ModelBox 是一个适用于端边云场景的 AI 推理应用开发框架，提供了基于 Pipeline 的并行执行流程，能帮助 AI 应用开发者较快的开发出高效，高性能，以及支持软硬协同优化的 AI 应用。

ModelBox 特点

1. 易于开发

   AI 推理业务可视化编排开发，功能模块化，丰富组件库；c++，python, Java 多语言支持。

2. 易于集成

   集成云上对接的组件，云上对接更容易。

3. 高性能，高可靠

   pipeline 并发运行，数据计算智能调度，资源管理调度精细化，业务运行更高效。

4. 软硬件异构

   CPU，GPU，NPU 多异构硬件支持，资源利用更便捷高效。

5. 全场景

   视频，语音，文本，NLP 全场景，专为服务化定制，云上集成更容易，端边云数据无缝交换。

6. 易于维护

   服务运行状态可视化，应用，组件性能实时监控，优化更容易。

ModelBox 解决的问题

目前 AI 应用开发时，训练完成模型后，需要将多个模型和应用逻辑串联在一起组成 AI 应用，并上线发布成为服务或应用。在整个过程中，需要面临复杂的应用编程问题：

1. 需要开发 AI 应用的周边功能：比如 AI 应用编译工程，应用初始化，配置管理接口，日志管理口，应用故障监控等功能。
2. 需要开发 AI 常见的前后处理：音视频加解码，图像转换处理，推理前处理，后处理 YOLO 等开发。
3. 需要开发和云服务对接的周边功能：比如 HTTP 服务开发，云存储，大数据服务，视频采集服务对接开发。
4. 需要开发出高性能的推理应用：需要基于多线程，内存池化，显存池化，多 GPU 加速卡，模型 batch 批处理，调用硬件卡的 API 等手段开发应用。
5. 需要开发验证 docker 镜像：需要开发 docker 镜像，集成必要的 ffmpeg，opencv 软件，CUDA, MindSpore，TensorFlow 等软件，并做集成测试验证。
6. 多种 AI 业务，需要共享代码，降低维护工作：需要复用不同组件的代码，包括 AI 前后处理代码，AI 应用管理代码，底层内存，线程管理代码等。
7. 模型开发者，验证模型功能比较复杂：模型开发者完成模型训练后，需要编写 python 代码验证，之后，再转成生产代码；在高性能，高可靠场景改造工作量大。

ModelBox 的目标是解决 AI 开发者在开发 AI 应用时的编程复杂度，降低 AI 应用的开发难度，将复杂的数据处理，并发互斥，多设备协同，组件复用，数据通信，交由 ModelBox 处理。开发者主要聚焦业务逻辑本身，而不是软件细节。 在提高 AI 推理开发的效率同时，保证软件的性能，可靠性，安全性等属性。

ModelBox 支持两种方式运行，一种是服务化，一种是 SDK，开发者可以按照下表选择相关

的开发模式。
1. 服务化：ModelBox 为独立的服务，适合云服务，端侧服务的 AI 推理开发场景，包括了后台服务，运维工具，docker 镜像等服务化组件
2. SDK：ModelBox 提供了 ModelBox 开发库，使用于扩展现有应用支持高性能 AI 推理，专注 AI 推理业务，支持 c++，Python 集成
在开发 AI 推理应用时，可以按照第一个应用的流程开发 AI 应用。