

顺景 ERP 与第三方接口协议标准

2024 年 11 月

1. 概要

1.1. 概述

本技术文档是溯源系统与商家系统对接的接口（API）描述。

目标读者是商家的 IT 开发人员。

1.2. 重要说明

为了保证数据可以顺利交互，任何数据的交互都以 UTF-8 的编码进行。

2. 接口规范

2.1. 接口技术规范

1. 所有接口均采用 WebAPI，输入输出均使用 JSON 字符串，读取采用 HTTP GET 请求方法，接口调用 URL 示例：“<http://xxx.xxx.xxx.xxx:8080/webapi/业务对象名/Getxxx>”；

更新采用 HTTP PUT 请求方法，接口调用 URL 示例：“<http://xxx.xxx.xxx.xxx:8080/webapi/业务对象名/DBUpdate>”；

2. 和第三方的对接，不支持除 WebAPI 接口的其它类型，商户系统提供信息的对接方案，由商家通过 HTTP 发送请求，我方返回 JSON 格式的数据；

3. 当物料有多个计量单位时，所有接口中涉及到的数量字段，均使用物料的基本单位作为基准，APS 中仅存储基本单位数量；

4. 所有字段的命名采用驼峰命名规则，首字母小写；

5. 对于字段值为空字符串或 Null 的，在 JSON 数据包中也要包含，避免遗漏参数；

6. 考虑到与接口系统均安装在同一个局域网内，同时考虑接口的性能及安全性，采用预先授权 tokenId 方式进行接口权限校验，授权 tokenId 字符串由 APS 接口系统提供，APS 中维护 tokenId 的有效期；

7. 考虑到接口的通用性，所有 id 字段使用字符串类型。

2.2. 接口应用规范

1. ERP 接口 API 返回的 JSON 均含有：ErrorCode、ErrorMsg 属性，第三方可依此判断错误信息及错误代码；

2. 发起方负责调用的重复调用规避。ERP 接口 API 调用成功后 ErrorCode=0，标志成功，第三方应保证接口不重复调用、保证数据不重复传输。

2.3. 接口增量规范

1. 从 ERP 系统中基本资料的增量读取，须遵循 ERP 增量读取原则： ERPWebAPI 会提供 2 中增量读取方式： A.单笔 ID 更新， B.所有变更更新；
2. 业务数据类，比如采购订单、销售订单、生产订单等，这些数据 ERP 不提供建议增量调用，调用单据信息时需提供单据编号。

2.4. 组织编码（多帐套）约定

ERP 中和第三方对接过程中客户多帐套的组织编码在第一次登录验证中需指明（如果 ERP 客户在 Web.config 中配置了组织编码信息，则登录验证中可以不包含组织编码信息），后期组织编码由 ERP 提供的 Token 自动适配，在取得 Token 后第三方调用 WebAPI 不需提供组织编码。

3. 接口调用

3.1. 简单验证方式

3.1.1 生成密钥

调用前我们会对调用客户提供密钥 CorpNo 和 CorpId 格式如下：

- CorpNo: 020 (客户 CorpId 序号) ;
- CorpId: **IezQB0Esc1mN4Tf7Xw83U3tv7eEy33PJ**
- auth_sys_time: 当前时间戳，数字，精确到秒的 10 位数字 (参考 <https://tool.lu/timestamp/>) ;
- auth_sign_code: md5 加密 auth_sys_time 和 CorpId。
即：auth_sign_code=MD5("auth_sys_time&CorpId);
密钥=CorpNo&auth_sys_time&auth_sign_code
注意：每次生成密钥有效期有 2 分钟，根据 auth_sys_time 时间戳来判断的。

3.1.2 更新用户账号

- 请求 URL: <http://WebAPIServer/api/RTBusiness/UpdateCmsUser>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
userNo	帐号	string
userName	姓名	string
workNo	工号	string
cardNo	卡号	string
station	岗位	string
telmobile	联系电话	string
weixin	企业微信号	string
active	状态 0.停用;1.启用	int

注意: 需要在 headers 传入 Ticket, 参考如下代码:

```
method: 'post',
```

```
headers: {
```

```
'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code
```

```
'Accept': 'application/json',
```

```
'Content-Type': 'application/json'
```

```
}
```

```
//body 传入更新的 JSON 数据
```

```
body: JSON.stringify (传入 JSON)
```

- 输入示例:

```
{
```

```
    "userNo": "C001",
    "userName": "张三",
    "active": 1,
    "workNo": "41684122014",
    "cardNo": "",
    "station": "",
    "weiXin": "ZHANGSAN",
    "telmobile": "8208208820",
```

```
}
```

- 返回参数说明:

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码: 0 代表正确, 其它数代表不同的错误标识	int

- 返回示例:

```
{  
    "errorCode": 0,  
    "errorMsg": ""  
}
```

3.1.3 更新物料分类

- 请求 URL: <http://WebAPIServer/api/RTBusiness/UpdateCmsSpType>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
tList	存货类型 list	list
cList	存货和类型对照 list	list

tList:

typeNo	类型编码	string
typeName	类型名称	string
parentTypeNo	上一级分类编码	string

cList:

spNo	物料编码	string
typeNo	类型编码	string

注意: 需要在 headers 传入 Ticket, 参考如下代码:

method: 'post',

headers: {

'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code

```
'Accept': 'application/json',
'Content-Type': 'application/json'
}
```

//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)

- 输入示例:

```
{
  "tList": [
    {
      "typeNo": "M0001",
      "typeName": "原材料",
      "parentTypeNo": ""
    },
    {
      "typeNo": "M20",
      "typeName": "助剂",
      "parentTypeNo": "M0001",
    }
  ],
}
```

```
"cList":  
[  
  {  
    "typeNo": "M20",  
    "spNo": "C012"  
  },  
  {  
    "typeNo": "M20",  
    "spNo": "C116"  
  }  
]
```

● 返回参数说明：

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码：0 代表正确，其它数代表不同的错误标识	int

● 返回示例：

```
{  
  "errorCode": 0,  
  "errorMsg": ""  
}
```

3.1.4 更新物料档案

- 请求 URL: <http://WebAPIServer/api/RTBusiness/UpdateSp>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
typeNo	物料分类	string
spNo	物料编码	string
spTitle	物料名称	string
spName	物料牌号	string
spSpec	物料规格	string
colorNo	物料颜色	string
colorNumber	物料色号	string
shape	物料形态 0.粉体;1.颗粒;2.液体;3.其他	int
unitName	单位	string
canBom	可建 BOM 0.否;1.是	int
active	状态 0.停用;1.启用	int
brief	备注	string
spttNo	材质编码	string
ctrlLot	控制批号 0.否;1.是	int

注意: 需要在 headers 传入 Ticket, 参考如下代码:

method: 'post',

headers: {

'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code

```
'Accept': 'application/json',
'Content-Type': 'application/json'
}
```

//body 传入更新的 JSON 数据

body: JSON.stringify (传入 JSON)

- 输入示例:

```
{
    "typeNo": "M20",
    "spNo": "rtJK",
    "spTitle": "瑞堂的接口",
    "unitName": "个",
    "canBom": 0,
    "active": 1,
    "shape": 2,
    "spttNo": "AS001",
```

```
    "ctrlLot":1  
}
```

● 返回参数说明：

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码：0 代表正确，其它数代表不同的错误标识	int

● 返回示例：

```
{  
    "errorCode": 0,  
    "errorMsg": ""  
}
```

3.1.5 删除物料档案

- 请求 URL: <http://WebAPIServer/api/RTBusiness/DeleteSp>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
spNo	物料编码	string

注意: 需要在 headers 传入 Ticket, 参考如下代码:

```
method: 'post',
headers: {
```

```
'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code
```

```
'Accept': 'application/json',
'Content-Type': 'application/json'
}
```

```
//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)
```

- 输入示例:

```
{
  "spNo": "1122001",
}
```

- 返回参数说明:

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码: 0 代表正确, 其它数代表不同的错误标识	int

- 返回示例:

```
{
  "errorCode": 0,
  "errorMsg": ""
}
```

3.1.6 更新配方档案

- 请求 URL: <http://WebAPIServer/api/RTBusiness/UpdateBom>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
spNo	产品编码	string
bomRev	配方版本	string
brief	备注	string
dList	子表	list

dList:

matNo	材料编码	string
matPercent	用量占比%	decimal
brief	摘要	string

注意: 需要在 headers 传入 Ticket, 参考如下代码:

method: 'post',

headers: {

'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code

```
'Accept': 'application/json',
'Content-Type': 'application/json'
}
```

//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)

- 输入示例:

```
{
  "spNo": "1122001",
  "bomRev": "A3",
  "brief": "测试的配方版本",
  "dList":
  [
    {
      "matNo": "ST02",
      "matPercent": 31.24,
      "brief": "测试备注 fk"
    },
    {
      "matNo": "ST02",
      "matPercent": 16.77,
```

```
        "brief": "ffffffffffff2"
    }
]
}
```

● 返回参数说明：

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码：0 代表正确，其它数代表不同的错误标识	int

● 返回示例：

```
{
    "errorCode": 0,
    "errorMsg": ""
}
```

3.1.7 更新配方档案状态

- 请求 URL: <http://WebAPIServer/api/RTBusiness/UpdateBomState>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
spNo	产品编码	string
bomRev	配方版本	string
Active	状态 0.停用;1.启用	int

注意: 需要在 headers 传入 Ticket, 参考如下代码:

```
method: 'post',
```

```
headers: {
```

```
'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code
```

```
'Accept': 'application/json',
```

```
'Content-Type': 'application/json'
```

```
}
```

//body 传入更新的 JSON 数据

```
body: JSON.stringify (传入 JSON)
```

- 输入示例:

```
{
  "spNo": "1122001",
  "bomRev": "A3",
  "active": 1
}
```

- 返回参数说明:

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码: 0 代表正确, 其它数代表不同的错误标识	int

- 返回示例:

```
{
  "errorCode": 0,
  "errorMsg": ""
}
```

3.1.8 更新检验项目

- 请求 URL: <http://WebAPIServer/api/RTBusiness/UpdateQc>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
qcCatNo	分类编码	string
qcCatName	分类名称	string
brief	备注	string
dList	子表	list

dList:

qcItemNo	项目编码	string
qcItemName	项目名称	string
testStandard	检测要求	string
testcriterion	检测标准	string
testcondition	测试条件	string
valueType	类型 0.标准值;1.标准值+公差;2.上限值; 3.下限值;4.区间值;99.文字描述	int
stdValue	标准值	string
tolerance	公差	decimal
minValue	最小值	decimal
maxValue	最大值	decimal
qmUnit	单位	string

注意：需要在 headers 传入 Ticket，参考如下代码：

method: 'post',

headers: {

'Authorization': 3.1.1 中生成密钥，即：CorpNo&auth_sys_time&auth_sign_code

```

    'Accept': 'application/json',
    'Content-Type': 'application/json'
  }

```

//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)

- 输入示例：

```

{
  "qcCatNo": "qc002",
  "qcCatName": "物理性能",
  "dList":

```

```
[
  {
    "qcItemNo": "q01",
    "qcItemName": "颜色一致",
    "testStandard": "与客户提供的色版对比",
    "testcriterion": "",
    "testcondition": "",
    "valueType": 99,
    "stdValue": "合格",
    "tolerance": "",
    "minValue": "",
    "maxValue": "",
    "qmUnit": ""
  },
  {
    "qcItemNo": "q02",
    "qcItemName": "颗粒均匀度",
    "testStandard": "颗粒大小相差小于 0.1mm",
    "testcriterion": "",
    "testcondition": "",
    "valueType": 1,
    "stdValue": "3.2",
    "tolerance": 0.1,
    "minValue": 3.1,
    "maxValue": 3.3,
    "qmUnit": "毫米"
  }
]
```

● 返回参数说明：

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码：0 代表正确，其它数代表不同的错误标识	int

● 返回示例：

```
{
  "errorCode": 0,
  "errorMsg": ""
}
```

3.1.9 更新终检项目

- 请求 URL: <http://WebAPIServer/api/RTBusiness/UpdateQcEnd>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
spNo	产品编码	string

dList:

itemNo	项目编码	string
itemName	检验项目	string
qcCatNo	分类编码	string
testStandard	检测要求	string
testcriterion	检测标准	
testcondition	测试条件	
valueType	类型 0.标准值;1.标准值+公差;2.上限值; 3.下限值;4.区间值;99.文字描述	int
stdValue	标准值	string
tolerance	公差	decimal
minValue	最小值	decimal
maxValue	最大值	decimal
qmUnit	单位	string
brief	备注	string

注意：需要在 headers 传入 Ticket，参考如下代码：

method: 'post',

headers: {

'Authorization': 3.1.1 中生成密钥，即：CorpNo&auth_sys_time&auth_sign_code

'Accept': 'application/json',

'Content-Type': 'application/json'

}

//body 传入更新的 JSON 数据

body: JSON.stringify (传入 JSON)

● 输入示例：

{

 "spNo": "MT221",

 "dList":

 [

 {

```

    "itemNo": "e01",
    "itemName": "颜色一致",
    "qcCatNo": "qc002",
    "testStandard": "与客户提供的色版对比",
    "testcriterion": "",
    "testcondition": "",
    "valueType": 99,
    "stdValue": "合格",
    "tolerance": "",
    "minValue": "",
    "maxValue": "",
    "qmUnit": ""

},
{
    "itemNo": "q02",
    "itemName": "颗粒均匀度",
    "qcCatNo": "qc002",
    "testStandard": "颗粒大小相差小于 0.1mm",
    "testcriterion": "",
    "testcondition": "",
    "valueType": 1,
    "stdValue": "3.2",
    "tolerance": 0.1,
    "minValue": 3.1,
    "maxValue": 3.3,
    "qmUnit": "毫米"
},
]
}

```

● 返回参数说明：

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码：0 代表正确，其它数代表不同的错误标识	int

● 返回示例：

```
{
    "errorCode": 0,
    "errorMsg": ""
}
```

3.1.10 新增工令单

- 请求 URL: <http://WebAPIServer/api/RTBusiness/AddPlan>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
typeName	工单类型	string
planNo	工单号	string
planDate	工单日期	DateTime
mkNo	车间编号	string
spNo	产品编码	string
planQuan	计划数量	decimal
dueDate	需求日期	DateTime
lotNo	工单批号	string
soNos	销售订单	string
customerCode	客户	string
shipInfo	出货要求	string
labelInfo	标签要求	string
packagingInfo	包装要求	string
bomRev	Bom 版本	string
packNo	包装 Bom 编号	string
tailBatchFlag	尾料需配料 0.否;1.是	int
packWeight	每包重量	decimal
brief	备注	string
state	状态 0.未审核;1.已审核	int
isUrgent	紧急订单 0.否;1.是	int
khLevel	客户等级	string
codeTemplate	条码模板	string
erpKey	ERP 主键	string
spShape	产品形状	string
planLevel	工单等级	string
matList	材料明细	list
tailList	尾料明细	list
packList	包材明细	list

matList:

matNo	材料编码	string
matPercent	用量占比(%)	decimal
brief	备注	string

tailList:

参数名称	参数说明	类型
spNo	尾料编码	string
lotNo	尾料批号	string
useQty	尾料应发数量	decimal
brief	备注	string

packList:

matNo	包材编码	string
matQuan	单位用量	decimal
brief	备注	string

注意：需要在 headers 传入 Ticket，参考如下代码：

method: 'post',

headers: {

'Authorization': 3.1.1 中生成密钥，即：CorpNo&auth_sys_time&auth_sign_code

'Accept': 'application/json',

'Content-Type': 'application/json'

}

//body 传入更新的 JSON 数据

body: JSON.stringify (传入 JSON)

● 输入示例：

{

typeName:"量产订单",

planNo:"202411-00068",

planDate :"2024-11-28",

mkNo:"ATP001",

spNo:"4A34R",

planQuan:100,

dueDate:"2024-11-28",

lotNo:"1128001",

soNos:"",

customerCode:"",

shipInfo:"",

labelInfo:"",

packagingInfo:"",

bomRev:"",

packNo:"",

```
tailBatchFlag:1,
packWeight:20,
brief:"接口测试",
state:1,
isUrgent:0,
khLevel:"",
codeTemplate:"",
erpKey:"",
spShape:"",
planLevel: "",
matList:[
    {
        matNo:"AER001",
        matPercent:10,
        brief:"mt1"
    },
    {
        matNo:"BA001",
        matPercent:20,
        brief:"mt2"
    },
],
tailList:[
    {
        spNo:"111",
        lotNo:"111",
        useQty:10,
        brief:"tail1"
    },
    {
        spNo:"111",
        lotNo:"222",
        useQty:20,
        brief:"222tail"
    },
],
packList:[
    {
        matNo:"SJ0021",
        matQuan:3,
        brief:"bc"
    },
]
```

```
{  
    matNo:"MAT40-0001",  
    matQuan:2,  
    brief:"bc2"  
}  
]  
}
```

● 返回参数说明：

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码：0 代表正确，其它数代表不同的错误标识	int

● 返回示例：

```
{  
    "errorCode": 0,  
    "errorMsg": ""  
}
```

3.1.11 删除工令单

- 请求 URL: <http://WebAPIServer/api/RTBusiness/DeletePlan>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
planNo	工令单号	string

注意: 需要在 headers 传入 Ticket, 参考如下代码:

```
method: 'post',
headers: {
```

'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code

```
'Accept': 'application/json',
'Content-Type': 'application/json'
}
```

//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)

- 输入示例:

```
{
  "planNo": "202411-00068",
}
```

- 返回参数说明:

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码: 0 代表正确, 其它数代表不同的错误标识	int

- 返回示例:

```
{
  "errorCode": 0,
  "errorMsg": ""
}
```

3.1.12 无来源收料保存

- 请求 URL: <http://WebAPIServer/api/RTBusiness/UpdateNoReceiveTask>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
receiveDate	收料日期(yyyy-mm-dd)	string

dList:

spNo	存货编码	string
lotNo	批号	string
whNo	仓库编码	string
quan	收料数量	decimal
crtDate	批号日期(yyyy-mm-dd)	string

注意: 需要在 headers 传入 Ticket, 参考如下代码:

```
method: 'post',
headers: {
  'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code
  'Accept': 'application/json',
  'Content-Type': 'application/json'
}
```

```
//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)
```

- 输入示例:

```
{
  "receiveDate": "2025-01-02",
  "dList": [
    {
      "spNo": "AR002",
      "lotNo": "dx001",
      "whNo": "A5",
      "quan": 2,
      "crtDate": "2025-01-01",
    }
  ]
}
```

- 返回参数说明:

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码: 0 代表正确, 其它数代表不同的错误标识	int

- 返回示例:

```
{
```

```
"errorCode": 0,  
"errorMsg": ""  
}
```

3.1.13 备料收料保存

- 请求 URL: <http://WebAPIServer/api/RTBusiness/UpdateReceiveTask>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
recReqId	收料申请 id	int
receiveDate	收料日期(yyyy-mm-dd)	string
whNo	仓库编码	string

dList:

spNo	存货编码	string
lotNo	批号	string
quan	收料数量	decimal
crtDate	批号日期(yyyy-mm-dd)	string

注意: 需要在 headers 传入 Ticket, 参考如下代码:

```
method: 'post',
headers: {
  'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code
  'Accept': 'application/json',
  'Content-Type': 'application/json'
}
```

```
//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)
● 输入示例:
{
  "recReqId": "774654",
  "whNo": "A5",
  "receiveDate": "2025-01-02",
  "dList": [
    {
      "spNo": "AR002",
      "lotNo": "dx001",
      "quan": 2,
      "crtDate": "2025-01-01",
    }
  ]
}
```

- 返回参数说明:

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码: 0 代表正确, 其它数代表不同的错误标识	int

- 返回示例：

```
{  
    "errorCode": 0,  
    "errorMsg": ""  
}
```

3.1.14 锐派码垛完成

- 请求 URL: <http://WebAPIServer/api/cmsDeviceData/REVOPACFinishPalletizing>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明		类型
agvDeviceCode	AGV 设备标识, 非空		String
palletId	共享托盘 ID, 非空		String
deviceCode	码垛机设备标识, 非空		String
jobId	MES 生产任务 ID, 非空		Int
list	打包明细, 非空		
	参数名称	参数说明	object
	bagNums	包号	String
	Quan	数量	Decimal

注意: 需要在 headers 传入 Ticket, 参考如下代码:

method: 'post',

headers: {

'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code

```

    'Accept': 'application/json',
    'Content-Type': 'application/json'
  }
//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)

```

- 输入示例:

```

{
  "agvDeviceCode": "AGV001",
  "palletId": "P00001",
  "deviceCode": "Palletizing",
  "jobId": 10000,
  "list": [
    {
      "bagNums": "A001",
      "quan": 20
    },
    {
      "bagNums": "A002",
      "quan": 30
    }
  ]
}

```

● 返回参数说明：

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码：0 代表正确，其它数代表不同的错误标识	int

● 返回示例：

```
{  
    "errorCode": 0,  
    "errorMsg": ""  
}
```

3.1.15 锐派托盘缺少

- 请求 URL: <http://WebAPIServer/api/cmsDeviceData/REVOPACLackPallet>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
agvDeviceCode	AGV 设备标识, 非空	String
palletType	托盘类型 1.表示母托盘, 2 表示子托盘, 非空	Int

注意: 需要在 headers 传入 Ticket, 参考如下代码:

```
method: 'post',
headers: {
  'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code
  'Accept': 'application/json',
  'Content-Type': 'application/json'
}
//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)
```

- 输入示例:

```
{
  "agvDeviceCode": "AGV001",
  "palletType": 1
}
```

- 返回参数说明:

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码: 0 代表正确, 其它数代表不同的错误标识	int

- 返回示例:

```
{
  "errorCode": 0,
  "errorMsg": ""
}
```

3.1.16 AGV 任务完成

- 请求 URL: <http://WebAPIServer/api/cmsDeviceData/FinishAGVTask>
- 请求方式: Post
- 输入参数说明:

参数名称	参数说明	类型
taskId	MES 系统生成的 AGV 任务 ID, 非空	Int

注意: 需要在 headers 传入 Ticket, 参考如下代码:

```
method: 'post',
headers: {
```

'Authorization': 3.1.1 中生成密钥, 即: CorpNo&auth_sys_time&auth_sign_code

```
'Accept': 'application/json',
'Content-Type': 'application/json'
}
//body 传入更新的 JSON 数据
body: JSON.stringify (传入 JSON)
```

- 输入示例:

```
{
    "taskId": 7924601
}
```

- 返回参数说明:

参数名称	参数说明	类型
errorMsg	错误信息	string
errorCode	错误代码: 0 代表正确, 其它数代表不同的错误标识	int

- 返回示例:

```
{
    "errorCode": 0,
    "errorMsg": ""
}
```