部署拼的是运维体力
- 运维经验程度决定了发布变更的成功率和效率
- 新人很难快速上手，交接难度高
- 长年累月的重复性劳动



现实与理想之间，你其实只差一

个 **调度编排**

调度编排能给我们带来什么？

操作自助化

Web执行降低操作门槛

可视化模板配置

**跨系统调度自动化**

发布变更流程固化

去Console化安全操作

批量并发执行

可视化流程交接

# 目录

1 传统发布变更之痛

➔ 2 初识调度编排

3 如何构建调度编排的5步曲

4 收益

## 何为调度编排？

可视化  
配置模板

业务配置  
全局变量  
节点变量

任务引擎

根据模板创建任务  
分支、重试&出错机制  
更新维护资源栈

资源栈

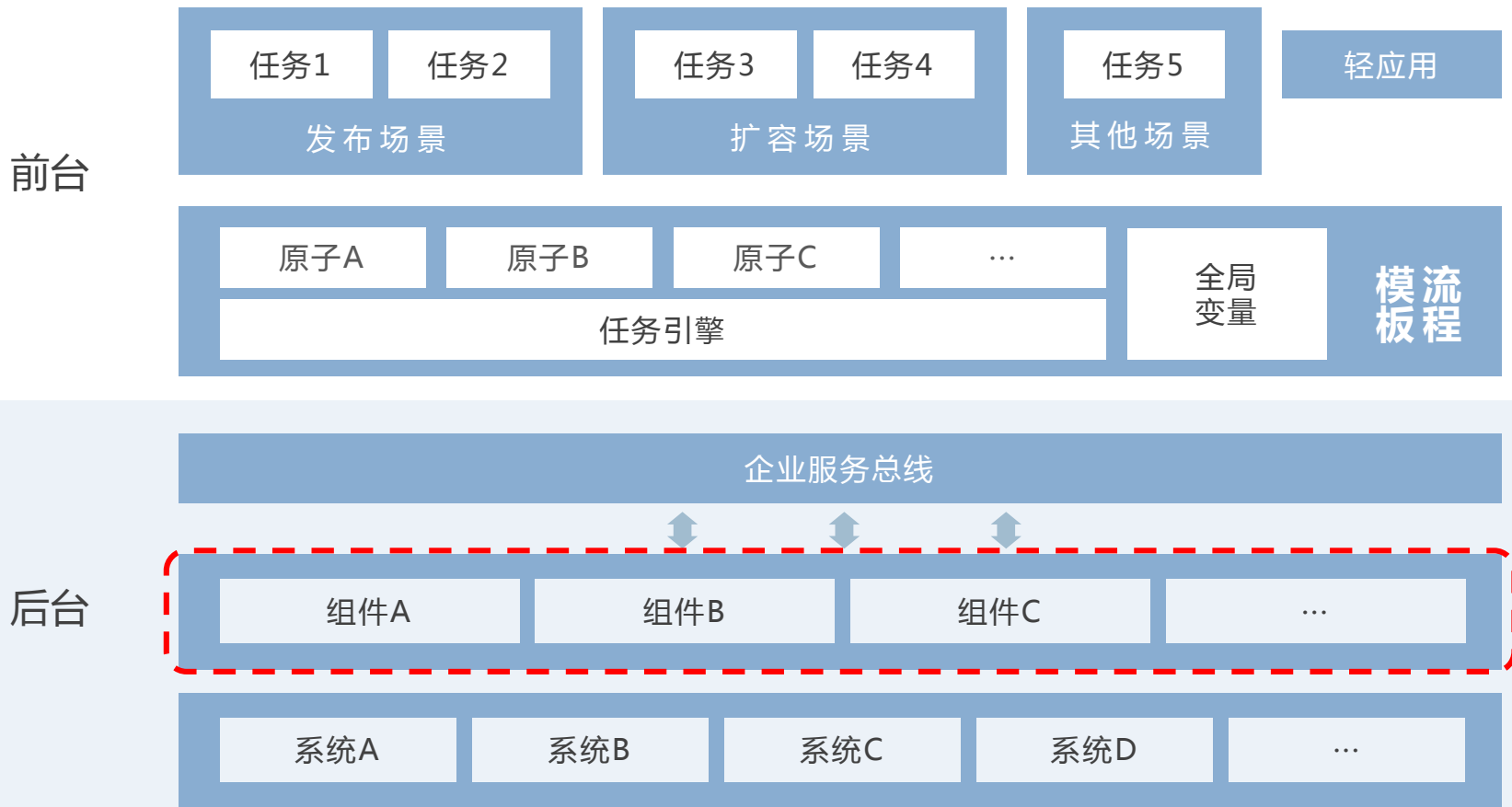
可视化原子节点  
IaaS资源

# 调度编排中的一些概念

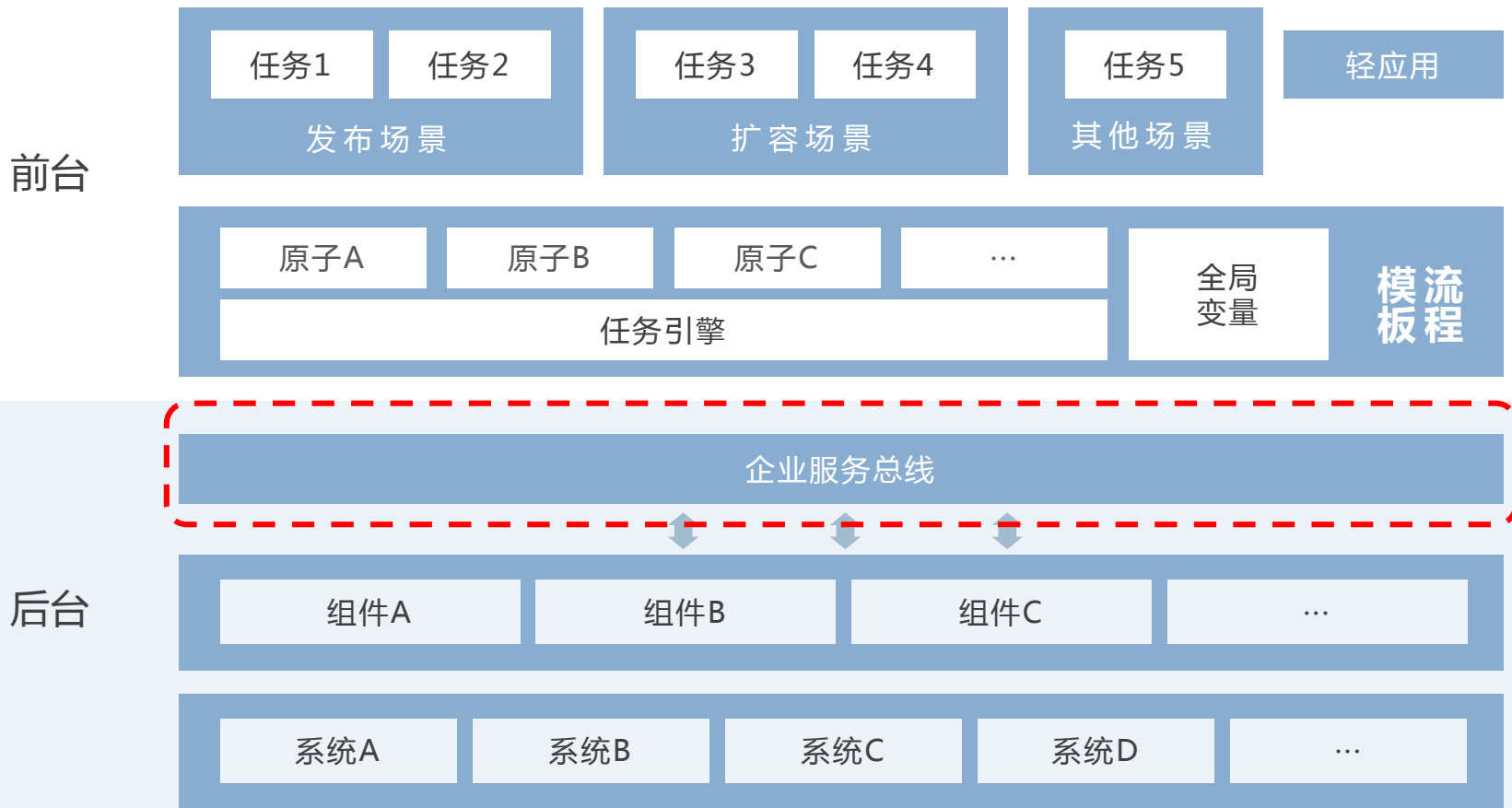




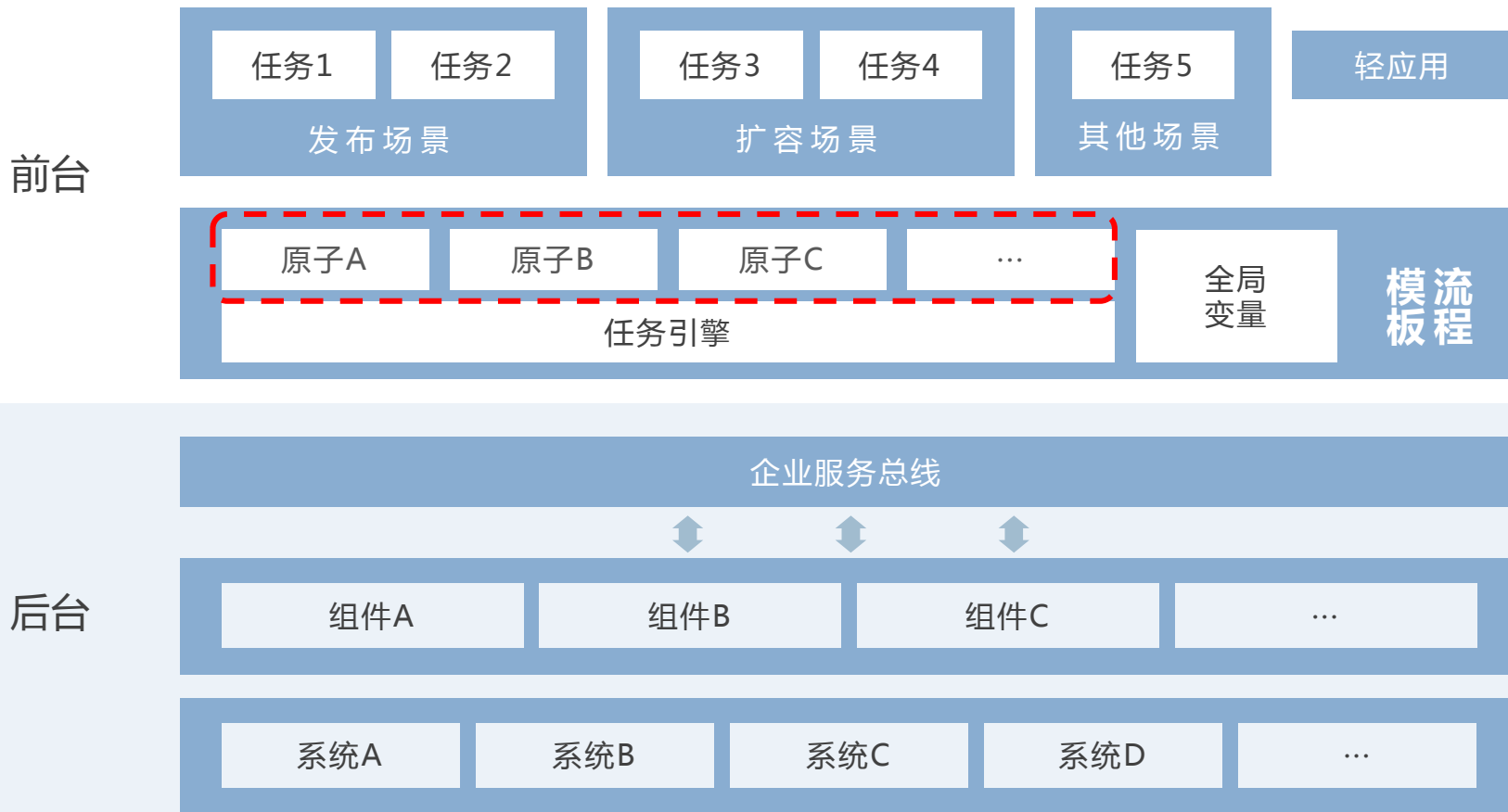
# 调度编排框架



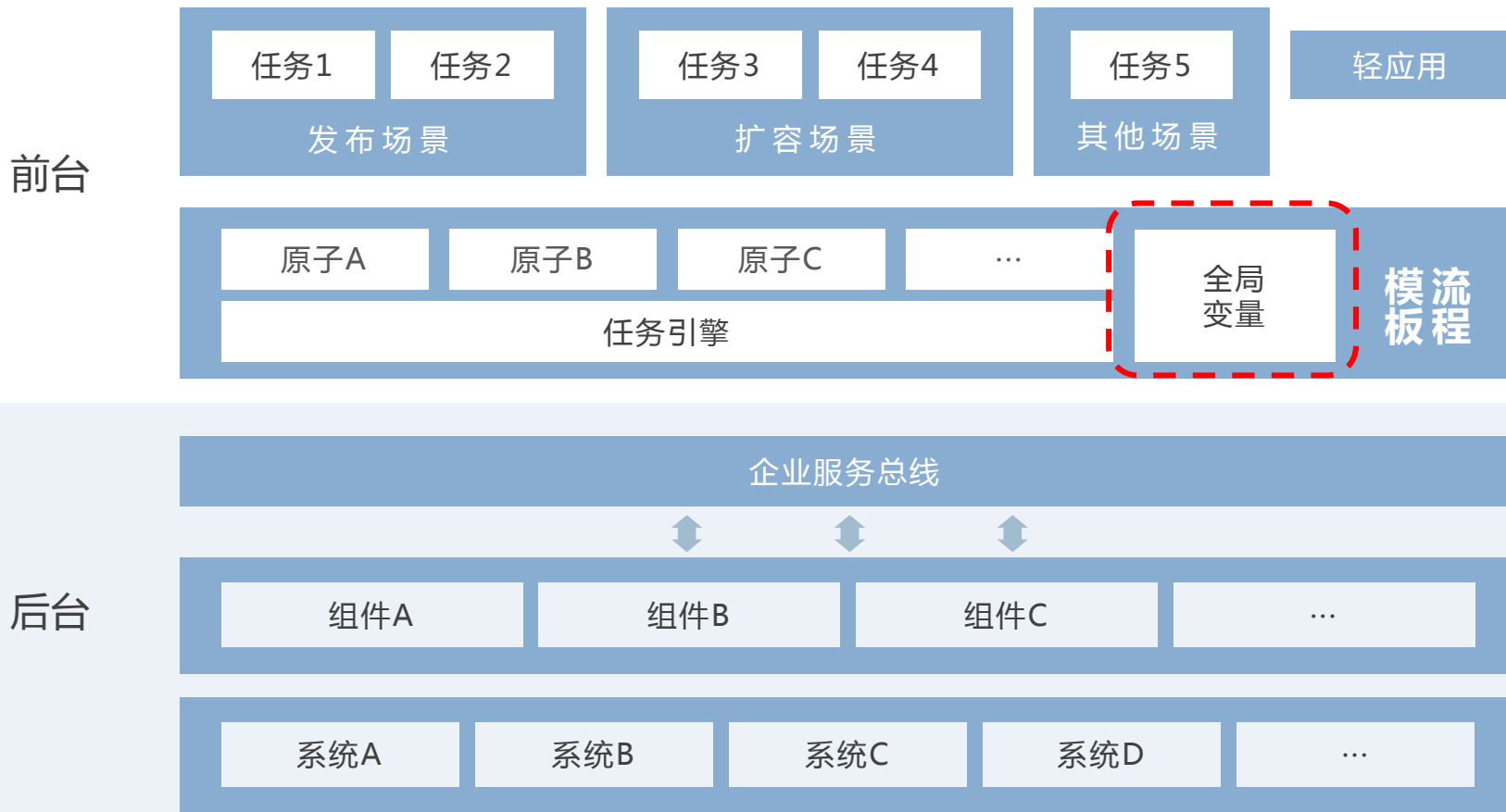
# 调度编排框架



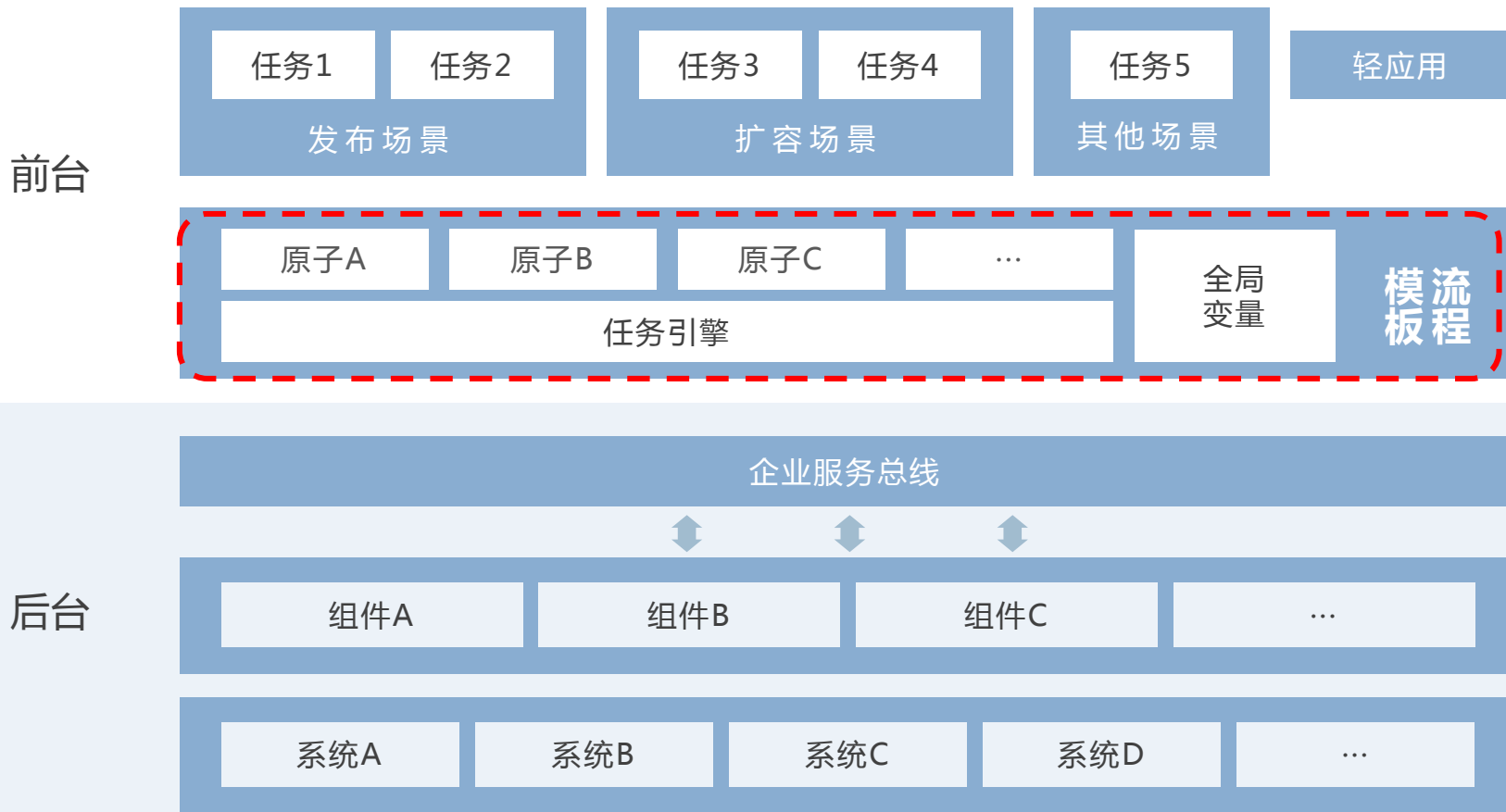
# 调度编排框架



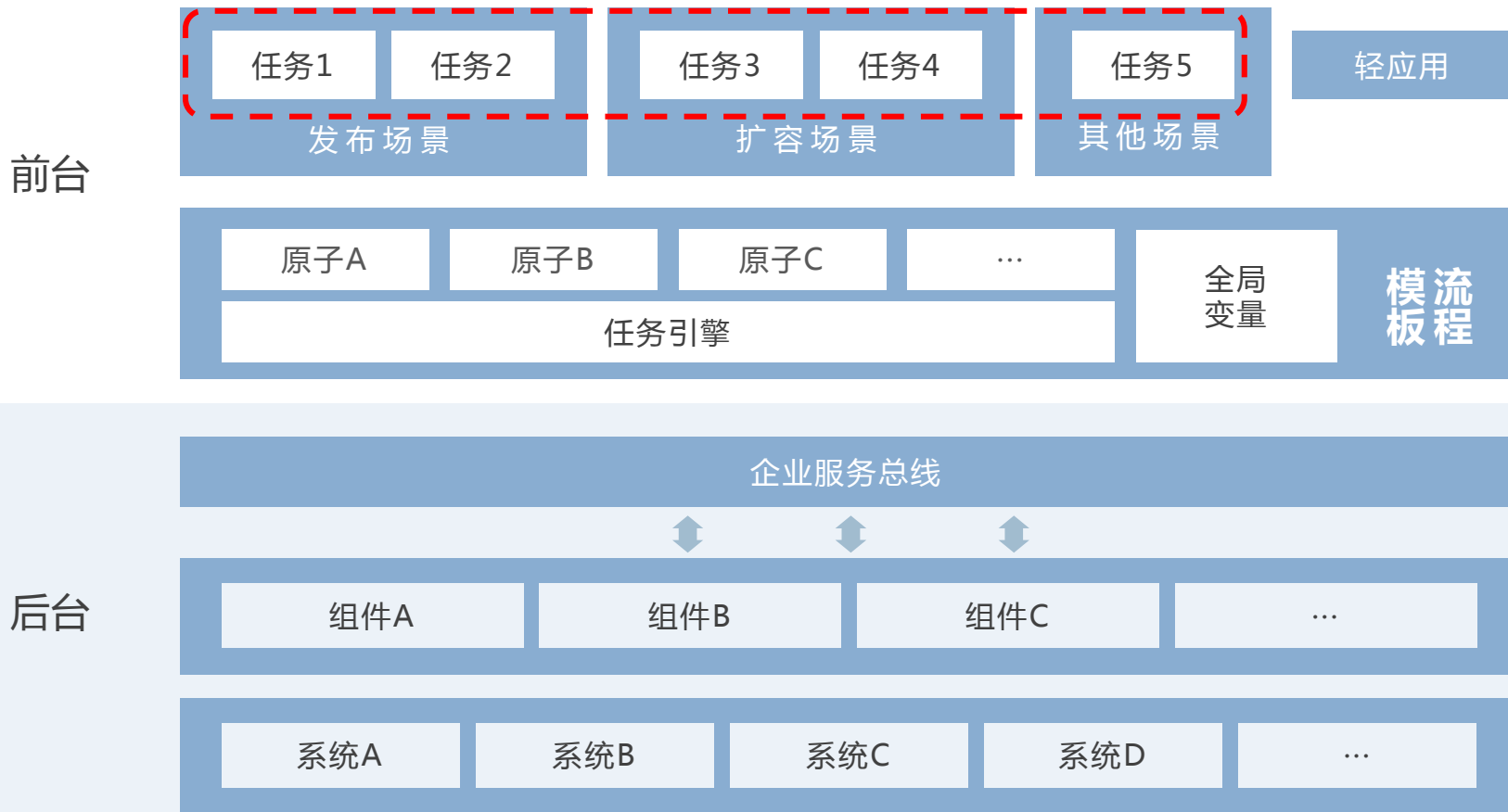
# 调度编排框架



# 调度编排框架



# 调度编排框架





## 调度编排服务的群体



运维

开发

测试

产品

外包

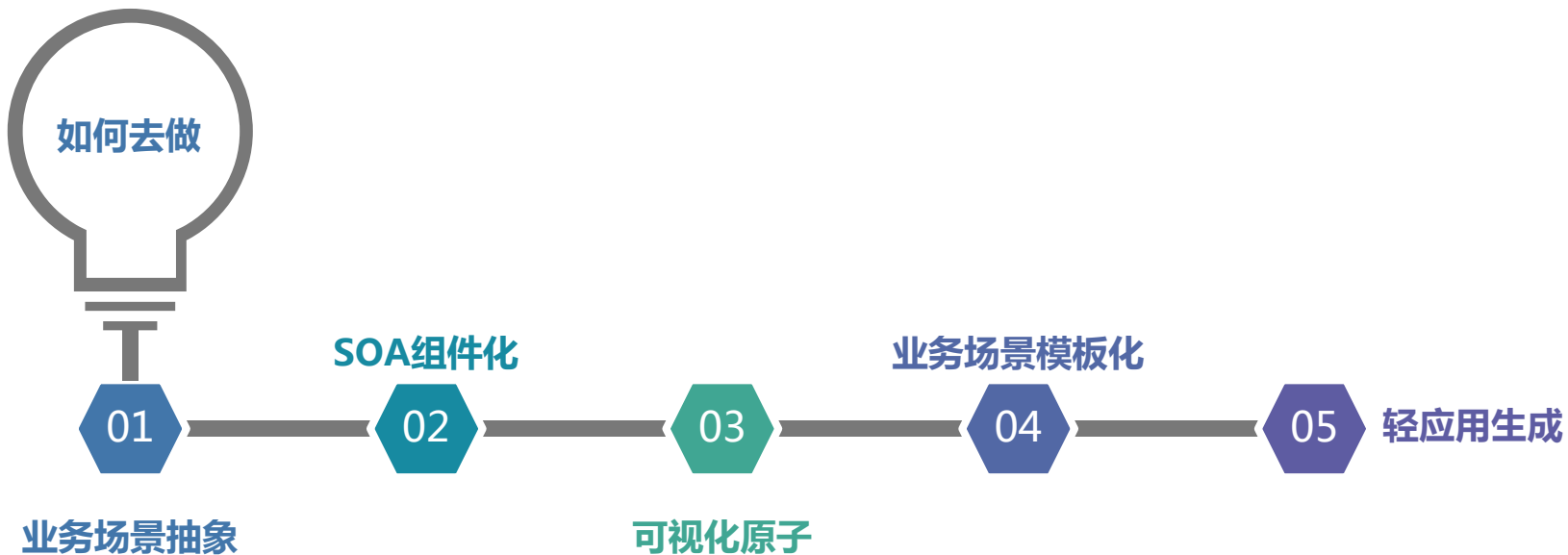
# 目录

1 传统发布变更之痛

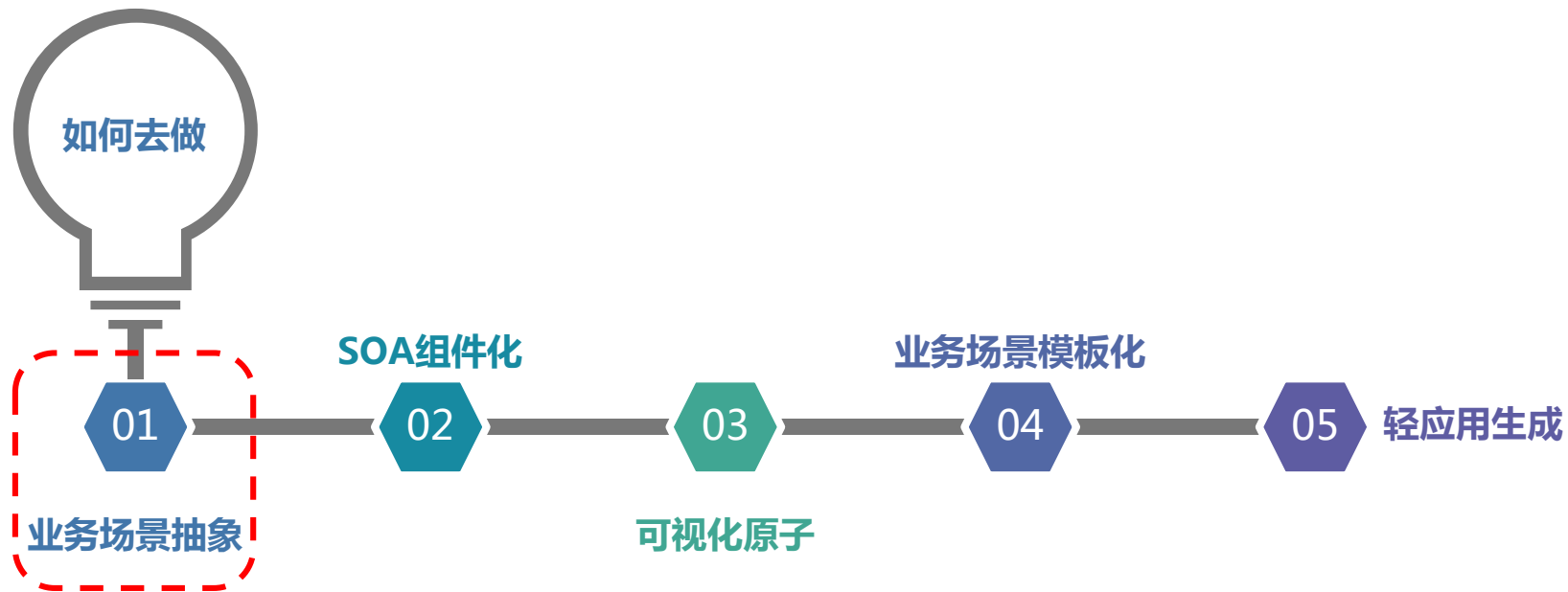
2 初识调度编排

➔ 3 如何构建调度编排的5步曲

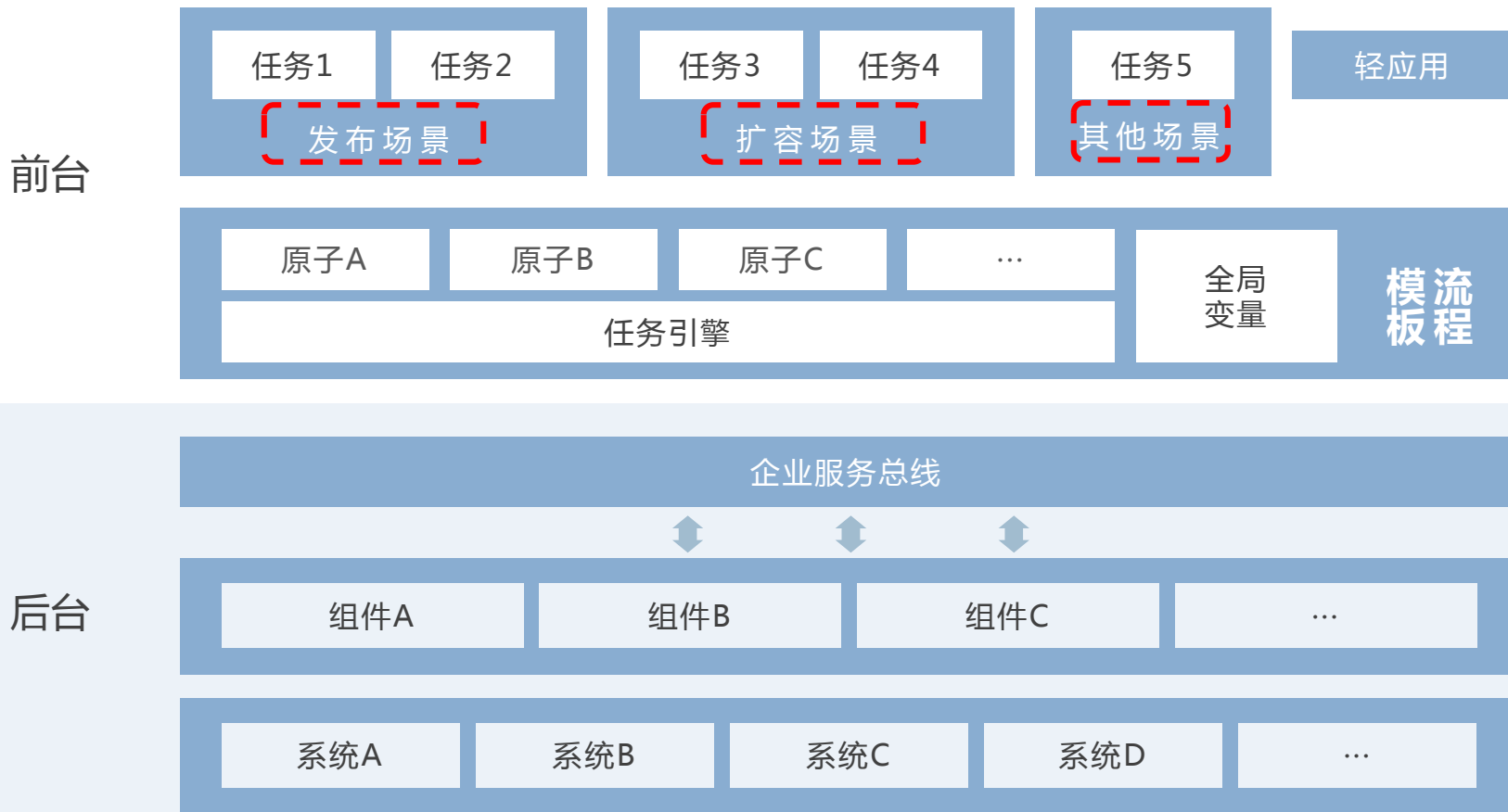
4 收益



## Step 1 : 针对业务场景抽象模型



# 调度编排框架





手游、端游、页游、平台类游戏，架构差异大

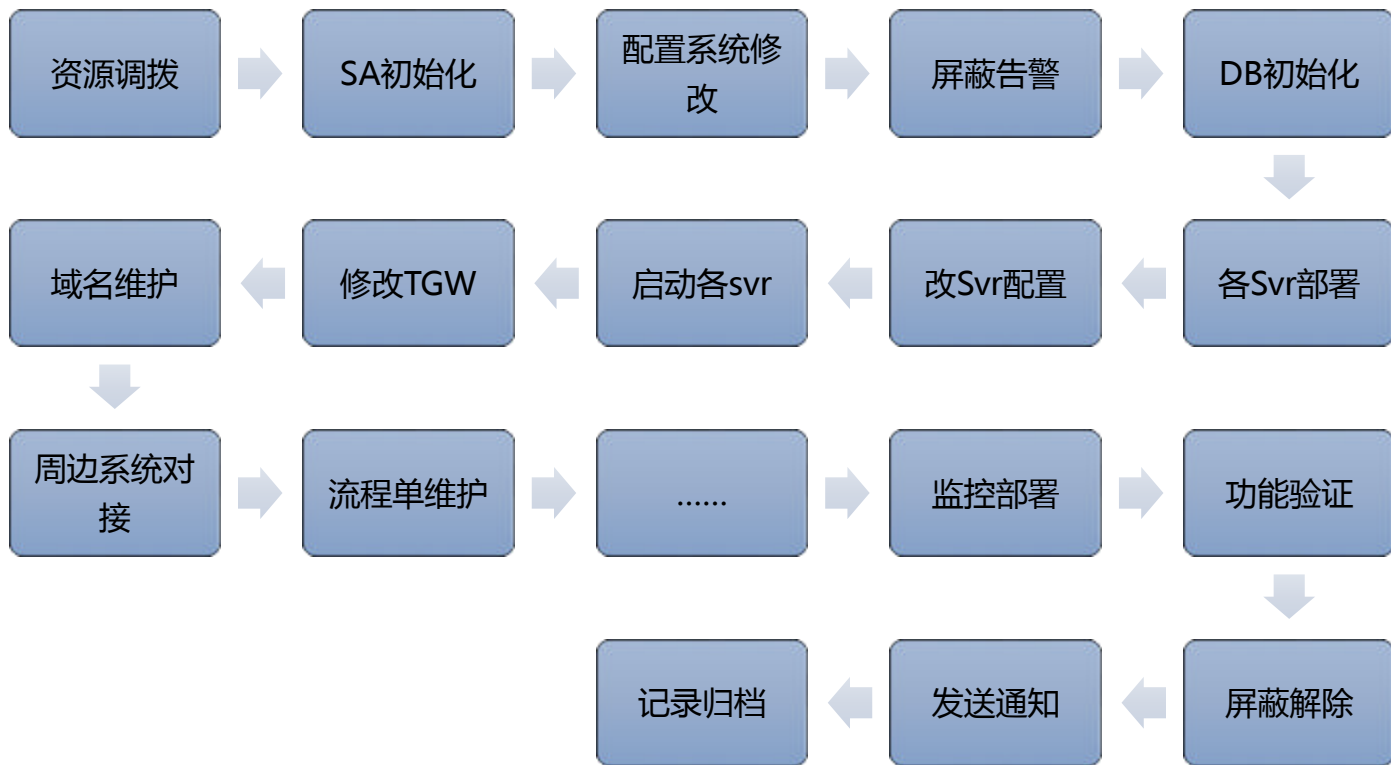
全球不同的开发商，语言及运行平台差异大

相互独立，相同场景流程差异大

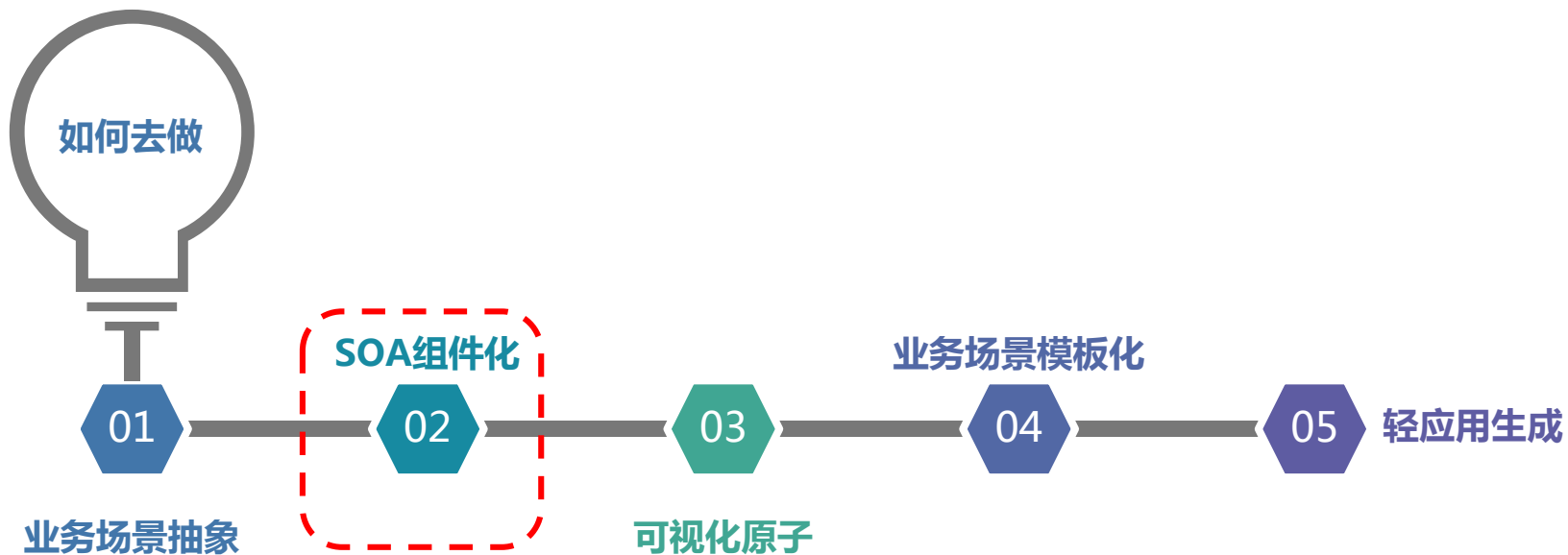
同业务下同一场景的操作流程差异大



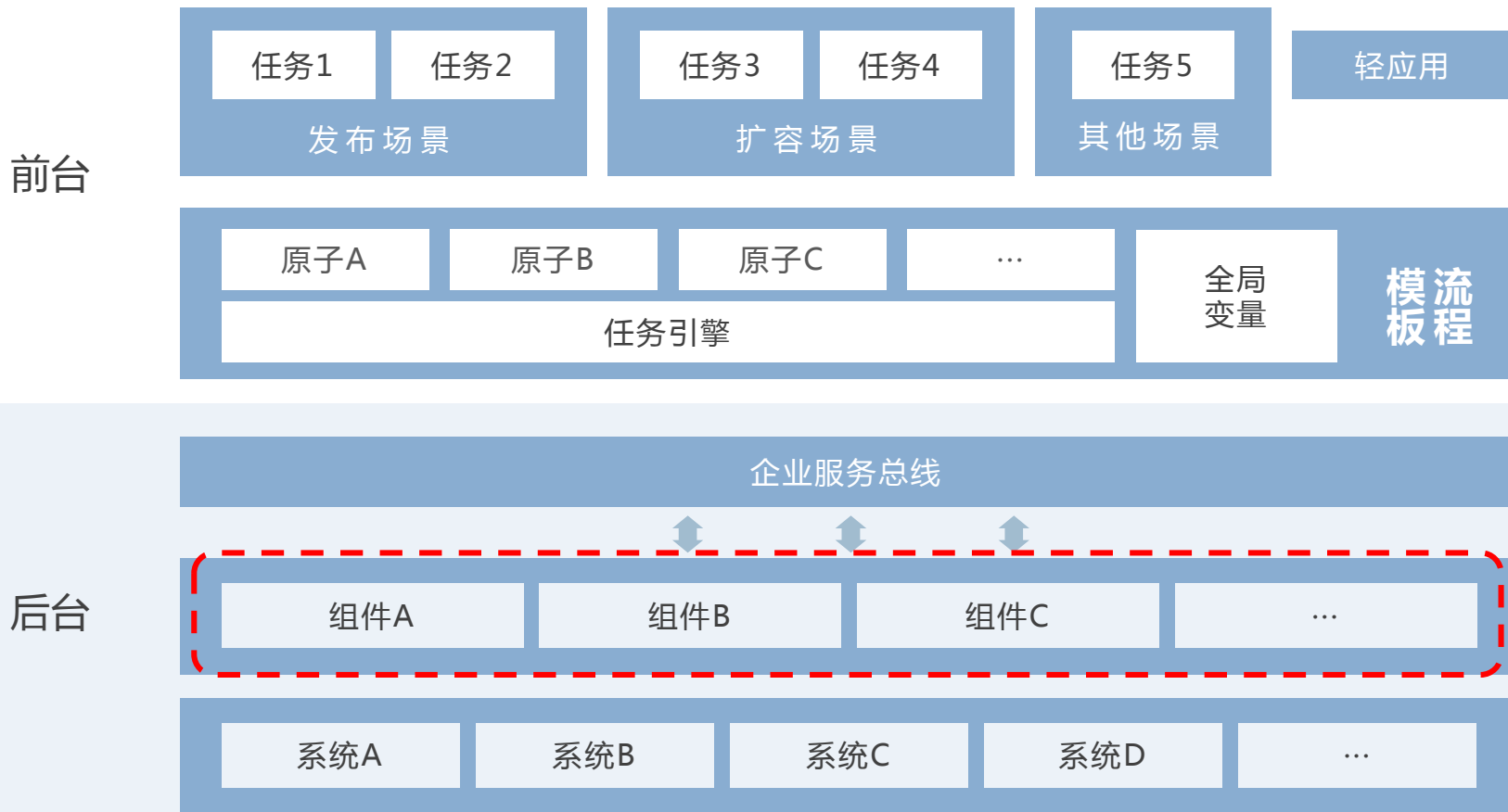
# 腾讯游戏新开大区场景抽象



## Step 2 : 以SOA理念对接企业内的系统 ( 组件接入 )



# 调度编排框架

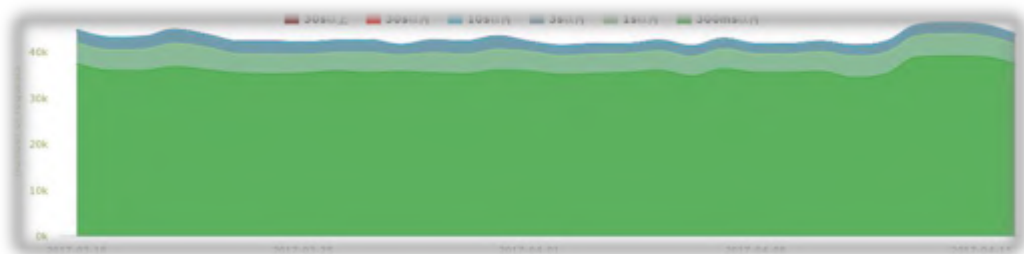


## 组件接入



- 1、统一平台管理、托管和维护
- 2、统一的用户认证和用户鉴权
- 3、页面自助接入
- 4、统一协议

## 组件接入后的质量保证

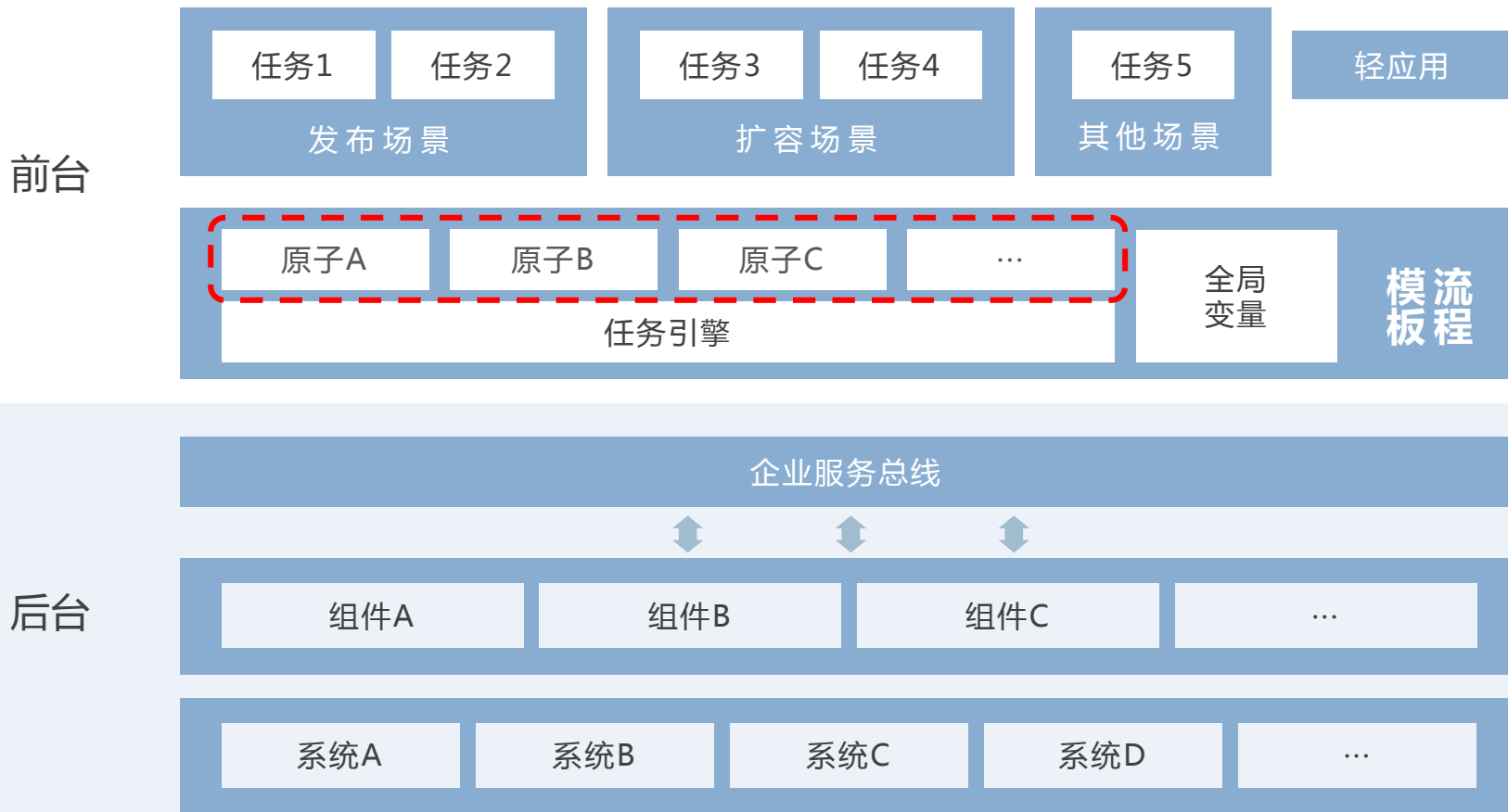


- 1、组件调用出错监控告警
- 2、组件稳定性、性能、热度周报
- 3、频率控制、过载保护
- 4、统一信息推送

## Step3 : 可视化原子接入和低成本开发



# 调度编排框架





# 平台已有原子，可视化配置



## 作业平台

执行类

脚本执行  
文件传输  
本地文件上传  
.....



## 配置平台

资源类

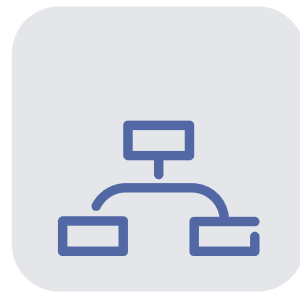
主机所属集群、模块修改  
自定义属性修改  
故障机替换  
.....



## 通知、暂停、定时

基础类

邮件、短信、微信通知  
Timer计时器  
暂停文本步骤  
.....



## ITIL

流程类

发布变更单创建  
工单审批  
结单  
.....

## 自定义原子，低成本开发

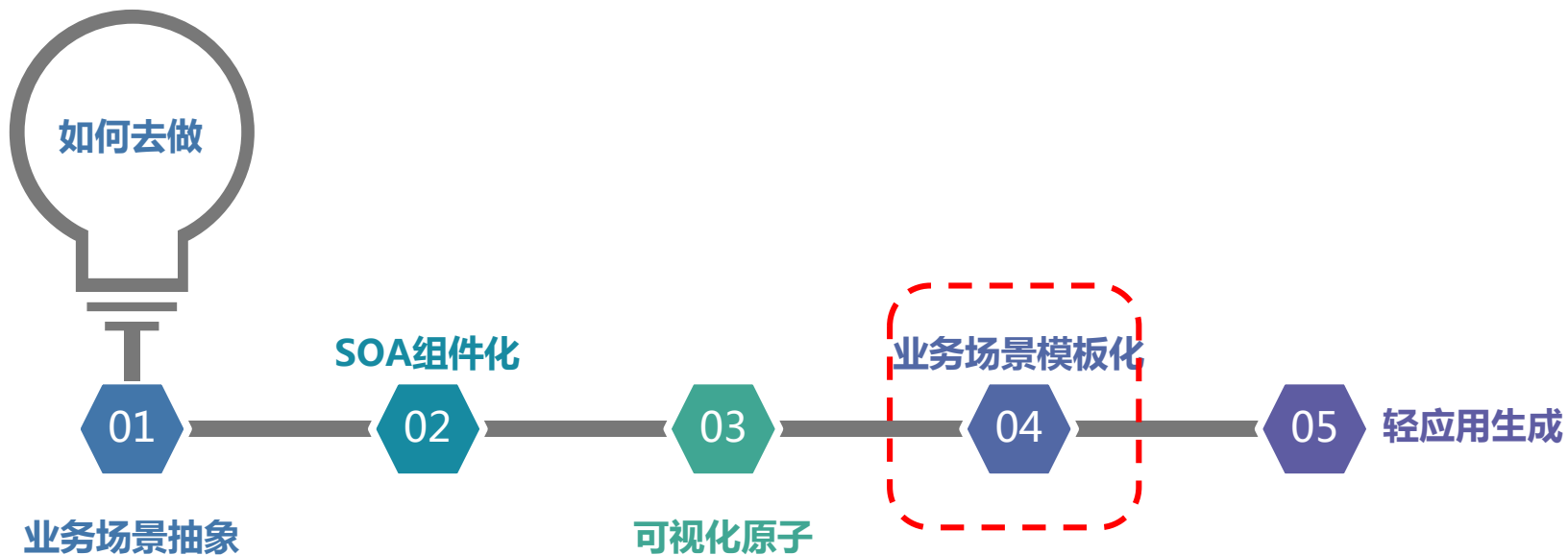
**开发方式：**  
配置

**涉及语言：**  
仅python,将前端的  
html、js全部转化为后  
台配置python类的对象

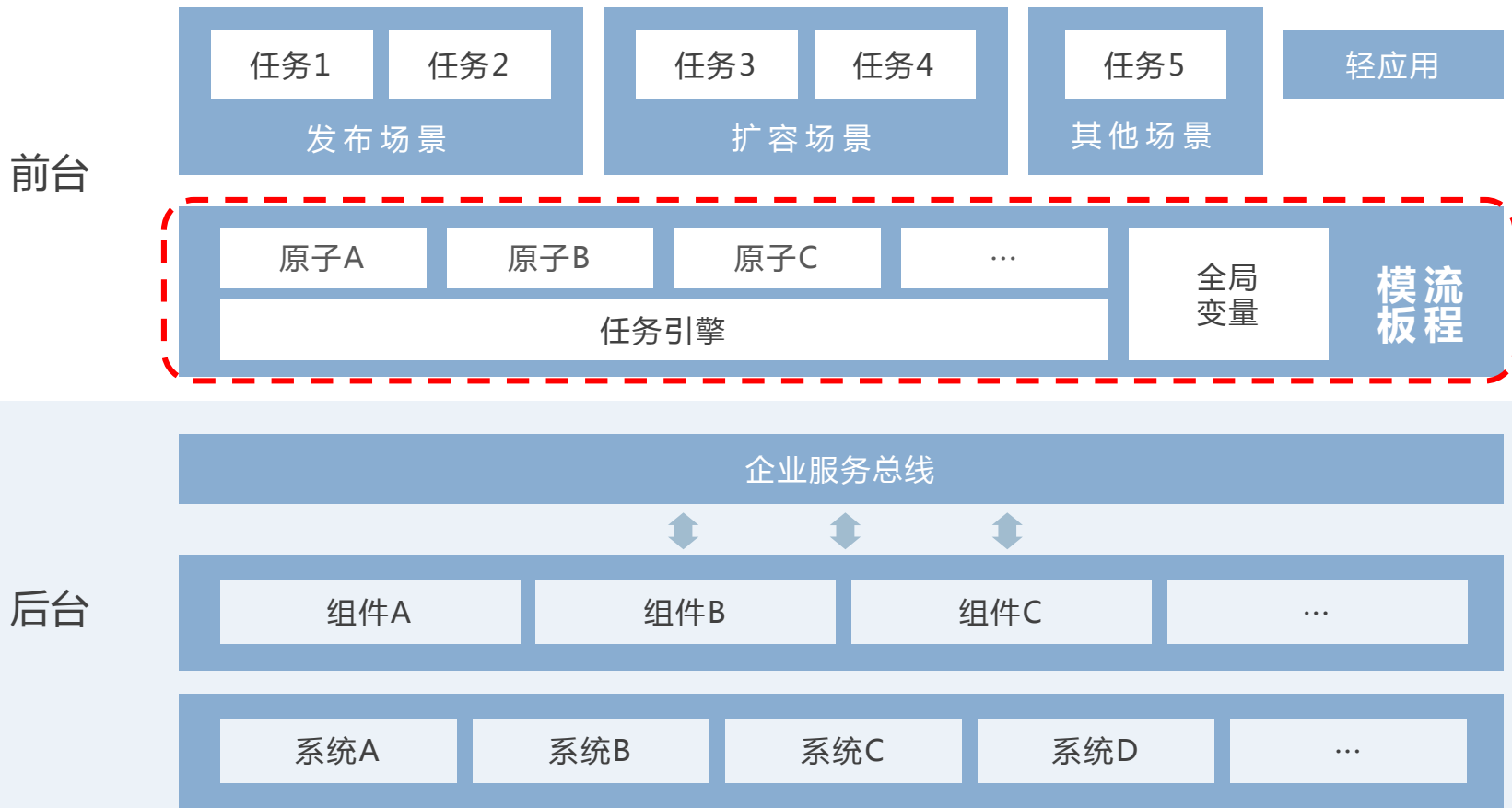
**代码量：**  
简单的原子只需50+行  
代码

**开发耗时：** 10+min

## Step 4 : 将业务梳理好的场景程固化为模板



# 调度编排框架



1 提交职能化需求

2 职能化认领

3 操作执行

4 职能化结单

5 运维评价

推送或定时默认通知方式:  全选  Email  RTX  短信  语音  微信

定时默认时长(单位:分钟):

推送或定时默认通知人员: 分组:  附加:

步骤名称: 发布前准备

串行 并行

完成或暂停

节点组件	节点别名	操作
ODM-变更提单到GCS	ODM-变更提单到GCS	<input type="checkbox"/> 忽略错误 <input checked="" type="checkbox"/> 可选项
GCS-变更提单(MySQL)	GCS变更提单(DB变更)	<input type="checkbox"/> 忽略错误 <input checked="" type="checkbox"/> 可选项
TNM2-各管屏蔽	全服大区各管屏蔽	<input checked="" type="checkbox"/> 忽略错误 <input type="checkbox"/> 可选项
Dobs-任务执行(CC传参)	脚本提取到达挂机	<input type="checkbox"/> 忽略错误 <input type="checkbox"/> 可选项
Version-分发脚本文件	Version-分发脚本文件【全服】	<input type="checkbox"/> 忽略错误 <input type="checkbox"/> 可选项
Dobs-任务执行(CC传参)	全服大区脚本文件检查	<input type="checkbox"/> 忽略错误 <input type="checkbox"/> 可选项
BK-文本分面(暂停)	【暂停】准备工作完毕	<input checked="" type="checkbox"/> 忽略错误 <input checked="" type="checkbox"/> 可选项

新增组件

步骤名称: 发布前检查版本【发布前】

串行 并行

节点组件	节点别名	操作
Timer-定时器	Timer-定时器(7:45开始执行)	<input type="checkbox"/> 忽略错误 <input checked="" type="checkbox"/> 可选项
Dobs-任务执行(CC传参)	[Jobs-【检查基次大区脚本和拉取】	<input type="checkbox"/> 忽略错误 <input type="checkbox"/> 可选项

名称	KEY	
发布范围	{SetName_LIST}	🔒
大区ID列表	{WORLD_ID_LIST}	🔒
SET中文名	{SetCheName_LIST}	🔒
发布起始时间	{task_start_time}	🔒
发布结束时间	{task_end_time}	🔒
此次版本发布的文件列表	{patch_file}	×
OOM-对比上传xml文件	{gcs_odm_file}	×
GCS_DB变更文件	{gcs_sql_import_sql_file}	×
脚本文件	{_variable_files}	×
解面时间	{start_end_time}	×
Timer-定时器-凌晨(7点45)	{_timer_time_picker}	×
Timer-定时器-执行(8点)	{_timer_time_picker79108}	×

添加变量

1

### 发布前准备

- 正式版大区告警屏蔽
- 在线告警屏蔽
- 正式版大区发布版本提取
- 逐鹿天下跨服战发布版本提取
- 问鼎中原跨服战发布版本提取
- GCS变更提单

II 步骤执行完成后暂停

2

### 大区停服

- 正式版大区停服
- 逐鹿天下跨服战进程停服
- 问鼎中原跨服战进程停服

II 步骤执行完成后暂停

3

### 文件更新及DB变更

- 执行GCS变更单据
- **手动执行部分大区bug修复的GCS变更单据**
- 正式版大区旧版本备份
- 正式版大区版本更新
- 逐鹿天下跨服旧版本备份
- 逐鹿天下跨服版本更新
- 问鼎中原跨服战旧版本备份
- 问鼎中原跨服战版本更新

II 步骤执行完成后暂停

4

### 大区起服

- 前台资源更新for回滚
- 逐鹿天下跨服战进程启动
- 问鼎中原跨服战进程启动
- 正式版大区启动

II 步骤执行完成后暂停

5

工单节点  
 校验节点  
 交互节点  
 锁控节点  
 配置节点  
 探测节点  
 延迟节点

- 逐鹿天下跨服战大区发布是否成功
- 解除在线告警屏蔽
- 逐鹿天下跨服战发布成功
- 问鼎中原跨服战发布成功
- 逐鹿天下跨服战发布后暂停
- 问鼎中原跨服战发布后暂停

.....

# 一切皆模板

新建模板						
id	模板名称	环境类型	更新时间	更新人	所属场景	模板状态
8202	20170324临时重刷全区+限时跨服【5052大区需将zoneid改为非5052，执行完后再改回来】	全区环境	2017-03-24 17:10:52		自定义	已生效
8201	20170324临时重刷全区和限时跨服战	全区环境	2017-03-24 15:45:12		发布	已生效
8182	【不load数据】版本测试服更新（含跨服战更新，不删除旧版本）	测试环境	2017-03-22 10:16:25		自定义	已生效
8181	【不load数据】版本测试服更新（含跨服战更新）（带删除旧版本）	测试环境	2017-03-21 18:32:43		自定义	已生效
8160	20170317临时使用	正式环境	2017-03-17 17:35:57		自定义	已生效
8089	game扩容	正式环境,测试环境,全区环境,灰度环境	2017-03-10 18:04:52		扩容	已生效
8088	【 】问鼎中原停服更新	灰度环境	2017-03-10 17:20:25		发布	已生效
7850	test只作为测试使用	测试环境	2017-03-08 19:22:44		自定义	已生效
7602	全服和跨服战的前台资源更新（职能化）	全区环境	2016-12-26 18:51:02		发布	已生效
7584	全服大区停服和sql变更（职能化）	全区环境	2016-12-22 22:11:00		发布	已生效
7583	全服大区停服sql变更（职能化）	全区环境	2016-12-22 21:49:23		发布	已生效
7291	灰度大区和灰度问鼎跨服战的版本发布（职能化）--云石	灰度环境	2017-03-13 16:01:05		发布	已生效
7096	七雄争霸——职能化测试服DB变更	测试环境	2016-11-01 11:28:28		自定义	已生效
6992	双区合服【同一个实例下面】	正式环境,测试环境,全区环境,灰度环境	2016-10-20 09:07:51		自定义	已生效

测试服发布模板

灰度发布模板

正式服发布模板

扩容模板

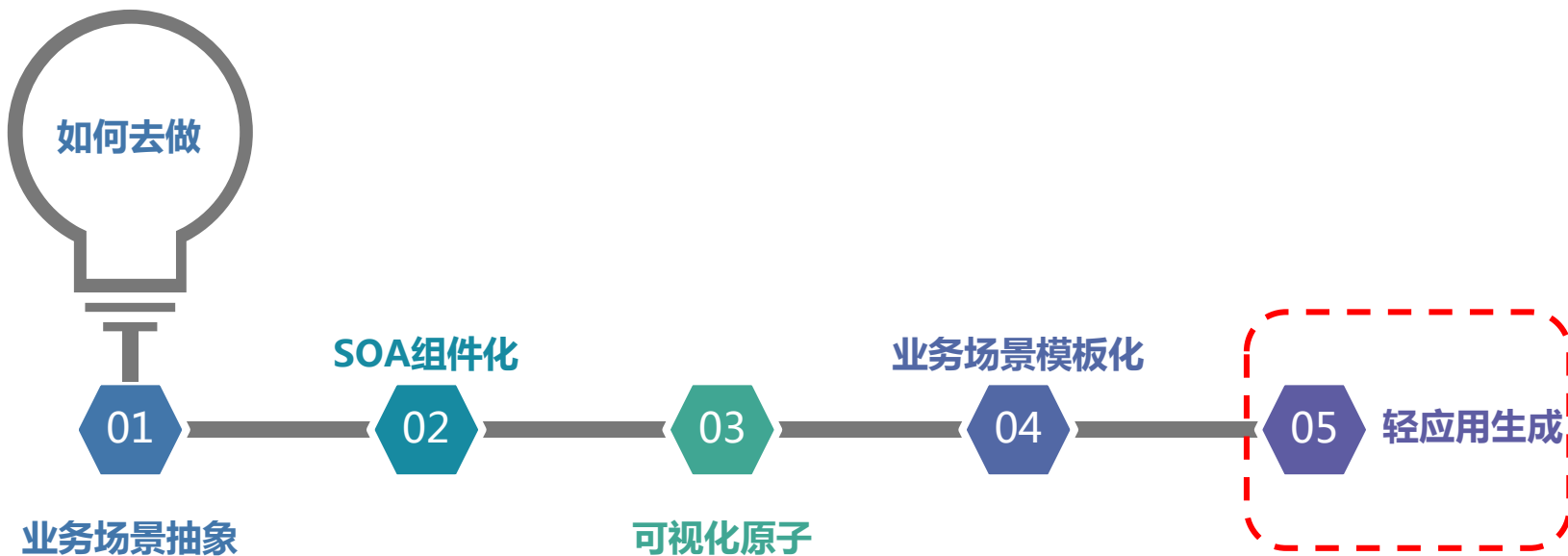
开区模板

合服模板

自定义模板

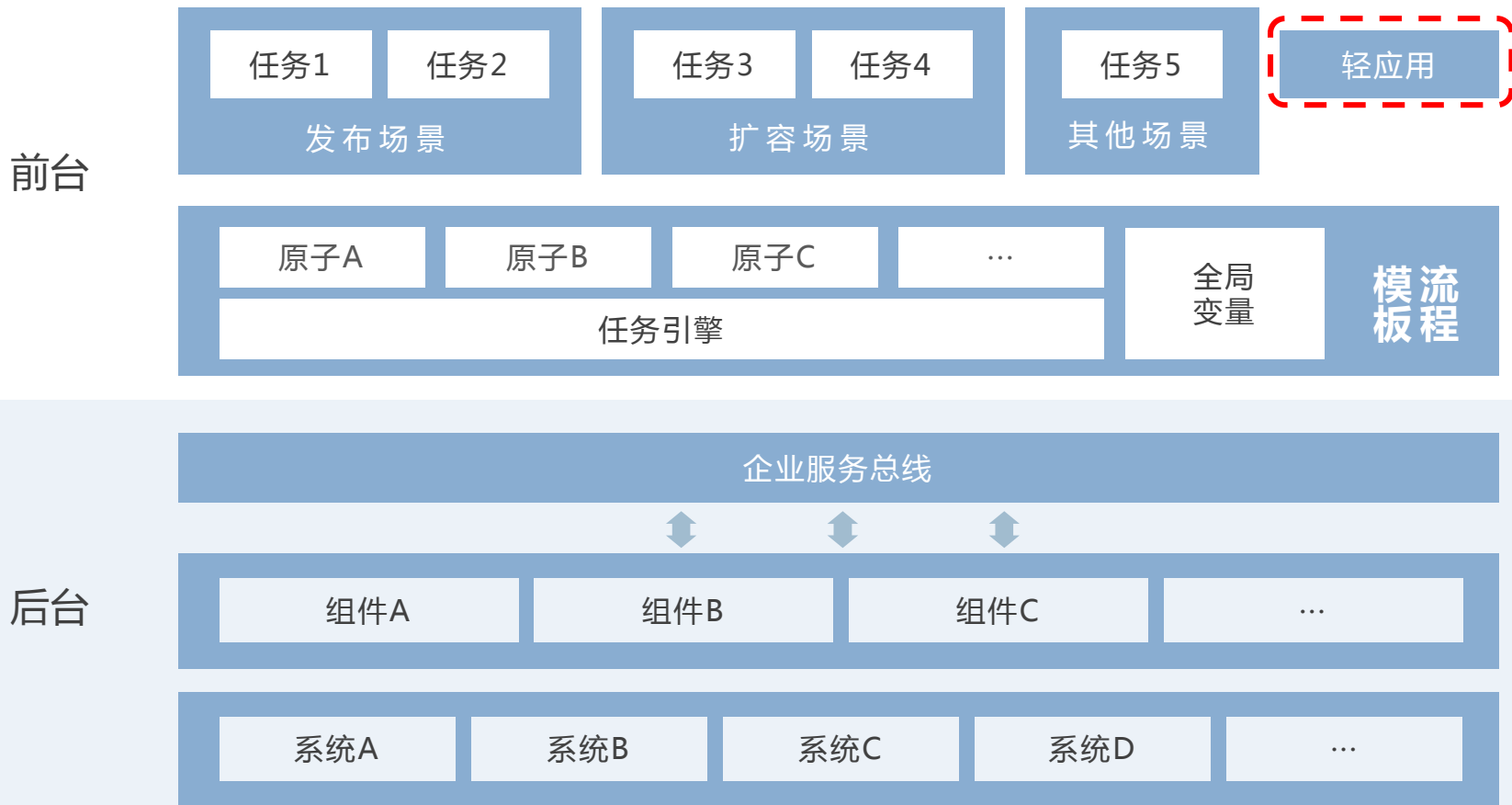
.....

## Step 5 : 轻应用生成





# 调度编排框架



## 通用工具

开发成本低

无法满足个性化需求

## 定制工具

开发成本高

满足个性化需求

轻应用



免开发



个性化需求

1 提交职能化需求

2 职能化认领

3 操作执行

4 职能化结单

5 需求评价

#### 任务信息

任务名称

范围\*  [Set详情](#)

任务时间\* 

开始时间	结束时间
<input type="text" value="2017-04-21 11:00"/> <input type="button" value="📅"/>	<input type="text" value="2017-04-21 11:00"/> <input type="button" value="📅"/>

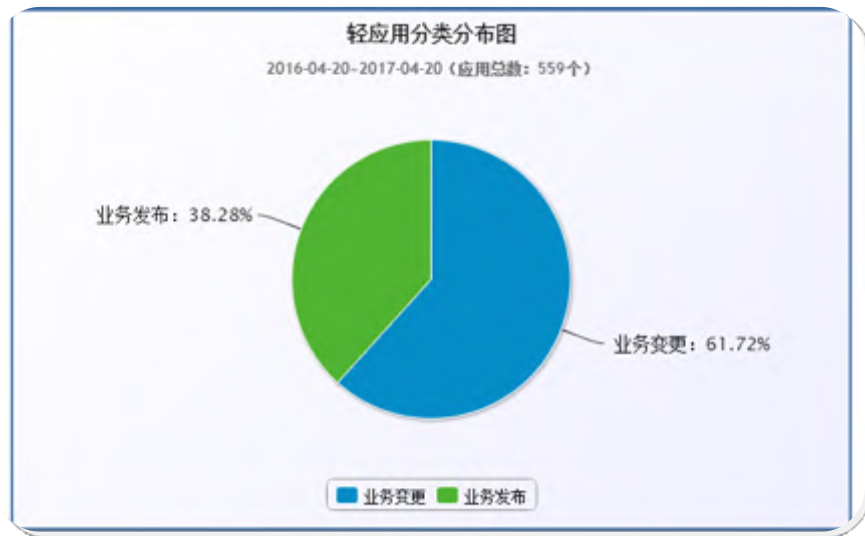
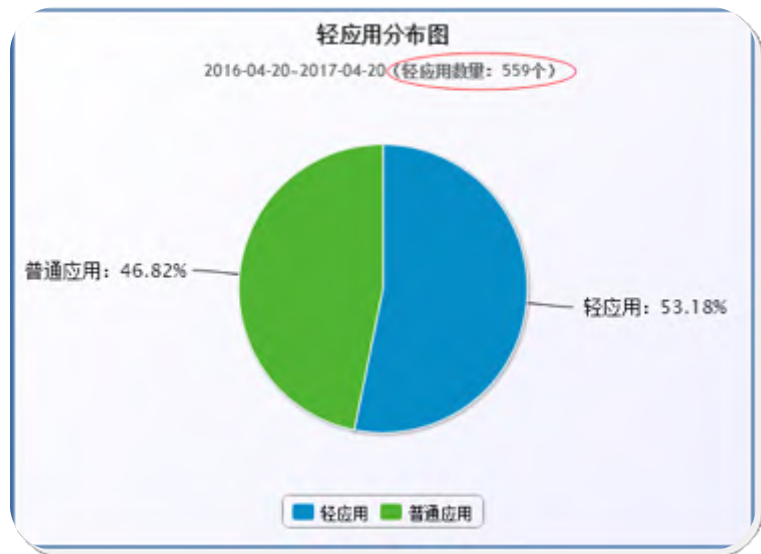
是否强制开始时间前无法执行任务

任务说明 [>> 编辑](#)

#### 参数填写

发布版本号

提交



559个轻应用    ≈≈    559个应用工具    ≈≈    559个开发人力

# 目录

1 传统发布变更之痛

2 初识调度编排

3 如何构建调度编排的5步曲

➔ 4 收益

# 总收益



## 连接一切

打通各平台的连通性，  
实现操作的全流程自  
动化



## 精准量化

精准量化运维工作，为业  
务优化点提供精准的依据

## 标准化

支持不同架构运维操  
作的标准化



## 工作转产

将运维日常重复操作  
固化成模板，转自助  
或外包



300+

接入业务量

150+

对接原子数

70% 13000+

自助率

执行量

## 调度编排对于传统发布变更的改变

步骤流程模板化，业务变动只需调整模板

操作统一调度，无需手动切换等待

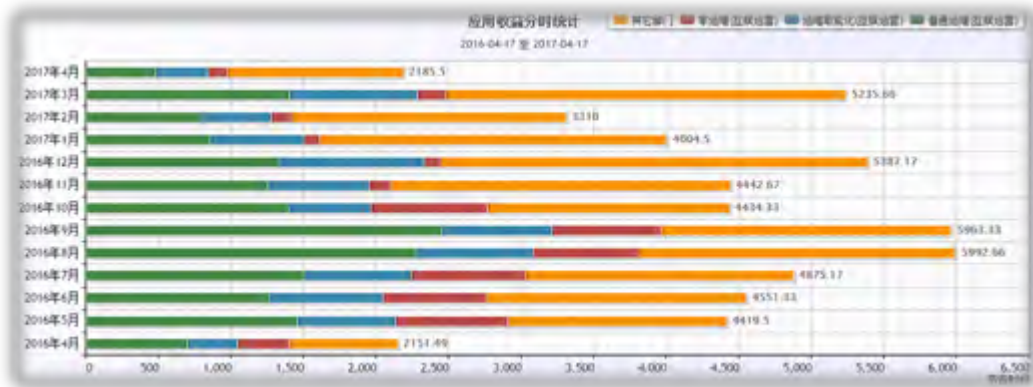
全局变量减少输入，任务参数强校验，多重确保操作安全

支持任务批量并发执行，适配大规模集中化业务布署

可视化模板传承，交接成本低

所有操作有据可查，便于事后安全审计

## 收益1——人力成本节省，业务停机时长



通过调度编排，运维操作转产自助或外包人员代替运维执行

传统方式：  
2小时以上

业务停机  
时长

调度编排：  
30分钟以内

减少了各环节切换等待、确认时间，  
业务平均停机时长从2小时降至30分钟



## 收益2——针对耗时节点，QC推动深入优化



精准统计发布变更的每一个操作节点耗时，  
便于业务针对性优化。

### 收益3——为其它智能决策类应用提供操作服务







# Thanks

扫一扫·关注蓝鲸智云公众号