沙钢平台CMS数据接口标准

# 数据集成采集方式

沙钢平台所需数据全部通过WebAPI（REST）接口服务方式，接收各个分厂现场不同CMS厂家上传的数据，数据统一采用Json格式按定义的协议规则，调用WebAPI中相应的服务接口进行数据的读写操作。

采用WebAPI这种接口通讯方式，可以有效的减少不同CMS厂家在数据对接时的开发难度，加快数据集成的过程。

上传数据基地址格式如下：

[http://IP:8124/api/services](http://IP:8019/DataPlatform/V1/Updater)/app/VDiagnosis/

其中IP是值用于部署接口服务的服务器IP地址，8019是端口号（可调整）

# 主数据编码规则

1. 集团、厂、产线编码

根据现场给定的唯一固定编码；

集团：固定2位，沙钢01

厂：固定2位，炼钢厂01

产线：固定2位，转炉特钢01、高线02、棒三03、棒四04

例如：010102，表示沙钢炼钢厂高线

1. 设备编码

根据现场给定的设备唯一固定编码；提供4位编码0001；

需给出设备清单表（示例）：

|  |  |
| --- | --- |
| **设备名称** | **编码** |
| 预精轧机 | 0101020001 |

1. 部件编码

提供2位固定编码，根据现场的部件做相应的增加；

01 主轴

02 齿轮箱

03 发电机

04 塔筒

05 泵

06 电机

07 压缩机

08 减速箱

09 增速箱

10 辊箱

11 分配箱

12 风机

13 锥箱

14 联轴器

15 风冷

16 搅拌机

17 液力耦合器

18 轧辊

1. 测点编码

提供4位固定编码，第一位为测点类型，后面3位按照顺序从001依次递增即可；

测点类型：

0-振动测点

1. 工艺测点

测点完整编码：

0001 振动测点1

0002 振动测点2

1003 工艺测点1

1004 工艺测点2

1. 测点方向

提供1位固定编码，根据现场的实际情况做相应的增加；

0-无方向

1-H 水平方向

2-V 垂直方向

3-A 轴向

4-R 径向

注：若同一传感器安装位置，水平方向、径向、垂直方向有2个或者更多个传感器，从机舱面向风轮，参照二维坐标轴0~360°，依次命名为1、2、3等。

1. 采集定义

提供1位固定编码，从0-9依次递增；

|  |  |  |
| --- | --- | --- |
| **名称** | **编码** | **备注** |
| 8K速度波形(10~1000) | 0 | 采集定义名称必须按格式包含采集长度、信号类型(加速度、速度、位移)、波形类型(波形、频谱、包络)、上下限频率 |
| 128K加速度波形(0~10000) | 1 |  |
| 4K位移波形(0-10000) | 2 |  |
| 1600线加速度频谱(2-2000) | 3 |  |
| 200线加速度包络(5000-15000) | 4 |  |

# 数据接口规范

此处的数据指各种采样数据，包括振动、温度、转速以及各种设备运行的过程数据。将根据数据的种类分别定义不同的数据结构，基本原则都是通过将数据打包成Json格式后进行数据的上传,所有接口通过HTTP发起请求，返回状态码为200时，代表HTTP请求接口正常。

1. 振动数据

振动数据是指由CMS系统采集的波形数据，包括时域波形，频谱，长波形等数据。振动数据分为元数据和波形块，元数据用于描述波形块的主要信息，如波形长度，采样频率，信号类型等，波形块部分应转换为16进制整形并以二进制方式进行压缩，减小上传时对网络带宽的要求以及减小存储时需要的存储空间。

由于振动数据较大，如长波形，单条数据可能达到8Mbyte,我们采取分包上传的方式进行数据的传输，支持断点续传，上传振动数据时，首先以Json格式上传元数据，如果服务端接收成功，就会返回一个唯一的标识以及当前已上传过的波形块大小(若已上传过，中途断了)，作为接下来分包上传波形块的参数，波形块以二进制方式上传。

* 数据结构定义

**元数据**

|  |  |  |
| --- | --- | --- |
| **名称** | **数据类型** | **说明** |
| Code | String | 编码，根据编码规则解析出来对应的设备、测点、测量定义等信息；例如010102000102000121 |
| FullPath | String | 完整名称路径，以\\分割，例如，沙钢\\炼钢厂\\高线\\预精轧机\\齿轮箱1\\振动测点1V\\128K加速度波形(0~10000)10000指的是分析频率，采用数值显示方式） |
| MeasDate | DateTime | 采样时间(北京时间转化成字符串:”yyyy-MM-dd HH:mm:ss” 如“2019-08-27 14:04:05”) |
| MeasValue | Float | 采样值(小数点后长度2位)为采集定义波形对应的采样值，加速度-有效值，速度-有效值，位移-峰峰值) |
| WaveLength | Int | 波形长度，采样点数 |
| SignalType | Int | 信号类型 0-加速度 1-速度 2-位移 |
| SampleRate | Float | 采样频率(小数点后长度最多2位) |
| RPM | Float | 运行转速，小数2位 |
| Unit | String | 工程单位，速度：mm/s，加速度：m/s²，位移：um |
| ConvertCoef | Float | 转换系数(固定为0.39) |
| Resolver | Int | 解析方式 0-缺省解析方式（高低位互换）,1-没有进行高低位互换(鸣志要求固定为0，别的厂家可自由选择) |

* 接口调用方式

接口地址:

[http://IP:8124/api/services](http://IP:8019/DataPlatform/V1/Updater)[/app/VDiagnosis/VibMetaData](http://IP:8019/DataPlatform/V1/Updater/VibMetaData)

参数：无

方法：post

body内容：按结构定义将数据打包成Json格式，

示例：

{

"Code": "010102000102000121",

"FullPath": "沙钢\\炼钢厂\\高线\\预精轧机\\齿轮箱1\\振动测点1V\\128K加速度波形(0~10000)",

"MeasDate": "2019-08-27 14:04:05",

"MeasValue": 3.82,

"WaveLength": 131072,

"SignalType": 0,

"SampleRate": 5120,

"RPM": 1000.00,

"Unit": "m/s^2",

"ConvertCoef": 0.39,

"Resolver": 0

}

返回值示例：

{

"WaveTag": "9e3e009a-f138-dcd6-6323-c768dc533b2f",

"CurrLength": 0

}

返回值说明：

|  |  |  |
| --- | --- | --- |
| **名称** | **数据类型** | **说明** |
| WaveTag | Guid | 表示此波形的唯一标识 |
| CurrLength | Int | 表示此波形在服务器上已接收的长度 |

**波形块**

波形块内容较长，如一个8K长度的时域波形，如果按浮点数存储，那么每个点需要4个字节，总共需要32KB的存储空间，在这里需要对波形块进行压缩处理，将浮点数转换成16位整形数字，并转换成2进制格式，单条数据可以抽取一个转换系数(ConvertCoef),那么波形块只需要16KB的存储空间，可以极大降低对带宽和存储的压力。

如波形中第一个16位整形数字为876，转换成二进制后得到两个字节的数组，其中arr[0]=108,arr[1]=3,以此类推，8K波形需要arr[16384]长度的字节数组空间，如果进行高低位互换，请在元数据定义中设置Resolver=0,不互换的设置Resolver=1

* 接口调用方式

循环调用此接口，如1个1M长波形，二进制格式的数组为2M，若按128KB每包拆分，总共需要16次可以上传完毕。（波形长度128K及小于128K时不分包，超过则分包）

接口地址:

[http://IP:8124/api/services](http://IP:8019/DataPlatform/V1/Updater)[/app/VDiagnosis/VibWaveData](http://IP:8019/DataPlatform/V1/Updater/VibWaveData)

请求参数说明：

|  |  |  |
| --- | --- | --- |
| **名称** | **数据类型** | **说明** |
| WaveTag | Guid | 表示此波形的唯一标识 |
| Length | Int | 内容总长度(byte[]的长度) |
| CurrIndex | Int | 当前上传的索引，即在总内容包中的索引地址 |
| BlockSize | Int | 当前上传的块大小 |

请求示例：

[http://IP:8124/api/services](http://IP:8019/DataPlatform/V1/Updater)[/app/VDiagnosis/VibWaveData](http://IP:8019/DataPlatform/V1/Updater/VibWaveData)?WaveTag=9e3e009a-f138-dcd6-6323-c768dc533b2f&Length=131072&CurrIndex=0&BlockSize=131072

方法：post

body内容：将二进制格式的波形块，拆分成合适大小的块，如128KB一个包

请求上传包示例：

[1,0,1,0,1,0,1,0,1]

返回值示例：

{

"WaveTag": "9e3e009a-f138-dcd6-6323-c768dc533b2f",

"CurrLength": 131072

}

返回值说明：

|  |  |  |
| --- | --- | --- |
| **名称** | **数据类型** | **说明** |
| WaveTag | Guid | 表示此波形的唯一标识 |
| CurrLength | Int | 表示此波形在服务器上已接收的长度 |

1. 时标数据

时标数据是指转速时标/转速脉冲，AD时标，用于对数据进行阶次分析（特别是轧机分析）时所必须的数据，多通道同步采集的振动数据应共享一组时标数据，时标数据和振动数据需要建立对应关系，如8通道在某时刻同步采集的振动数据，只会有一组时标数据，这8个通道的的振动数据都指向同一组时标数据。这样可以避免因为重复存储造成的存储空间浪费。

* 数据结构定义

|  |  |  |
| --- | --- | --- |
| **名称** | **数据类型** | **说明** |
| Code | string | 编码，根据编码规则解析出来对应的设备、测点、测量定义等信息；例如010102000102000121 |
| FullPath | String | 完整名称路径，以\\分割，例如，沙钢\\炼钢厂\\高线\\预精轧机\\齿轮箱1\\振动测点1V\\128K加速度波形(0~10000) |
| MeasDate | DateTime | 采样时间(北京时间转化成字符串:”yyyy-MM-dd HH:mm:ss” 如“2019-08-27 14:04:05”) |
| ADTS | Long[] | AD时标，采集振动波形每2048个点时读取一次计数器值记录 |
| RPMTS | Long[] | 转速时标，转一圈读取一次计数器值记录 |

* 接口调用方式

接口地址:

[http://IP:8124/api/services](http://IP:8019/DataPlatform/V1/Updater)[/app/VDiagnosis/T](http://IP:8019/DataPlatform/V1/Updater/ProcessDatas)imeIndexData

参数：无

方法：post

body内容：按结构定义将数据打包成Json格式

请求上传包示例：

{

"Code": "010102000102000121",

"FullPath": "沙钢\\炼钢厂\\高线\\预精轧机\\齿轮箱1\\振动测点1V\\128K加速度波形(0~10000)",

"MeasDate": "2019-08-27 14:04:05",

"ADTS": [1, 2, 3, 4, 5, 6, 5, 4, 2],

"RPMTS": [3, 4, 5, 3, 5, 7, 2, 4, 8]

}

返回值：-1表示失败，其他表示成功

1. 工艺数据

工艺数据是指由CMS系统采集的工艺量数据，如风速，油温，功率，转速等。工艺数据由于单条数据比较小并且采集密度大，一般只包含测量点编码、采样时间和采样值，如果单条上传，不能很好的利用网络带宽，并且会造成上传速度慢等情况，需要通过打包上传的方式进行数据上传，也就是，一次将1批数据，如多台设备下在一定时间范围内所有的工艺数据都打包在一起后进行上传，规定为针对1个测点1分钟打包发送一次数据。

* 结构定义

PointProcessData结构

|  |  |  |
| --- | --- | --- |
| **名称** | **数据类型** | **说明** |
| Code | string | 编码，根据编码规则解析出来对应的设备、测点、测量定义等信息；例如01010200010210030 |
| FullPath | String | 完整名称路径，以\\分割，例如，沙钢\\炼钢厂\\高线\\预精轧机\\齿轮箱1\\工艺测点1 |
| TSDatas | 数组 自定义 | 测点包含的工艺数据集合（时序数据） |

TSData结构

|  |  |  |
| --- | --- | --- |
| **名称** | **数据类型** | **说明** |
| MeasDate | DateTime | 采样时间 |
| MeasValue | float | 采样值(小数点后长度2位) |

* 接口调用方式

接口地址:

[http://IP:8124/api/services](http://IP:8019/DataPlatform/V1/Updater)[/app/VDiagnosis/ProcessDatas](http://IP:8019/DataPlatform/V1/Updater/ProcessDatas)

参数：无

方法：post

body内容：按结构定义将数据打包成Json格式

请求上传包示例：

{

"Code": "010102000102000121",

"FullPath": "沙钢\\炼钢厂\\高线\\预精轧机\\齿轮箱1\\振动测点1V\\128K加速度波形(0~10000)",

"TSDatas": [{

"MeasDate": "2019-08-27 14:04:05",

"MeasValue": 32.65

}, {

"MeasDate": "2019-08-27 14:04:05",

"MeasValue": 40.43

}]

}

返回值：-1表示失败，其他表示成功

1. 指标采样值数据

指标采样值数据是指通过高密度采集时域波形（分钟级），计算出的各种特征指标，如总值，峭度，歪度，峰值等。与采集定义波形数据计算的指标不同，属于指标数值的采集（采集波形计算完指标后，波形不保存，保留指标），但指标采集比采集定义的间隔更加密集。此类数据由于单条数据比较小并且采集密度大，一般只包含测量点编码、采样时间，指标类型和采样值等参数，如果单条上传，不能很好的利用网络带宽，并且会造成上传速度慢等情况，需要通过打包上传的方式进行数据上传，也就是，一次将1批数据，如多台设备下在一定时间范围内所有的采样值数据都打包在一起后进行上传，针对一个测点采样值5分钟打包发送一次。

* 结构定义

PointSampleData结构

|  |  |  |
| --- | --- | --- |
| **名称** | **数据类型** | **说明** |
| Code | string | 编码，根据编码规则解析出来对应的设备、测点、测量定义等信息；例如01010200010200012 |
| FullPath | String | 完整名称路径，以\\分割，例如，沙钢\\炼钢厂\\高线\\预精轧机\\齿轮箱1\\振动测点1V |
| SampleDatas | 数组 自定义 | 测点包含的采样值数据集合（时序数据） |

SampleData结构

|  |  |  |
| --- | --- | --- |
| **名称** | **数据类型** | **说明** |
| MeasDate | DateTime | 采样时间 |
| DateType | Int | 指标类型 1-总值指标（加速度-有效值，速度-有效值，位移-峰峰值），2-峭度，3-歪度，4-频带能量1,5-频带能量2，6-峰值 …… |
| Condition | int | 工况(固定为0) |
| MeasValue | float | 值(小数点后长度2位) |

* 接口调用方式

接口地址:

[http://IP:8124/api/services](http://IP:8019/DataPlatform/V1/Updater)[/app/VDiagnosis/SampleDatas](http://IP:8019/DataPlatform/V1/Updater/SampleDatas)

参数：无

方法：post

body内容：按结构定义将数据打包成Json格式

请求上传包示例：

{

"Code": "010102000102000121",

"FullPath": "沙钢\\炼钢厂\\高线\\预精轧机\\齿轮箱1\\振动测点1V\\128K加速度波形(0~10000)",

"SampleDatas": [{

"MeasDate": "2019-08-27 14:04:05",

"DateType": 1,

"Condition": 0,

"MeasValue": 3.58

}, {

"MeasDate": "2019-08-27 14:04:05",

"DateType": 2,

"Condition": 0,

"MeasValue": 4.58

}]

}

返回值：-1表示失败，其他表示成功

# 数据接口通讯要求

通讯协议按照列出的WebAPI的方式；

CMS数据方需要按照每1小时内的数据质量要求送到沙钢数据平台(即6点钟采集的数据必须在7点之前发送给平台)。同时在可能出现断网或者故障的情况下，出现数据积压后以保证一天的数据在2小时内传输完毕，前期测试性能需要测试这块；

在整个链路、程序出现故障的情况下，CMS需要进行数据缓存以保障数据不丢失，在故障恢复后继续将数据传输到数据平台。