**卡调系统部署说明**

修订记录

类别：A-增加；M-修改；D-删除

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 修订日期 | 版本号 | 修订类别 | 描述 | 修改人 |
| 2022-09-09 | V1.1 | A | 新增依赖库sqlite3，requests，新增拓扑路网更新方式 |  |
| 2022-11-15 | V.2 | A | 二次调度与周期调度分离，启动方式变化 |  |
| 2023-3-20 | V2.1 | A | 增加穿越装载区调度，二次调度无需停车 |  |

# 1.矿卡调度系统

使用 Python3 编写的程序，用于使用启发式算法在露天矿山作业中管理矿卡周期及实时调度。

# 2.如何安装

## 2.1 环境要求

至少安装Python 3.6

依赖项：sqlalchemy、redis、numpy、datetime、mysql-connector、psycopg2

## 2.2 安装python3.6

（如下介绍是针对于Linux操作系统）

下载并安装python3.6.5版本。

1. 首先安装python依赖的库

Centos:

yum install zlib-devel bzip2-devel openssl-devel ncurses-devel sqlite-devel readline-devel tk-devel gcc make

ubuntu:

sudo apt-get install zlib1g-dev libbz2-dev libssl-dev libncurses5-dev libsqlite3-dev libreadline-dev tk-dev libgdbm-dev libdb-dev libpcap-dev xz-utils libexpat1-dev liblzma-dev libffi-dev libc6-dev

1. 把Python3.6下载到/usr/local目录

# wget https://www.python.org/ftp/python/3.6.5/Python-3.6.5.tgz

# tar -xzvf Python-3.6.5.tgz -C /tmp

# cd /tmp/Python-3.6.5/

# ./configure --prefix=/usr/local

1. 修改配置，防止出现pip3找不到TLS/SSL的问题出现：



参考链接：

https://blog.csdn.net/Pc620/article/details/86163275

https://blog.csdn.net/hobohero/article/details/54381475?utm\_source=copy%A0

1. 执行make编译命令

# make

# make altinstall

(make altinstall用于防止替换/usr/bin中默认的python软连接)

如果在安装过程中遇到zipimport.ZipImportError: can't decompress data; zlib not available的问题，执行下面两条命令，安装zlib、zlib-devel依赖库即可。

# yum install zlib zlib

# yum install zlib zlib-devel

1. 更改/usr/bin/python链接和pip3链接

# cd /usr/bin

# ln -s /usr/local/bin/python3.6 /usr/bin/python3

# ln -s /usr/local/bin/pip3.6 /usr/bin/pip3

至此，python 3.6安装完毕，但是此时在命令窗口输入python3，即可打开python3的命令窗口界面。

## 2.4 创建虚拟环境（可选）

virtualenv是一个能创建隔绝的独立的Python虚拟环境的工具。它能够建立多个相对独立，互不影响的Python工作环境。也可以直接在python主环境进行配置，请跳过2.4, 直接安装依赖库(2.5)。

1. virtualenv 安装

# pip install virtualenv

1. 创建虚拟环境

安装好virtualenv后，使用virtualenv创建一个虚拟环境：

# virtualenv watodisp # watodisp为虚拟环境的目录名称，名称可以自定义

1. 激活虚拟环境

进入到虚拟环境bin 目录下：

# cd watodisp/bin

执行以下命令启动虚拟环境：

# source activate

启动之后命令行界面会出现一个(watodisp)，这就说明我们启动虚拟环境成功了，可以在当前虚拟环境下安装依赖库。



## 2.5 安装依赖库

**pip3 install** sqlalchemy

pip3 install redis

pip3 install numpy

pip3 install datetime

pip3 install mysql-connector

pip3 install psycopg2

pip3 install pulp

pip3 install flask

pip3 install flask-caching

pip3 install bidict

pip3 install requests

**安装 sqlite3**

sudo apt-get install sqlite3

或者：

访问 <https://www.sqlite.org/download.html>，从源代码区下载 sqlite-autoconf-\*.tar.gz，执行：

$ tar xvzf sqlite-autoconf-3071502.tar.gz

$ cd sqlite-autoconf-3071502

$ ./configure --prefix=/usr/local

$ make

$ make install

# 3.配置文件

1. 修改config.json



log\_path 修改为：服务器调度日志文件，所在文件夹位置

mysql 修改IP地址及端口 账户密码

postgresql 修改IP地址及端口 账户密码

redis 修改IP地址 账户密码

gothrough 配置穿越装载区服务信息：closer\_area\_name 为近处的装载区名称，further\_area\_name 为远处的装载区名称

# 4.程序运行

1. **前端调度界面更新路网（重要）**
2. **启动调度（重要）**
   1. 在依赖齐全的(虚拟)环境下，进入项目目录下执行：

**sh start.sh**

* 1. 注：该方式将启动全部调度功能，包括：周期调度，二次调度，实时调度，穿越装载区调度；

1. 特殊情况处理：

若脚本启动失败，可按照需求手动执行如下命令：

* 1. 仅需启动周期调度：

nohup python3 realtime\_dispatch.py >/dev/null 2>&1 &

* 1. 需启动周期调度及二次调度：

nohup python3 realtime\_dispatch.py --active >/dev/null 2>&1 &

* 1. 需启动实时调度或穿越装载区：

nohup flask run --host=0.0.0.0 >/dev/null 2>&1 &