工大项目公共组件指标接口文档

**文档版本信息**

|  |  |  |  |
| --- | --- | --- | --- |
| 修订日期 | 修订内容 | 版本号 | 编写人 |
| 2017-02-07 | 初稿 | 草拟 | 张晨 |
| 2017-02-16 | 整理格式，补充必要内容 | 初稿 | 张晨 |
| 2017-02-28 | 修正出行强度章节4.3 | 初稿 | 张晨 |
| 2017-03-03 | 修正人口分布（位置点）章节4.1 | 初稿 | 张晨 |
| 2017-03-16 | 修改空间区域编码章节4.5 和 4.2 | 初稿 | 侯捷西 |
| 2017-03-16 | 修正4.4章节，增加入参及排序功能 | 初稿 | 张晨 |
| 2017-03-27 | 修正4.6章节、增加4.7章节 | 初稿 | 张晨 |
| 2017-03-29 | 修正4.2章节、增加4..3章节 | 初稿 | 侯捷西 |
| 2017-04-01 | 4.5章节增加dataView:nest | 初稿 | 张晨 |
| 2017-04-07 | 更新目录结构和入参说明 | 初稿 | 乔良闱 |

# 目录

[1 目录 3](#_Toc479349220)

[2 接口版本 5](#_Toc479349221)

[3 接口调用 5](#_Toc479349222)

[3.1 客户端请求 5](#_Toc479349223)

[3.2 服务端响应 6](#_Toc479349224)

[4 指标接口 7](#_Toc479349225)

[4.1 人口分布（位置点）-/population/point 7](#_Toc479349226)

[4.2 人口分布（区域） 7](#_Toc479349227)

[4.2.1 请求方式 7](#_Toc479349228)

[4.2.2 请求参数 7](#_Toc479349229)

[4.2.3 请求参数样例 9](#_Toc479349230)

[4.2.4 返回格式“dataView”:”default”格式 9](#_Toc479349231)

[4.2.5 返回样例“dataView”:”default”样例 9](#_Toc479349232)

[4.2.6 返回格式 “dataView”:”range” 10](#_Toc479349233)

[4.2.7 返回样例“dataView”:”range” 10](#_Toc479349234)

[4.3 实时人口分布（区域） 11](#_Toc479349235)

[4.3.1 请求方式 11](#_Toc479349236)

[4.3.2 请求参数 11](#_Toc479349237)

[4.3.3 请求参数样例 12](#_Toc479349238)

[4.3.4 返回格式 “dataView”:”default” 12](#_Toc479349239)

[4.3.5 返回样例“dataView”:”default” 13](#_Toc479349240)

[4.4 区域出行 14](#_Toc479349241)

[4.4.1 请求方式 14](#_Toc479349242)

[4.4.2 请求参数样例 15](#_Toc479349243)

[4.4.3 返回格式 “dataView”:”default” 15](#_Toc479349244)

[4.4.4 返回样例“dataView”:”default” 16](#_Toc479349245)

[4.4.5 返回格式 “dataView”:”echarts3-migrate” 17](#_Toc479349246)

[4.4.6 返回样例“dataView”:”echarts3-migrate” 17](#_Toc479349247)

[4.5 交通走廊 18](#_Toc479349248)

[4.5.1 请求方式 18](#_Toc479349249)

[4.5.2 请求参数 18](#_Toc479349250)

[4.5.3 请求参数样例 19](#_Toc479349251)

[4.5.4 返回格式 “dataView”:”default” 19](#_Toc479349252)

[4.5.5 返回样例“dataView”:”default” 20](#_Toc479349253)

[4.5.6 返回格式 “dataView”:”range” 20](#_Toc479349254)

[4.5.7 返回样例“dataView”:”range” 21](#_Toc479349255)

[4.5.8 返回格式 “dataView”:”nest” 21](#_Toc479349256)

[4.5.9 返回样例“dataView”:”nest” 22](#_Toc479349257)

[4.6 编码-空间区域 23](#_Toc479349258)

[4.6.1 请求方式 23](#_Toc479349259)

[4.6.2 请求参数 23](#_Toc479349260)

[4.6.3 请求参数样例 23](#_Toc479349261)

[4.6.4 返回格式 “dataView”:”default” 24](#_Toc479349262)

[4.6.5 返回样例“dataView”:”default” 24](#_Toc479349263)

[4.6.6 返回格式 “dataView”:”nest” 24](#_Toc479349264)

[4.6.7 返回样例“dataView”:”nest” 25](#_Toc479349265)

[4.7 字典-类型组数据 25](#_Toc479349266)

[4.7.1 请求方式 25](#_Toc479349267)

[4.7.2 请求参数 25](#_Toc479349268)

[4.7.3 请求参数样例 26](#_Toc479349269)

[4.7.4 返回格式 “dataView”:”default” 26](#_Toc479349270)

[4.7.5 返回样例“dataView”:”default” 26](#_Toc479349271)

[4.8 字典-业务组数据 28](#_Toc479349272)

[4.8.1 请求方式 28](#_Toc479349273)

[4.8.2 请求参数 28](#_Toc479349274)

[4.8.3 请求参数样例 29](#_Toc479349275)

[4.8.4 返回格式 “dataView”:”default” 29](#_Toc479349276)

[5 附录 30](#_Toc479349277)

[5.1 HTTP1.1常用状态码及方法 30](#_Toc479349278)

# 接口版本

* 版本号：V01

# 接口调用

此接口基于RESTful风格完成客服端、服务端的交互。接口调用继承HTTP1.1协议（rfc2616）中规定的“方法”、“状态码”。

此版本的接口调用中只支持使用POST方法进行请求，并且正常调用返回的HTTP状态码只使用200。在实际开发中，接收到服务端响应后，应首先检查HTTP状态码是否为200，对状态码为正常的响应数据再进行业务解析。其他状态码可参考[附录——常用状态码及方法](#_HTTP1.1常见状态码及方法)。

## 客户端请求

* URI请求格式

|  |
| --- |
| POST http|https://{host}/{version}/{唯一标识} |

* POST Http post 方法
* http|https
* {host} 服务端URI地址
* {version} 此接口版本号
* {唯一标识} 业务接口唯一标识
* JSON参数对象

请求数据需使用“Content-Type:application/json”且使用UTF-8字符集。

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 类型 | 必填 | 说明 |
| para | Object | 是 | 具体参数需要看考指标接口 |
| para.dataView | Object | 否 | 指定数据返回的具体JSON格式，缺省值为”default”,具体格式需参考各接口指标 |
| pageObj | Object | 否 | 分页对象，需要分页。保留对象，暂不实现。 |
| pageObj.start | number | 是 | 起始记录数。保留对象，暂不实现。 |
| pageObj.size | number | 是 | 每页记录数。保留对象，暂不实现。 |
| pageObj.sort | string | 是 | 排序字段。保留对象，暂不实现。 |
| pageObj.order | string | 是 | 顺序。保留对象，暂不实现。 |

* 样例

|  |
| --- |
| **{**  **"para": {**  **“dataView”:”default” //或“dataView”:” echarts3- migrate”**  **……**  **},**  **"pageObj": {**  **"start": 10,**  **"size": 10,**  **"sort": “age”,**  **"order": “asc”**  **}**  **}** |

## 服务端响应

* JSON返回对象

服务端返回为“Content-Type:application/json”且使用UTF-8字符集的报文。解析服务端返回的JSON对象之前，需判断HTTP状态码是否为200。

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 类型 | 必填 | 说明 |
| data | Object|Array | 是 | 数据类型视具体指标接口而定 |
| pageObj | Object | 否 | 分页对象。保留对象，暂不实现。 |

* 样例

|  |
| --- |
| **{**  **"data": [**  **{**  **"id": "10",**  **"name": "人口分布",**  **"deploymentTime": "2010-10-13T14:54:26.750+02:00",**  **"category": "examples",**  **"url": "http://localhost:8081/service/repository/deployments/10",**  **"tenantId": null**  **}**  **],**  **"pageObj": {**  **"start": 10,**  **"size": 10,**  **"sort": “age”,**  **"order": “asc”**  **}**  **}** |

# 指标接口

## 人口分布（位置点）-/population/point

* 说明：统计位置点（基站），单位时间段内“流出人员数量”、“流入人员数量”、“固定人员数量”
* 唯一标识：/population/point
* 客户端请求

|  |
| --- |
| POST http://{host}/{version}**/population/point** |

## 人口分布（区域）

### 请求方式

* 说明：统计区域内，单位时间段内“人口数”
* 唯一标识：/population/region
* 客户端请求

|  |
| --- |
| POST http://{host}/{version}**/population/region** |

### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 必填 | 数据类型 | 参数说明 | 界面入参说明 |
| userType | 否 | string | 用户类型（常住、流动、其他），参考字典数接口 | 人口类型 |
| distributionId | 否 | string | 人口分布（工作地分布、居住地分布） | 分布类型 |
| range | 否 | string | 选择生成几组range值  只有在 dataView为range时加入该入参 | 传递颜色分为几个级别，例如：5就是分为5级 |
| regionType | 是 | string | 交通小区、功能区、方格网、环带、任意划定区域（任意类型中可多选小区），参考字典数接口 | 区域类型 |
| subRegionCode | 否 | Array<string> | 区域编码 | 区域对应编码，树状结构返回值。 |
| timeSize | 否 | string | 粒度：2日、3 周、4月、5 季度，参考字典数接口 | 日期类型 |
| timeObj | 否 | Object | 时间值的封装对象，在timeSzie属性不为空时才生效，包含format、vals[]。两个子对象。例如：”timeObj”:{ “format”:” "yyyy-MM-dd HH”, “vals”:[“2009-02-15 14”,“2009-02-15 22”] } | 包括（时间类型，时间范围） |
| timeObj.format | 否 | string | 数据格式，后台使用 java.text.SimpleDateFormat处理，可参考相关API进行相关数值设定 | 时间类型（timeObj的一个属性） |
| timeObj.vals | 否 | Array<string> | 时间数值字符串数组 | 时间范围（timeObj的一个属性） |
| dataView | 否 | string | 除默认值“default”外，目前支持“range”，用于前端框架生成图形 | 界面没有相关的控制按钮 需要前台代码控制 |

### 请求参数样例

{ "para": { "dataView": "default", "range": "5", "regionType": "3", "subRegionCode": [ "21-30-1", "21-30-2", "21-30-3", "21-30-4", "21-30-5", "21-30-6", "21-30-7", "21-30-9", "21-30-10", "21-30-11", "21-30-13", "21-30-14", "21-30-12", "21-30-2428", "21-30-2424", "21-30-2907", "21-30-2908" ], "timeSize": "2", "timeObj": { "format": "yyyyMMdd", "vals": [ "20170221" ] } } }

### 返回格式“dataView”:”default”格式

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| regionName | 是 | string | 区域名称，和入参的regionType相关 |
| userSum | 是 | number | 人员数量 |
| density | 是 | number | 区域密度 |
| objectId | 是 | number | 下钻关联使用 |

### 返回样例“dataView”:”default”样例

|  |
| --- |
| **{**  **"server timestamp": 1491452145473,**  **"data": [**  **{**  **"density": 50,**  **"objectId": 6,**  **"userSum": 100,**  **"regionName": "昌平区"**  **},**  **{**  **"density": 50,**  **"objectId": 3,**  **"userSum": 100,**  **"regionName": "顺义区"**  **},**  **{**  **"density": 48.5,**  **"objectId": 4,**  **"userSum": 97,**  **"regionName": "通州区"**  **},**  **……**      **]**  **}** |

### 返回格式 “dataView”:”range”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| data.dataArray.[i].  regionName | 是 | string | 区域名称，和入参regionType相关 |
| data. dataArray. [i] userSum | 是 | number | 总人次数 |
| data. dataArray. [i] rangeIdx | 是 | number | 人数所属范围 |
| data. dataArray. [i] density | 是 | number | 密度 |
| data. dataArray. [i] objectId | 是 | number | 下钻关联使用 |
| data. range | 是 | number | 人数范围数组 |

### 返回样例“dataView”:”range”

|  |
| --- |
| **{**  **"server timestamp": 1491452234043,**  **"data": {**  **"dataArray": [**  **{**  **"density": 50,**  **"objectId": 6,**  **"userSum": 100,**  **"regionName": "昌平区",**  **"rangeIdx": 4**  **},**  **{**  **"density": 50,**  **"objectId": 3,**  **"userSum": 100,**  **"regionName": "顺义区",**  **"rangeIdx": 4**  **},**  **{**  **"density": 48.5,**  **"objectId": 4,**  **"userSum": 97,**  **"regionName": "通州区",**  **"rangeIdx": 2**  **},**  **…….**  **],**  **"range": [**  **"91-93",**  **"94-95",**  **"96-97",**  **"98-99",**  **"100-101"**  **]**  **}**  **}** |

### 返回格式 “dataView”:” TimeArray”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| data. timeData .[i].  regionName | 是 | string | 区域名称，和入参regionType相关 |
| data. timeData . [i] userSum | 是 | number | 总人次数 |
| data. timeData . [i] density | 是 | number | 区域人口密度 |
| data. timeData . [i] objectId | 是 | number | 下钻关联使用 |
| data. time | 是 | number | 时间粒度下的所有时间 |

### 返回样例“dataView”:” TimeArray”

|  |
| --- |
| **{**  **"server timestamp": 1491897396785,**  **"data": {**  **"times": [**  **1,**  **2,**  **….**  **],**  **"timeData": [**  **[],**  **[**  **{**  **"density": 50,**  **"userSum": 100,**  **"objectId": 7,**  **"regionName": "门头沟区"**  **},**  **{**  **"density": 48.5,**  **"userSum": 97,**  **"objectId": 5,**  **"regionName": "朝阳区"**  **},**  **…..**    **{**  **"density": 45.5,**  **"userSum": 91,**  **"objectId": 9,**  **"regionName": "石景山区"**  **}**  **],**  **[],**  **[**  **{**  **"density": 50,**  **"userSum": 100,**  **"objectId": 7,**  **"regionName": "门头沟区"**  **},**  **{**  **"density": 48.5,**  **"userSum": 97,**  **"objectId": 5,**  **"regionName": "朝阳区"**  **},**  **……**  **],**  **[],**  **[],**  **[],**  **[],**  **[],**  **[],**  **[],**  **[]**  **]**  **}**  **}** |

## 实时人口分布（区域）

### 请求方式

* 说明：统计区域内，单位时间段内“流出人员数量”、“流入人员数量”、“固定人员数量”
* 唯一标识：/ population /realtime
* 客户端请求

|  |
| --- |
| POST http://{host}/{version}**/population/realtime** |

### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 必填 | 数据类型 | 说明 | 界面入参说明 |
| regionType | 是 | string | 区域类型 1：省市县 | 区域类型 |
| subRegionCode | 否 | Array<string> | 区域编码 | 区域对应编码，树状结构返回值。 |
| timeSize | 否 | string | 粒度：1 历史时间 ，2 实时时间，参考字典数接口 | 包括（时间类型，时间范围） |
| timeObj | 否 | Object | 时间值的封装对象，在timeSzie属性不为空时才生效，包含format、vals[]。两个子对象。例如：”timeObj”:{ “format”:” "yyyy-MM-dd HH”, “vals”:[“2009-02-15 14”,“2009-02-15 22”] } | 时间类型（timeObj的一个属性） |
| timeObj.format | 否 | string | 数据格式，后台使用 java.text.SimpleDateFormat处理，可参考相关API进行相关数值设定 | 时间范围（timeObj的一个属性） |
| timeObj.vals | 否 | Array<string> | 时间数值字符串数组 | 界面没有相关的控制按钮 需要前台代码控制 |

### 请求参数样例

{ "para": { "dataView": "default", "range": "5", "regionType": "3", "subRegionCode": [ "21-30-7", "21-30-1", "21-30-2", "21-30-3", "21-30-4", "21-30-4" ], "timeSize": "1", "timeObj": { "format": "yyyyMMddHHmm", "vals": [ "201702221330" ] } } }

### 返回格式 “dataView”:”default”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| regionName | 是 | string | 区域名称，和入参的regionType相关 |
| loc | 是 | Array | 位置的经纬度，loc[0]经度，loc[1]纬度 |
| userSum | 是 | number | 人员数量 |
| density | 是 | number | 区域密度 |
| inflowUserSum | 是 | number | 流入人员数量（当选择实时或历史人口时 有该输出项） |
| outflowUserSum | 是 | number | 流出人员数量（当选择实时或历史人口时 有该输出项） |
| fixUserSum | 是 | number | 固定人员数量（当选择实时或历史人口时 有该输出项） |

### 返回样例“dataView”:”default”

|  |
| --- |
| **{**  **"server timestamp": 1491443110225,**  **"data": [**  **{**  **"outflowUserSum": 9,**  **"density": 1,**  **"userSum": 1,**  **"regionName": "延庆县",**  **"loc": [**  **"1",**  **"1"**  **],**  **"fixUserSum": 345,**  **"inflowUserSum": 1**  **},**  **{**  **"outflowUserSum": 785,**  **"density": 1,**  **"userSum": 1,**  **"regionName": "门头沟区",**  **"loc": [**  **"1",**  **"1"**  **],**  **"fixUserSum": 654,**  **"inflowUserSum": 1**  **},**  **….**  **]**  **}** |

## 区域出行

### 请求方式

* 说明：统计OD区域，单位时间段内的“出行强度数值”、“人员出行次数”、“平均出行距离”、“平均出行时长”、“出行总距离”、“出行总时长”。
* 唯一标识：/travel/intensity
* 客户端请求

|  |
| --- |
| POST http://{host}/{version}**/travel/intensity** |

* 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必填 | 数据类型 | 说明 |
| userType | 否 | string | 用户类型（常住、流动、其他），参考字典数接口 |
| regionType | 是 | string | 交通小区、功能区、方格网、环带、任意划定区域（任意类型中可多选小区），参考字典数接口 |
| subRegionCode | 否 | Array | 数组中存放一个或多个数值，regionType的二级数据项，参考regionType，如regionType为行政区划，则此处需填写区域代码。 |
| timeSize | 否 | string | 粒度：小时、日、周、月、季度，参考字典数接口 |
| timeObj | 否 | Object | 时间值得封装对象，在timeSzie属性不为空时才生效，包含format、vals[]。两个子对象。例如：  ”timeObj”:{  “format”:” "yyyy-MM-dd HH”,  “vals”:[“2009-02-15 14”,“2009-02-15 22”]  } |
| timeObj.format | 是 | string | 数据格式，后台使用 java.text.SimpleDateFormat处理，可参考相关API进行相关数值设定 |
| timeObj.vals | 是 | Array<string> | 时间数值字符串数组 |
| odGoal | 否 | string | 出行目的（通勤、旅游、办公、出差），参考字典数接口 |
| odType | 否 | string | 出行方式(汽车、地铁、自行车、走路、飞机、普快、高铁)，参考字典数接口 |
| showFields | 否 | Array<string> | 返回的数据字段，设定的字段会做为输出字段出现在返回结果中，可选值：出行目的odGoal、出行方式odType、出行时间odTime |
| topN | 否 | number | 返回记录的条数，通常和orderBy入参结合使用，实现TopN功能 |
| orderBy | 否 | Array<string> | 排序指标，可传入出现在返回值中的参数，连接 asc、desc。例如： "orderBy":["distAvg asc","oName"]。返回结果中会出现排名字段orderRank. |
| dataView | 否 | string | 除默认值“default”外，目前支持“echarts3- migrate”，用于前端框架（如echarts）生成迁徙图形 |

### 请求参数样例

{"para":{"dataView":"default","regionType":"3","userType":"0","showFields":["odGoal","odType","odTime"],"subRegionCode":["21-30-1","21-30-3"],"orderBy":["timeAvg desc","oName"],"timeSize":"2"}}

### 返回格式 “dataView”:”default”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| val | 是 | number | 出行强度数值 |
| timeSum | 是 | number | 出行总时长 |
| distSum | 是 | number | 出行总距离 |
| timeAvg | 是 | number | 平均出行时长 |
| distSum | 是 | number | 平均出行距离 |
| odCnt | 是 | number | 出行总人次 |
| oName | 是 | string | O点名称 |
| oObjectId | 是 | number |  |
| dName | 是 | string | D点名称 |
| dObjectId | 是 | number |  |
| odGoal | 否 | string | 出行目的，参考入参 showFields |
| odType | 否 | string | 出行方式，参考入参 showFields |
| odTime | 否 | string | 出行时间，参考入参 showFields |
| orderRank | 否 | number | 排序字段，参考入参 orderBy |

### 返回样例“dataView”:”default”

|  |
| --- |
| **{**  **"data": [**  **{**  **"val": 345.54, //出行强度数值**  **"timeSum": 240, //出行总时长**  **"distSum": 20, //出行总距离**  **"odGoal": "娱乐", //出行目的**  **"dName": "丰台区", //D点名称**  **"dObjectId":3,**  **"distAvg": 0.0005788041905423395, //平均出行距离**  **"oName": "朝阳区", //O点名称**  **"oObjectId": 1,**  **"timeAvg": 0.0069456502865080745, //平均出行时长**  **"odType": "小汽车", //出行方式**  **"orderRank": 1, //排序字段**  **"odCnt": 34554 //出行总人次**  **},**  **{**  **"val": 45.65,**  **"timeSum": 240,**  **"distSum": 20,**  **"odGoal": "上学",**  **"dName": "西城区",**  **"dObjectId":1,**  **"distAvg": 0.004381161007667032,**  **"oName": "朝阳区",**  **"oObjectId":3,**  **"timeAvg": 0.05257393209200438,**  **"odType": "公交",**  **"orderRank": 2,**  **"odCnt": 4565**  **}**  **],**  **"server timestamp": 1489651715622**  **……**  **}** |

### 返回格式 “dataView”:”echarts3-migrate”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| data[i][j][0].name | 是 | string | O点名称，和入参regionType相关 |
| data[i][j][1].name | 是 | string | D点名称，和入参的regionType相关 |
| data[i][j][1].value | 是 | number | 出行强度数值 |
| data[i][j][1].odCnt | 是 | number | 总出行人次数 |

### 返回样例“dataView”:”echarts3-migrate”

|  |
| --- |
| **{**  **"data":[**  **[**  **[{"name":"北京"},{"name":"上海","value":95,odCnt:950}],**  **[{"name":"北京"},{"name":"广州","value":90,odCnt:900}],**  **],**  **[**  **[{"name":"上海"},{"name":"广州","value":80,odCnt:800}],**  **[{"name":"上海"},{"name":"大连","value":10,odCnt:100}]**  **],**  **[**  **[{"name":"广州"},{"name":"福州","value":95,odCnt:950}],**  **[{"name":"广州"},{"name":"太原","value":90,odCnt:90}],**  **[{"name":"广州"},{"name":"北京","value":30,odCnt:30}]**  **]**  **]**  **}** |

## 交通走廊

### 请求方式

* 说明：统计区域内，单位时间段内交通情况
* 唯一标识：/metric/traffic/corridor
* 客户端请求

|  |
| --- |
| POST http://{host}/{version} **/metric/traffic/corridor** |

### 请求参数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 必填 | 数据类型 | 说明 | 界面入参说明 |
| userType | 否 | string | 用户类型（常住、流动、其他），参考字典数接口 | 人口类型 |
| corridor | 否 | Array<string> | 交通走廊标识参考字典数接口 | 暂时无用 |
| range | 否 | string | 选择生成几组range值 只有在 dataView为range时加入该入参 | 传递颜色分为几个级别，例如：5就是分为5级 |
| road | 是 | Array<string> | 交通路段标识，参考字典数接口 | 交通走廊筛选框对应的路段编码，树状结构返回值。 |
| odPurpose | 否 | Array<string> | 出行目的标识 | 出行目的 |
| odWay | 否 | Array<string> | 出行方式标识 | 出行方式 |
| timeSize | 否 | string | 粒度：小时、日、周、月、季度，参考字典数接口 | 时间类型 |
| timeObj | 否 | Object | 时间值的封装对象，在timeSzie属性不为空时才生效，包含format、vals[]。两个子对象。例如：”timeObj”:{ “format”:” "yyyy-MM-dd HH”, “vals”:[“2009-02-15 14”,“2009-02-15 22”] } | 包括（时间类型，时间范围） |
| timeObj.format | 否 | string | 数据格式，后台使用 java.text.SimpleDateFormat处理，可参考相关API进行相关数值设定 | 时间类型（timeObj的一个属性） |
| timeObj.vals | 否 | Array<string> | 时间数值字符串数组 | 时间范围（timeObj的一个属性） |
| dataView | 否 | string |  | 界面没有相关的控制按钮 需要前台代码控制 |

### 请求参数样例

{ "para": { "dataView": "nest", "range": "5", "userType": "1", "corridor": [ "1", "2" ], "road": [ "1", "2", "3", "4" ], "odPurpose": [ "1", "2" ], "odWay": [ "1", "2" ], "timeSize": "2", "timeObj": { "format": "yyyyMMdd", "vals": [ "20170330" ] } } }

### 返回格式 “dataView”:”default”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| importance | 是 | string | 重要度 |
| userSum | 是 | number | 人员数量 |
| corridorName | 是 | number | 交通走廊名称，和入参的 corridor 相关 |
| roadName | 是 | string | 交通路段名称，和入参的 road 相关 |
| tideAttribute | 是 | number | 潮汐性1表示是、2表示否 |

### 返回样例“dataView”:”default”

|  |
| --- |
| **{**  **"server timestamp": 1491545827228,**  **"data": [**  **{**  **"importance": 1,**  **"userSum": 12,**  **"corridorName": "长安街",**  **"roadName": "长安街南",**  **"tideAttribute": 1**  **},**  **{**  **"importance": 1,**  **"userSum": 32,**  **"corridorName": "西二环",**  **"roadName": "西二环南",**  **"tideAttribute": 1**  **},**    **……..**    **]**  **}** |

### 返回格式 “dataView”:”range”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| data.dataArray.[i].  importance | 是 | string | 重要度 |
| data. dataArray. [i] userSum | 是 | number | 人员数量 |
| data. dataArray. [i] corridorName | 是 | number | 交通走廊名称，和入参的 corridor 相关 |
| data.dataArray.[i].  rangeIdx | 是 | string | 人数所属rang数组的哪个范围 |
| data. dataArray. [i] roadName | 是 | number | 交通路段名称，和入参的 road 相关 |
| data. dataArray. [i] corridorName | 是 | number | 潮汐性1表示是、2表示否 |
| data. range | 是 | number | 人数范围数组 |

### 返回样例“dataView”:”range”

|  |
| --- |
| **{**  **"server timestamp": 1491015907934,**  **"data": {**  **"dataArray": [**  **{**  **"importance": 1,**  **"userSum": 100,**  **"corridorName": "长安街东",**  **"rangeIdx": 4,**  **"roadName": "长安街",**  **"tideAttribute": 1**  **},**  **{**  **"importance": 1,**  **"userSum": 98,**  **"corridorName": "西二环东",**  **"rangeIdx": 4,**  **"roadName": "西二环",**  **"tideAttribute": 1**  **},**  **…**  **"range": [**  **"12-30",**  **"31-48",**  **"49-66",**  **"67-84",**  **"85-102"**  **]**  **}**  **}** |

### 返回格式 “dataView”:”nest”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| data.key[i].  importance | 是 | string | 重要度 |
| data. key . [i] userSum | 是 | number | 人员数量 |
| data. key . [i] corridorName | 是 | number | 交通走廊名称，和入参的 corridor 相关 |
| data. key . [i] roadName | 是 | number | 交通路段名称，和入参的 road 相关 |
| data. dataArray. [i] tideAttribute | 是 | number | 潮汐性1表示是、2表示否 |

### 返回样例“dataView”:”nest”

|  |
| --- |
| **{**  **"server timestamp": 1491466133825,**  **"data": {**  **"西二环": [**  **{**  **"importance": 1,**  **"userSum": 98,**  **"tideAttribute": 1,**  **"roadName": "西二环东"**  **},**    **{**  **"importance": 1,**  **"userSum": 32,**  **"tideAttribute": 1,**  **"roadName": "西二环南"**  **}**  **],**  **"长安街": [**  **{**  **"importance": 1,**  **"userSum": 100,**  **"tideAttribute": 1,**  **"roadName": "长安街东"**  **},**    **{**  **"importance": 1,**  **"userSum": 45,**  **"tideAttribute": 1,**  **"roadName": "长安街北"**  **}**  **]**  **}**  **}** |

## 编码-空间区域

### 请求方式

* 唯一标识：/code/region/level
* 客户端请求

|  |
| --- |
| **POST** http://{host}/{version}**/code/region/level** |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | | 必填 | | 数据类型 | 说明 |
| regionType | | 是 | | string | 空间区域类型，如行政区划、交通小区、旅游区等。 |
| regionName | | 否 | | String | 区域名称 （用于筛选框联动模块 文字搜索） |
| subRegionCode | | 否 | | Array | 区域编码 （如果没传默认显示所有目录） |
| subLevel | | 否 | | string | 期望下钻的层数（比如 subLevel值为2， regionCode 值为011011， 后台代码判断为2级 输出参数 的区域等级为 2， 3，4级。 subLevel值为0时 表示 查询所有下级数据）只有regionCode值不为空 subLevel值才可以不为空 |
| dataView | 否 | | string | | 除默认值“default”外，目前支持“nest”，用于前端框架生成树状结构 |

### 请求参数样例

{"para": { "dataView": "nest", "regionType":"7"}}

### 返回格式 “dataView”:”default”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| regionCode | 是 | String | 区域编码 用于业务查询 |
| regionName | 是 | String | regionCode 对应的区域的名称 |
| regionArea | 是 | String | regionCode 对应的区域的面积 |

### 返回样例“dataView”:”default”

|  |
| --- |
| **{**  **"data": [**  **{**  **"regionArea": "12",**  **"regionCode": "011001014",**  **"regionName": "昌平区"**  **},**  **{**  **"regionArea": "12",**  **"regionCode": "011001017",**  **"regionName": "平谷区"**  **}**  **]** |

### 返回格式 “dataView”:”nest”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| id | 是 | String | 区域编码 用于业务查询 |
| name | 是 | String | regionCode 对应的区域的名称 |
| children | 是 | Array | 下级数据 |

### 返回样例“dataView”:”nest”

|  |
| --- |
| **"data": [**  **{**  **"name": "北京市",**  **"id": "00210030"**  **"children": [**  **{**  **"name": "平谷区",**  **"id": "002100302428"**  **},**  **{**  **"name": "西城区",**  **"id": "002100302907"**  **},**  **{**  **"name": "东城区",**  **"id": "002100302908"**  **}**  **]**  **}**  **],**  **"server timestamp": 1491043385526**  **}** |

## 字典-类型组数据

### 请求方式

* 说明：根据传入的数据字典类型值，返回相关的数据字典键值对
* 唯一标识：/code/dict/type
* 客户端请求

|  |
| --- |
| POST http://{host}/{version}**/code/dict/type** |

### 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必填 | 数据类型 | 说明 |
| dicTypes | 否 | Array<String> | 根据数据字典类型获取相关字段数据，入参为空时，返回数据字典的元数据，即数据类型为”metadata”的数据 |

### 请求参数样例

{"para":{"dicTypes":["metadata","cxmd","cxfs","rklx","sjlx","dldj","sjlx-h24","sjlx-xq","qylx","fblx","zswd","cxms","sjlx-h24","sjlx-xq","jtzl","lyjd","cjlj","cxms","xzqy","bdbs","sfzzcbs","tqbs","xbbs"]}}

### 返回格式 “dataView”:”default”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| data.x | 是 | string | 字典数据类型 |
| data.x .[i].name | 是 | string | 字典数据代表的名称 |
| data.x .[i].value | 是 | string | 字典数据代表的数值 |

### 返回样例“dataView”:”default”

|  |
| --- |
| **{**  **"data": {**  **"rklx": [ //人口类型**  **{**  **"name": "全部",**  **"value": "0"**  **},**  **{**  **"name": "常住人口",**  **"value": "1"**  **},**  **{**  **"name": "流动人口",**  **"value": "2"**  **},**  **{**  **"name": "实时人口",**  **"value": "3"**  **}**  **],**  **"metadata": [ //元数据**  **{**  **"name": "出行方式",**  **"value": "cxfs"**  **},**  **{**  **"name": "出行目的",**  **"value": "cxmd"**  **},**  **{**  **"name": "出行模式",**  **"value": "cxms"**  **},**  **{**  **"name": "人口分布类型",**  **"value": "fblx"**  **},**  **{**  **"name": "空间区域",**  **"value": "kjqy"**  **},**  **{**  **"name": "路况图例",**  **"value": "lktl"**  **},**  **{**  **"name": "人口类型",**  **"value": "rklx"**  **},**  **{**  **"name": "时间分类（实时、历史）",**  **"value": "sjfl"**  **},**  **{**  **"name": "时间类型",**  **"value": "sjlx"**  **},**  **{**  **"name": "时间类型-24小时",**  **"value": "sjlx-h24"**  **},**  **{**  **"name": "时间类型-星期",**  **"value": "sjlx-xq"**  **}**  **],**  **"cxms": [**  **{**  **"name": "短距离通勤",**  **"value": "1"**  **},**  **{**  **"name": "长距离通勤",**  **"value": "2"**  **},**  **{**  **"name": "嵌套出行",**  **"value": "3"**  **}**  **]**  **},**  **"server timestamp": 1490585824913**  **}** |

## 字典-业务组数据

### 请求方式

* 说明：根据传入的预定义字典业务类型，返回相关的数据字典键值对
* 唯一标识：/code/dict/busi
* 客户端请求

|  |
| --- |
| POST http://{host}/{version}**/code/dict/busi** |

### 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必填 | 数据类型 | 说明 |
| busiTypes | 是 | Array<String> | 根据预定义的业务类型标识获取相关字段数据。 |

### 请求参数样例

{"para":{"busiTypes":["default-init","ssrk-szwd","fssrk-szwd","czrk-fblx","ldrk-fblx","xs-shlx","default-sjlx","jtzl-rklx","fssrk-rklx","ssrk-rklx"]}}

### 返回格式 “dataView”:”default”

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 必填 | 数据类型 | 说明 |
| data.x | 是 | string | 预定义的业务场景类型名称 |
| data.x .[i].name | 是 | string | 字典数据代表的名称 |
| data.x .[i].value | 是 | string | 字典数据代表的数值 |
| data.x .[i].type | 是 | string | 字典数据的字典类型，参考4.5章节 |

* 返回样例

|  |
| --- |
| **{**  **"data": {**  **"czrk-fblx": [ //业务场景数据——依赖于常驻人口的分布类型**  **{**  **"name": "居住地分布",**  **"type": "fblx",**  **"value": "1"**  **},**  **{**  **"name": "工作地分布",**  **"type": "fblx",**  **"value": "2"**  **}**  **]**  **},**  **"default-init": [ //业务自定义的初始化字典数据**  **{**  **"name": "全部",**  **"type": "cxfs",**  **"value": "0"**  **},**  **{**  **"name": "地铁",**  **"type": "cxfs",**  **"value": "1"**  **},**  **{**  **"name": "公交",**  **"type": "cxfs",**  **"value": "2"**  **},**  **{**  **"name": "小汽车",**  **"type": "cxfs",**  **"value": "3"**  **},**  **{**  **"name": "步行/自行车",**  **"type": "cxfs",**  **"value": "4"**  **},**  **{**  **"name": "全部",**  **"type": "cxmd",**  **"value": "0"**  **}**  **],**  **"server timestamp": 1490597582230**  **}** |

# 附录

## HTTP1.1常用状态码及方法

* 常用状态码

|  |  |
| --- | --- |
| 状态码 | 说明 |
| 200 | 请求已成功，请求所希望的响应头或数据体将随此响应返回。 |
| 201 | 请求已经被实现，而且有一个新的资源已经依据请求的需要而建立，且其 URI 已经随Location 头信息返回。假如需要的资源无法及时建立的话，应当返回 '202 Accepted'。 |
| 204 | 在并没有新文档的情况下，204 (SC\_NO\_CONTENT)确保浏览器继续显示先前的文档。 |
| 400 | (Bad Request/错误请求)；(SC\_BAD\_REQUEST)指出客户端请求中的语法错误。 |
| 401 | 未授权 |
| 403 | 禁止，服务器拒绝提供所请求的资源。 |
| 404 | 未找到，请求资源不存在。 |
| 405 | 方法未允许，(SC\_METHOD\_NOT\_ALLOWED)指出请求方法(GET, POST, HEAD, PUT, DELETE, 等)对某些特定的资源不允许使用。 |
| 409 | 冲突，状态通常与PUT请求一同使用，409 (SC\_CONFLICT)状态常被用于试图上传版本不正确的文件时。 |
| 415 | 不支持的媒体格式，(SC\_UNSUPPORTED\_MEDIA\_TYPE)意味着请求所带的附件的格式类型服务器不知道如何处理。 |
| 500 | 服务器内部错误。 |

* 常用方法

|  |  |
| --- | --- |
| 方法 | 说明 |
| GET | 向特定的资源发出请求。注意：GET方法不应当被用于产生“副作用”的操作中，例如在web app.中。其中一个原因是GET可能会被网络蜘蛛等随意访问 |
| POST | 向指定资源提交数据进行处理请求（例如提交表单或者上传文件）。数据被包含在请求体中。POST请求可能会导致新的资源的建立和/或已有资源的修改 |
| PUT | 向指定资源位置上传其最新内容 |
| OPTIONS | 返回服务器针对特定资源所支持的HTTP请求方法。也可以利用向Web服务器发送'\*'的请求来测试服务器的功能性 |
| DELETE | 请求服务器删除Request-URI所标识的资源 |