

# rhinoceros系统 — webservice 接口定义

Revisions:

```
20160501 zhangbin created  
version: 0.2
```

## 目录

### 1. 常数定义

#### 1.1 [错误代码定义](#)

### 2. 接口定义

#### 2.1 [用户登入 login](#)

#### 2.2 [车辆在线状态统计 vehicle\\_statistics](#)

#### 2.3 [车辆热区查询 vehicle\\_hotspots](#)

#### 2.4 [地理可视区域车辆状态查询 vehicles\\_by\\_geometry\\_rect](#)

#### 2.5 [车辆名称筛选 vehicle\\_filter\\_names](#)

#### 2.6 [查询指定车辆状态查询 vehicles\\_status](#)

### 3. 测试

```
curl -X PUT/POST/DELETE url
```

## 1. 常数定义

### 1.1 错误代码定义

code	msg
-----	
0	成功
1001	参数不完整或数据内容损坏
1002	目标对象不存在
1003	无权限操作
1004	令牌错误
1005	会话错误
1006	用户账号或密码错误
1007	应用授权错误
2001	系统内部运行故障

## 2. 数据封包

接口调用返回格式：

```
{
  status  状态码    0 : succ; others : error
  errcode 错误码
  errmsg  错误信息
  result  数据内容
}
```

采用json编码(UTF-8)

## 2. 接口定义

### 2.1 用户登录

**名称:**

login( app\_key , app\_secret , user\_id )

**描述:**

应用系统登录，每当客户登录时到应用系统时，应用系统应向rhino系统进行用户登录注册，并获得访问票据 ticket。

**Request**

```
URL: /rhino/auth/tickets
Method: POST
Headers:
Character Encoding: utf-8
Content-Type: x-www-form-urlencoded
Query Parameters:
  - app_key      应用key
  - app_secret   应用登录密码
  - user_id      应用用户标识
```

## Response

```
Headers:
Character Encoding: utf-8
Content-Type: application/json
Data:
  - status  状态码 0 : succ; others : error
  - errcode 错误码
  - errmsg  错误信息
  - result  登录token
```

## Examples:

```
Request:
  /rhino/auth/tickets
Response:
  {
    status:0,
    result: oiwurwurioqweuirqwerjqwewriu==
  }
```

## Remarks

## 2.2 车辆在线状态统计

名称:

vechicle\_statistics ( )

描述:

获得用户所有在线车辆状态信息

## Request

```
URL: /rhino/vehicles/statistics
Method: GET
Headers:
Character Encoding: utf-8
Content-Type: x-www-form-urlencoded
Query Parameters:
  - ticket : access ticket 登录时获取的票据
```

## Response

```
Headers:
Character Encoding: utf-8
Content-Type: application/json
Data: (object)
  - running 运行数
  - stopped 停止数
  - offline 离线数

examples:
```

## Examples:

```
Request:
  /rhino/vehicles/statistics/?ticket=12312j%3D
Response:
  {
    status:0,
    result:{
      running: 200,
      stopped: 10,
      offline: 5
    }
  }
```

## Remarks

## 2.3 车辆热区

名称:

vehicle\_hotspots()

## 描述:

获取地理可视区域的车辆热点数

## Request

```
URL: /rhino/vehicles/hotspots
Method: GET
Headers:
Character Encoding: utf-8
Content-Type: multipart/form-data
Query Parameters:
- ticket      access ticket
- scale       地图缩放级别 (1-25)
- rect        地图可视区域 x,y,w,h 格式
- limit       返回最大热区数量
```

## Response

```
Headers:
Character Encoding: utf-8
Content-Type: application/json
Data:(array)
- id  热区编号
- lon 经度
- lat 纬度
- scale 缩放级别
- running 运行数量
- stopped 停止数量
- offline 离线数量
```

将热区的车辆总数降序输出

## Examples:

```
Request:
/rhino/vehicles/hotspots/?
ticket=12312j%3D&scale=20&rect=121.101,31.22,2,2.5&limit=20

Response:
{
  status:0,
  result:[
    {
      id:    1001,
      lon:   121.22,
      lat:   31.10,
      scale: 18,
      running: 200,
      stopped: 10,
      offline: 5
    },
    ...
  ]
}
```

## Remarks

## 2.4 地理可视区域车辆状态查询

名称:

get\_vehicles\_by\_geometry\_rect()

描述:

获取地理可视区域的车辆状态列表

## Request

```
URL: /rhino/vehicles/geometry
Method: GET
Headers:
Character Encoding: utf-8
Content-Type: multipart/form-data
Query Parameters:
- ticket    access ticket
- scale     地图缩放级别 (1-25)
- rect      地图可视区域 x,y,w,h 格式
- circle    地图可视区域 x,y,radius 格式
- limit     [可选] 返回最大数量 ( 0: 所有 )
```

## Response

```
Headers:
Character Encoding: utf-8
Content-Type: application/json
Data:(array)
  - id          车辆标识(车牌)
  - lon         经度
  - lat         纬度
  - speed       速度
  - direction   方向
  - altitude    高度
  - time        gps定位时间( timestamp 1970~)
  - status      1:在线, 2:停止,3: 离线
  - address     地址
  - text        其他描述
```

## Examples:

```
Request:
  可视矩形区域车辆查询
  /rhino/vehicles/geometry/?
  ticket=12312j%3D&scale=20&rect=121.101,31.22,2,2.5&limit=0
```

```
  可视圆形区域车辆查询
  /rhino/vehicles/geometry/?
  ticket=12312j%3D&scale=20&circle=121.101,31.22,2&limit=0
```

```
Response:
{
  status:0,
  result:[
    {
      id: 贵K56712,
      lon: 121.22,
      lat: 31.10,
      speed: 18,
      direction: 200,
      altitude: 10,
      time: 1493702543
      status: 1
      address: 上海市长宁之路200号
      text: ACC/ON 发动机怠速:2000
    },
    ...
  ]
}
```

## Remarks

## 2.5 车辆名称筛选

### 名称:

vehicle\_filter\_names ( )

### 描述:

获得指定车牌名称列表

## Request

```
URL: /rhino/vehicles/names
Method: GET
Headers:
Character Encoding: utf-8
Content-Type: x-www-form-urlencoded
Query Parameters:
- ticket :    access ticket 登录时获取的票据
- keyword:    车牌关键字
- limit:      返回数量
```

## Response

```
Headers:
Character Encoding: utf-8
Content-Type: application/json
Data: (array)
- name    车牌
```

## Examples:

```
Request:
/rhino/vehicles/names/?ticket=12312j%3D
Response:
{
  status:0,
  result:[
    '沪DC8795',
    '粤AV1919',
  ]
}
```



## Remarks

## 2.6 查询指定车辆状态查询

### 名称:

get\_vehicles\_status()

### 描述:

根据车牌标识查询车辆运行状态信息

## Request

```
URL: /rhino/vehicles
Method: GET
Headers:
Character Encoding: utf-8
Content-Type: multipart/form-data
Query Parameters:
- ticket    access ticket
- ids       车牌列表, 多车查询以','分隔
```

## Response

```
Headers:
Character Encoding: utf-8
Content-Type: application/json
Data:(array)
- id          车辆标识(车牌)
- lon         经度
- lat         纬度
- speed       速度
- direction   方向
- altitude    高度
- time        gps定位时间( timestamp 1970~)
- status      1:在线, 2:停止,3: 离线
- address     地址
- text        其他描述
```

## Examples:

**Request:**

可视矩形区域车辆查询

/rhino/vehicles/?ticket=12312j%3D&ids=贵K56712,沪A23412

**Response:**

```
{
  status:0,
  result:[
    {
      id: 贵K56712,
      lon: 121.22,
      lat: 31.10,
      speed: 18,
      direction: 200,
      altitude: 10,
      time: 1493702543
      status: 1
      address: 上海市长宁之路200号
      text: ACC/ON 发动机怠速:2000
    },
    ...
  ]
}
```

**Remarks**