

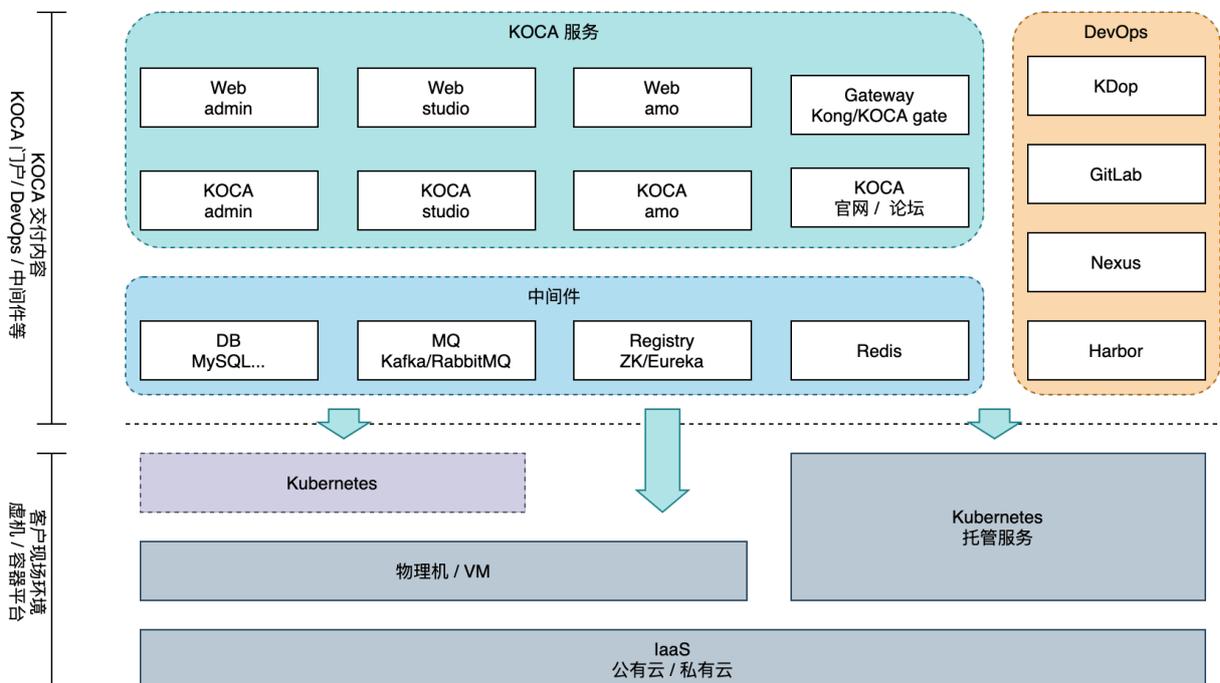
# KOCA 交付方案设计

## 交付内容

KOCA 交付主要任务是在客户现场，基于虚拟机或容器环境完成 KOCA 交付内容的部署，交付内容主要分为三类：

- KOCA 门户以及网关、官网等研发中心交付内容
- DevOps 相关服务，包括 KDop、GitLab、Nexus 等
- 中间件服务，既包括 KOCA 交付所依赖的服务，也有客户业务开发需要的服务

具体支持的服务列表见 [附录一：服务清单](#)，部署的服务可以根据客户具体需求进行选择交付。



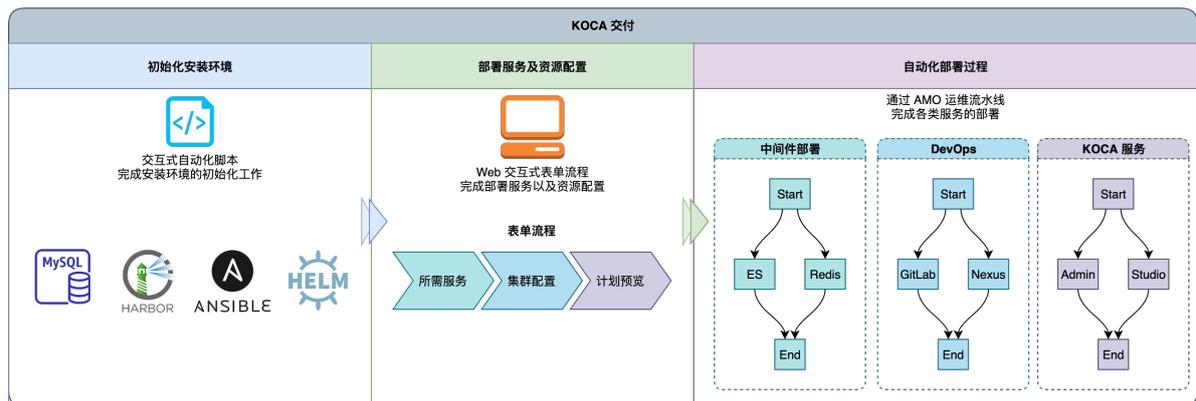
## 核心需求

- 满足虚拟机和容器环境部署
  - 在虚拟机或者 Kubernetes 容器平台上完成 KOCA 交付内容的部署
- 自动化，减少人工参与
- 定制化安装
  - 服务可选，如：仅部署 KOCA Studio 服务
  - 资源可配，如：单节点 ES 或 多节点 ES 集群

## 整体框架

KOCA 交付整体设计在 AMO 运维组件基础上构建，所有服务的部署过程使用 AMO 运维流水线完成，并为用户提供一个 Web 交互式的表单流程，完成所要部署服务的选择以及相关基础资源的配置，配置完成后自动生成 AMO 运维流水线，运维人员仅需要执行流水线，并对执行结果以及服务运行状态进行检查即可完成 KOCA 交付服务的部署。这个过程大概分为以下三个阶段：

1. 初始化环境的安装，主要完成 AMO 运维组件以及部署过程所依赖的服务，如数据库、镜像仓库、Ansible、Helm 工具等；
2. 启动基于 AMO 运维组件开发的部署工具，通过一个表单流程勾选所要部署的服务，分配的虚拟机或容器集群资源，最后形成 N 个 AMO 的运维流水线；
3. 根据服务间的依赖关系，运行 AMO 流水线，完成各个服务的部署；

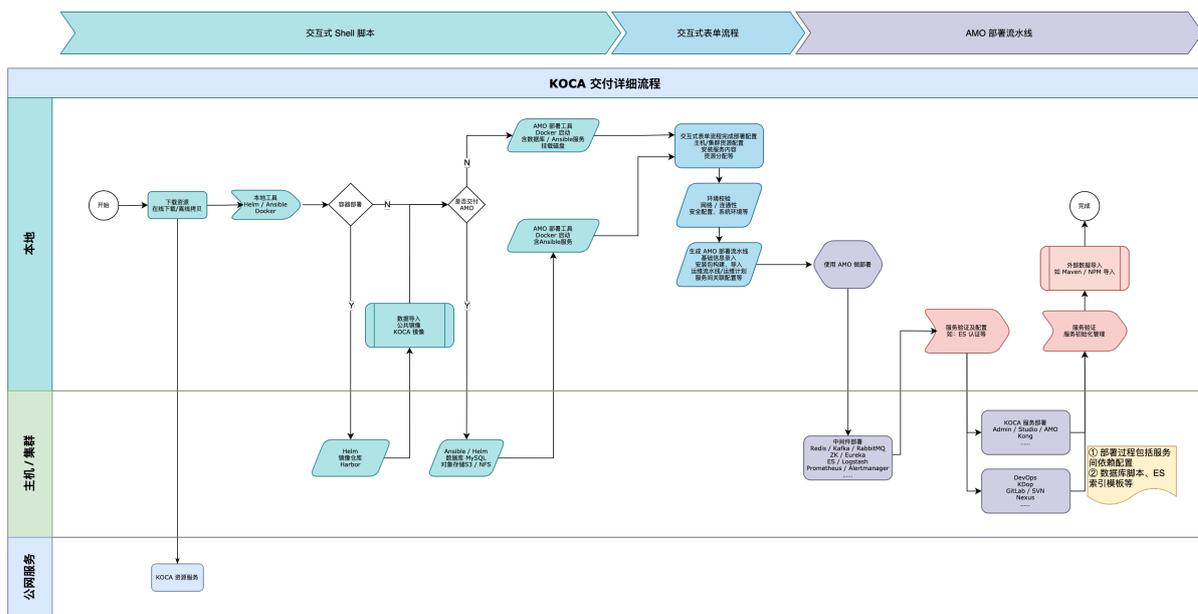


## docker-compose 部署

为满足用户快速体验需求，提供 `docker-compose` 方式一键部署 KOCA 交付的服务，包括 Admin、Studio、AMO、Kong、KDop 等

- 仅包括 KOCA 交付的服务以及依赖
- 不包括 GitLab、SVN 等非 KOCA 服务内容
- 数据库及数据自动初始化
- 可做定制裁剪
  - 基于模板的方式动态生成 `docker-compose.yml` 配置

## 详细流程



## 流程说明

- 资源下载
- 具体资源见 [附录二：资源清单](#)
  - 做资源校验，避免发布包、镜像等资源有损坏或篡改
  - 支持两种模式：
    - 1. FTP 提供一个完整离线安装包，可直接拿到现场使用，特点是使用简单
    - 2. KOCA 运维人员先下载交付工具，再通过下载脚本完成二进制包、镜像等资源的下载，特点是便于使用管理
- AMO 部署工具支持 `java -jar` 或 `docker-compose` 两种方式运行
  - `java -jar` 需要本地自备数据库
- 如果虚拟机部署过程需要以 Docker 方式部署服务，镜像通过离线方式导入到主机
- 如果存在堡垒机需要将“本地”的 Ansible 移到堡垒机
  - 或者在处理完离线资源后，将“本地”移到内网主机进行
- 服务的运维部署每个服务对应一套标准的资源，内容包括：
  - 服务描述文件，如：服务名称、版本等
  - 二进制 / 镜像文件
  - 容器 / 虚机的运维脚本及流水线
  - 虚拟机 服务运行配置模板
  - 容器 Helm Chart
  - 可选 监控相关仪表等配置

- 1 # 完整服务资源结构
- 2 # 数据来自两部分
- 3 # ① 交付工具仓库
- 4 # ② 制品库: binary、image

```

5 | ./service/amo
6 | └─ service.yml # 服务描述文件, 名称、版本等
7 | └─ binary      # 二进制部署文件, KOCA 制品/三方依赖 Release 包, 包括数据
   |             # 库脚本
8 | └─ image       # 镜像
9 | └─ chart       # Helm Chart, Kubernetes 部署使用
10 |   └─ value.yml
11 | └─ config      # 配置模板, 虚拟机
12 |   └─ jdbc.yml.tpl
13 | └─ monitor     # 监控相关仪表配置
14 | └─ pipeline    # 运维流水线, 部署、升级、启停等,
15 |   └─ 01.install.yml
16 |   └─ 02.upgrade.yml
17 | └─ task
18 |   └─ 01.task-1.yml
19 |   └─ 02.task-2.yml

```

。

## 交付管理

KOCA 交付相关资源主要包括两部分，一是交付工具，二是制品。交付工具主要包括初始化安装脚本，基于 AMO 运维组件开发的 KOCA 交付工具，以及各个服务运维监控相关的脚本资源；制品则主要包含两方面一是二进制的发布包，二是用于容器部署的镜像，其中二进制包根据服务类型不同有不同结构，三方依赖一般是官方发布的 Release 版本，KOCA 门户则有统一的交付内容，详见制品管理。

## 交付工具仓库

- 基于 AMO 运维组件开发的 KOCA 交付工具
  - 提供界面化操作，简化运维配置的复杂性
- 部署过程的自动化脚本、工具
  - 资源下载工具
    - 客户现场或内部使用
  - 本地环境安装工具
  - 环境检查工具等
- 通过文件方式维护服务列表，每个服务的资源结构如下：
  - 服务描述文件，如：服务名称、版本等
  - 容器 / 虚拟机的运维脚本及流水线
    - 流水线包括数据的初始化/升级
    - 以文件目录组织服务列表，通过代码仓库对脚本等资源进行管理，流水线等相关配置可以通过 AMO 平台导入/导出，甚至包括监控采集、仪表等相关配置
      - 为以后服务市场或者远程自助升级等打好基础
  - 虚拟机 服务运行配置模板
  - 容器 Helm Chart

- 可选 监控相关仪表等配置

```
1  ./
2  └─ Makefile
3  └─ README.md
4  └─ operation          # 基于 AMO 运维组件开发的 KOCA 交付工具
5  |   └─ src
6  |   └─ .....
7  |   └─ pom.xml
8  └─ service
9  |   └─ admin
10 |   |   └─ .....
11 |   └─ amo
12 |   |   └─ service.yml # 服务描述信息，如包含哪些组件，关联组件等相关配置
13 |   |   └─ config      # 部署配置模板
14 |   |   |   └─ jdbc.yml.tpl
15 |   |   └─ chart      # Helm Chart
16 |   |   |   └─ value.yml
17 |   |   └─ monitor    # 监控相关仪表配置
18 |   |   └─ pipeline   # 运维流水线，安装、部署等
19 |   |   |   └─ 01.install.yml
20 |   |   |   └─ 02.upgrade.yml
21 |   |   └─ task
22 |   |       └─ 01.task-1.yml
23 |   |       └─ 02.task-2.yml
24 |   └─ elasticsearch
25 |   └─ .....
26 └─ tools              # 自动化工具，shell、playbook、jar等
    └─ check.sh
    └─ download.sh
    └─ install.sh
```

## 制品管理

- 三方依赖发布包统一维护，并提供下载服务
- KOCA 门户在发版后根据服务交付的粒度，构建前后端发布包以及相应镜像，并上传到制品库（资源中心 / FTP）
  - 发布包包括对应的全量、增量数据库脚本
  - 每个门户独立部署，基础数据不共享
- 前端直接打包构建的 `dist` 文件，后端发布包结构如下：

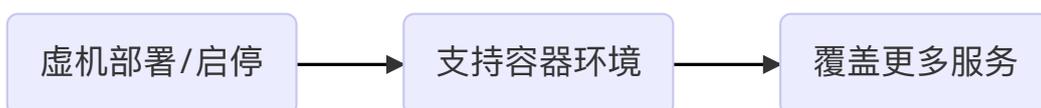
```
1  ./
2  └─ bin
3  |   └─ shutdown.sh
4  |   └─ startup.sh
5  └─ conf
6  └─ application.yml
```

```
7 | | | banner.txt
8 | | | config
9 | | | | jdbc.yml
10 | | | | mybatis.yml
11 | | | | logback-spring.xml
12 | | db
13 | | | mysql
14 | | | | data
15 | | | | | all
16 | | | | | | update
17 | | | | | table
18 | | | | | | all
19 | | | | | | update
20 | | | | oracle
21 | | lib
22 | | | amo-2.2.0-RELEASE.jar
```

## 主要工作

- 交付工具
  - 交付流程、资源描述、目录结构等规范设计
  - 基于 AMO 运维组件的交付工具
  - 服务运维脚本及流水线
    - 根据优先级逐步建设，先满足 KOCA 门户需求，以及网关 和 DevOps 产品
  - 流程自动化工具开发
  - AMO 运维能力的补充
    - 编排能力
    - 服务间数据关联
    - .....
- 制品管理
  - 流水线
  - 制品发布到 FTP

## 分阶段建设



- 阶段一
  - 虚拟机部署
- 阶段二

- 容器环境部署
- 阶段三
  - 支持更多服务

## More

- ✨ 中间件、KOCA 服务等作为服务市场提供
  - 初始仅需要部署 AMO，其它都可以在平台自助完成
  - 相当于在一个 PaaS 平台，安装 SaaS 服务
- ✨ 对 KOCA 组件级定制化交付问题，提供类似 Spring Initializr 产品
  - 作为交付工具使用，资源中心仅提供标准版，Initializer 可以根据需求直接定制，并直接构建出发布包和镜像
- ✨ 服务监控
- DB 等可以根据需求做混合部署
- 自助升级、灰度发布
- 更多中间件支持，各种集群、高可用模式的部署
- .....

## 附录

### 附录一：服务清单

- KOCA 服务

服务	版本	端口占用	升级支持	备注
Admin	-		✓	微服务部署
Studio	-		✓	
AMO	-		✓	
Kong 网关	-		✓	
KOCA 官网	-		✓	

- DevOps

服务	版本	端口占用	升级支持	备注
KDop	-		✓	
GitLab			-	
Subservation			-	
Harbor	1.10.0		-	可以外部提供
Nexus	3.25.1		-	

- 中间件类

服务	版本	端口占用	升级支持	备注
MySQL	v5.7	3306	-	可以外部提供Oracle、SQLServer数据库
Redis	5.0.9	6379	-	
Zookeeper			-	
Eureka	1.9.13		-	
RabbitMQ			-	
Kafka			-	
Elasticsearch	7.8.1	9200	-	可外部提供
Logstash	7.8.1	5044	-	
Kibana	7.8.1	5601		可选
Prometheus	2.21.0	9090	-	
Grafana				可选
Alertmanager	0.21.0	9093	-	
Ansible	2.9.10	-	-	
MinIO		9000		可选 AMO 发布包管理的一种方案，尤其容器部署

## 附录二：资源清单

服务	资源类型	来源	
KOCA 交付工具	交付工具发布包	Git仓库	
AMO	前后端发布包	流水线构建	
	前后端镜像	流水线构建	
Studio			
Admin			
Kong			
KDop			
GitLab			
Nexus			
MySQL			
Redis			
Kafka			
RabbitMQ			
Zookeeper			
Eureka			
Ansible			
Helm			
Elasticsearch			
Logstash			
Prometheus			
Alertmanager			
Grafana			
Kibana			

KOCA Agent			