# 调度平台代码解释文档

程序包含主文件main4-3.py和ui.py、message.py两个由PyQt5文件自动生产的ui界面描述文件。

main程序中除主要功能类外，还有三个函数：

1. applyYaml(file)

部署yaml文件，file字符串类型，为yaml文件路径。

1. deleteYaml(file)

删除该yaml部署的应用，file字符串类型，为yaml路径。

1. convert\_bps(bps)

将输入的以bps为单位的速率值（整型），转成合适的单位，返回内容：速率值（整型），单位（字符串）。

## 用户交互界面

该部分相关函数主要位于MainDialog类，该类负责ui界面数据显示、刷新及与用户进行逻辑交互。此外还有newWindow类负责弹出客户端请求；QChartViewPlot类显示实时曲线图；networkTopology类绘制网络拓扑图。

## MainDialog类

1. \_\_init\_\_(self, parent=None)

初始化ui界面，创建曲线图和消息窗口，并将其隐藏；声明使用到的变量。

1. setSocket(self, Socket)

储存与客户端建立的socket连接

1. updateTable(self, table, datas)

将数据更新到ui的表格中，table为对应的表格，属于QTableWidget类型；

data为对应的数据，数组（list类型），一维为行，二维为列

1. updateNodeTable(self, datas)

将数据更新到节点信息表格，数据要求同（3）

双击打开曲线图后，数据同步更新到曲线图。

1. updateAccessTable(self, datas)

将数据更新到接入设备信息表格，数据要求同（3）

双击打开曲线图后，数据同步更新到曲线图。

1. updatePodsTable(self, datas)

将数据更新到pod业务应用信息表格，数据要求同（3）

双击打开曲线图后，数据同步更新到曲线图。

1. showPlot(self, row)

由双击表格某行触发事件，展示曲线图，row为对应触发事件的行的序数，触发时自动填入。

1. bitrate(self)

在显示空口速率曲线图时，点击速率切换按钮，触发此事件，用于切换显示的上下行速率。

1. RB(self)

预留函数，暂未使用，用于显示rb利用率曲线图。

1. hidePlot(self)

曲线图显示状态时，点击退出按钮触发事件，隐藏曲线图。

1. showPods(self, row)

单击node节点表格的某一行触发，显示该节点的所有pod的信息，row为所点击行的序数，自动填写。

1. createTopo(self, nodes)

创建拓扑图，nodes为二维数组类型，输入node节点的连接情况。输入的数据为节点对，既相互连接的两个节点。输入节点为[[node1,’’],[node2,’’],[node3,’’]...]时，这些节点均与处于图像中心的交换机相连。

1. changeAi(self)

切换ai模型，将ai业务由k8s-node3迁移到k8s-node1，若该业务请求为客户端发起时，将新的拉流地址反馈给客户端。

1. update(self)

数据更新函数，将数据获取模块传输过来的数据定时更新到ui界面上，该函数的触发由Qtimer定时器控制，定时器声明位于\_init\_函数。

1. init(self)

初始化函数，由ui界面的初始化按钮触发，清除所有数据，清空ui界面，同时关闭所有业务。

1. addRecord(self, content)

日志更新函数，content为字符串，在ui界面右下角有一个日志显示窗口，用于显示程序启动之后或初始化之后的所有状态改变记录，该函数用于将文本在该窗口中显示。

1. setting(self)

业务部署函数，由ui界面的部署按钮触发，部署下拉列表中选中的应用。

1. record(self, content)

将需要在日志窗口输出的文字加入到待输出序列，content为字符串，该函数的创建用于保证日志按顺序依次输出。

* 1. **newWindow类**

1. \_\_init\_\_(self, main)

初始化客户端请求显示类，main为主页面类，main的引入用于与主类的数据通信。

1. showMessage(self, request)

由主界面类update函数调用，当客户端发来业务请求后，运行该函数，用于弹出客户端请求窗口，request为字符串。

1. accept(self)

用户点击同意后触发，分别根据“同意转译”和“同意部署”两个过程进行不同的功能执行。

同意转译后进行转译，同意部署后调用部署函数。

1. send\_task(self, task, send='')

用于部署转译后的业务，业务部署完成后将结果返回至客户端。

task和send均为字符串，task为转译后的业务名称，send为给客户端的回复内容。

1. reject(self)

拒绝转译或部署时，关闭请求弹窗，返回客户端结果。

* 1. **QChartViewPlot类**

1. \_\_init\_\_(self, parent=None)

初始化画布和坐标轴。

1. setText(self, text)

更新图表文字，text为长度为4的数组，需要设置的文字:[标题，X轴标签，Y轴标签，曲线图例]

1. reset(self)

重置画布内容和曲线

1. handle\_update(self, ydata, resize=False)

更新曲线数据，ydata为y轴更新的数据，resize为是否根据曲线的数值改变y轴的范围，若为否，则y轴的范围在0至100之间。

* 1. **networkTopology类**

1. \_\_init\_\_(self, parent=None)

初始化拓扑图画布。

1. initCenter(self)

在画布中心画出交换机节点。

1. setSize(self, width, height)

设置画布的大小。

1. updateNode(self, node, pos)

在画布上画出一个节点，node为字符串，表示节点名称；pos为整型一维数组，表示节点的位置。

1. updateLine(self, node)

在画布上画出节点连接线，node为一维数值，存有两个节点的名称，若数值为：[‘center’,node]，将node节点与交换机连接。

1. changeState(self, node, state)

更新节点的状态，node为字符串，代表节点名称；state为节点状态，1代表在线，0代表掉线。

## 数据获取

数据获取由update类负责，主要功能为实时获取各项数据用于ui界面显示，并实现调用基站api对基站进行参数调整。

1. \_\_init\_\_(self, window)

初始化变量，window为主界面所在的类，定义业务名称与对应yaml代码的关系，建立ssh连接用于获取ai时延信息。

1. start\_loop(self)

开启获取数据的循环程序。

1. update\_data(self)

获取信息的主循环函数，负责调用各类数据获取函数，处理数据并将数据更新到主界面。同时具有调度功能，同步各业务的状态，对业务进行部署和删除。

1. pods\_status(self)

获取业务pod的状态，返回二维数组，包含所有运行的pod，数据类型为多组[pod名称,pod状态]数组。

1. getNodeInformation(self)

获取node节点的信息，通过调用k8s的api，获取节点的名称、ip地址、cpu核数、节点准备状态、cpu使用个数、总内存、cpu利用率等信息，每个节点以字典类型储存，返回包含多个节点信息的数组。

1. getPodsInformation(self)

获取pod的信息，通过调用k8s的api，获取pod的名称、所属node节点的名称、cpu使用核数、内存使用大小、cpu利用率、内存利用率，每个pod以字典类型储存，返回包含多个pod信息的数组。

1. connectByWebsocket(self, url, data, isRecv=True)

异步函数，与基站建立websocket通信连接。输入参数中，url为建立连接的网址，data为发送信息内容，url和data均为字符串。isRecv表示是否等待接收对端回复信息，默认为是。

1. ue\_get(self, n)

获取用户设备的信息，n为整型，表示基站编号，目前为1和2两个基站。通过向对应基站发送信息，获取连接该基站的用户设备。返回结果包含ueid、imsi、上行速率、下行速率，每个ue使用字典保存信息，返回包含多个ue信息的数组。

1. resourceLocks\_get(self)

获取基站的资源块设置，返回上行和下行的资源块，获取数据暂未使用。

1. setRate(self)

设置基站最大空口速率。

1. decode\_delay(self, line)

从ai程序的性能指标记录文件中解析出时延，line为文件中的一行，字符串类型。返回结果为字典类型，包含时延和识别到的物品。

1. decode\_item(self, line)

从ai程序的性能指标记录文件中解析出每个物品的置信度，line为文件中的一行，字符串类型。返回内容为包含物品和置信度的二维数组。

1. AI\_get(self)

使用ssh获取ai性能指标记录文件最后一行内容，调用（11）（12）函数解析结果，将得出指标输出，目前返回结果为时延。

1. update\_task(self, datas)

用于更新ftp业务和ai业务的性能指标，datas为（8）获取的ue数据，用于获取ftp服务的下行空口速率；ai时延使用（13）获取。在ftp或ai业务部署之后，该函数的指标展示功能生效。

1. getEnb(self)

获取基站连接的ue数量，返回结果为ue个数。该结果会作为关闭基站的参考，在基站迁移过程中，待关闭基站的ue个数为0代表用户迁移完成，可以关闭。

1. setDB(self, tx\_gain, n)

设置基站的信号增益，tx\_gain为想要设置的信号增益，n为基站编号（目前为1和2）。

## 客户端通信

负责与客户端通信的类为listen，功能为监听客户端请求和回复客户端信息。

1. \_\_init\_\_(self, window)

创建socket服务器，初始化变量。

1. start\_loop(self)

绑定服务器的ip地址和端口号，开始监听循环。

1. listenRequest(self)

实时监听客户端信息，收到信息后记录客户端地址和端口，若收到信息为”connect”表示信息用于测试连接，不做反应；若不为”connect”，代表接收到的为客户端的业务请求，将请求上传到主界面，开启弹窗询问事件。

1. sendMsg(self, msg)

向客户端回复信息，msg为向客户端回复的信息，字符串类型，发送地址为函数（3）中记录的地址。

1. isConnect(self)

测试与客户端连接是否正常，若通信正常返回True，否则返回Flase。

## 主函数

该部分包含类的实例化，重要参数的定义，和线程的启动。其中MainDialog为主线程，用于用户界面的展示与交互，update类的运行和listen类的运行分别为另外两个线程。