预备知识：linux基础、docker

Python算法docker打包步骤：

1. 安装docker

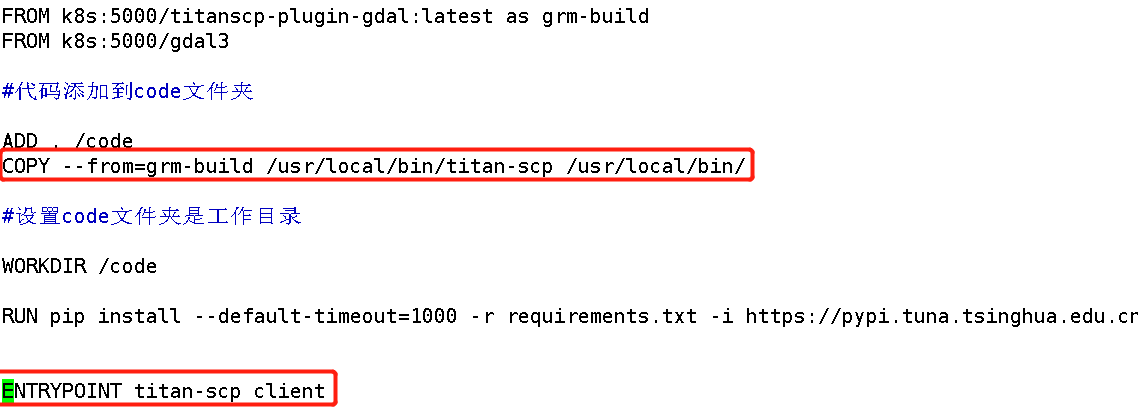
这个就不说了，根据操作系统版本自行安装，注意使用GPU加速的项目需要安装nvidia-docker，后面的步骤也是基于这个。

1. 将Python算法打包为docker镜像
2. 把写好的python算法应整理在一个文件夹内。进入到你运行python项目的虚拟环境（这里的话首先要在linux上安装python环境，为了方便使用可以安装Anaconda，用conda来创建一个虚拟环境），使用pip freeze > requirements.txt导出运行依赖的第三方库。（这一步也可以自己手写，总之要把得到的requirements.txt文件放入项目文件夹内，方便下一步操作）requirements.txt这个文件就是用来放运行当前python算法所要用到的包以及包的版本号。
3. 编写Dockerfile（重点）

这里提供一个例子，需要特别注意的是合适的基础镜像很关键！！比如你的程序是基于某个深度学习框架，那么直接拉取这个框架的镜像就好了。一般会帮你配置好相关环境。我们打包的Python算法通常是基于下图中的两个镜像。k8s:5000/gdal3这个基础镜像要提前在本机构建（构建的时候需要用gdal3的项目文件和Dockerfile）。还有两个注意点，第一个注意点是要在拉取基础镜像的时候，配置/etc/hosts文件，添加k8s 139.198.13.149。第二个注意点是在/etc/docker下打开daemon.json文件，添加"insecure-registries":["k8s:5000"]。



下面这两行也是要在Dockerfile中必须加的，这里的意思是要把这个titanscp的宿主程序放进去，才能使我们的算法在云平台上去运行起来。



1. 制作镜像

docker build -t imagename Dockerfilepath

Dockerfilepath：Dockerfile 所在文件夹名称，通常我们使用当前名录为 “.”

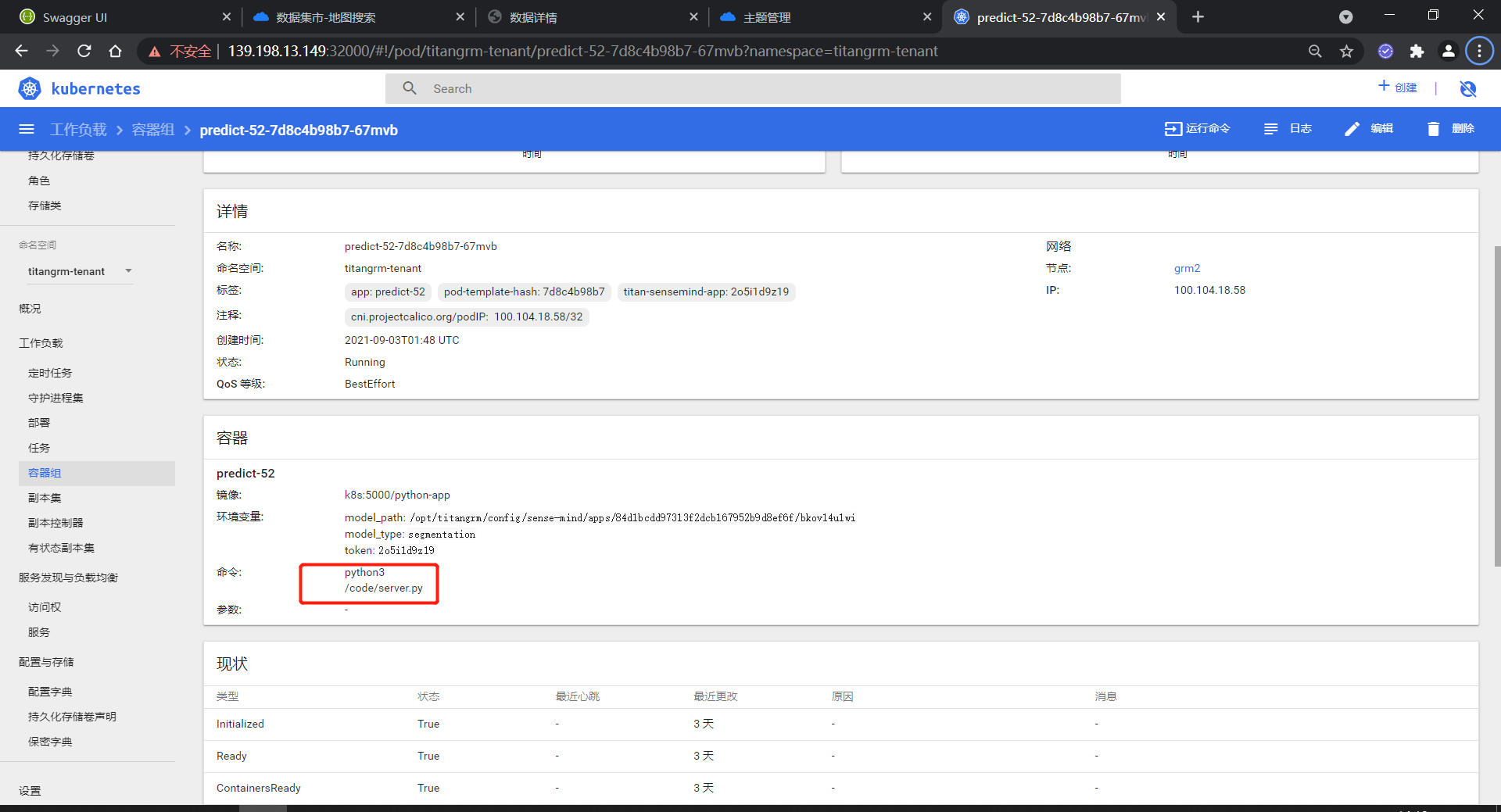
Imagename：要以k8s:5000/开头，比如k8s:5000/python-app:latest这种格式

1. 上传镜像

镜像制作好以后，就可以上传到服务器了，执行docker push imagename，上传成功之后，然后在云平台上添加json文件进行算法注册，注册成功之后就可以在云平台上进行算法测试了。

算法调试：

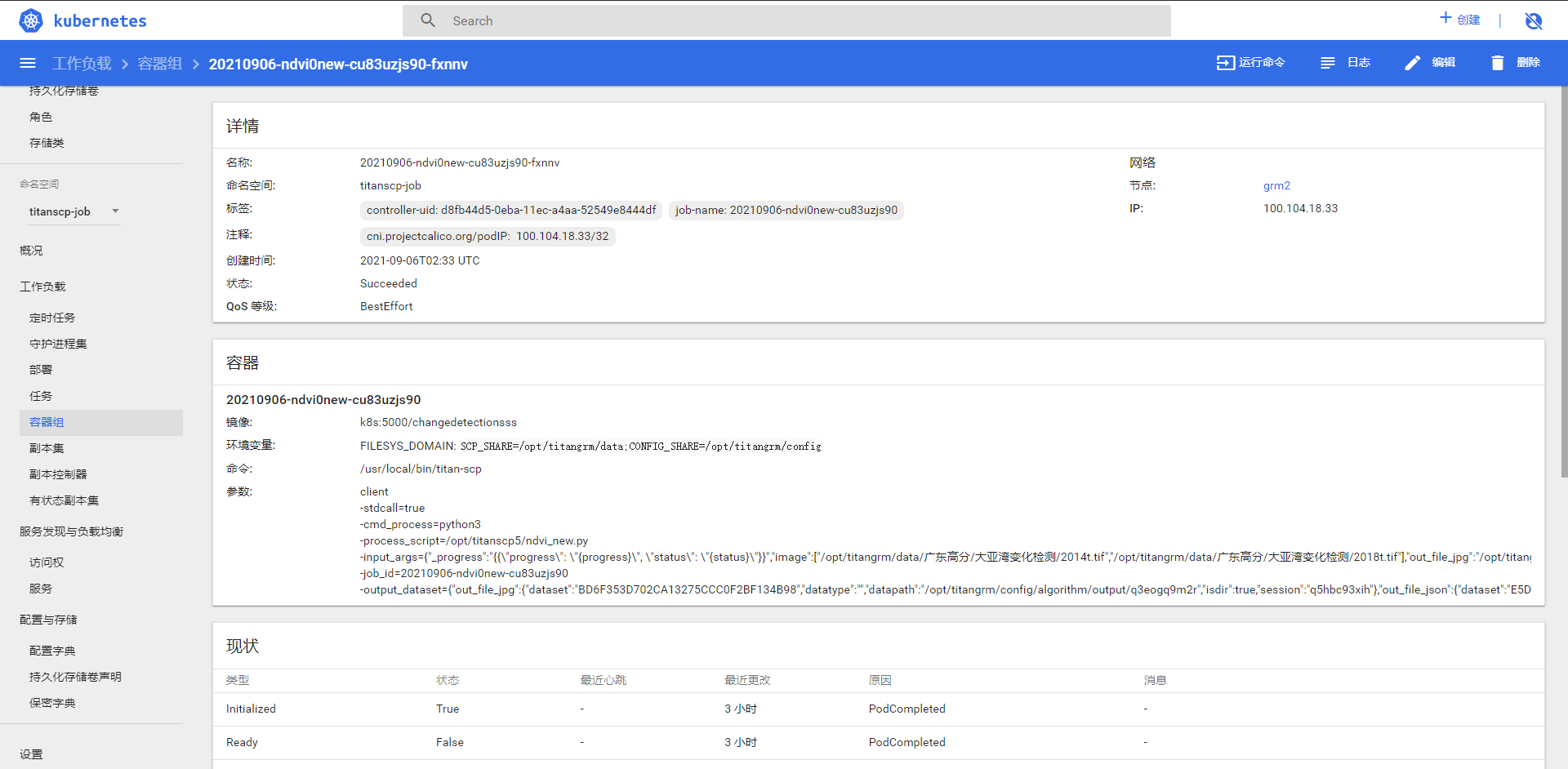
1. 解译算法运行环境：



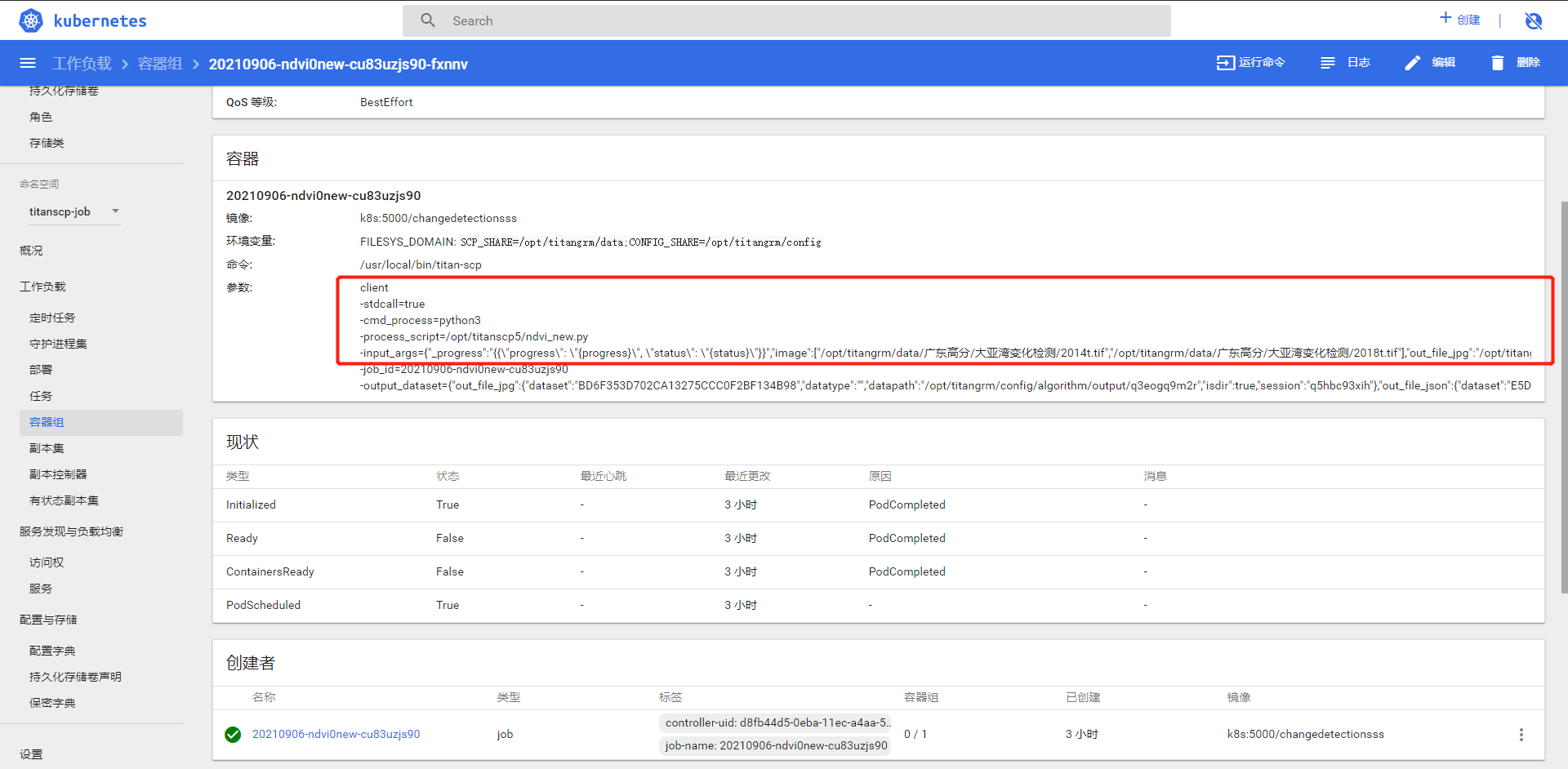
Server.py 文件：



1. 算法2.0运行环境



参数写法：



内部调用:

