

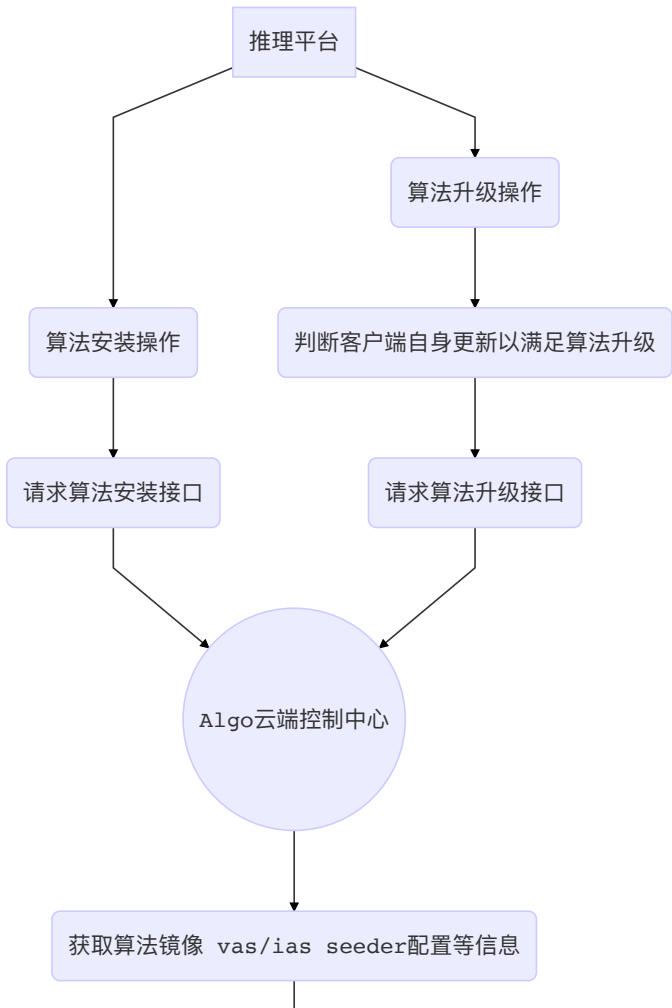
算法升级与安装

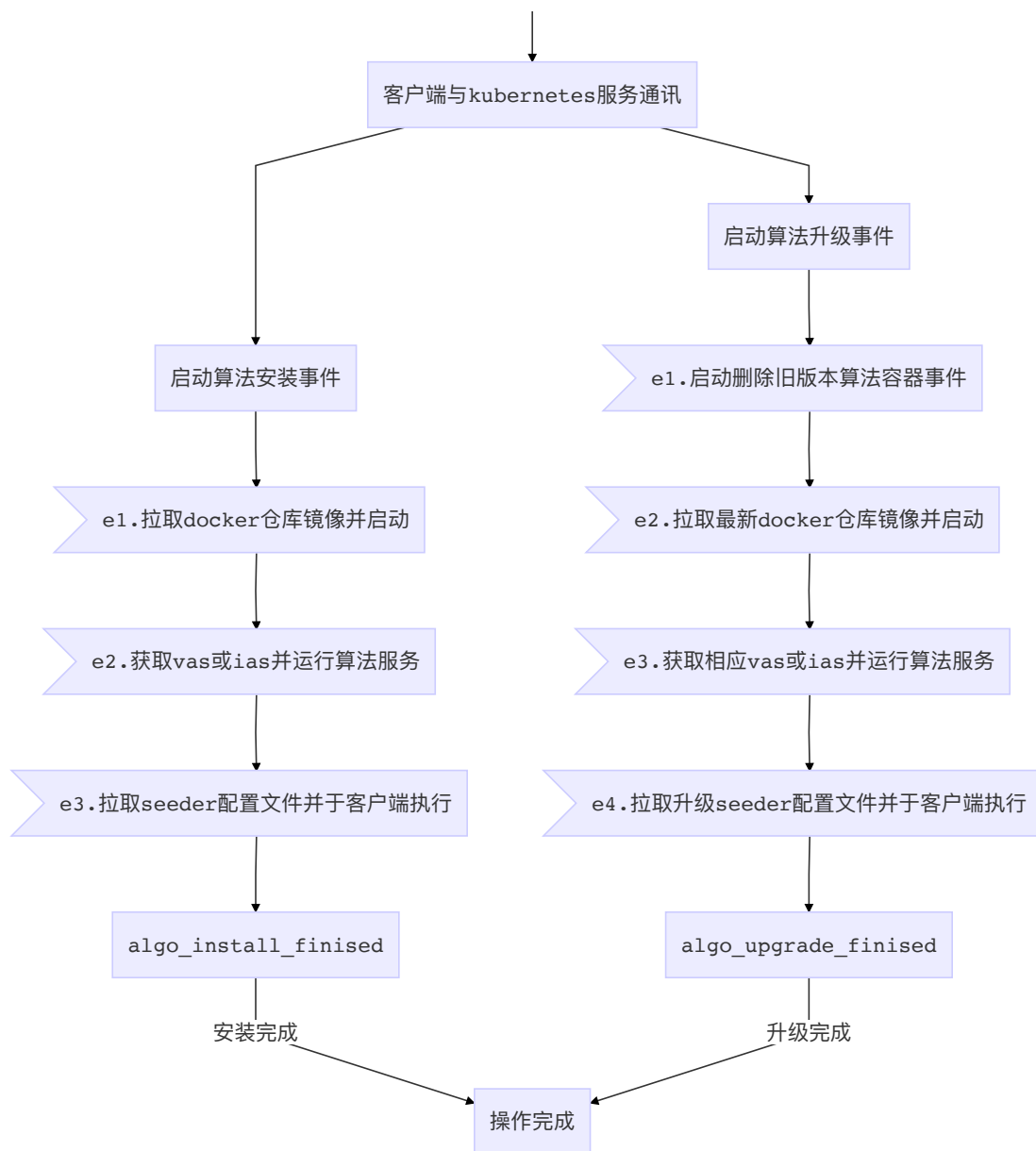
前言

推理平台 或 需求方到底是如何对算法进行安装 与 升级的呢？无论是安装还是升级算法都涉及好些依赖模块以及平台，包括推理平台自身客户端、算法云端控制中心、企业对象存储平台、`Docker` 仓库、`Kubernetes` 集群等。算法部署支持单机、集群等部署方式，同时可部署于内网或外网。

- ✓ 算法工程师开发的算法，算力达到预期值后会将训练后的模型转成 `SDK`，最后封装成 `Docker` 镜像并上传与极视角镜像仓库，提供 `Kubernetes` 集群部署算法使用，即 `Docker` 仓库提供镜像存储的服务。
- ✓ 对于算法云端控制中心主要职责是对算法相关的管理，比如算法版本管理、`ias` 管理、`vas` 管理、算法调度资源管理等。
- ✓ 借助 `cos` 对象存储优势，为算法的配置 `seeder` 文件、`vas`、`ias` 等文件提供对象储存服务，它核心是针对算法相关的文件进行灵活、稳定的资源管理平台。

流程图





流程阐述理解

算法的安装与算法的升级操作流程总体类似，而算法升级的流程稍微复杂一些，两者的处理流程如下：

算法安装流程

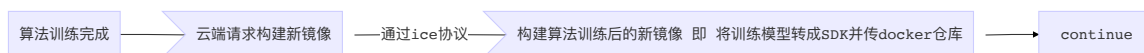
- **推理平台**：根据客户端可安装的镜像 或 用户输入期望的算法与版本，根据算法的唯一标识和版本信息请求云端获取算法安装详情信息接口
- **云端控制中心**：根据推理平台客户端请求的算法与版本参数返回对应涉及安装的数据信息，包括镜像地址、算法运行参数(承载于 `seeder` 文件)、`vas` 或 `ias` 信息、算法基本信息等
- **推理平台**：根据云端控制中心返回的算法报文信息后处于算法安装的就绪状态
 - 与 `kubernetes` 中的 `api-server` 通讯，根据算法镜像信息与环境信息在 `k8s` 集群构起安装的算法运行容器
 - 在容器启动时，算法容器会在 `cos` 存储服务拉取 `vas` 或 `ias` 文件，部署并启动服务

- 算法容器启动完成，系统继续请求 `cos` 储存云服务下拉 `seeder` 文件，程序会在执行此 `seeder` 文件，此时推理平台对此算法的运行参数就储存于客户端了，至此、算法就安装完成。

算法更新流程

- **推理平台**：客户端实时请求云端控制中心，根据本地已经安装的算法或用户输入期望的版本请求云端获取版本最新的算法安装详情接口
- **云端控制中心**：根据推理平台客户端请求的算法与版本参数与云端最新的版本比较，倘若是存在最新的版本，则返回对应涉及安装的数据信息，包括镜像地址、算法运行参数(承载于 `seeder` 文件)、`vas` 或 `ias` 信息、算法基本信息等
- **推理平台**：根据云端控制中心返回的最新算法报文信息后处于算法安装的就绪状态
 - 与 `Kubernetes` 进行通讯，清除当前安装并正在运行旧版本算法
 - 旧版本算法容器删除完成后，再次与 `kubernetes` 中的 `api-server` 通讯，根据算法镜像信息与环境信息在 `k8s` 集群构起安装的算法运行容器
 - 在容器启动时，算法容器会在 `cos` 存储云服务拉取 `vas` 或 `ias` 文件，部署并启动服务
 - 算法容器启动完成后，系统先清除就版本的配置信息，清除完成后继续请求 `cos` 储存云服务下拉 `seeder` 文件，程序会在执行此 `seeder` 文件，此时推理平台对此算法的运行参数就储存于客户端了，至此、算法就更新完成

模型转成 SDK



模型作为一个算法的记忆储存，要将一个算法正常在程序运行、那就需要将模型转成 `SDK`。算法基于训练数据集经过联邦学习的上万级、甚至以亿级量次的训练后，会生成算法训练的新模型，接下来需要将出来的新模型构建、转换成 `SDK`，此步骤对于算法封装成新的算法镜像非常重要，属于联邦训练的核心流程。云端接收到联邦学习完成回调通知后，此时算法云端中心会通过 `ice` 协议请求算法打包构建的服务器，此服务执行的流程相当严谨、复杂，简单而言呢，就是打包服务器拉取训练后的最新模型替换算法旧版本的模型、然后将镜像打包并提交到算法 `Docker` 仓库。