IIOT 产品对接文档

文档修改记录

版本号	版本描述	责任人	日期	审核人	备注
V1.5.3	第一版	梁佳辉	2025-08-19	林宁	1. 增加 5.2 主动查询 1 和 2 说明 2. 删除 5.5 使用物标识指令执行（带参数） 3. 修改 5.9、5.16、5.17、5.18 示例参数 4. 增加 5.10.报警历史记录时间筛选参数 5. 补充 5.13 历史工况聚合查询 6. 增加工厂实体历史工况和实时工况查询
V1.5.4		黄鹏波	2025-08-27	林宁	1. 补充查询实体属性接口
V1.5.5		李侨	2025-10-15	林宁	1、补充数据编织查询数据接口
V1.5.6		黄鹏波	2025-10-31	林宁	1. 补充创建物实体接口
V1.5.7		梁佳辉	2025-11-03	林宁	1. 补充查询物实体 MQTT 认证标识和密钥接口 2. 完善回补历史工况接口参数说明
V1.5.8		梁佳辉	2025-11-06	林宁	1. 补充 MQTT 连接地址获取接口
V1.5.9		梁佳辉	2025-11-07	林宁	2. 补充创建数据建模接口、新增建模字段接口、新增建模数据接口
V1.5.10		李侨	2025-11-10	林宁	补充数据建模设置唯一标识和启用的接口

目录

1. 文档说明

此文档用于指导开发人员进行 iiot 数据对接使用。

2. 连接方式

2.1. HTTP 连接

- 获取 token：http://ip:port/api/root/rpc/login
- 通用接口：http://ip:port/api/root/rpc/service/

2.2. MQTT 连接地址获取

2.2.1.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

2.2.1.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/entityId	String	查询的设备实体	"sagjf3435sfa"	是
params/args/channel	String	查询的频道	固定为"mqtt"	是
params/model	String	模型	"iiot_thing_entity"	是
params/service	String	服务	"getIpAddressById"	是

2.2.1.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	"192.168.174.169:31882"	

2.2.1.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
        "entityId":"046bzfu5vnq5e",
        "channel":"mqtt"
    },
    "context": {
      "uid": "",
      "timeZone": "UTC+8",
      "lang": "zh-CN"
    },
    "model": "iiot_thing_entity",
    "tag": "master",
    "service": "getIpAddressById",
    "app": "iiot_thing"
  }
}
```

2.2.1.5. 返回示例

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "result": {
        "data": "192.168.174.169:31882",
        "context": {
            "@Class": "java.lang.String",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    }
}
```

2.3. MQTT 连接

- 服务器地址:mqtt://ip:port,根据 2.2 章节获取 ip 和端口

2.4. 转发配置

- 通过数据转发模块的转发协议和转发配置，配置接收数据的服务信息。
- 支持 Mqtt、Rabbitmq、WebApi、OPCUA。

3. API 接口说明

- 请求语法：POST 接口地址 HTTP/2.0 协议
- 一般情况下，接口用同一个地址请求：http://ip:port/api/root/rpc/service/
- 一般情况下，需要添加请求头：Authorization:Bearer token 字符串
- 请求/响应的数据格式：JSON 格式
- 一般情况下，请求参数包括

名称	类型	描述	示例	必填
id	String	请求 ID	"guid"（固定值）	是
jsonrpc	String	协议版本	"2.0"（固定值）	是
method	String	接口行为方法	"service"（固定值）	是
params	JSON	API 参数		是
params/app	String	APP 名称	"iiot_thing"	是
params/args	JSON	服务参数		是
params/args/filter	JSON	过滤参数	[["key","like","%server%"]]	否
params/args/xxx		xxx 参数	30	否
params/context	JSON	上下文	{"uid": "","lang": "zh_CN"}（固定值）	是
params/model	String	模型	"iiot_thing_entity_history_tagvalue"	是
params/service	String	服务	"search"	是
params/tag	String	版本	"master"（固定值）	是

- 一般情况下，响应参数包括

名称	类型	描述	示例	备注
id	String	请求 ID	"guid"（固定值）	
jsonrpc	String	协议版本	"2.0"（固定值）	
result	JSON	响应结果		
result/context	JSON	上下文	{"uid": "","lang": "zh_CN"}	
result/data	JSON	结果数据		
error	JSON	错误		
error/code	String	错误码		
error/data	JSON	错误数据		
error/message	String	错误信息		

4. 获取 token

4.1. 请求语法

POST /api/root/rpc/login HTTP/2.0

4.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"base"	是
params/args	JSON	参数	{"useDisplayForModel": true}	是
params/login	String	账号	"testUserName"	是
params/password	String	密码，Base64 加密	"QWRtaW4xMjM="	是
params/model	String	模型	""	是

4.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	{
			"name": "管理员",
			"pwd_expire_day": 90,
			"id": "sgakfa786",
			"login": "testUserName",
			"roles": [
			{
			"id": "sklkfshf6",
			"name": "开发主管",
			"code": null,
			"remark": "3235345"
			}
			],
			"token":
			"cae26ebfc0a54a39b2e0da35633d19b6cd11cb19efe6457ce5c8305f32fe206a59763a247662f8ce53932676274d68774f8917fe4727ce9ce6683e2858afb96cb7e61a5d594f8456ea8c3b0ce79e66db9fa09d230277c41541323ceca7ec63e1bb81a414f6d1287b54173fba5fd7029f2993807d54341a01453a469d0774ffe2d04ee4e8ddb606c5eaf665533c3f5cf8eb96777e10cffee06449cf777521c9d9eb064583b761b806ad07c685a5e642f5d66cdeb596c57ca3"
			}	

4.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "useDisplayForModel": true
    },
    "context": {
      "lang": "zh_CN",
      "uid": ""
    },
    "model": "",
    "tag": "master",
    "app": "base",
    "login": "admin",
    "password": "QWRtaW4xMjM=",
    "remember": false
  }
}
```

4.5. 返回示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "result": {
    "data": {
      "name": "管理员",
      "pwd expire day": 90,
      "id": "rbac user admin",
      "login": "admin",
      "roles": [
        {
          "id": "rbac role admin",
          "name": "开发主管",
          "code": null,
          "remark": "3235345"
```

```
      }
    ],
    "token":
"cae26ebfc0a54a39b2e0da35633d19b6cd11cb19efe6457ce5c8305f32fe206a59763a247662f8ce53932
676274d68774f8917fe4727ce9ce6683e2858afb96cb7e61a5d594f8456ea8c3b0ce79e66db9fa09d23027
7c41541323ceca7ec63e1bb81a414f6d1287b54173fba5fd7029f2993807d54341a01453a469d0774ffe2d
04ee4e8ddb606c5eaf665533c3f5cf8eb96777e10cffee06449cf777521c9d9eb064583b761b806ad07c68
5a5e642f5d66cdeb596c57ca3"
  }
 }
}
```


5. 平台端对接

5.1. （物资源总览）查询所有项目

注意：
使用平台通用查询语法，通过 filter 参数筛选数据，具体语法可参考官网了解

5.1.1.请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.1.2.请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_project"	是
params/model	String	模型	"iiot_project"	是
params/service	String	服务	"searchIncludeAll"	是
params/args/properties	String	查询值	按照示例固定值，无需修改	是

5.1.3.返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	[
			{
			"name": "全部",
			"project_code": null,
			"id": "0",
			"desc": "全部"
			},
			{
			"name": "系统项目",
			"project_code": "system_project",
			"desc": "系统默认项目",
			"id": "system_proj"
			}
			]
				name:项目名；
				project_code:项目编码；
				id:项目 Id；
				desc:描述；

5.1.4.请求示例

只需关注请求参数中提到的参数部分，其他参数固定无需修改

示例：

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "properties": [
        "name",
        "project_code",
        "desc"
      ],
    },
    "context": {
      "uid": "",
      "lang": "zh_CN"
    },
    "model": "iiot_thing_property",
    "tag": "master",
    "service": "searchNodeData",
    "app": "iiot_thing"
  }
}
```

5.1.5.返回示例

```
{
    "result": {
        "data": [
            {
                "name": "全部",
                "project_code": null,
                "id": "0",
                "desc": "全部"
             },
            {
                "name": "系统项目",
                "project_code": "system_project",
                "desc": "系统默认项目",
                "id": "system_proj"
             }
        ],
        "context": {
            "@Class": "java.util.HashMap",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.2. （物资源总览）查询数据

注意：
使用平台通用查询语法，通过 filter 参数筛选数据，具体语法可参考官网了解

5.2.1.请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.2.2.请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/filter/project_id	String	项目 id，参考 5.1	"03efoznia1hc0"，不填查询全部项目	否
params/args/properties	String	查询值	按照示例固定值，无需修改	是
params/model	String	模型	"iiot_thing_overview_model"	是
params/service	String	服务	"search"	是

5.2.3.返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	[{ "tags_total_count", "thing_entity_total_count", "manu_entity_total_count", "direct_device_total_count", "online_direct_device_total_count", "offline_direct_device_total_count", "inactive_direct_device_total_count", "stop_direct_device_total_count", "smdc_device_total_count", "online_smdc_device_total_count", "offline_smdc_device_total_count", "inactive_smdc_device_total_count", "stop_smdc_device_total_count", "gateway_device_total_count", "online_gateway_device_total_count", "offline_gateway_device_total_count", "inactive_gateway_device_total_count", "stop_gateway_device_total_count", "thing_model_total_count", "device_thing_model_total_count", "gateway_thing_model_total_count", "smdc_thing_model_total_count", "manu_thing_model_total_count", "smdc_child_device_total_count", "online_smdc_child_device_total_count", "offline_smdc_child_device_total_count", "inactive_smdc_child_device_total_count", "stop_smdc_child_device_total_count", "gateway_child_device_total_count", "online_gateway_child_device_total_count", "offline_gateway_child_device_total_count", "inactive_gateway_child_device_total_count", "stop_gateway_child_device_total_count", "alarms_week_count", "alarms_crux_week_count",	thing_entity_total_count: 物实体总数 direct_device_total_count: 直连设备总数 online_direct_device_total_count: 直连设备总数-在线 offline_direct_device_total_count: 直连设备总数-离线 inactive_direct_device_total_count: 直连设备总数-未激活 stop_direct_device_total_count: 直连设备总数-停用 smdc_device_total_count: SMDC 设备总数 online_smdc_device_total_count: SMDC 设备总数-在线 offline_smdc_device_total_count: SMDC 设备总数-离线

			"alarms_important_week_count", "alarms_commonly_week_count", "alarms_secondary_week_count", "alarms_tips_week_count" }]	inactive_smdc_device_total_count: SMDC 设备总数-未激活 stop_smdc_device_total_count: SMDC 设备总数-停用 gateway_device_total_count: 网关设备总数 online_gateway_device_total_count: 网关设备总数-在线 offline_gateway_device_total_count: 网关设备总数-离线 inactive_gateway_device_total_count: 网关设备总数-未激活 stop_gateway_device_total_count: 网关设备总数-停用 thing_model_total_count: 物模型总数 device_thing_model_total_count: 设备模型总数 gateway_thing_model_total_count: 网关模型总数 smdc_thing_model_total_count: SMDC 模型总数 manu_thing_model_total_count: 工厂模型总数 smdc_child_device_total_count: SMDC 子设备总数 online_smdc_child_device_total_count: SMDC 子设备在线总数 offline_smdc_child_device_total_count: SMDC 子设备离线总数 inactive_smdc_child_device_total_count: SMDC 子设备未激活总数

				stop_smdc_child_device_total_count: SMDC 子设备停用总数 gateway_child_device_total_count: 网关子设备总数 online_gateway_child_device_total_count: 网关子设备在线总数 offline_gateway_child_device_total_count: 网关子设备离线总数 inactive_gateway_child_device_total_count: 网关子设备未激活总数 stop_gateway_child_device_total_count: 网关子设备停用总数 tags_total_count: 测点总数 project_id: 项目 ID alarms_important_week_count: 重要报警周计数 alarms_week_count: 报警周计数 alarms_crux_week_count: 关键报警周计数 alarms_commonly_week_count: 常见报警周计数 alarms_secondary_week_count: 次要报警周计数 alarms_tips_week_count: 提示报警周计数 manu_entity_total_count: 工厂实体总数

5.2.4.请求示例

只需关注请求参数中提到的参数部分，其他参数固定无需修改

注：project_id 参数可选，不需要时整体删除这部分

```
    [
        "project_id",
        "=",
        "03efoznia1hc0"
    ],
```

示例：

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
     "useDisplayForModel": true,
     "filter": [
       [
         "project_id",
         "=",
         "03efoznia1hc0"
       ]
     ],
     "properties": [
       "tags total count",
       "thing entity total count",
       "manu entity total count",
       "direct device total count",
       "online direct device total count",
       "offline direct device total count",
       "inactive direct device total count",
       "stop direct device total count",
       "smdc device total count",
       "online smdc device total count",
       "offline smdc device total count",
       "inactive smdc device total count",
       "stop smdc device total count",
       "gateway device total count",
       "online gateway device total count",
       "offline gateway device total count",
       "inactive gateway device total count",
       "stop gateway device total_count",
       "thing model total count",
       "device thing model total count",
       "gateway thing model total count",
       "smdc thing model total count",
       "manu thing model total count",
       "smdc child device total count",
       "online smdc child device total count",
       "offline smdc child device total count",
       "inactive smdc child device total count",
       "stop smdc child device total count",
       "gateway child device total count",
       "online gateway child device total count",
       "offline gateway child device total count",
       "inactive gateway child device total count",
       "stop gateway child device_total_count",
       "alarms week count",
       "alarms crux week count",
       "alarms important week count",
       "alarms commonly week count",
       "alarms secondary week count",
       "alarms_tips_week_count"
     ]
    },
    "context": {
      "uid": "",
      "lang": "zh_CN"
    },
    "model": "iiot thing_overview_model",
    "tag": "master",
    "service": "search",
    "app": "iiot_thing"
  }
}
```

5.2.5.返回示例

```
{
    "result": {
```

```
        "data":
            [
                {
                    "gateway thing model total count": 0,
                    "alarms important week count": 0,
                    "alarms tips week count": 0,
                    "device thing model total count": 0,
                    "smdc thing model total count": 0,
                    "alarms secondary week count": 0,
                    "alarms week count": 0,
                    "alarms crux week count": 0,
                    "manu thing model total_count": 0,
                    "tags total count": 0,
                    "alarms commonly week count": 0,
                    "thing_model_total_count": 0
                }
             ],
        "context": {
            "@Class": "java.util.HashMap",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.3. （Websocket 订阅）订阅地址获取

5.3.1.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.3.1.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/entityId	String	查询的设备实体	"sagjf3435sfa"	是
params/args/channel	String	查询的频道	固定为"websocket"	是
params/model	String	模型	"iiot_thing_entity"	是
params/service	String	服务	"getIpAddressById"	是

5.3.1.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	"192.168.174.169:31882"	

5.3.1.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
        "entityId":"046bzfu5vnq5e",
        "channel":"websocket"
    },
    "context": {
      "uid": "",
      "timeZone": "UTC+8",
      "lang": "zh-CN"
    },
    "model": "iiot_thing_entity",
    "tag": "master",
    "service": "getIpAddressById",
    "app": "iiot_thing"
  }
}
```

5.3.1.5. 返回示例

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "result": {
        "data": "192.168.174.169:31882",
        "context": {
            "@Class": "java.lang.String",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    }
}
```

5.4. （Websocket 订阅）订阅

连接上 websocket 后发送订阅请求,订阅成功后会持续返回数据，只需订阅一次即可

5.4.1.1. 请求参数

```
{
  "ts": 16297912232543,
  "reporter": "iiot",
  "type": "Subscribe",
  "data": {
    "dataType": "Request",
    "models": [
      {
        "id": "ad4ad320",/ 实体 id（entity_id） or 模型 id（model_id） */
        "mode": "entity",/*订阅模式: （物实体）entity or （ model*/
        "subps": {
```

```
            "valueChanged": /*上报模式：（值变化）valueChanged or （实时）time*/
            {
              "0": ["3a0401d4"]/*key：订阅参数；value：属性 ID 集合*/
            }
          }
        }
      ]
    }
}
```

5.4.1.2. 返回参数

```
{
  "ts": 16297888953494,
  "reporter": "iiot",
  "type": "Subscribe",
  "data": {
    "dataType": "Response",
    "models": [
      {
        "Id": "3a0401d4"/*设备 Id*/,
        "ps": {/*key:时间戳；value:属性值列表*/
          "16297888953469": {
            "temperature ": "40"/*key:属性 Id；value:属性值*/,
            "property2": "660",
            "property3": "5",
            "qs": {/*key:qs；value:异常质量戳*/
              "bad": ["temperature"]/*key:质量戳名称；value:属性 Id*/
            }
          }
        }
      }
    ]
  }
}
```

5.4.1.3. 请求示例

```
{
  "seq": 1719538961001,
  "ts": 1719538961001,
  "reporter": "iot",
  "type": "Subscribe",
  "data": {
    "dataType": "Request",
```

```
    "models": [
        {
          "id": "044vxpr6hklom",
          "mode": "entity",
          "subps": {
                "time": {
                "0": [
                    "IO6"
                    ]
                }
            }
        }
    ]
  }
}
```

5.4.1.4. 返回示例

```
{
     "seq": 11043,
     "ts": 1755767747354,
     "reporter": "iiot",
     "type": "DataReport",
   "data": {
      "dataType": "ReportData",
      "models": [
          {
             "id": "04gcda931k6r7",
             "old value": {
                "IO6": 111
             },
             "ps": {
                "1755767746898": {
                    "IO6": 111,
                    "qs": {}
                }
             }
          }
      ]
   }
}
```

5.5. （Websocket 订阅）取消订阅

连接上 websocket 后发送取消订阅请求

5.5.1.1. 请求参数

三方应用向平台取消订阅属性上报事件：

```
{
  "ts": 16297888953494,
  "reporter": "ebox1",
  "type": "UnSubscribe",
  "data": {
    "dataType": "Request",
    "models": [
      {
        "id": "ad4ad320", /*（物实体 id) entity_id or （物模型 id） model_id*/
       "mode": "entity", /*订阅模式：（物实体） entity or （物模型） model*/
        "unsubps": ["8b9edd0b"] /*取消订阅的属性 Id*/
      }
```

```
      ]
   }
}
```

5.5.1.2. 返回参数

```
{
   "ts": 16297888953494,
   "reporter": "ebox1",
   "type": "UnSubscribe",
   "data": {
      "dataType": "Response",
      "value": "true"
   }
}
```

5.5.1.3. 请求示例

```
{
    "ts": 16297888953494,
    "reporter": "ebox1",
    "type": "UnSubscribe",
    "data": {
        "dataType": "Request",
        "models": [
            {
                "id": "04gcda931k6r7",
                "mode": "entity",
                "unsubps": [
                    "IO6"
                ]
            }
        ]
    }
}
```

5.5.1.4. 返回示例

```
{
   "seq": 200976,
   "ts": 1755771066447,
   "reporter": "iiot",
   "type": "UnSubscribe",
   "data": {
      "dataType": "Response",
      "value": "true"
   }
}
```

5.6. 查询历史工况

注意：
每次查询最多只能查询一个设备的历史数据；

每次查询只能查询同类型的测点数据。

5.6.1.请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.6.2.请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/propertyPaths	String[]	属性路径	["03bv8e4gl883k.temperature"]	是
params/args/startTime	Date	开始时间	"2023-11-07 08:54:45"	是
params/args/endTime	Date	结束时间	"2023-11-07 09:54:45"	是
params/args/limit	Integer	查询行数，最大 100000	30	是
params/args/offset	Integer	跳过行数	0	是
params/model	String	模型	"iiot_thing_entity_history_tagvalue"	是
params/service	String	服务	"searchHistoryData"	是

5.6.3.返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	[{
			"03bv8e4gl883k": [{
			"temperature": [{
			"cloud_time": "2023-11-07 09:39:41",
			"qs": "good",
			"entity_name": "电表 A-1",
			"name": "温度",
			"generation_time": "2023-11-07 09:39:42",
			"custom_property_id": "temperature",
			"current_value": 46,
			"basic_time": "2023-11-07 09:39:42"
			}]
			}]
			}]
				cloud_time:入云时间；
				qs:质量戳；
				entity_name:设备名称；
				name:属性名称；
				generation_time:生产时间；
				custom_property_id:自定义属性 ID；
				current_value: 实时值；
				basic_time:基准时间

5.6.4.请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
```

```
      "propertyPaths": [
       "04a1a0tv6fon0.property500",
      "04a1a0tv6fon0.property499"
       ],
       "startTime": "2024-10-18 08:54:45",
       "endTime": "2024-10-18 09:54:45",
      "offset": 0,
      "limit": 1
    },
    "context": {
      "uid": "",
      "lang": "zh_CN"
    },
    "model": "iiot_thing_entity_history_tagvalue",
    "tag": "master",
    "service": "searchHistoryData",
    "app": "iiot_thing"
  }
}
```

5.6.5.返回示例

```
{
    "result": {
        "data": {
            "04a1a0tv6fon0": {
                "property499": [
                    {
                        "cloud time": "2024-10-18 09:47:32.147",
                        "qs": "good",
                        "entity name": "涂布机 4",
                        "name": "property499",
                        "generation time": "2024-10-18 09:47:30.771",
                        "custom property id": "property499",
                        "entity id": "04a1a0tv6fon0",
                        "current value": 6065742,
                        "basic_time": "2024-10-18 09:47:30.771"
                    }
                ],
                "property500": [
                    {
                        "cloud time": "2024-10-18 09:47:32.147",
                        "qs": "good",
                        "entity name": "涂布机 4",
                        "name": "property500",
                        "generation time": "2024-10-18 09:47:30.771",
                        "custom property id": "property500",
                        "entity id": "04a1a0tv6fon0",
                        "current value": 6065742,
                        "basic_time": "2024-10-18 09:47:30.771"
                    }
                ]
            }
        },
        "context": {
            "@Class": "java.util.HashMap",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.7.查询实时工况

支持主动查询（webapi）和被动接收（转发配置）。

5.7.1.主动查询 1

适用于批量查询具有相同属性的多个实体的数据

注意：

- 主动查询返回的结果为最近一次上报的设备工况。
- SMDC/网关离线期间采集的数据属于历史数据，无法通过查询实时数据获取。

5.7.1.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.7.1.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/entityIds	String[]	查询的设备实体	["sagjf3435sfa"]	是
params/args/customPropertyIds	String[]	查询的测点自定义属性 ID，为 null 或空时，查询设备实体的所有测点的实时工况	["EPz","Pa"]	是
params/model	String	模型	"iiot_thing_property"	是
params/service	String	服务	"searchRealData"	是

5.7.1.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	[
			{
			"cloud_time": "2023-11-07 09:39:41.688",
			"time": "2023-11-07 09:39:41.488",
			"qs": "good",
			"entity_id": "sagjf3435sfa",
			"property_id": "sedfoi35sgh",
			"custom_property_id": "EPz",
			"value": 46183857.34
			}
			]
				cloud_time:入云时间；
				time:时间；
				qs:质量戳；
				entity_id:设备实体 ID；
				property_id:属性 ID；
				custom_property_id:自定义属性 ID；
				value: 实时值；

5.7.1.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
```

```
  "params": {
    "args": {
      "entityIds": ["04a1a0tv6fon0"],
      "customPropertyIds": ["property499","property500"]
    },
    "context": {
      "uid": "",
      "lang": "zh_CN"
    },
    "model": "iiot_thing_property",
    "tag": "master",
    "service": "searchRealData",
    "app": "iiot_thing"
  }
}
```

5.7.1.5. 返回示例

```
{
    "result": {
        "data": [
            {
                "time": "2024-10-18 11:02:03.432",
                "cliud time": "2024-10-18 11:02:03.632",
                "entity id": "04a1a0tv6fon0",
                "custom property id": "property500",
                "value": 6074654,
                "qs": "good",
                "property_id": "049xgivbrn6f2"
            },
            {
                "time": "2024-10-18 11:02:03.432",
                "time": "2024-10-18 11:02:03.632",
                "entity id": "04a1a0tv6fon0",
                "custom property id": "property499",
                "value": 6074654,
                "qs": "good",
                "property_id": "049xgd553vb2t"
            }
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.7.2.主动查询 2

适用于批量查询不同属性的多个实体的数据

注意：

- 主动查询返回的结果为最近一次上报的设备工况。
- SMDC/网关离线期间采集的数据属于历史数据，无法通过查询实时数据获取。

5.7.2.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.7.2.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是

params/args/entityIdCustomPropertyIds	JsonArray	查询的设备实体及其属性，属性为空时，查询设备的所有测点的实时工况	[
			{
			"entityId": "sagjf3435sfa",
			"customPropertyId": [
			"EPz",
			"Pa"
			]
			}
			]
				是

params/model	String	模型	"iiot_thing_property"	是
params/service	String	服务	"searchPropertyRealData"	是

5.7.2.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	[
			{
			"time": "2023-11-07 09:39:41.488",
			"qs": "good",
			"entity_id": "sagjf3435sfa",
			"property_id": "sedfoi35sgh",
			"custom_property_id": "EPz",
			"value": 46183857.34
			}
			]
				time:时间;
				qs:质量戳;
				entity_id:设备实体 ID;
				property_id:属性 ID;
				custom_property_id:自定义属性 ID;
				value: 实时值;

5.7.2.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "entityIdCustomPropertyIds": [
        {
          "entityId": "04a1a0tv6fon0",
          "customPropertyId": [
            "property499",
            "property500"
          ]
        }
      ]
    },
    "context": {
      "uid": "",
      "lang": "zh_CN"
    },
    "model": "iiot_thing_property",
    "tag": "master",
    "service": "searchPropertyRealData",
    "app": "iiot_thing"
  }
}
```

5.7.2.5. 返回示例


```
{
    "result": {
        "data": [
            {
                "time": "2024-10-18 11:18:59.437",
                "entity id": "04a1a0tv6fon0",
                "custom property id": "property500",
                "value": 6076678,
                "property_id": "049xgivbrn6f2"
            },
            {
                "time": "2024-10-18 11:18:59.437",
                "entity id": "04a1a0tv6fon0",
                "custom property id": "property499",
                "value": 6076678,
                "property_id": "049xgd553vb2t"
            }
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.7.3.被动接收

被动接收配置下，当有设备上报测点数据时，会实时接收到测点数据。

支持 Mqtt、Rabbitmq、OPCUA 和 WebApi 方式，配置方法请参考"IIOT 用户操作手册"的第五章"数据转发"。

Mqtt：转发到 Mqtt 服务，支持缓存固定时间，超过缓存时间的数据将不再转发；

Rabbitmq：转发到 Rabbitmq 服务，支持缓存固定时间，超过缓存时间的数据将不再转发；

WebApi：转发到 WebApi 服务，支持缓存固定时间，超过缓存时间的数据将不再转发；

OPCUA：通过订阅 OPCUA 服务，或定时调用 OPCUA 服务获取数据，不支持缓存，需要保持连接，断开后会丢失数据；

5.8.通用指令执行/下发

在平台上给物模型配置指令，生成指令 ID，在系统对接时，可通过指令 ID 调用接口执行该指令。

5.8.1.请求语法

POST /api/root/rpc/service/ HTTP/2.0

Header：Authorization:Bearer token 字符串

5.8.2.请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/orderId	String	指令 ID	"sie_NTP"	是
params/args/entityId	String	设备实体 ID	"03q02whxzyf7z"	是
params/model	String	模型	"iiot_thing_order"	是
params/service	String	服务	"executeOrder"	是

5.8.3.返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	true	true：成功； false：失败

5.8.4.请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
       "entityId": "03q02whxzyf7z",
       "orderId": "sie_NTP"
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_thing_order",
    "tag": "master",
    "service": "executeOrder",
    "app": "iiot_thing"
  }
}
```

5.8.5.返回示例

```
{
    "result": {
        "data": true,
        "context": {
            "@Class": "java.lang.Boolean",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.9. 指令执行（带参数）

5.9.1.请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.9.2.请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/orderId	String	指令 ID	"sie_NTP"	是
params/args/entityId	String	设备实体 ID	"03q02whxzyf7z"	是
params/args/parameterMap	Json	参数	{"temperature":34}	是
params/model	String	模型	"iiot_thing_order"	是
params/service	String	服务	"executeOrderWithParamsById"	是

5.9.3.返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	true	true：成功； false：失败

5.9.4.请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
       "entityId": "03q02whxzyf7z",
       "orderId": "sie NTP",
       "parameterMap": {
         "temperature": 34
       }
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_thing_order",
    "tag": "master",
    "service": "executeOrderWithParamsById",
    "app": "iiot_thing"
  }
}
```

5.9.5.返回示例


```
{
    "result": {
        "data": true,
        "context": {
            "@Class": "java.lang.Boolean",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.10. 执行属性点写值

5.10.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.10.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/orderId	String	指令 ID	"sie_NTP"	是
params/args/entityId	String	设备实体 ID	"03q02whxzyf7z"	是
params/args/parameterMap	Json	参数	{"temperature":34}	是
params/model	String	模型	"iiot_thing_order"	是
params/service	String	服务	"propertyUpdateOrderWithParamsByOrderId"	是

5.10.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	true	true：成功； false：失败

5.10.4. 请求示例


```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
```

```
  "params": {
    "args": {
      "entityId": "03q02whxzyf7z",
      "orderId": "sie NTP",
      "parameterMap": {
        "temperature": 34
      }
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_thing_order",
    "tag": "master",
    "service": "propertyUpdateOrderWithParamsByOrderId",
    "app": "iiot_thing"
  }
}
```

5.10.5. 返回示例

```
{
    "result": {
        "data": true,
        "context": {
            "@Class": "java.lang.Boolean",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.11. 执行自定义指令

5.11.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.11.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/orderId	String	指令 ID	"sie_NTP"	是
params/args/entityId	String	设备实体 ID	"03q02whxzyf7z"	是
params/args/parameterMap	Json	参数	{"temperature":34}	是
params/model	String	模型	"iiot_thing_order"	是
params/service	String	服务	"customOrderWithParamsByOrderId"	是

5.11.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	true	true：成功； false：失败

5.11.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
       "entityId": "03q02whxzyf7z",
       "orderId": "sie NTP",
       "parameterMap": {
         "temperature": 34
       }
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_thing_order",
    "tag": "master",
    "service": "customOrderWithParamsByOrderId",
    "app": "iiot_thing"
  }
}
```

5.11.5. 返回示例

```
{
    "result": {
        "data": true,
        "context": {
            "@Class": "java.lang.Boolean",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.12. 停止指令执行/下发

如果该指令为周期性执行指令，可通过调用本接口停止该指令执行。

5.12.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.12.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/orderId	String	指令 ID	"sie_NTP"	是
params/args/entityId	String	设备实体 ID	"03q02whxzyf7z"	是
params/model	String	模型	"iiot_thing_order"	是
params/service	String	服务	"stopOrderById"	是

5.12.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	true	true：成功； false：失败

5.12.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
       "entityId": "03q02whxzyf7z",
       "orderId": "sie_NTP"
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_thing_order",
    "tag": "master",
    "service": "stopOrderById",
    "app": "iiot_thing"
  }
}
```

5.12.5. 返回示例

```
{
    "result": {
        "data": true,
        "context": {
            "@Class": "java.lang.Boolean",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.13. 报警规则创建

5.13.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.13.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_alarm"	是
params/args/valuesList	String	新建的数据	[{xxx}]	是
params/model	String	模型	"iiot_alarm_rules"	是
params/service	String	服务	"create"	是

valuesList 参数说明

名称	类型	描述	示例	必填
model_id	String	物模型 id	051rqn0vn84va	是
name	String	报警名称	报警 1	是
alarm_type	String	报警级别，0:关键,1:重要,2:一般,3:次要,4:提示	2	是
is_alarm_release_confirm	Boolean	是否报警触发后需要手动确认	false	是
alarm_mode	String	报警模式 0:仅报警 1 次，1:持续报警	0	否
set_alarm	Integer	持续报警的间隔值	10	否
date_type	String	持续报警时的时间类型，second：秒,minute: minute, hour：小时,day:天	minute	否
delay_alarm	Integer	delay_alarm:延迟报警时间，单位：秒	30	否
attribute_value	String	与报警同时上报的属性值	a,b	否
alarm_reason	String	报警原因	扫码枪离线,请检查网络	否
solution	String	解决方案	如断网，检查下网络等等	否

label	String	报警标签		否
complex_rules	Boolean	是否设置复杂规则	false	是
alarm_easy_rules	String	简单规则，complex_rules 为 false 时必填。格式参考平台的 filter，如果是单一条件，格式为两层数据，里面数组的第一个和第二个参数为 null，第三个为报警条件，具体参数看下方部分	[[null,null, {xxx:xxx}]]	否
alarm_custom_rules	String	复杂规则，complex_rules 为 true 时必填，格式为 groovy 表达式	$model.报警测试模型.a.value$>10	否
alarm_release	false	是否设置报警解除条件	false	是
alarm_release_value	String	报警解除内容, 格式为 groovy 表达式	$model.报警测试模型.a.value$<10	否
alarm_conditions	String	报警前置条件，格式和 alarm_easy_rules 一样		否

alarm_easy_rules 参数说明

名称	类型	描述	示例	必填
alarm_linkage_type	String	报警联动的类型 0：属性 1：报警规则，报警条件联动才用	0	否
property_id	String	自定义的属性 ID	a	是
property_type	String	属性类型(0:数值类型,1:其他类型)，属性的类型如果是 Integer、Double、Long 则填 0，其他类型的属性填 1	0	是
alarm_id	String	报警规则 ID	051rqn0vn84va	否
dead_zone	Integer	死区		否
left_symbol	String	左符号(>,<,>=,<=,==,!=)	>	否
lvalue	String	左侧条件值	10	否
right_symbol	String	右符号(>,<,>=,<=,==,!=)	>	否
rvalue	String	右侧条件值	10	否
rate_of_change_alarm	Boolean	是否变化率报警	false	否
hl_change_rate	String	//高低变化率(0:高变化率,1:低变化率)	0	否

value_type	String	变化率值类型[Percent（百分比）/AbsoluteValue（绝对值）]	Percent	否
value	Integer	变化率值		否
unidirectional	Boolean	单方向(true,false)	true	否
direction_type	String	方向类型 Increase（增加）/Deincrease（减少）	Increase	否
time_unit	String	时间单位 Second/Minute/Hour (秒/分钟/小时)	Second	否
max_value	Double	最大值	100	否
min_value	Double	最小值	10	否
no_change_timeout	Integer	无变化超时，单位：秒	100	否

5.13.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	{
			"result": {
			"data": ["054c3mizdiwjx"]
			}
			}	

5.13.4. 请求示例


```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "params": {
        "args": {
            "valuesList": [{
                "model id": "051rqn0vn84va",
                "name": "alarm a",
                "alarm type": "2",
                "is alarm release confirm": false,
                "is reference": false,
                "alarm template": null,
                "alarm mode": "0",
                "set alarm": 10,
                "date type": "minute",
                "delay alarm": null,
                "attribute value": "a",
                "alarm reason": null,
                "solution": null,
                "label": null,
                "complex rules": false,
                "alarm easy rules":
"[[null,null,{\"property id\":\"a\",\"property value type\":0,\"siqu\":null,\"dead zon
e\":null,\"lvalue\":\"1\",\"left symbol\":\"==\",\"property name\":null,\"right symbol
\":null,\"rvalue\":null,\"rate of change alarm\":false,\"hl change rate\":0,\"value ty
pe\":\"Percent\",\"value\":null,\"unidirectional\":false,\"direction type\":null,\"tim
e unit\":\"Second\",\"min_value\":\"\",\"max_value\":\"\",\"no_change_timeout\":10,\"p
roperty type\":1}]]",
                "alarm custom rules": null,
                "alarm release": false,
                "alarm release value": null,
                "alarm conditions":
"[[null,null,{\"alarm_linkage_type\":null,\"property_id\":null,\"alarm_id\":null,\"pro
```

```
perty value type\":0,\"siqu\":null,\"dead zone\":null,\"lvalue\":null,\"left symbol\":
null,\"property_name\":null,\"right_symbol\":null,\"rvalue\":null,\"property_type\":1}]
]"
            }]
        },
        "app": "iiot alarm",
        "service": "create",
        "context": {
            "uid": "",
            "lang": "zh_CN"
        },
        "model": "iiot alarm_rules",
        "tag": "master"
    },
    "method": "service"
}
```


5.13.5. 返回示例


```
{
    "result": {
        "data": [
            "054c3mizdiwjx"
        ],
        "context": {
            "@Class": "com.sie.snest.engine.data.RecordSet",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "error": null,
    "jsonrpc": "2.0"
}
```


5.14. 报警规则修改

5.14.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.14.2. 请求参数

请求参数和上一节【报警规则创建】里的请求参数保持一致

名称	类型	描述	示例	必填
params/args/ids	String	报警规则 id	054c3mizdiwjx	是
params/args/values	Map	报警规则内容	上一节【报警规则创建】里的请求参数保持一致	是

5.14.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果
		数据
			{
			"result": {
			"data": ["054c3mizdiwjx"]
			}
			}	

5.14.4. 请求示例

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "params": {
        "args": {
            "ids":["054c3mizdiwjx"],
            "values": {
                "model id": "051rqn0vn84va",
                "name": "alarm a",
                "alarm type": "2",
                "is alarm release confirm": false,
                "is reference": false,
                "alarm template": null,
                "alarm mode": "0",
                "set alarm": 10,
                "date type": "minute",
                "delay alarm": null,
                "attribute value": "a",
                "alarm reason": null,
                "solution": null,
                "label": null,
                "complex rules": false,
                "alarm easy rules":
"[[null,null,{\"property id\":\"a\",\"property value type\":0,\"siqu\":null,\"dead zon
e\":null,\"lvalue\":\"1\",\"left symbol\":\"==\",\"property name\":null,\"right symbol
\":null,\"rvalue\":null,\"rate of change alarm\":false,\"hl change rate\":0,\"value ty
pe\":\"Percent\",\"value\":null,\"unidirectional\":false,\"direction type\":null,\"tim
e unit\":\"Second\",\"min_value\":\"\",\"max_value\":\"\",\"no_change_timeout\":10,\"p
roperty type\":1}]]",
                "alarm custom rules": null,
                "alarm release": false,
                "alarm release value": null,
                "alarm conditions":
"[[null,null,{\"alarm linkage type\":null,\"property id\":null,\"alarm id\":null,\"pro
perty value type\":0,\"siqu\":null,\"dead zone\":null,\"lvalue\":null,\"left symbol\":
null,\"property_name\":null,\"right_symbol\":null,\"rvalue\":null,\"property_type\":1}]
]"
            }
        },
        "app": "iiot alarm",
        "service": "update",
        "context": {
            "uid": "",
            "lang": "zh_CN"
        },
        "model": "iiot alarm_rules",
        "tag": "master"
    },
    "method": "service"
}
```

5.14.5. 返回示例


```
{
    "result": {
        "data": [
            "054c3mizdiwjx"
        ],
        "context": {
            "@Class": "com.sie.snest.engine.data.RecordSet",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "error": null,
    "jsonrpc": "2.0"
}
```


5.15. 报警历史记录

5.15.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.15.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/entityId	String	设备实体 ID	"03q02whxzyf7z"	是
params/args/isDeal	Boolean	是否已解除	true	是
params/args/limit	Integer	查询行数，最大 5000	30	是
params/args/offset	Integer	跳过行数	0	是
params/args/order	String	排序	"trigger_time desc"	否
params/args/triggerStartTime	String	触发开始时间	"2025-08-18 00:00:00"	否
params/args/triggerEndTime	String	触发结束时间	"2025-08-19 00:00:00"	否
params/model	String	模型	"iiot_alarm_history"	是
params/service	String	服务	"searchHistory"	是

5.15.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果
		数据
			[
			{
			"alarm_name": "温度过高",
			"alarm_state": "0",
			"relieve_time": null,
			"label": "",
			"confirmation_time": null,
			"trigger_time": "2024-09-13 14:43:37",
			"alarm_type": "0",
			"solution": "",
			"time": "2024-09-03T00:28:04.478Z",
			"alarm_id": "0459h0xdorciw",
			"attribute_value": "temperature=99",
			"state": "0",
			"id": "045a1gnpgk58w",
			"alarm_reason": "",
			"initial_trigger_time": "2024-09-03 08:28:04"
			}
			]
				alarm_name: 名称
				alarm_state: 状态
				relieve_time: 解除时间
				label: 标签
				confirmation_time: 确认时间
				trigger_time: 触发时间
				alarm_type: 报警级别
				solution: 解决方案
				time: 存储时间
				alarm_id: 报警 ID
				attribute_value: 同时上报的属性
				state: 手动 or 自动解除
				id: ID
				alarm_reason: 报警原因
				initial_trigger_time: 初次触发时间

5.15.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "entityId": "049xgus8ujdnq",
      "isDeal": false,
      "limit": 30,
      "offset": 0,
      "order": "trigger time desc",
      "triggerStartTime": "2025-08-18 00:00:00",
      "triggerEndTime": "2025-08-19 00:00:00"
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_alarm_history",
    "tag": "master",
    "service": "searchHistory",
    "app": "iiot_thing"
  }
}
```

5.15.5. 返回示例

```
{
    "result": {
        "data": [
            {
                "alarm name": "温度过高",
                "alarm state": "0",
                "relieve time": null,
                "label": "",
                "confirmation time": null,
                "trigger time": "2024-10-18 14:09:10.341",
                "alarm type": "2",
                "solution": "加水",
                "time": "2024-10-18T06:08:15.666Z",
                "alarm_id": "04a7vg4wikpcc",
```

```
                "attribute value": "temperature=99",
                "state": "0",
                "id": "04a7vg8m5ah95",
                "alarm reason": "温度过高",
                "initial_trigger_time": "2024-10-18 14:08:15.666"
            }
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.16. 查询初次报警时的触发值

5.16.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.16.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/alarmIds	String	报警历史 ID，注意不是报警 ID，从查询报警历史记录结果的 id 字段可获得	["045a1gnpgk58w"]	是
params/model	String	模型	"iiot_alarm_history"	是
params/service	String	服务	"searchAlarmValue"	是

5.16.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	[
			{
			"id": "045a1gnpgk58w",
			"alarmValue": "temperature=99"
			}
			]	

5.16.4. 请求示例


```
{
  "id": "guid",
```

```
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
       "alarmIds": ["04a7vg8m5ah95"]
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_alarm_history",
    "tag": "master",
    "service": "searchAlarmValue",
    "app": "iiot_thing"
  }
}
```

5.16.5. 返回示例

```
{
    "result": {
        "data": [
            {
                "alarmValue": "temperature=99",
                "id": "04a7vg8m5ah95"
            }
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.17. 查询聚合历史

5.17.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.17.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_aggregation"	是
params/args/entityId	String	设备实体 ID	"042yveivbkltv"	是
params/args/ruleId	String	聚合规则 ID	"A1_1_spread"	是
params/args/startTime	Date	开始时间	"2024-08-06 11:30:00.000"	是
params/args/endTime	Date	结束时间	"2024-08-06 11:45:00.000"	是
params/args/limit	Integer	查询行数	30	是
params/args/offset	Integer	跳过行数	0	是

params/model	String	模型	"iiot_aggregation_history"	是
params/service	String	服务	"searchAggregationHistory"	是

5.17.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果
		数据
			[
			{
			"start_time": "2024-08-06 11:30:00.000",
			"end_time": "2024-08-06 11:45:00.000",
			"exec_time": "2024-08-06 11:52:08.094",
			"value": "87",
			"entity_id": "042yveivbkltv",
			"custom_property_id": "A1_1_spread"
			}
			]
				start_time: 开始时间
				end_time: 结束时间
				exec_time: 执行时间
				value: 值
				entity_id:设备实体 ID
				custom_property_id: 自定义属性 ID

5.17.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "entityId": "042yveivbkltv",
      "ruleId": "A1 1 spread",
      "startTime": "2024-08-06 11:30:00.000",
      "endTime": "2024-08-06 11:45:00.000",
      "limit": 30,
      "offset": 0
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot aggregation_history",
    "tag": "master",
    "service": "searchAggregationHistory",
    "app": "iiot_aggregation"
  }
}
```

5.17.5. 返回示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "result": {
    "data": [
      {
        "start time": "2024-08-06 11:30:00.000",
        "end time": "2024-08-06 11:45:00.000",
        "exec time": "2024-08-06 11:52:08.094",
        "value": "87",
        "entity id": "042yveivbkltv",
        "custom_property_id": "A1_1_spread"
      }
    ],
    "context": {
      "uid": "",
      "lang": "zh CN",
      "token": "cae26ebfc0a54a39b2e0da35633d19b6cd11c",
      "tenant": "root"
    }
  }
}
```

5.18. 历史工况聚合查询

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.18.1. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/function	String	查询方法	根据实际查询条件填写，查看 5.13.5 开始的说明。例："last"	是
params/args/entityIds	String[]	查询的设备实体	["sagjf3435sfa"]	是
params/args/customPropertyIds	String[]	查询的属性 id	["IO4"]	是
params/args/startTime	Date	开始时间	支持时间戳 或者日期格式的字符串 yyyy-MM-dd HH:mm:ss.SSS 例： 日期格式"2024-08-06 11:45:00.000" 时间戳格式 1755702000000	是
params/args/endTime	Date	结束时间	支持时间戳 或者日期格式的字符串 yyyy-MM-dd HH:mm:ss.SSS 例： 日期格式"2024-08-06 11:45:00.000" 时间戳格式 1755702000000	是
params/model	String	模型	"iiot_thing_entity_history_tagvalue"	是
params/service	String	服务	"aggData"	是

5.18.2. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	[
			{
			"time": "2025-08-20 00:28:16.000",
			"entity_id": "04gcda931k6r7",
			"custom_property_id": "IO4",
			"value": 334.0
			}
			]
				time: 开始时间
				value: 值
				entity_id:设备实体 ID
				custom_property_id: 自定义属性 ID

5.18.3. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
```

```
    "args": {
      "entityId": "042yveivbkltv",
      "ruleId": "A1 1 spread",
      "startTime": "2024-08-06 11:30:00.000",
      "endTime": "2024-08-06 11:45:00.000",
      "limit": 30,
      "offset": 0
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_aggregation_history",
    "tag": "master",
    "service": "searchAggregationHistory",
    "app": "iiot_aggregation"
  }
}
```


5.18.4. 返回示例


```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "result": {
    "data": [
      {
          "time": "2025-08-20 00:28:16.000",
          "entity id": "04gcda931k6r7",
          "custom property id": "IO4",
          "value": 334.0
      }
    ],
    "context": {
      "uid": "",
      "lang": "zh CN",
      "token": "cae26ebfc0a54a39b2e0da35633d19b6cd11c",
      "tenant": "root"
    }
  }
}
```


5.18.5. 查询最后一个值

请求参数中 function 设置为"last"

5.18.6. 查询最大值

请求参数中 function 设置为"max"

5.18.7. 查询最小值

请求参数中 function 设置为"min"

5.18.8. 查询平均值

请求参数中 function 设置为"avg"

5.18.9. 查询总和值

请求参数中 function 设置为"sum"

5.18.10. 查询中间值

请求参数中 function 设置为"middle"

5.18.11. 查询总数值

请求参数中 function 设置为"count"

5.18.12. 查询去重总数值

请求参数中 function 设置为"discount"

5.18.13. 查询极差值

请求参数中 function 设置为"spread"

5.18.14. 查询方差值

请求参数中 function 设置为"variance"

5.18.15. 查询标准差值

请求参数中 function 设置为"stddev"

5.18.16. 查询相邻和值

请求参数中 function 设置为"adjacentSum"

5.18.17. 查询相邻差值

请求参数中 function 设置为"adjacentDiff"

5.18.18. 查询相邻积值

请求参数中 function 设置为"adjacentMulti"

5.18.19. 查询相邻除值

请求参数中 function 设置为"adjacentDiffRate"

5.18.20. 查询状态聚合

请求参数中 function 设置为"state"

注意：返回值会有区别，增加了持续时长 duration（单位 ms）


```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "result": {
    "data": [
      {
          "duration": 1038,
          "time": "2025-08-20 00:28:16.000",
          "entity id": "04gcda931k6r7",
          "custom property id": "IO4",
          "value": 334.0
      }
    ],
    "context": {
      "uid": "",
      "lang": "zh CN",
      "token": "cae26ebfc0a54a39b2e0da35633d19b6cd11c",
      "tenant": "root"
    }
  }
}
```


5.19. 设备上下线日志查询

5.19.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.19.2. 请求参数

名称	类型	描述	示例	必填

params/app	String	APP 名称	"iiot_base"	是

params/args/labels	JSON	标签，iiot_thing_entity 标签值固定为 onoffline	{
			"entity_id": "03zlg99lb6jm6",
			"iiot_thing_entity": "onoffline"
			}
				是

params/args/start	long	开始时间戳（纳秒级）	1726383602000000000	是
params/args/end	long	结束时间戳（纳秒级）	1728975602000000000	是
params/args/limit	long	查询行数，最大 5000	30	是
params/args/direction	String	排序，backward（时间倒序），forward（时间倒序）	backward	是
params/args/line	String	关键词	"ip"	是
params/model	String	模型	"iiot_log_loki"	是
params/service	String	服务	"queryRange"	是

5.19.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	{
			"result": [
			{
			"stream": {
			"level": "info",
			"iiot_thing_entity": "onoffline",
			"entity_id": "049xgus8ujdnq"
			},
			"values": [
			[
			"1729234895265000000",
			"{\n \"sie_online\":0,\n
			\"duration\":559,\n
			\"gateway_id\":\"wjuji_smdc2\",\n
			\"service\":\"iiot-0\",\n
			\"reason\":\"associated_gateway_offline\",\n
			\"sie_last_report_time\":\"1729234335587\",\n
			\"operation\":\"下线\"\n}"
			]
			]
			}
			]
			}
				stream：标签和标签值
				values：日志内容

5.19.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "labels": {
        "entity id": "049xqus8ujdnq",
        "iiot_thing_entity": "onoffline"
      },
      "start": "1729234746000000000",
      "end": "1729234896000000000",
      "limit": 30,
      "direction": "backward",
      "line": ""
    },
```

```
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot log_loki",
    "tag": "master",
    "service": "queryRange",
    "app": "iiot_base"
  }
}
```

5.19.5. 返回示例

```
{
    "result": {
        "data": {
            "result": [
                {
                    "stream": {
                        "level": "info",
                        "iiot thing entity": "onoffline",
                        "entity_id": "049xgus8ujdnq"
                    },
                    "values": [
                        [
                            "1729234895265000000",
                            "{\n  \"sie online\":0,\n  \"duration\":559,\n
\"gateway id\":\"wjuji smdc2\",\n  \"service\":\"iiot-0\",\n
\"reason\":\"associated gateway offline\",\n
\"sie last report time\":\"1729234335587\",\n  \"operation\":\"下线\"\n}"
                        ]
                    ]
                }
            ]
        },
        "context": {
            "@Class": "com.alibaba.fastjson.JSONObject",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.20. 备份数据库实例数据库清单查询

5.20.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.20.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_backup"	是
params/model	String	模型名称	"iiot_data_backup_query"	是
params/service	String	服务名称	"searchBackupDbInstDbs"	是

params/args/dbInstanceName	String	数据库实例名称	"TDengine"	是

5.20.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果
		数据
			{
			"result": {
			"data": [
			"sie_iiot_oracle",
			"sie_iiot",
			"sie_alarm",
			"sie_iiot_iidp"
			]
			}
			}	

5.20.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "dbInstanceName": "TDengine"
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot data_backup_query",
    "tag": "master",
    "service": "searchBackupDbInstDbs",
    "app": "iiot_data_backup"
  }
}
```

5.20.5. 返回示例

```
{
    "result": {
        "data": [
            "sie iiot oracle",
            "sie iiot",
            "sie alarm",
            "aie_iiot_iidp"

        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.21. 备份数据库实例某数据库表清单查询

5.21.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.21.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_backup"	是
params/model	String	模型名称	"iiot_data_backup_query"	是
params/service	String	服务名称	"searchBackupDbInstTables"	是
params/args/dbInstanceName	String	数据库实例名称	"TDengine"	是
params/args/dbName	String	数据库名称	"sie_iiot"	是

5.21.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果
		数据
			{
			"result": {
			"data": [
			"device_tag_value_boolean",
			"device_tag_value_double",
			"device_tag_value_int",
			"device_tag_value_string"
			]
			}
			}	

5.21.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "dbInstanceName": "TDengine",
      "dbName": "sie_iiot"
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_data_backup_query",
    "tag": "master",
    "service": "searchBackupDbInstTables",
    "app": "iiot_data_backup"
  }
}
```

5.21.5. 返回示例


```
{
    "result": {
        "data": [
            "device tag value boolean",
            "device tag value double",
            "device tag value int",
            "device_tag_value_string"
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.22. 备份数据库实例历史工况数据查询（有问题）

5.22.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_backup"	是
params/model	String	模型名称	"iiot_data_backup_query"	是
params/service	String	服务名称	"searchHistoryTagValue"	是
params/args/dbInstanceName	String	数据库实例名称	"TDengine"	是
params/args/dbName	String	数据库名称	"sie_iiot"	是
params/args/entityId	String	实体 ID	"04aipawi423wj"	是
params/args/customPropertyIds	JSON	属性列表	["property500"]	是
params/args/queryStartTime	String	查询起始时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-18 00:00:00"	是
params/args/queryEndTime	String	查询结束时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-18 23:59:59"	是
params/args/properties	JSON	查询字段，"*"表示全部字段	["*"]	是

params/args/limit	Integer	查询行数，最大值：10000	1000	是
params/args/offset	Integer	跳过行数	0	是
params/args/order	String	排序	"time desc"	否

5.22.2. 请求参数

5.22.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	{ "result": { "data": [{ "qs": "good", "cloud_ts": 1729169211788, "time": "2024-11-18 20:45:20.135", "custom_property_id": "sie_WorkState", "ts": 1729169120135, "current_value": "0", "property_name": "冗余系统工作状态" }] } }	

5.22.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "dbInstanceName": "TDengine",
      "dbName": "sie iiot",
      "entityId": "04aipawi423wj",
      "customPropertyIds": ["property500"],
      "queryStartTime": "2024-11-18 00:00:00",
      "queryEndTime": "2024-11-18 23:59:59",
      "properties": ["*"],
      "limit": 1,
      "offset": 0,
      "order": ""
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_data_backup_query",
    "tag": "master",
    "service": "searchHistoryTagValue",
    "app": "iiot_data_backup"
  }
}
```

5.22.5. 返回示例


```
{
    "result": {
        "data": [
            {
                "qs": "good",
                "cloud ts": 1729169211788,
                "time": "2024-10-17 20:45:20.135",
                "custom_property_id": "sie_WorkState",
                "ts": 1729169120135,
                "current value": "0",
                "property_name": "冗余系统工作状态"
            }
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.23. 备份数据库实例历史工况数据量查询

5.23.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_backup"	是
params/model	String	模型名称	"iiot_data_backup_query"	是
params/service	String	服务名称	"countHistoryTagValue"	是
params/args/dbInstanceName	String	数据库实例名称	"TDengine"	是
params/args/dbName	String	数据库名称	"sie_iiot"	是
params/args/entityId	String	实体 ID	"04aipawi423wj"	是
params/args/customPropertyIds	JSON	属性列表	["property500"]	是
params/args/queryStartTime	String	查询起始时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-18 00:00:00"	是
params/args/queryEndTime	String	查询结束时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-18 23:59:59"	是

5.23.2. 请求参数

5.23.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果
		数据
			{
			"result": {
			"data": 589
			}
			}	

5.23.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "dbInstanceName": "TDengine",
      "dbName": "sie iiot",
      "entityId": "04aipawi423wj",
      "customPropertyIds": ["property500"],
      "queryStartTime": "2024-11-18 00:00:00",
      "queryEndTime": "2024-11-18 23:59:59"
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_data_backup_query",
    "tag": "master",
    "service": "countHistoryTagValue",
    "app": "iiot_data_backup"
  }
}
```

5.23.5. 返回示例

```
{
    "result": {
        "data": 589,
        "context": {
            "@Class": "java.lang.Long",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.24. 备份数据库实例历史聚合数据查询

5.24.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_backup"	是
params/model	String	模型名称	"iiot_data_backup_query"	是
params/service	String	服务名称	"searchHistoryAggregation"	是
params/args/dbInstanceName	String	数据库实例名称	"TDengine"	是
params/args/dbName	String	数据库名称	"sie_iiot"	是
params/args/entityId	String	实体 ID	"03z76vcb0sicu"	是
params/args/aggregationRuleId	String	聚合规则 ID	"TD_integer_two"	是
params/args/queryStartTime	String	查询起始时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-18 00:00:00"	是
params/args/queryEndTime	String	查询结束时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-18 23:59:59"	是
params/args/properties	JSON	查询字段，"*"表示全部字段	["*"]	是
params/args/limit	Integer	查询行数，最大值：10000	1000	是
params/args/offset	Integer	跳过行数	0	是
params/args/order	String	排序	"time desc"	否

5.24.2. 请求参数

5.24.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	{
			"result": {
			"data": [
			{
			"start_time": "2024-10-17	

			12:00:00.000", "end_time": "2024-10-17 13:00:00.000", "exec_time": "2024-10-17 14:00:07.926", "time": "2024-10-17 12:00:00.000", "entity_id": "03z76vcb0sicu", "custom_property_id": "mysql_integer_A1_min_min", "value": 12345 }] } }	

5.24.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "dbInstanceName": "TDengine",
      "dbName": "sie iiot",
      "entityId": "03z76vcb0sicu",
      "aggregationRuleId": "TD integer two",
      "queryStartTime": "2024-11-18 00:00:00",
      "queryEndTime": "2024-11-18 23:59:59",
      "properties": ["*"],
      "limit": 1,
      "offset": 0,
      "order": ""
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot data_backup_query",
    "tag": "master",
    "service": "searchHistoryAggregation",
    "app": "iiot_data_backup"
  }
}
```

5.24.5. 返回示例

```
{
    "result": {
        "data": [
            {
                "start time": "2024-10-17 12:00:00.000",
                "end time": "2024-10-17 13:00:00.000",
                "exec time": "2024-10-17 14:00:07.926",
                "time": "2024-10-17 12:00:00.000",
                "entity id": "03z76vcb0sicu",
                "custom property_id": "mysql_integer_A1_min_min",
                "value": 12345
            }
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.25. 备份数据库实例历史聚合数据量查询

5.25.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_backup"	是
params/model	String	模型名称	"iiot_data_backup_query"	是
params/service	String	服务名称	"countHistoryAggregation"	是
params/args/dbInstanceName	String	数据库实例名称	"TDengine"	是
params/args/dbName	String	数据库名称	"sie_iiot"	是
params/args/entityId	String	实体 ID	"03z76vcb0sicu"	是
params/args/aggregationRuleId	String	聚合规则 ID	"TD_integer_two"	是
params/args/queryStartTime	String	查询起始时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-18 00:00:00"	是
params/args/queryEndTime	String	查询结束时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-18 23:59:59"	是

5.25.2. 请求参数

5.25.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	{
			"result": {
			"data": 120
			}
			}	

5.25.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
```

```
  "params": {
    "args": {
      "dbInstanceName": "TDengine",
      "dbName": "sie iiot",
      "entityId": "03z76vcb0sicu",
      "aggregationRuleId": "TD integer two",
      "queryStartTime": "2024-11-18 00:00:00",
      "queryEndTime": "2024-11-18 23:59:59"
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_data_backup_query",
    "tag": "master",
    "service": "countHistoryAggregation",
    "app": "iiot_data_backup"
  }
}
```

5.25.5. 返回示例

```
{
    "result": {
        "data": 120,
        "context": {
            "@Class": "java.lang.Long",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.26. 备份数据库实例历史报警数据查询

5.26.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_backup"	是
params/model	String	模型名称	"iiot_data_backup_query"	是
params/service	String	服务名称	"searchHistoryAlarm"	是
params/args/dbInstanceName	String	数据库实例名称	"TDengine"	是
params/args/dbName	String	数据库名称	"sie_alarm"	是
params/args/entityId	String	实体 ID	"03vy68yj17g93"	是
params/args/queryStartTime	String	查询起始时间，格式：yyyy-MM-dd	"2024-11-01 00:00:00"	是

		HH:mm:ss		
params/args/queryEndTime	String	查询结束时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-31 23:59:59"	是
params/args/properties	JSON	查询字段，"*"表示全部字段	["*"]	是
params/args/limit	Integer	查询行数，最大值：10000	1000	是
params/args/offset	Integer	跳过行数	0	是
params/args/order	String	排序	"time desc"	否

5.26.2. 请求参数

5.26.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	{ "result": { "data": [{ "alarm_name": "CPU 使用率报警", "alarm_state": null, "relieve_time": 1729829764783, "label": "", "entity_id": "03vy68yj17g93", "confirmation_time": 1729829764783, "link_custom_property_ids": "sie_CPULoad", "is_not": "1", "trigger_time": 1729829762690, "solution": "", "alarm_type": "0", "alarm_trigger_value": null, "attribute_value_define": "sie_CPULoad", "time": "2024-10-25 12:16:00.599", "id": "04awljza9awuo", "attribute_value": "CPU 使用率(%)=69.14", "state": "0", "alarm_id": "alarm_CPULoad", "alarm_reason": "CPU 使用率(%)>70%", "initial_trigger_time": 1729829760599 }] } }	

5.26.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
```

```
  "params": {
    "args": {
      "dbInstanceName": "TDengine",
      "dbName": "sie_alarm",
      "entityId": "03vy68yj17g93",
      "queryStartTime": "2024-11-01 00:00:00",
      "queryEndTime": "2024-11-31 23:59:59",
      "properties": ["*"],
      "limit": 1,
      "offset": 0,
      "order": ""
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_data_backup_query",
    "tag": "master",
    "service": "searchHistoryAlarm",
    "app": "iiot_data_backup"
  }
}
```

5.26.5. 返回示例

```
{
    "result": {
        "data": [
            {
                "alarm name": "CPU 使用率报警",
                "alarm state": null,
                "relieve time": 1729829764783,
                "label": "",
                "entity id": "03vy68yj17g93",
                "confirmation time": 1729829764783,
                "link custom property_ids": "sie_CPULoad",
                "is not": "1",
                "trigger time": 1729829762690,
                "solution": "",
                "alarm type": "0",
                "alarm trigger value": null,
                "attribute value define": "sie CPULoad",
                "time": "2024-11-15 12:16:00.599",
                "id": "04awljza9awuo",
                "attribute value": "CPU 使用率(%)=69.14",
                "state": "0",
                "alarm id": "alarm CPULoad",
                "alarm reason": "CPU 使用率(%)>70%",
                "initial_trigger_time": 1729829760599
            }
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.27. 备份数据库实例历史报警数据量查询

5.27.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_backup"	是
params/model	String	模型名称	"iiot_data_backup_query"	是
params/service	String	服务名称	"countHistoryAlarm"	是
params/args/dbInstanceName	String	数据库实例名称	"TDengine"	是
params/args/dbName	String	数据库名称	"sie_alarm"	是
params/args/entityId	String	实体 ID	"03vy68yj17g93"	是
params/args/queryStartTime	String	查询起始时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-01 00:00:00"	是
params/args/queryEndTime	String	查询结束时间，格式：yyyy-MM-dd HH:mm:ss	"2024-11-31 23:59:59"	是

5.27.2. 请求参数

5.27.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	{
			"result": {
			"data": 68
			}
			}	

5.27.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "dbInstanceName": "TDengine",
      "dbName": "sie_alarm",
      "entityId": "03vy68yj17g93",
      "queryStartTime": "2024-11-01 00:00:00",
      "queryEndTime": "2024-11-31 23:59:59"
    },
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_data_backup_query",
    "tag": "master",
    "service": "countHistoryAlarm",
    "app": "iiot_data_backup"
  }
}
```

5.27.5. 返回示例


```
{
    "result": {
        "data": 542,
        "context": {
            "@Class": "java.lang.Long",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.28. (工厂实体)查询历史工况

注意：
每次查询最多只能查询一个工厂实体的一个属性的历史数据；
使用平台通用查询语法，通过 filter 参数筛选数据，具体语法可参考官网了解

5.28.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.28.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/filter/entity_id	String	实体 id，具体格式看示例	"03bv8e4gl883k"	是
params/args/filter/custom_property_id	String	属性 id，具体格式看示例	"机床"	是
params/args/filter/basic_time	Date	开始时间，具体格式看示例，带>=符号的	"2023-11-07 09:54:45"	是
params/args/filter/basic_time	Date	结束时间，具体格式看示例，带<=符号的	"2023-11-07 09:54:45"	是
params/args/limit	Integer	查询行数，最大值: 10000	1000	是
params/args/offset	Integer	跳过行数	0	是
params/model	String	模型	"iiot_thing_entity_history_tagvalue"	是
params/service	String	服务	"searchByCondition"	是

5.28.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	{ "机床": [{ "cloud_time": "2025-08-20 14:00:34.648", "qs": "good", "entity_name": "湖北编织工厂", "name": "机床", "generation_time": "2025-08-20 14:00:34.648", "custom_property_id": "机床", "entity_id": "051nr7oe86vns", "current_value": 51786, "basic_time": "2025-08-20 14:00:34.648" }, { "cloud_time": "2025-08-20 14:00:14.554", "qs": "good", "entity_name": "湖北编织工厂", "name": "机床", "generation_time": "2025-08-20 14:00:14.554", "custom_property_id": "机床", "entity_id": "051nr7oe86vns", "current_value": 51746, "basic_time": "2025-08-20 14:00:14.554" }] }	cloud_time:入云时间； qs:质量戳； entity_name:设备名称； name:属性名称； generation_time:生产时间； custom_property_id:自定义属性 ID； current_value: 实时值； basic_time:基准时间

5.28.4. 请求示例

只需关注请求参数中提到的参数部分，其他参数固定无需修改

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "useDisplayForModel": true,
      "filter": [
        "&",
        "&",
        "&",
        [
          "custom_property_id",
          "=",
          "机床"
        ],
        [
          "entity_id",
          "=",
          "051nr7oe86vns"
        ],
        [
          "basic_time",
          ">=",
          "2025-08-20 00:00:00"
        ],
        [
          "basic_time",
          "<=",
          "2025-08-20 23:59:59"
        ]
```

```
      ],
      "limit": 31,
      "offset": 0,
      "order": "",
      "properties": [
        "basic time",
        "generation time",
        "cloud time",
        "name",
        "custom property_id",
        "current_value",
        "qs"
      ],
      "condition": "[]"
    },
    "context": {
      "uid": "",
      "lang": "zh_CN"
    },
    "model": "iiot_thing_entity_history_tagvalue",
    "tag": "master",
    "service": "searchByCondition",
    "app": "iiot_thing"
  }
}
```

5.28.5. 返回示例

```
{
    "result": {
        "data": {
            "机床": [
                {
                    "cloud time": "2025-08-20 14:37:46.989",
                    "qs": "good",
                    "entity name": "湖北编织工厂",
                    "name": "机床",
                    "generation time": "2025-08-20 14:37:46.989",
                    "custom property id": "机床",
                    "entity id": "051nr7oe86vns",
                    "current value": 56157,
                    "basic_time": "2025-08-20 14:37:46.989"
                },
                {
                    "cloud time": "2025-08-20 14:37:26.817",
                    "qs": "good",
                    "entity name": "湖北编织工厂",
                    "name": "机床",
                    "generation time": "2025-08-20 14:37:26.817",
                    "custom property id": "机床",
                    "entity id": "051nr7oe86vns",
                    "current value": 56117,
                    "basic_time": "2025-08-20 14:37:26.817"
                }
            ]
        },
        "context": {
            "@Class": "java.util.HashMap",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```

5.29. (工厂实体)查询实时工况

注意：
每次查询最多只能查询一个工厂实体的实时数据；

使用平台通用查询语法，通过 filter 参数筛选数据，具体语法可参考官网了解

5.29.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.29.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/filter/entity_id	String	实体 id，具体格式看示例	"03bv8e4gl883k"	是
params/args/filter/custom_property_id	String	属性 id，具体格式看示例	"%机床%" ,like 语法属性前后要增加% 不填查询所有属性	否
params/model	String	模型	"iiot_thing_property"	是
params/service	String	服务	"searchNodeData"	是

5.29.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	[
			{
			"name": "在线状态",
			"data_type": "String",
			"description": "在线状态",
			"custom_property_id": "sie_online",
			"property_source": "1",
			"id": "sie_online_factory",
			"basic_timeStr": "2025-08-20 13:53:21.538",
			"value": "在线"
			},
			{
			"name": "电功耗能",
			"data_type": "Integer",
			"description": null,
			"custom_property_id": "电功耗能",
			"property_source": "2",
			"id": "051o6h4hcqw7y",
			"basic_timeStr": "2025-08-20 14:58:54.257",
			"value": 5864800
			}
			]
				basic_timeStr:基准时间；
				data_type:数据类型；
				description:描述；
				name:属性名称；
				generation_time:生产时间；
				custom_property_id:自定义属性 ID；
				value: 实时值；
				id:属性 id；
				property_source:属性值来源；0：直接变量，1：逻辑变量；2：常量

5.29.4. 请求示例

只需关注请求参数中提到的参数部分，其他参数固定无需修改

注：custom_property_id 参数可选，不需要时整体删除这部分

```
[
        "custom_property_id",
        "like",
        "%机床%"
],
```

示例：

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "useDisplayForModel": true,
      "filter": [
        "&",
        [
          "custom_property_id",
          "like",
          "%机床%"
        ],
        [
          "entity_id",
          "=",
          "051nr7oe86vns"
        ]
      ],
      "properties": [
        "basic timeStr",
        "name",
        "custom property_id",
        "value",
        "data type",
        "property source",
        "description"
      ],
      "condition": "[]"
    },
    "context": {
      "uid": "",
      "lang": "zh_CN"
    },
    "model": "iiot thing_property",
    "tag": "master",
    "service": "searchNodeData",
    "app": "iiot_thing"
  }
}
```

5.29.5. 返回示例

```
{
    "result": {
        "data": [
            {
                "name": "在线状态",
                "data type": "String",
                "description": "在线状态",
                "custom property id": "sie_online",
                "property source": "1",
                "id": "sie online factory",
                "basic timeStr": "2025-08-20 13:53:21.538",
                "value": "在线"
            },
            {
                "name": "电功耗能",
                "data type": "Integer",
                "description": null,
                "custom property id": "电功耗能",
                "property source": "2",
                "id": "051o6h4hcqw7y",
                "basic timeStr": "2025-08-20 14:58:54.257",
                "value": 5864800
            }
        ],
        "context": {
```

```
            "@Class": "java.util.HashMap",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.30. 查询物实体属性列表

5.30.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.30.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/model	String	模型	"iiot_thing_property"	是
params/service	String	服务	"search"	是
params/args/filter	JSON	筛选字段，entity_id 表示物实体 ID，property_type 表示属性类型，取值关系：{“0”: “全部属性”, “1”: “模型属性”, “2”: “实体属性”}	[“&”,[“entity_id”,“=”,”05075tb09k9pi”],[“property_type”,”=”,”0”]]	是
params/args/properties	JSON	查询字段，"*"表示全部字段	["*"]	是
params/args/limit	Integer	查询行数，最大值: 10000	1000	是
params/args/offset	Integer	跳过行数	0	是
params/args/order	String	排序	"time desc"	否

5.30.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	[{ "id": "sie_online_device", "name": "在线状态", "custom_property_id": "sie_online", "data_type": "String", "description": "在线状态" },	id:属性 ID； name:属性名称; custom_property_id:自定义属性 ID data_type:数据类型；

			{
			"id": "sie_location_device",
			"name": "当前设备位置",
			"custom_property_id": "sie_location",
			"data_type": "String",
			"description": "当前设备位置"
			}
			]
				description:描述

5.30.4. 请求示例

示例：

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "useDisplayForModel": true,
      "filter": [
        "&",
        [
          "entity_id",
          "=",
          "0558mm0d8phwa"
        ],
        [
          "property_type",
          "=",
          "0"
        ]
      ],
      "properties": [
        "id",
        "name",
        "custom property_id",
        "data type",
        "description"
      ],
      "limit": 1000,
      "offset": 0,
      "order": "",
    },
    "context": {
      "uid": "",
      "lang": "zh_CN"
    },
    "model": "iiot thing_property",
    "tag": "master",
    "service": "search",
    "app": "iiot_thing"
  }
}
```

5.30.5. 返回示例

```
{
    "result": {
        "data": [
            {
                "id": "sie online device",
                "name": "在线状态",
                "custom property id": "sie_online",
                "data type": "String",
                "description": "在线状态"
            },
            {
                "id": "sie location device",
                "name": "当前设备位置",
                "custom property id": "sie_location",
                "data type": "String",
                "description": "当前设备位置"
            }
```

```
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": "03jetedjgruy6",
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.31. 查询数据编织数据

5.31.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.31.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	`iiot_data_fabric`	是
params/model	String	模型	`iiot_data_model`	是
params/service	String	服务	`searchData`	是
params/args/`identify`	String	编织模型	devicexxx	
params/args/type	String	模型类型，thingDevice:物联设备, businessData:业务数据, dataDictionary:数据字典	thingDevice	是
params/args/filter	JSON	筛选字段，entity_id 表示物实体 ID，可根据具体的字段过滤，编织建模的字段都可以放 filter 里进行过滤查询	[”&”,[”entity_id”,“=”,”05075tb09k9pi”],[“property_type”,”=”,”0”]]	否
params/args/properties	JSON	查询字段，"*"表示全部字段	["*"]	是
params/args/limit	Integer	查询行数	30	是
params/args/offset	Integer	跳过行数	0	是
params/args/order	String	排序	"time desc"	否

5.31.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	[
			{
			"a2": 168423,
			"a4": 1500.0,
			"time": "2025-10-15 16:21:19.820",
			"entity_id": "04gcda931k6r7",
			"tags": null
			},
			{
			"a2": 168421,
			"a4": 1498.0,
			"time": "2025-10-15 16:21:18.774",
			"entity_id": "04gcda931k6r7",
			"tags": null
			}
			]
				如果编织模型是时序存储，肯定会有 time 字段，如果是关系型数据库存在，则无 time 字段，字段根据建模字段展示

5.31.4. 请求示例

示例：

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "method": "service",
    "params": {
        "args": {
            "useDisplayForModel": true,
            "type": "thingDevice",
            "identify":"abc",
            "filter": [

            ],
            "limit": 2,
            "offset": 0,
            "order": "",
            "properties": [
                "*"
            ]
        },
        "context": {
            "uid": "",
            "timeZone": "UTC+8",
            "lang": "zh-CN"
        },
        "model": "iiot_data_model",
        "tag": "master",
        "service": "searchData",
        "app": "iiot_data_fabric"
    }
}
```

5.31.5. 返回示例


```
{
    "result": {
        "data": [
            {
                "a2": 168423,
                "a4": 1500.0,
                "time": "2025-10-15 16:21:19.820",
                "entity id": "04gcda931k6r7",
                "tags": null
            },
            {
                "a2": 168421,
                "a4": 1498.0,
                "time": "2025-10-15 16:21:18.774",
                "entity id": "04gcda931k6r7",
                "tags": null
            }
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": "03trs8al95hse",
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "error": null,
    "jsonrpc": "2.0"
}
```


5.32. 创建数据建模

5.32.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.32.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	`iiot_data_fabric`	是
params/model	String	模型	`iiot_data_model`	是
params/service	String	服务	`batchCreate`	是
params/args/`valuesList`	String	模型参数数组 参数含义： name: 模型名称,支持大小写英文字母、数字和下划线、中文 identify: 模型标识，支持大小写英文字母、数字和下划线	[{ "name": "批量创建 1", "identify": "batch1", "type": "thingDevice", "description": null, "dest_source": "TDengine", "time_series_store": true,	

		type: 模型类型，物联设备（thingDevice）、业务数据（businessData）、数据字典（dataDictionary） description: 模型描述, 可以为空 dest_source: 目标数据源 time_series_store: 是否时序库存储 use_all_entity: 应用全实体 enable: 是否启用 use_all_properties:应用全属性 thing_model_id:订阅的物模型 id thing_property_ids: 订阅的物模型属性 id 列表 thing_entity_ids: 订阅的实体 id 列表 thing_entity_property_ids: 订阅的物实体扩展属性 id 列表，没有可以填空数组，列表每个元素的格式：实体 id|属性 id subscribe_working_data: 是否订阅实时工况 store_delay_duration: 落库时延（s） store_delay: 是否启用时间窗口	"thing_model_id": "053xpdvud8q5e", "use_all_entity": true, "enable": true, "use_all_properties": true, "thing_property_ids": ["operatior", "product", "line", "barcode"], "thing_entity_property_ids": ["055tfsp51xplh|总耗时", "055tfsp51xplh|test_double"], "subscribe_working_data": true, "thing_entity_ids": ["055tfsp51xplh"], "store_delay_duration": 10, "store_delay": true }]	

5.32.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	{
			"result": {
			"data": [
			"05ccxfl7ajjrw"
			],
			"context": {
			"@Class":
				data 中返回的模型 id 列表

			"com.sie.snest.engine.data.RecordSet", "uid": "", "tenantId": null, "skin": null, "lang": "zh-CN", "curSg": null } }, "id": "guid", "error": null, "jsonrpc": "2.0" }	

5.32.4. 请求示例

示例：

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "method": "service",
    "params": {
        "args": {
            "valuesList": [
  {
    "name": "批量创建 1",
    "identify": "batch1",
    "type": "thingDevice",
    "description": null,
    "dest source": "TDengine",
    "time series store": true,
    "thing model id": "053xpdvud8q5e",
    "use all entity": true,
    "enable": true,
    "use all properties": true,
    "thing_property_ids":
    [
        "operatior",
        "product",
        "line",
        "barcode"
    ],
    "thing_entity_property_ids":
    [
        "055tfsp51xplh|总耗时",
        "055tfsp51xplh|test_double"
    ],
    "subscribe working data": true,
    "thing_entity_ids":
    [
        "055tfsp51xplh"
    ],
    "store delay duration": 10,
    "store_delay": true
  }
]
        },
        "context": {
            "uid": "",
            "timeZone": "UTC+8",
            "lang": "zh-CN"
        },
        "model": "iiot data_model",
        "tag": "master",
        "service": "searchData",
        "app": "iiot_data_fabric"
    }
}
```

5.32.5. 返回示例


```
{
    "result": {
        "data": [
           "05ccxfl7ajjrw"
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": "03trs8al95hse",
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "error": null,
    "jsonrpc": "2.0"
}
```


5.33. 新增数据建模字段

5.33.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.33.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	`iiot_data_fabric`	是
params/model	String	模型	`iiot_data_model`	是
params/service	String	服务	`batchCreate`	是
params/args/`valuesList`	String	模型参数数组 参数含义： name：字段名称,支持大小写英文字母、数字和下划线、中文 identify：字段标识，支持大小写英文字母、数字和下划线 type：字段类型，模型属性（ModelProperty） init_value：初始值，可以为空 string_length：字符串长度，模型属性是字符串时需要填写长度	[{ "name": "字段 1", "identify": "field1", "type": "ModelProperty", "init_value": null, "string_length": 50, "min_value": null, "max_value": null, "decimal": 2, "custom_property_id": "line", "fill_strategy": "None", "not_blank": false, "visible": true, "queryable": false,	

		min_value: 最小值，可以为空，数字类型需要填写
		max_value: 最大值，可以为空，数字类型需要填写
		decimal: 小数位数，数字类型需要填写
		custom_property_id:关联的属性 id
		fill_strategy:补数策略，无（None）、补 0（Zero）、补前值（PreviousValue）
		not_blank: 是否不能为空
		visible: 是否可见
		queryable: 是否可查询
		sequence： 排序号
		unique_field: 是否唯一字段
		property_entity_id: 属性所属实体 id，可以为空，扩展属性需要填写，模型属性为空
		property_data_type: 属性类型，查看物模型或物实体属性所对应的类型。String、Boolean、Integer、JSON、Double、DateTime、Date
		data_model_id：关联的数据建模 id
			"sequence": 1,
			"unique_field": false,
			"property_entity_id": null,
			"data_type_extend":
			{
			"property_data_type": "String"
			},
			"data_model_id": "05ccxfl7ajjrw"
			}
			]	

5.33.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	{
			"result": {
			"data": [
			"05ccyxm66ugrh"
			],
			"context": {
			"@Class": "com.sie.snest.engine.data.RecordSet",
			"uid": "",
			"tenantId": null,
			"skin": null,
				data 中返回的字段 id 列表

			"lang": "zh-CN", "curSg": null } }, "id": "guid", "error": null, "jsonrpc": "2.0" }	

5.33.4. 请求示例

示例：

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "method": "service",
    "params": {
        "args": {
            "valuesList": [
  {
  "name": "字段 1",
  "identify": "field1",
  "type": "ModelProperty",
  "init value": null,
  "string length": 50,
  "min value": null,
  "max value": null,
  "decimal": 2,
  "custom property id": "line",
  "fill strategy": "None",
  "not blank": false,
  "visible": true,
  "queryable": false,
  "sequence": 1,
  "unique field": false,
  "property entity id": null,
  "data_type_extend":
  {
    "property_data_type": "String"
  },
  "data_model_id": "05ccxfl7ajjrw"
},
{
  "name": "字段 2",
  "identify": "field2",
  "type": "ModelProperty",
  "init value": null,
  "string length": 50,
  "min value": null,
  "max value": null,
  "decimal": 2,
  "custom property id": "line",
  "fill strategy": "None",
  "not blank": false,
  "visible": true,
  "queryable": false,
  "sequence": 1,
  "unique field": false,
  "property entity id": null,
  "data_type_extend":
  {
    "property_data_type": "Double"
  },
  "data_model_id": "05ccxfl7ajjrw"
}
]
        },
        "context": {
            "uid": "",
            "timeZone": "UTC+8",
            "lang": "zh-CN"
        },
        "model": "iiot_data_model",
```

```
        "tag": "master",
        "service": "searchData",
        "app": "iiot_data_fabric"
    }
}
```

5.33.5. 返回示例

```
{
    "result": {
        "data": [
           "05ccyxm66ugrh",
           "05cd07yz593mf"
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": "03trs8al95hse",
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "error": null,
    "jsonrpc": "2.0"
}
```

5.34. 新增数据建模数据

5.34.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.34.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	`iiot_data_fabric`	是
params/model	String	模型	`iiot_thing_device`	是
params/service	String	服务	`create`	是

params/args/`valuesList`	String	模型参数数组
		参数含义：
		data_model_id：模型 id
		data_field_values：插入的 field1："test"，字段和值，key 是字段标识，value 是值
		time：时间戳
		model_id：所属物模型 id
		entity_id：所属物实体 id
			[
			{
			"data_model_id": "05ccxfl7ajjrw",
			"data_field_values":
			{
			"field1": "test",
			"time": "1762481283449",
			"model_id": "053xpdvud8q5e",
			"entity_id": "053xpj1xqwy00"
			}	

			}]	

5.34.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	{
			"result": {
			"data": [
			"05ccyxm66ugrh"
			],
			"context": {
			"@Class":
			"com.sie.snest.engine.data.RecordSet",
			"uid": "",
			"tenantId": null,
			"skin": null,
			"lang": "zh-CN",
			"curSg": null
			}
			},
			"id": "guid",
			"error": null,
			"jsonrpc": "2.0"
			}
				data 中返回的数据 id 列表

5.34.4. 请求示例

示例：

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "method": "service",
    "params": {
        "args": {
            "valuesList": [
  {
  "data model id": "05ccxfl7ajjrw",
  "data_field_values":
  {
    "field1": "test",
    "time": "1762481283449",
    "model id": "053xpdvud8q5e",
    "entity_id": "053xpj1xqwy00"
  }
}
]
        },
        "context": {
            "uid": "",
            "timeZone": "UTC+8",
            "lang": "zh-CN"
```

```
        },
        "model": "iiot_data_model",
        "tag": "master",
        "service": "searchData",
        "app": "iiot_data_fabric"
    }
}
```

5.34.5. 返回示例

```
{
    "result": {
        "data": [
           "1762481283449_053xpj1xqwy00"
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": "03trs8al95hse",
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "error": null,
    "jsonrpc": "2.0"
}
```

5.35. 创建物实体

5.35.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.35.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	iiot_thing	是
params/model	String	模型	iiot_thing_entity	是
params/service	String	服务	create	是
params/args/valuesList	JsonArray	待创建物实体信息列表	[{ "name": "设备名称", "entity_type": "1", "model_id": “sagjf3435sfa”, "mark": "物标识", "labels": "标签", "description": "描述",	是

			"networkmode": "directConn", "auth_type": "password", "conn_account": “123456”, "conn_password": “654321”, "project_id": “rwgs5hwdtfh1” }]	
params/args/valuesList/name	String	实体名称	"name":"物实体名称"	是
params/args/valuesList/entity_type	String	实体类型，取值：{"1":"设备","2":"网关","3":"SMDC","4":"工厂实体"}，默认值: 1	"entity_type":"1"	是
params/args/valuesList/model_id	String	物模型 ID	"model_id":"sagjf3435sfa"	是
params/args/valuesList/mark	String	物标识	"mark":"物标识"	是
params/args/valuesList/gateway	String	关联网关（即关联网关实体 ID），连接方式选择【通过网关连接】/【通过 SMDC 连接】使用	"gateway":"rwgs5hwdtfh2"	连接方式选择【直接连接】则否，选择【通过网关连接】/【通过 SMDC 连接】则是
params/args/valuesList/labels	String	标签，多个标签以英文逗号（,）分隔	"labels":"标签 1,标签 2"	否
params/args/valuesList/description	String	描述	"description":"物实体描述"	否
params/args/valuesList/networkmode	String	连接方式，取值:{"directConn":"直接连接","gatewayConn":"通过网关连接","smdcConn":"通过 SMDC 连接"}，默认值: directConn	"networkmode":"directConn"	否
params/args/valuesList/auth_type	String	认证类型，取值:{"password":"密码认证","cert":"证书认证"}，默	"auth_type":"password"	否

		认值: password，连接方式选择【直接连接】使用		
params/args/valuesList/conn_account	String	认证标识，连接方式选择【直接连接】且认证类型选择【密码认证】使用	"conn_account":"123456"	连接方式选择【直接连接】则是，否则否
params/args/valuesList/conn_password	String	认证密钥，连接方式选择【直接连接】且认证类型选择【密码认证】使用	"conn_password":"654321"	连接方式选择【直接连接】且认证类型选择【密码认证】则是，否则否
params/args/valuesList/cert_file_key	String	证书路径，连接方式选择【直接连接】且认证类型选择【证书认证】使用	"cert_file_key":"/2025-11-01/cert.zip"	连接方式选择【直接连接】且认证类型选择【证书认证】则是，否则否
params/args/valuesList/message_mark	String	通讯标识，连接方式选择【通过网关连接】/【通过SMDC 连接】使用	"message_mark":"thing_25_11_01_1024152"	否
params/args/valuesList/protocol_type	String	协议类型，，取值: {"MQTT":"MQTT 协议", "OPCUA":"OPCUA 协议","COAP":"COAP 协议", "HTTP":"HTTP 协议"}, 默认值: MQTT	"protocol_type":"MQTT"	否
params/args/valuesList/data_format	String	数据格式，取值: {"Json":"Json 格式", "Binary":"二进制"}，默认值: Json	"data_format":"Json"	否
params/args/valuesList/i	String	启用状态，取值: {"1":"启	"is_enable": "1"	否

s_enable		用","0":"停用"}，默认值: 1		
params/args/valuesList/project_id	String	所属项目 ID	"project_id": “rwgs5hwdtfh1”	是
params/args/valuesList/group_id	List<String>	分组，若不填写，默认分组【全部分组】下【未分组】	"group_id": [["entity_group_all", “zxcvbnmasdfg”]]	否
params/args/valuesList/register_type	String	注册类型，取值: {1:"自动注册", 2:"手动注册"}，默认值: 1	"register_type": 1	否

5.35.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	{ "result": { "data": ["poiuytrewqas"], "context": { "@Class": "com.sie.snest.engine.data.RecordSet", "uid": "", "tenant_id": "03trs8al95hse", "skin": null, "long": "zh_CN", "curSg": null }, }, "id": "guid", "error": null, "jsonrpc": "2.0" }	批量创建则 data 按序返回实体 ID

5.35.4. 请求示例

示例:

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "method": "service",
    "params": {
        "args": {
            "valuesList": [
```

```
                {
                    "name": "设备名称",
                    "entity type": "1",
                    "model id": "sagjf3435sfa",
                    "mark": "物标识",
                    "labels": "标签 1,标签 2",
                    "description": "描述",
                    "networkmode": "directConn",
                    "auth type": "password",
                    "conn account": "123456",
                    "conn password": "654321"
                    "protocol type": "MQTT",
                    "data format": "Json",
                    "project_id": "rwgs5hwdtfh1"
                }
            ]
        },
        "context": {
            "uid": "",
            "timeZone": "UTC+8",
            "lang": "zh-CN"
        },
        "model": "iiot thing_entity",
        "tag": "master",
        "service": "create",
        "app": "iiot_thing"
    }
}
```

5.35.5. 返回示例

```
{
    "result": {
        "data": [
            "poiuytrewqas"
        ],
        "context": {
            "@Class": "com.sie.snest.engine.data.RecordSet",
            "uid": "",
            "tenantId": "03trs8al95hse",
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "error": null,
    "jsonrpc": "2.0"
}
```

5.36. 查询物实体 MQTT 认证标识和密钥

5.36.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.36.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是

params/model	String	模型	"iiot_thing_entity"	是
params/service	String	服务	"search"	是
params/args/filter	JSON	筛选字段，id 表示物实体 ID	[[“id”,“=”,”05075tb09k9pi”]]	是
params/args/properties	JSON	查询字段，固定填写无需修改	["name","id","conn_account","conn_password""]	是
params/args/limit	Integer	查询行数，最大值: 10000	1	是
params/args/offset	Integer	跳过行数	0	是
params/args/order	String	排序	"id desc"	否

5.36.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	[
			{
			"name": "dsxw_test2",
			"conn_password": "Hlq6X7fF",
			"id": "055tfsp51xplh",
			"conn_account": "0l9tRmBR"
			}
			]
				id:实体 ID;
				name:实体名称;
				conn_account:认证标识;
				conn_password:认证密钥;

5.36.4. 请求示例

示例:

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "useDisplayForModel": true,
      "filter": [
        [
          "id",
          "=",
          "0558mm0d8phwa"
        ]
      ],
      "properties": [
        "id",
        "name",
        "conn account",
        "conn_password"
      ],
      "limit": 1,
      "offset": 0,
      "order": ""
    },
    "context": {
      "uid": "",
      "lang": "zh_CN"
    },
    "model": "iiot thing_entity",
    "tag": "master",
    "service": "search",
    "app": "iiot_thing"
  }
}
```

5.36.5. 返回示例


```
{
    "result": {
        "data": [
            {
                "id": "sie_online_device",
                "name": "在线状态",
                "conn_account": "Hlq6x7ff",
                "conn_password": "0l9trmTb"
            }
        ],
        "context": {
            "@Class": "java.util.ArrayList",
            "uid": "",
            "tenantId": "03jetedjgruy6",
            "skin": null,
            "lang": "zh_CN",
            "curSg": null
        }
    },
    "id": "guid",
    "jsonrpc": "2.0",
    "error": null
}
```


5.37. 数据编织-设置唯一标识

5.37.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.37.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_fabric"	是
params/model	String	模型	"iiot_data_model"	是
params/service	String	服务	"setUniqueIdentify"	是
params/args/modelId	String	数据建模的 id	"05075tb09k9pi"	是
params/args/ids	List<String>	需要设置成唯一标识的字段的 id，如果是时序库存储，至少要包含 time 列的 id，如果是时序库并且物联设备类型，则至少包含 time、entity_id 这两列的 id，这两列的 id 可以通过查询 iiot_data_field 拿到。数据建模创建模型后，unique_identify 列默认就是 time、entity_id 这两列	["05cdxmaox1sux","05cdxmamcnl23","05cnofculdlhd"]	是

		的 id，逗号分割		

5.37.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	{ "result": { "data": true, "context": { "@Class": "java.lang.Boolean", "uid": "", "tenantId": "03trs8al95hse", "skin": null, "lang": "zh-CN", "curSg": null } }, "id": "guid", "error": null, "jsonrpc": "2.0" }	data 的内容, true 代表成功, false 失败

5.37.4. 请求示例

示例：

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "method": "service",
    "params": {
        "args": {
            "modelId":"05cdxmaj5s8z2",
            "ids": ["05cdxmaox1sux","05cdxmamcnl23","05cnofculdlhd"],
            "useDisplayForModel": true
        },
        "context": {
            "uid": "",
            "timeZone": "UTC+8",
            "lang": "zh-CN"
        },
        "model": "iiot_data_model",
        "tag": "master",
        "service": "setUniqueIdentify",
        "app": "iiot_data_fabric"
    }
}
```

5.37.5. 返回示例

```
{
    "result": {
        "data": true,
        "context": {
            "@Class": "java.lang.Boolean",
            "uid": "",
            "tenantId": "03trs8al95hse",
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "error": null,
    "jsonrpc": "2.0"
```

```
}
```

5.38. 数据编织-建模启用

5.38.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

5.38.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_data_fabric"	是
params/model	String	模型	"iiot_data_model"	是
params/service	String	服务	"startUsing"	是
params/args/model_ids	List<String>	要启用的数据建模的 id	["05075tb09k9pi"]	是

5.38.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	{ "result": { "data": true, "context": { "@Class": "java.lang.Boolean", "uid": "", "tenantId": "03trs8al95hse", "skin": null, "lang": "zh-CN", "curSg": null } }, "id": "guid", "error": null, "jsonrpc": "2.0" }	data 的内容, true 代表成功, false 失败

5.38.4. 请求示例

示例：

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
```

```
      "model ids": [
        "05cdxmaj5s8z2"
      ],
      "useDisplayForModel": true
    },
    "context": {
      "uid": "",
      "timeZone": "UTC+8",
      "lang": "zh-CN"
    },
    "model": "iiot_data_model",
    "tag": "master",
    "service": "startUsing",
    "app": "iiot_data_fabric"
  }
}
```

5.38.5. 返回示例

```
{
    "result": {
        "data": true,
        "context": {
            "@Class": "java.lang.Boolean",
            "uid": "",
            "tenantId": "03trs8al95hse",
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
    "error": null,
    "jsonrpc": "2.0"
}
```

6. 边缘端对接

6.1. 上报实时工况

6.1.1. 配置准备

- 在 iiot 平台-接入与建模-物模型菜单分别创建网关模型，设备模型；
- 在 iiot 平台-接入与建模-物实体菜单分别创建网关实体、设备实体；
- 配置方法请参考"IIOT 用户操作手册"的第三章"接入与建模"。

6.1.2. 通讯协议

6.1.2.1. Mqtt

- 固定地址：单个 iiot 服务或单机时，使用固定地址连接。

Host and Port：mqtt://ip:port

Client ID：网关实体 id（通过网关连接时的网关），或设备实体 id（直接连接时）

Username：网关实体的认证标识（通过网关连接时的网关），或设备实体的认证标识（直

接连接时）
Password：网关实体的认证密钥（通过网关连接时的网关），或设备实体的认证秘钥（直接连接时）
Topic：网关实体 id（通过网关连接时的网关），或设备实体 id（直接连接时）

- 动态地址：分布式部署多个 iiot 服务时，使用动态地址连接。

Host and Port：通过接口获取（见 6.1.3 和 6.1.4）
Client ID：网关实体 id（通过网关连接时的网关），或设备实体 id（直接连接时）
Username：网关实体的认证标识（通过网关连接时的网关），或设备实体的认证标识（直接连接时）
Password：网关实体的认证密钥（通过网关连接时的网关），或设备实体的认证秘钥（直接连接时）
Topic：网关实体 id（通过网关连接时的网关），或设备实体 id（直接连接时）

6.1.3.获取指定设备的 Mqtt 连接地址

6.1.3.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

6.1.3.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/entityId	String	查询的设备实体	"sagjf3435sfa"	是
params/args/channel	String	查询的频道	固定为"mqtt"	是
params/model	String	模型	"iiot_thing_entity"	是
params/service	String	服务	"getIpAddressById"	是

6.1.3.3. 返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	"192.168.174.169:31882"	

6.1.3.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
```

```
  "params": {
    "args": {
        "entityId":"046bzfu5vnq5e",
        "channel":"mqtt"
    },
    "context": {
      "uid": "",
      "timeZone": "UTC+8",
      "lang": "zh-CN"
    },
    "model": "iiot_thing_entity",
    "tag": "master",
    "service": "getIpAddressById",
    "app": "iiot_thing"
  }
}
```

6.1.3.5. 返回示例

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "result": {
        "data": "192.168.174.169:31882",
        "context": {
            "@Class": "java.lang.String",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    }
}
```

6.1.4.批量获取 Mqtt 连接地址

6.1.4.1. 请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

6.1.4.2. 请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/entityId	String	查询的设备实体	["sagjf3435sfa"]	是
params/args/channel	String	查询的频道	固定为"mqtt"	是
params/model	String	模型	"iiot_thing_entity"	是
params/service	String	服务	"getIpAddressByIds"	是

6.1.4.3. 返回参数

名称	类型	描述	示例	备注

result/data	JSON	结果数据	"192.168.174.169:31882": ["046bzfu5vnq5e","046bze3gg3qlb"]	

6.1.4.4. 请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
        "entityIds":["046bzfu5vnq5e","046bze3gg3qlb"],
        "channel":"mqtt"
    },
    "context": {
      "uid": "",
      "timeZone": "UTC+8",
      "lang": "zh-CN"
    },
    "model": "iiot_thing_entity",
    "tag": "master",
    "service": "getIpAddressByIds",
    "app": "iiot_thing"
  }
}
```

6.1.4.5. 返回示例

```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "result": {
        "data": {
            "192.168.174.169:31882": [
                "046bzfu5vnq5e",
                "046bze3gg3qlb"
            ]
        },
        "context": {
            "@Class": "java.util.HashMap",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    }
}
```

6.1.5.上报工况消息格式

6.1.5.1. 报文参数

名称	类型	描述	示例	备注
seq	Long	消息序号	1	
ts	Long	消息时间戳	1700461108881	消息的当前时间戳
reporter	String	上报者 ID，网关或者直连设备 ID	02ushf98d21a8	网关实体 ID（通过网关连接）或设备实体 ID（直接连接）
type	String	上报类型，固定	DataReport	固定值
data/dataType	String	数据类型，固定	ReportData	固定值

data/models	Array	工况数据	[{
			"id": "02usqyax1sbuo",
			"ps": {
			"1700461108881": {
			"Uca": 405.4,
			"Ucb": -404.5,
			"qs": {
			"bad": [
			"Ucb"
			]
			}
			}
			}
			}]	

data/models 参数说明

名称	类型	描述	示例	备注
id	String	设备 id，工况所属的设备的 id	02usqyax1sbuo	需要先在 iot 平台创建对应的物实体

ps	Object	工况内容	{
			"1700461108881": {
			"Uca": 405.4,
			"Ucb": -404.5,
			"qs": {
			"bad": [
			"Ucb"
			]
			}
			}
			}
				key（1700461108881）是设备测点值生成的时间戳，value 是具体的工况内容，例如 Uca 则是测点 Uca 的值，需要在 iot 平台的物模型创建对应的属性 Uca（Uca 为自定义属性 ID）。qs 是质量戳，标识该属性值的质量，如果所有测点值的质量戳均为 good，则传 qs:{}

6.1.5.2. 报文示例

Mqtt Topic: 02ushf98d21a8

```
{
  "seq": 1,----------------------------消息序号
  "ts": 1700461108881,-----------------------------消息时间戳
  "reporter": "02ushf98d21a8",---上报者 ID（连接到平台的设备实体 ID，在平台创建生成）
  "type": "DataReport",--------------------------固定
  "data": {
     "dataType": "ReportData",-------------------------固定
     "models": [
     {
       "id": "02usqyax1sbuo",------实体 ID（在平台上创建物生成）
       "ps": {
        "1700461108881": {----------------------------测点值生成的时间戳
          "Uca": 405.4,---------------------------属性 ID：测点值
          "Ucb": -404.5,---------------------------属性 ID：测点值
          "qs": {"bad":["Ucb"]}---质量戳，如果所有的质量戳都是 good，则为{}
        }
       }
     }
   ]
  }
}
```

6.2. 回补历史工况

建议：
每批次回补测点值数量少于 50000 性能较佳；
每批次回补同类型数据的测点值性能较佳；

6.2.1.请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

6.2.2.请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_thing"	是
params/args/valuesList	JsonArray	回补的测点值集合。 entity_id:物实体 id property_id:属性 ID ts:时间戳 qs：质量戳，没特殊需求填写默认值 good value：上报的属性值	[{ "entity_id": "03q02whxzyf7z", "property_id": "sie_online", "ts": 1715349972669, "qs": "good", "value": 32.45 }]	是
params/model	String	模型	"iiot_thing_entity_history_tagvalue"	是
params/service	String	服务	"reportHistory"	是

6.2.3.返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	true	true：成功； false：失败

6.2.4.请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "args": {
      "valuesList": [
        {
          "entity id": "03q02whxzyf7z",
          "property_id": "03q02whxyhnl9",
```

```
        "ts": 1715349972669,
        "qs": "good",
        "value": 32.45
      }
    ]
  },
  "context": {
    "uid": "",
    "zoneOffset": 8,
    "lang": "zh-CN"
  },
  "model": "iiot_thing_entity_history_tagvalue",
  "tag": "master",
  "service": "reportHistory",
  "app": "iiot_thing"
 }
}
```


6.2.5.返回示例


```
{
    "id": "guid",
    "jsonrpc": "2.0",
    "result": {
        "data":true,
        "context": {
            "uid": "",
            "lang": "zh CN",
            "token": "cae26ebfc0a54a39b2e0da35633d19b6cd1",
            "tenant": "root"
        }
    }
}
```


6.3. 指令下发对接

6.3.1.配置准备

- 在 iiot 平台-接入与建模-物模型菜单给物模型配置指令；
- 配置方法请参考"IIOT 用户操作手册"的第三章""。

6.3.2.通讯协议

通讯协议为 Mqtt,
请求 Topic：SIIOT/ ORDER /{物标识}
响应 Topic：SIIOT/ORDER/ACK

6.3.3.获取指定设备的 Mqtt 连接地址

参考 6.1.3 和 6.1.4

6.3.4.指令下发消息格式

6.3.4.1. 报文参数

名称	类型	描述	示例	备注
requestId	String	指令 ID	"1729143744868"	
ts	Long	消息时间戳	1700461108881	消息的当前时间戳
reporter	String	消息下发者 ID	"iiot"	固定值
type	String	消息类型	"OrderIssued"	固定值
data/dataType	String	请求 or 响应	"Request"	

data/params	Json	下发参数	{
			"id":"034klgrw9wlxc",
			"client_id":"034klgrw9wlxc",
			"ordertype":"standard",
			"sn":"thing_23_08_25_1004627",
			"order":"sie_NTP"
			}	

data/params 参数说明

名称	类型	描述	示例	备注
id	String	设备 id	"02usqyax1sbuo"	需要先在 iot 平台创建对应的物实体
client_id	String	网关或者直连设备 ID	"048bl6uuxofs4"	网关实体 ID（通过网关连接）或设备实体 ID（直接连接）
ordertype	String	指令类型	"standard"	standard：标准指令 propertyupdate：属性下发 cmd：自定义指令
sn	String	物标识	"SMDC0001"	
order	String	指令 ID	"sie_NTP"	下发标准指令时必填，可选值:

ps	Json	属性值	{
			"temperature":"34"
			}
				属性下发时必填

request_method	String	请求方法	"POST"	下发自定义指令时必填
parameter	Object[]	请求参数		下发自定义指令时必填

6.3.4.2. 报文示例

```
Mqtt Topic: SIIOT/ ORDER /{物标识}
```

```
// 标准指令
{
  "requestId": "abc",       // 注意，这里是字符串
  "ts":16297888953494,      // 注意，这里是 long 数值类型
  "reporter": "iiot",
  "type": "OrderIssued",
  "data": {
    "dataType":"Request",
       "params":{
                  "client id":"034klgrw9wlxc",
              "id":"034klgrw9wlxc",
              "order":"sie_NTP",
              "ordertype":"standard",
```

```
            "sn":"SMDC0001"
        }
  }
}

// 属性点写值
{
  "requestId": "abc",       // 注意，这里是字符串
  "ts":16297888953494,// 注意，这里是 long 数值类型
  "reporter":"iiot",
  "type":"OrderIssued",
  "data":{
     "dataType":"Request",
     "params": {
            "client id":"034klgrw9wlxc",
            "id":"034klgrw9wlxc",
            "ordertype":"propertyupdate",
            "ps":{
               "temperature":"34"
            },
            "sn":"SMDC0001"
        }
  }
}

// 自定义指令
{
  "requestId": "abc",       // 注意，这里是字符串
  "ts":16297888953494,// 注意，这里是 long 数值类型
  "reporter":"iiot",
  "type":"OrderIssued",
  "data":{
     "dataType":"dataService/webapi/IO/ReadIOTag",// 指令路径，如 api 路径
     "params":[
        {
            "client id":"034klgrw9wlxc",
            "id":"034klgrw9wlxc",
            "ordertype":"cmd",
            "sn":"SMDC0001",
              "request method":"POST",
              "parameter": [["temperature","34"]]
        }
     ]
  }
}
```

6.4. 指令执行与响应

接收到指令消息后，解析消息并执行相应操作，执行完成后，上报响应消息，响应消息格式为：

Mqtt Topic: SIIOT/ORDER/ACK

```
{
  "reporter":"SMDC0001",
  "ts":1693278400285,
  "requestId":"abc",    // 注意，这里是字符串，与下发的 requestId 相同
  "type":"OrderAck",
  "data":{
     "dataType":"ack",
     "params":{
        "code":"201",
        "msg":"执行成功",
        "error":"",
          "response":"",

       },
     "extData":null
    }
}
```

返回值 code 定义：

```
201   成功
400   失败
```

6.5. 固件升级下发

6.5.1.配置准备

- 需要提前将固件升级文件上传到服务器并拿到文件下载地址；

6.5.2.通讯协议

通讯协议为 Mqtt,
请求 Topic：SIIOT/ Edge/{物标识}
响应 Topic：SIIOT/Edge/ACK

6.5.3.获取指定设备的 Mqtt 连接地址

参考 6.1.3 和 6.1.4

6.5.4.指令下发消息格式

6.5.4.1. 报文参数

名称	类型	描述	示例	备注
requestId	String	指令 ID	"1729143744868"	
ts	Long	消息时间戳	1700461108881	消息的当前时间戳
reporter	String	消息下发者 ID	"iiot"	固定值
type	String	消息类型	"OrderIssued"	固定值
data/dataType	String	请求 or 响应	"Request"	

data/params	Json	下发参数	{
			"url":"文件下载地址",
			"entity_id":"实体 Id",
			"version":"版本号",
			"sign":"md5 签名",
			"firmware_type":"固件类型",
			"disk_size":"磁盘建议大小（M）
			"
			}	

data/params 参数说明

名称	类型	描述	示例	备注
url	String	文件下载地址	""	
entity_id	String	实体 ID	"048bl6uuxofs4"	网关实体 ID（通过网关连接）或设备实体 ID（直接连接）

version	String	版本号	"11.0.337_x64"	
sign	String	md5 签名	"a3924ca373acea578e6f7b319497e83a"	
firmware_type	String	固件类型：整包 or 补丁包；wholepackage or servicepack	"wholepackage"	
disk_size	String	磁盘建议大小（M)	25	

6.5.4.2. 报文示例

Mqtt Topic: SIIOT/ EDGE /{物标识}

```
// 固件升级
{
    "ts":1688017517002,
    "reporter":"iiot",
    "requestId":"0310yymqvnxfk",
    "type":"FirmwareUpgrade",
    "data":{
        "dataType":"IOTService/AppUpgrade",
        "params":{
            "url":"文件下载地址",
            "entity id":"实体 Id",
            "version":"版本号",
            "sign":"md5 签名",
            "firmware type":"固件类型",
            "disk_size":"磁盘建议大小（M） "
        }
    }
}
```

```
// 固件升级应答

{
    "id":"9efbd5056db04bb48344befb9990c94f",
    "ts":1693278400285,
    "reporter":"03471cfcy1hc0",
    "requestId":"034z0iemtodmo",
    "type":"EdgeFirmwareAck",
    "data":{
        "dataType":"ack",
        "params":{
            "code":"201",
            "msg":"固件升级成功",
            "error":"",
            "process":100
        },
        "extData":null
    }
}
```

返回值 Code 定义：

100 接收到消息通知
200 进行中
201 成功
400 失败

6.6.边缘日志对接

6.6.1.配置准备

- 在 iiot 平台-边缘节点-边缘管理选择需要配置边缘日志的网关；
- 在运行时日志 tab 页配置日志开关、级别等；
- 配置方法请参考"IIOT 用户操作手册"的第三章""。

6.6.2.通讯协议

通讯协议为 Mqtt，请求 Topic：SIIOT/ ORDER /{物标识}

6.6.3.获取指定设备的 Mqtt 连接地址

参考 6.1.3 和 6.1.4

6.6.4.开启边缘日志消息格式

6.6.4.1. 报文参数

名称	类型	描述	示例	备注
requestId	String	指令 ID	"1729143744868"	
ts	Long	消息时间戳	1729143744868	消息的当前时间戳
reporter	String	消息下发者 ID	"iiot"	固定值
type	String	消息类型	"OrderIssued"	固定值
data/dataType	String	请求 or 响应	"Request"	
data/params	Json	下发参数	{ "parameter": ["Warn", "Error"], "id": "03ucg25ock3of", "sn": "config_smdc", "client_id": "03ucg25ock3of", "ordertype": "edgeLog", "order": "close" }	parameter：日志级别 id：设备 ID sn：设备的物标识 client_id：网关 ID ordertype：固定为 edgeLog order：open（开启）、close（关闭）

data/params 参数说明

名称	类型	描述	示例	备注
parameter	String	设备 id	["Warn", "Error"	日志级别，支持 Trace、Debug、Info、Fatal、Error、

			]	Warn
id	String	设备 id	"03ucg25ock3of"	设备 ID
sn	String	设备的物标识	"config_smdc"	设备的物标识
client_id	String	网关 ID	"03ucg25ock3of"	网关实体 ID（通过网关连接）或设备实体 ID（直接连接）
ordertype	String	指令类型	"edgeLog"	固定为 edgeLog
order	String	指令 ID	"close"	open（开启）、close（关闭）

6.6.4.2. 报文示例

Mqtt Topic: SIIOT/ ORDER /{物标识}

```
{
  "data": {
    "dataType": "Request",
    "params": {
      "parameter": [
        "Warn",
        "Error"
      ],
      "id": "03ucg25ock3of",
      "sn": "config_smdc",
      "client_id": "03ucg25ock3of",
      "ordertype": "edgeLog",
      "order": "open"
    }
  },
  "reporter": "iiot",
  "requestId": "1729143718801",
  "ts": 1729143718801,
  "type": "OrderIssued"
}
```

6.7. 日志开启与响应

接收到边缘日志消息后，解析消息并执行相应操作，执行完成后，上报响应消息，响应消息格式为：

Mqtt Topic: SIIOT/ORDER/ACK

```
{
  "reporter":"config_smdc",
  "ts":1693278400285,
  "requestId":"1729143718801",    // 注意，这里是字符串，与下发的 requestId 相同
  "type":"OrderAck",
  "data":{
      "dataType":"ack",
      "params":{
          "code":"201",
          "msg":"执行成功",
          "error":"",
            "response":"",

        },
      "extData":null
    }
}
```

返回值 code 定义：

201 成功
400 失败

6.8.查询日志上报地址

6.8.1.请求语法

POST /api/root/rpc/service/ HTTP/2.0
Header：Authorization:Bearer token 字符串

6.8.2.请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_base"	是
params/model	String	模型	"iiot_log_loki"	是
params/service	String	服务	"getLokiUrl"	是

6.8.3.返回参数

名称	类型	描述	示例	备注
result/data	JSON	结果数据	"http://192.168.174.169:32245"	

6.8.4.请求示例

```
{
  "id": "guid",
  "jsonrpc": "2.0",
  "method": "service",
  "params": {
    "context": {
      "uid": "",
      "zoneOffset": 8,
      "lang": "zh-CN"
    },
    "model": "iiot_log_loki",
    "tag": "master",
    "service": "getLokiUrl",
    "app": "iiot_base"
  }
}
```

6.8.5.返回示例

```
{
    "result": {
        "data": "http://192.168.174.169:32245",
        "context": {
            "@Class": "java.lang.String",
            "uid": "",
            "tenantId": null,
            "skin": null,
            "lang": "zh-CN",
            "curSg": null
        }
    },
    "id": "guid",
```

```
    "error": null,
    "jsonrpc": "2.0"
}
```

6.9.边缘日志上报

6.9.1.请求语法

POST [loki 地址]/loki/api/v1/push HTTP/2.0

注意：loki 地址从 6.7 获取。

6.9.2.请求参数

名称	类型	描述	示例	必填
params/app	String	APP 名称	"iiot_base"	是
params/model	String	模型	"iiot_log_loki"	是
params/service	String	服务	"getLokiUrl"	是

6.9.3.返回参数

无返回值，HTTP 响应码定义：

204 No Content：成功

400 Bad Request：请求中有语法错误或无法完成请求。

404 Not Found：请求的资源不存在。

500 Internal Server Error：服务器遇到了意外的情况，无法完成请求。

6.9.4.请求示例

```
{
  "streams": [
    {
      "stream": {
        "entity_id": "047a931coaqeo",
        "system": "SMDCRun",
        "level": "Info"
      },
      "values": [
          [ "1693278400285000000", "mqtt 客户端已连接" ]
      ]
    }
  ]
}
```

6.9.5.返回示例

204 No Content