通用数采程序架构设计方案python

1. 目的

为了方便数采程序统一管理，实现快速部署和轻松运维。

1. 程序架构
2. 数采程序项目名称统一为“edge\_iot”，内部文件均以其实现功能命名，每个项目必须创建readme文件说明程序功能和详细使用方法；
3. 配置信息模块：

系统运行配置：独立的一个settting.py文件参数管理

系统数据初始化：配置数据清单以xlsx表格形式创建，每行一个配置信息，程序启动运行第一步则是将配置数据清单导入sqlite数据库，使用配置数据前则使用sqlite查询语句将配置数据加载入内存变量中；

1. 采集模块：python开源依赖库，例如pymosbus等；每个协议独立的一份代码文件
2. 数据存储模块：sqlite数据库存储配置数据，influxdb时序数据库存储过程数据，MySQL关系数据库存储计算结果数据；
3. 数据转发模块：kafka服务器；需要结合不同项目类型的数据结构进行包装。
4. 心跳检测
5. 日志文件
6. 虚拟环境包
7. 程序版本requirements.txt清单
8. 部署与运维
9. 项目文件统一放置在系统用户根目录下；
10. 将系统python环境安装为python3.8版本；
11. 环境依赖包统一放在IOT/依赖包文件夹内；
12. Window系统将项目所需运行程序文件在任务计划程序中配置为开机自启动；
13. Linux系统将项目所需运行程序文件使用crontab -e配合keep\_alive.sh创建定时任务检查任务的启动情况，发现程序未启动需重启程序。
14. 参考项目结构：

edge\_iot ##项目名称

├─apps ##数采封装

│ └─apis **##采集业务封装**

└─\_art\_collector.py

└─models ##采集参数据ORM模型

└─\_measurement\_model.py

└─services  **##采集业务实现**

└─\_art\_service.py

└─connect ##采集协议链接

└─\_ art\_connect.py

├─env ##环境包

│ └─\_\_paho\_mqtt-2.1.0.tar.gz

├─lib ##第三方库

│ └─HslCommunication.dll

├─utils ##工具函数

│ └─\_\_baseLogger.py

│ └─\_\_util.py

├─logs ##日志

├─databases ##日志等其他基本信息

│ |─\_\_db.sqlite3

└─settings.py ##程序启动相关参数

└─run.py ##程序入口

└─requirements.txt ##组件版本库

└─README.md ##使用说明

└─通用数采程序设计方案v2.docx ##方案说明