# 监控管理

说明：

下面查询接口分为实时查询接口和历史查询接口,历史查询返回数组,实时查询返回最新的一条数据(当前时间减去一分钟---数据延时)

## 查询接口

### 1.1单个物理机cpu使用率监控

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/cpu | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 此物理机一段时间内cpu使用率的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| cpu使用率 | usage | []object | | cpu使用率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集cpu的值 | |
| 举例 | {  "usage": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.5  },  {  "timestamp": 1449061299,  "value": 0.5  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 1.2单个物理机cpu使用率的月监控

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/cpu\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 此物理机一个月cpu使用率的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| cpu使用率 | usage | []object | | cpu使用率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "usage": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.5  },  {  "timestamp": 1449061299,  "value": 0.5  }  ]  }  说明:将一月的数据进行压缩,1个小时数据压成一个点. | | | | |
| 是否同步 | true | | | | | |

### 2.1单个物理机内存使用率监控

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/mem | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 此物理机一段时间内内存使用率的监控数据 | | | | | |
| 输入参数 | 字段名 | 参数名称 | | 参数类型 | | 备注 |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {"startTime": 1449061299,"endTime": 1449061310} | | | | |
| 200返回参数 | 字段名 | 参数名称 | 参数类型 | | 备注 | |
| 内存使用率 | usage | []object | | 内存使用率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {"usage": [{"timestamp": 1449061299,"value": 0.4},{"timestamp": 1449061299,"value": 0.5},{"timestamp": 1449061299,"value": 0.5}]} | | | | |
| 是否同步 | true | | | | | |

### 2.2单个物理机内存使用率月监控

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/mem\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 此物理机一个月内存使用率的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| id | node\_id | | String | | 物理机注册id |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 内存使用率 | usage | []object | | 内存使用率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "usage": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.5  },  {  "timestamp": 1449061299,  "value": 0.5  }  ]  }  说明:将一月的数据进行压缩,1个小时数据压成一个点 | | | | |
| 是否同步 | true | | | | | |

### 3.1物理机网卡流量监控

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/network | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 此物理机一段时间内网卡流量的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 设备名 | <dev name> | Object | | 设备名,最多三个设备（bond0,bond1.bond2） | |
| 网络流入量 | network\_input | []object | | 网络流入量 | |
| 网络流出量 | network\_output | []object | | 网络流出量 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "<dev name>":【 {  "network\_intput": [  {  "timestamp": 1449061299,  "value": 0.4  }，  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "network\_output": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  },  "<dev name >": {  "network\_input": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "network\_output": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  },  "< dev name >": {  "network\_input": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "network\_output": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  }  } | | | | |
| 是否同步 | true | | | | | |

### 3.2物理机网卡流量月监控

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/network\_month | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 此物理机一个月内网卡流量的监控数据（今天到上一个月的今天） | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 起始时间 | startTime | int64 | | 查询开始时间 |
| 截止时间 | endTime | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 设备名 | <dev name> | Object | 设备名,最多三个(bond0,bond1,bond2) | |
| 网络流入量 | network\_input | []object | 网络流入量 | |
| 网络流出量 | network\_output | []object | 网络流出量 | |
| 时间戳 | timestamp | int64 | 采集时间戳 | |
| 值 | value | float64 | 采集的值 | |
| 举例 | {  "<dev name>": {  "network\_intput": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "network\_output": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  },  "<dev name >": {  "network\_input": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "network\_output": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  },  "< dev name >": {  "network\_input": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "network\_output": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  }  }  说明:将一月的数据进行压缩,1个小时数据压成一个点 | | | |
| 是否同步 | true | | | | |

### 4.1文件Home目录使用率监控

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/homedir | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 此物理机一段时间内home目录使用率的监控数据 | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 起始时间 | startTime | int64 | | 查询开始时间 |
| 结束时间 | endTime | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 使用率 | application\_amount | []object | home目录使用率 | |
| 总量 | total | []object | home目录总量 | |
| 时间戳 | timestamp | int64 | 采集时间戳 | |
| 值 | value | float64 | 采集的值 | |
| 举例 | {  "application\_amount": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "total": [  {  "timestamp": 1449061299,  "value": 4  } ]  } | | | |
| 是否同步 | true | | | | |

### 4.2文件Home目录使用率月监控

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/homedir\_month | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 此物理机一个月内home目录使用率的监控数据（今天到上一个月的今天） | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 起始时间 | startTime | int64 | | 查询开始时间 |
| 截止时间 | endTime | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 使用率 | application\_amount | []object | home目录使用率 | |
| 总量 | total | []object | home目录总量 | |
| 时间戳 | timestamp | int64 | 采集时间戳 | |
| 值 | value | float64 | 采集的值 | |
| 举例 | {  " application\_amount ": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "total": [  {  "timestamp": 1449061299,  "value": 0.4  }  ]  }  说明:将一月的数据进行压缩,1个小时数据压成一个点 | | | |
| 是否同步 | true | | | | |

### 5.1文件Root目录使用率监控

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/rootdir | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 此物理机一段时间内root目录使用率的监控数据 | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 起始时间 | startTime | int64 | | 查询开始时间 |
| 截止时间 | endTime | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 使用率 | application\_amount | []object | root目录使用率 | |
| 总量 | total | []object | root目录总量 | |
| 时间戳 | timestamp | int64 | 采集时间戳 | |
| 值 | value | float64 | 采集的值 | |
| 举例 | {  " application\_amount ": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "total": [  {  "timestamp": 1449061299,  "value": 0.4  }  ]  } | | | |
| 是否同步 | true | | | | |

### 5.2文件Root目录使用率月监控

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/<node\_id>/rootdir\_month | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 此物理机一个月内root目录使用率的监控数据（今天到上一个月的今天） | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 起始时间 | startTime | int64 | | 查询开始时间 |
| 截止时间 | endTime | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 使用率 | application\_amount | []object | root目录使用率 | |
| 总量 | total | []object | root目录总量 | |
| 时间戳 | timestamp | int64 | 采集时间戳 | |
| 值 | value | float64&int | 采集的值 | |
| 举例 | {  " application\_amount ": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "total": [  {  "timestamp": 1449061299,  "value": 4  }  ]  }  说明:将一月数据进行压缩,1个小时数据压成一个点 | | | |
| 是否同步 | true | | | | |

### 6.1挂载容器目录使用率监控(upsql,swm,upproxy)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/dir | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 此物理机一段时间内挂载容器目录使用率的监控数据(upsql,swm,upproxy) | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 起始时间 | startTime | int64 | | 查询开始时间 |
| 截止时间 | endTime | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| data目录使用率 | datadir\_application\_amount | []object | 挂载容器data目录使用率 | |
| data目录总量 | datadir\_total | []object | 挂载容器data目录总量 | |
| log目录使用率 | logdir\_application\_amount | []object | 挂载容器log目录使用率 | |
| log目录总量 | logdir\_total | []object | 挂载容器log目录总量 | |
| 时间戳 | timestamp | int64 | 采集时间戳 | |
| 值 | value | float64 | 采集的值 | |
| 举例 | {  " datadir\_application\_amount ": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "logdir\_application\_amount ": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  “logdir\_total”:[  {  "timestamp": 1449061299,  "value": 0.4  }  ]  } | | | |
| 是否同步 | true | | | | |

### 6.2挂载容器目录使用率月监控(upsql,swm,upproxy)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/dir\_month | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 此物理机一个月内挂载容器目录使用率的监控数据（今天到上一个月的今天） | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 起始时间 | startTime | int64 | | 查询开始时间 |
| 截止时间 | endTime | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| data目录使用率 | datadir\_ application\_amount | []object | 挂载容器data目录使用率 | |
| data目录总量 | Datadir\_total | []object | 挂载容器data目录总量 | |
| log目录使用率 | logdir\_ application\_amount | []object | 挂载容器log目录使用率 | |
| log目录总量 | logdir\_total | []object | 挂载容器log目录总量 | |
| 时间戳 | timestamp | int64 | 采集时间戳 | |
| 值 | value | float64 | 采集的值 | |
| 举例 | {  "datadir\_ application\_amount ": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  “datadir\_total”:[  {  "timestamp": 1449061299,  "value": 0.4  }  ],  "logdir\_ application\_amount ": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ],  “logdir\_total”:[  {  "timestamp": 1449061299,  "value": 0.4  }  ]  }  说明:将一月的数据进行压缩,1个小时数据压缩成一个点 | | | |
| 是否同步 | true | | | | |

### 7.1挂载容器cpu使用率监控(upsql,swm,upproxy)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/cpu | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 挂载容器一段时间内cpu使用率的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询起始时间 |
| 截止时间 | endTime | | int64 | | 查询结束时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| cpu使用率 | usage | []object | | 所在容器cpu使用率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "usage": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 7.2挂载容器cpu使用率月监控(upsql,swm,upproxy)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/cpu\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 挂载容器一个月内cpu使用率的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询起始时间 |
| 截止时间 | endTime | | int64 | | 查询结束时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| cpu使用率 | usage | []object | | 所在容器cpu使用率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "usage": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  }  说明:将一月数据进行压缩,1个小时数据压成一个点 | | | | |
| 是否同步 | true | | | | | |

### 8.1挂载容器内存使用率监控(upsql,swm,upproxy)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/mem | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 挂载容器一段时间内内存使用率的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询起始时间 |
| 截止时间 | endTime | | int64 | | 查询结束时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 内存使用率 | usage | []object | | 所在容器mem使用率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "usage": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 8.2挂载容器内存使用率月监控

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/mem\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 挂载容器一个月内存使用率的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询起始时间 |
| 截止时间 | endTime | | int64 | | 查询结束时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 内存使用率 | usage | []object | | 所在容器mem使用率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "usage": [  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  },  {  "timestamp": 1449061299,  "value": 0.4  }  ]  }  说明:将一月数据进行压缩,1个小时数据压成一个点 | | | | |
| 是否同步 | true | | | | | |

### 9.1服务状态监控(upsql,swm,upproxy)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/status | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内服务状态的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | | **备注** |
| 起始时间 | startTime | int64 | | | 查询起始时间 |
| 截止时间 | endTime | int64 | | | 查询结束时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200返回参数 | **字段名** | **参数名称** | | **参数类型** | **备注** | |
| 服务状态 | service\_status | | []object | 服务状态 | |
| 时间戳 | timestamp | | int64 | 采集时间戳 | |
| 值 | value | | string | 采集的值 | |
| 举例 | {  "service\_status": [  {  "timestamp": 1449061299,  "value": "normal"  },  {  "timestamp": 1449061299,  "value": "normal"  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 10.1线程连接监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/thread | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内线程连接的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | | **备注** |
| 起始时间 | startTime | int64 | | | 查询开始时间 |
| 截止时间 | endTime | int64 | | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | | **参数类型** | **备注** | |
| 最大连接线程数 | connection\_total | | []object | 最大连接线程数 | |
| 当前已连接线程数 | connection\_attempts | | []object | 当前已连接线程数 | |
| 正在执行线程数 | exec\_thread | | []object | 正在执行线程数 | |
| 时间戳 | timestamp | | int64 | 采集时间戳 | |
| 值 | value | | int | 采集的值 | |
| 举例 | {  "connection\_total": [  {  "timestamp": 1449061299,  "value": 60  },  {  "timestamp": 1449061299,  "value": 60  }  ],  "connection\_attempts": [  {  "timestamp": 1449061299,  "value": 60  },  {  "timestamp": 1449061299,  "value": 60  }  ],  "exec\_thread": [  {  "timestamp": 1449061299,  "value": 40  },  {  "timestamp": 1449061299,  "value": 40  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 10.2线程连接月监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/thread\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一个月内线程连接的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | | **备注** |
| 起始时间 | startTime | int64 | | | 查询开始时间 |
| 截止时间 | endTime | int64 | | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | | **参数类型** | **备注** | |
| 最大连接线程数 | connection\_total | | []object | 最大连接线程数 | |
| 当前已连接线程数 | connection\_attempts | | []object | 当前已连接线程数 | |
| 正在执行线程数 | exec\_thread | | []object | 正在执行线程数 | |
| 时间戳 | timestamp | | int64 | 采集时间戳 | |
| 值 | value | | float64 | 采集的值 | |
| 举例 | {  "connection\_total": [  {  "timestamp": 1449061299,  "value": 60  },  {  "timestamp": 1449061299,  "value": 60  }  ],  "connection\_attempts": [  {  "timestamp": 1449061299,  "value": 60  },  {  "timestamp": 1449061299,  "value": 60  }  ],  "exec\_thread": [  {  "timestamp": 1449061299,  "value": 40  },  {  "timestamp": 1449061299,  "value": 40  }  ]  }  说明:将一月的数据进行压缩,1个小时数据压成一个点 | | | | |
| 是否同步 | true | | | | | |

### 11.1 thread cache 使用率监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/cache | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内thread cache使用率的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | | **备注** |
| 起始时间 | startTime | int64 | | | 查询开始时间 |
| 截止时间 | endTime | int64 | | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | | **参数类型** | **备注** | |
| thread cache使用率 | usage | | []object | thread cache使用率 | |
| 时间戳 | timestamp | | int64 | 采集时间戳 | |
| 值 | value | | float64 | 采集的值 | |
| 举例 | {  "usage": [  {  "timestamp": 1449061299,  "value": 0.6  },  {  "timestamp": 1449061299,  "value": 0.6  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 11.2 thread cache 使用率月监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/cache\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一个月内thread cache的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | | | **备注** |
| 起始时间 | startTime | int64 | | | 查询开始时间 |
| 截止时间 | endTime | int64 | | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | | **参数类型** | **备注** | |
| thread cache使用率 | usage | | []object | thread cache使用率 | |
| 时间戳 | timestamp | | int64 | 采集时间戳 | |
| 值 | value | | float64 | 采集的值 | |
| 举例 | {  "usage": [  {  "timestamp": 1449061299,  "value": 0.6  },  {  "timestamp": 1449061299,  "value": 0.6  }  ]  }  说明：将一月的数据进行压缩，1个小时的数据压成一个点 | | | | |
| 是否同步 | true | | | | | |

### 12.1文件大小监控(upsql,upproxy)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/file\_size | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内upproxy错误文件大小(upsql有错误文件大小和慢日志文件大小)的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 错误文件大小 | error\_file\_size | []object | | 错误文件大小 | |
| 慢日志文件大小 | slow\_query\_file\_size | []object | | 慢日志文件大小 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | int | | 采集的值 | |
| 举例 | Upproxy举例如下  {  "error\_file\_size": [  {  "timestamp": 1449061299,  "value": 20  },  {  "timestamp": 14490612399,  "value": 21  }  ]  }  Upsql举例如下:  {  "error\_file\_size": [  {  "timestamp": 1449061299,  "value": 20  },  {  "timestamp": 14490612399,  "value": 21  }  ],  “slow\_query\_file\_size”:[  {  "timestamp": 1449061299,  "value": 20  },  {  "timestamp": 14490612399,  "value": 21  }  ]  }  说明:前端url中unit\_id类型是upsql的话返回值就是错误文件大小和慢日志文件大小,类型是upproxy的话返回值只有错误文件大小 | | | | |
| 是否同步 | true | | | | | |

### 13.1表文件大小监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/table\_size | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内表文件大小的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 表 | table | []object | | 表对象 | |
| 表名称 | name | string | | 表文件名称 | |
| 表大小 | value | int | | 采集的值 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 所属数据库 | database | string | | 表所属数据库 | |
| 举例 | {  "table": [  {  "name": "table1",  “timestamp“: 1449061299,  "value": 100,  "database": "test"  },  {  "name": "table2",  “timestamp”: 1449061299,  "value": 80,  "database": "test"  },  {  "name": "table3",  “timestamp”: 1449061299,  "value": 90,  "database": "test"  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 14.1锁等待信息监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/wait | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内锁等待的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 锁等待信息 | lock\_wait\_info | []Object | | 锁等待具体信息 | |
| 等待锁 | wait\_lock | object | | 等待锁 | |
| 获得锁 | acquire\_lock | object | | 获得锁 | |
| Id号 | Id | string | | Id号 | |
| 用户 | user | string | | 用户 | |
| ip地址 | host | string | | ip地址 | |
| Db | db | string | | Db | |
| 命令 | command | string | | 命令 | |
| 时间 | time | int | | 时间 | |
| 状态 | status | string | | 状态 | |
| 信息 | info | string | | 信息 | |
| 举例 | {  "lock\_wait\_info": [  {  "wait\_lock": {  “id“:”1234”,  "user": "test1",  "host": " [192.168.212.1@1234](mailto:192.168.212.1@1234)",  “db”:”test”,  “command”:”query”,  "time": 100,  “status”:””,  "info": "delete from test where id=2"  },  "acquire\_lock": {  “id”:”2345”,  "user": "test2",  "host": "192.168.2.212@4567",  “db”:”test”,  “command”:”delete”,  "time": 100,  “status”:””,  "info": "update test SET name='aaa' where id=2"  }  },  {  "wait\_lock": {  “id”:”5678”,  "user": "test2",  "host": "192.168.212.2@5745",  “db”:”test”,  “command”:”update”,  "time": 100,  “status”:””,  "info": "update test SET name='aaa' where id=3"  },  "acquire\_lock": {  “id”:”45609”  "user": "test1",  "host": "192.168.212.1@6899",  “db”:”test”,  “command”:”insert”,  "time": 100,  “status”:””,  "info": "delete from test where id=3"  }  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 15.1负载监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/load | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内负载的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 结束时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 实例 | unit | object | | 实例 | |
| 插入 | insert | []object | | 每秒插入笔数 | |
| 查询 | select | []object | | 每秒查询笔数 | |
| 更新 | update | []object | | 每秒更新笔数 | |
| 删除 | delete | []object | | 每秒删除笔数 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | int | | 采集的值 | |
| 举例 | {  "unit": {  "insert": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ],  "select": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ],  "update": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ],  "delete": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ]  }  } | | | | |
| 是否同步 | true | | | | | |

### 15.2负载月监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/load\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一个月内负载的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 结束时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 实例 | unit | object | | 实例 | |
| 插入 | insert | []object | | 每秒插入笔数 | |
| 查询 | select | []object | | 每秒查询笔数 | |
| 更新 | update | []object | | 每秒更新笔数 | |
| 删除 | delete | []object | | 每秒删除笔数 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | int | | 采集的值 | |
| 举例 | {  "unit": {  "insert": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ],  "select": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ],  "update": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ],  "delete": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ]  }  }  说明:将一月数据进行压缩,1个小时数据压成一个点 | | | | |
| 是否同步 | true | | | | | |

### 16.1缓冲池大小监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/buffer\_size | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内缓冲池大小的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 结束时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 缓冲池大小 | buffer\_pool\_size | []object | | 缓冲池大小 | |
| 缓冲池剩余大小 | Buffer\_pool\_free\_size | []object | | 缓冲池剩余大小 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | int | | 采集的值 | |
| 举例 | {  "buffer\_pool\_size": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ],  "buffer\_pool\_free\_size": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 16.2缓冲池大小月监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/buffer\_size\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一个月内缓冲池大小的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 结束时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 缓冲池大小 | buffer\_pool\_size | []object | | 缓冲池大小 | |
| 缓冲池剩余大小 | buffer\_pool\_free\_size | []object | | 缓冲池剩余大小 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | int | | 采集的值 | |
| 举例 | {  "buffer\_pool\_size": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ],  "buffer\_pool\_free\_size": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ]  }  说明:将一月数据进行压缩,1个小时数据压缩成一个点 | | | | |
| 是否同步 | true | | | | | |

### 17.1缓冲池脏页数监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/buffer\_ dirty\_page | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内缓冲池脏页数的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 结束时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 缓冲池脏页数量 | buffer\_pool\_dirty\_page | []object | | 缓冲池脏页数 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "buffer\_pool\_dirty\_page": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 17.2缓冲池脏页数月监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/buffer\_ dirty\_page\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一个月内缓冲池脏页数的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | Date | | 查询开始时间 |
| 结束时间 | endTime | | Date | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 缓冲池脏页数量 | buffer\_pool\_dirty\_page | []object | | 缓冲池脏页数 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "buffer\_pool\_dirty\_page": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ]  }  说明:将一月数据进行压缩,1个小时数据压缩成一个点 | | | | |
| 是否同步 | true | | | | | |

### 18.1缓冲池命中率监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/hit | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内缓冲池命中率的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 缓冲池命中率 | buffer\_pool\_hit | []object | | 缓冲池命中率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "buffer\_pool\_hit": [  {  "timestamp": 1449061299,  "value": 0.6  },  {  "timestamp": 1449061299,  "value": 0.6  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 18.2缓冲池命中率月监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/hit\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一个月内缓冲池命中率的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 缓冲池命中率 | buffer\_pool\_hit | []object | | 缓冲池命中率 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | float64 | | 采集的值 | |
| 举例 | {  "buffer\_pool\_hit": [  {  "timestamp": 1449061299,  "value": 0.6  },  {  "timestamp": 1449061299,  "value": 0.6  }  ]  }  说明:将一月数据进行压缩,1个小时数据压成一个点 | | | | |
| 是否同步 | true | | | | | |

### 19.1打开表数监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/opentable | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一段时间内打开表数的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询结束时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 打开表数 | open\_table | []object | | 打开表的数量 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | int | | 采集的值 | |
| 举例 | {  "open\_table": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ]  } | | | | |
| 是否同步 | true | | | | | |

### 19.2打开表数月监控(upsql)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/opentable\_month | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 一个月内打开表数的监控数据（今天到上一个月的今天） | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询结束时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 打开表数 | open\_table | []object | | 打开表的数量 | |
| 时间戳 | timestamp | int64 | | 采集时间戳 | |
| 值 | value | int | | 采集的值 | |
| 举例 | {  "open\_table": [  {  "timestamp": 1449061299,  "value": 100  },  {  "timestamp": 1449061299,  "value": 100  }  ]  }  说明:将一月数据进行压缩，1个小时数据压成一个点 | | | | |
| 是否同步 | true | | | | | |

### 20.1主从复制监控(upsql)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/replication | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 一段时间内主从复制的监控数据 | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 起始时间 | startTime | int64 | 查询开始时间 | |
| 截止时间 | endTime | int64 | 查询结束时间 | |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 被连接的主服务器主机的ip | replication\_master\_host | string | | 被连接的主服务器主机的ip |
| 主服务器主机的端口号 | replication\_master\_port | String | | 主服务器主机的端口号 |
| 主服务器服务号 | replication\_master\_server\_id | string | | 的id号 |
| 运行状态 | replication\_running.status | string | | 复制状态 |
| 主机读写模式 | replication\_read\_write\_mode | string | | 读写模式 |
| 角色 | replication\_role | string | | 角色 |
| 从机Io是否被连接在主机上 | replication\_slave\_io\_running | string | | slave\_io的状态 |
| 从机SQL线程是否被启动 | replication\_slave\_sql\_running | string | | slave\_sql的状态 |
| 主从延迟时间 | replication\_seconds\_behind\_master | string | | 延迟时间 |
| 备库当前binlog文件名 | replication\_slave\_binlog\_filename | string | | 备库当前binlog文件名 |
| 备库当前binlog位置 | replication\_slave\_binlog\_pos | string | | 备库当前binlog位置 |
| 主库当前binlog文件名 | replication\_master\_binlog\_filename | string | | 主库当前binlog文件名 |
| 主库当前binlog文件位置 | replication\_master\_binlog\_pos | string | | 主库当前binlog文件位置 |
| 举例 | {  "replication\_master\_host": "192.168.212.1",  "replication\_master\_port": “3306”,  "replication\_master\_server\_id":”1”,  "replication\_running.status": "nomarl",  "replication\_read\_write\_mode": "readonly",  "replication\_slave\_io\_running": "Yes",  "replication\_slave\_sql\_running": "Yes",  "replication\_seconds\_behind\_master": “100”,  "replication\_slave\_binlog\_filename": "master-bin.000002",  "replication\_slave\_binlog\_pos": “1307”,  "replication\_master\_binlog\_filename": "master-bin.000001",  "replication\_master\_binlog\_pos": “1307”  } | | | |
| 是否同步 | true | | | | |

### 21.1进程和守护进程状态(upsql)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/process | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 一段时间内进程和守护进程的监控数据 | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 起始时间 | startTime | int64 | 查询开始时间 | |
| 截止时间 | endTime | int64 | 查询结束时间 | |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 进程状态 | upsqld\_process | []object | | 进程状态 |
| 时间戳 | timestamp | int64 | | 采集时间戳 |
| 值 | value | passing | | 采集值 |
| 举例 | {  " upsqld\_process ": [  {  "timestamp": 1449061299,  "value": “passsing”  },  {  "timestamp": 1449061299,  "value": “psaaing”  }  ]  } | | | |
| 是否同步 | true | | | | |

### 22.1复制sql线程状态(upsql)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/slave\_sql | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 一段时间内复制sql状态的监控数据 | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 起始时间 | startTime | int64 | 查询开始时间 | |
| 截止时间 | endTime | int64 | 查询截止时间 | |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| **upsqld核心进程** | **slave\_sql** | **[]object** | | **复制sql线程** |
| 时间戳 | timestamp | int64 | | 采集时间戳 |
| 值 | value | string | | 采集的值 |
| 举例 | **{**  **"slave\_sql": [**  **{**  **"timestamp": 1449061299,**  **"value": "warning"**  **},**  **{**  **"timestamp": 1449061299,**  **"value": "warning"**  **}**  **]**  **}** | | | |
| 是否同步 | true | | | | |

### 23.1复制io线程状态(upsql)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/unit/<unit\_id>/slave\_io | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 一段时间内复制io状态的监控数据 | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 起始时间 | startTime | int64 | 查询开始时间 | |
| 截止时间 | endTime | int64 | 查询截止时间 | |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| **upsqld核心进程** | **Slave\_io** | **[]object** | | **复制io线程** |
| 时间戳 | timestamp | int64 | | 采集时间戳 |
| 值 | value | string | | 采集的值 |
| 举例 | **{**  **"slave\_io": [**  **{**  **"timestamp": 1449061299,**  **"value": "warning"**  **},**  **{**  **"timestamp": 1449061299,**  **"value": "warning"**  **}**  **]**  **}** | | | |
| 是否同步 | true | | | | |

### 24.1平台组件状态(mg,registory,polymerizer)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口url | GET /v1/monitor/component/<Name> | | | |
| 调用逻辑 | 前端应用→后端监控 | | | |
| 功能描述 | 查询平台组件状态 | | | |
| url参数 | **参数名称** | **备注** | | |
| all | 查询平台所有组件的状态 | | |
| mg | 查询mg组件的状态 | | |
| registory | 查询registory组件的状态 | | |
| polymerizer | 查询polymerizer组件的状态 | | |
| 举例 | 1. <http://127.0.0.1:8080>/v1/monitor/component/all 2. <http://127.0.0.1:8080>/v1/monitor/component/mg 3. <http://127.0.0.1:8080>/v1/monitor/component/registory 4. <http://127.0.0.1:8080>/v1/monitor/component/polymerizer | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** |
| **mg组件状态** | **mg** | **[]object** | **总控组件状态** |
| **registory组件状态** | **registory** | **[]object** | **软件仓库组件状态** |
| **polymerizer组件状态** | **polymerizer** | **[]object** | **聚合器组件状态** |
| 名称 | name | string | 组件名称 |
| 状态 | status | string | 组件状态 |
| ip地址 | ip | String | 组件ip地址 |
| 端口号 | port | int | 组件的端口号 |
| 举例 | **url参数为all**  **{**  **"mg": [**  **{**  **"name": “mg1”,**  **"status": "critical",**  **“ip“:”127.0.0.1”,**  **“port”:2301**  **},**  **{**  **"name": “mg2”,**  **"status": "passing",**  **“ip“:”127.0.0.2”,**  **“port”:2301**  **},**  **"registory": [**  **{**  **"name": “registory1”,**  **"status": "critical",**  **“ip“:”127.0.0.1”,**  **“port”:2302**  **},**  **{**  **"name": “registroy2”,**  **"status": "passing",**  **“ip“:”127.0.0.2”,**  **“port”:2302**  **},**  **"polymerizer": [**  **{**  **"name": “poly,erizer1”,**  **"status": "critical",**  **“ip“:”127.0.0.3”,**  **“port”:2303**  **}**  **]**  **}**  **url参数为mg&registory&polymerizer**  **{**  **{**  **"name": “mg1”,**  **"status": "critical",**  **“ip“:”127.0.0.1”,**  **“port”:2301**  **},**  **{**  **"name": “mg2”,**  **"status": "critical",**  **“ip“:”127.0.0.2”,**  **“port”:2301**  **}**  **}** | | |
| 是否同步 | true | | | |

### 25.1查询监控项类别状态

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口url | GET /v1/metric/metricclass | | | |
| 调用逻辑 | 前端应用→后端监控 | | | |
| 功能描述 | 查询有哪些监控项类别 | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** |
| **监控项类别** | **metricclass** | **string** | **监控项类别名称** |
| **监控频率** | **Metrictime** | **Int** | **监控项类别的采集频率** |
| **状态** | **status** | **string** | **监控项类别状态** |
| 举例 | **[**  **{**  **"metricclass": "hostcpu ",**  **“metrictime“:120,**  **"status": "on"**  **},**  **{**  **"metricclass": "hostmem ",**  **“metrictime”:60,**  **"status": "on"**  **}**  **]** | | |
| 是否同步 | true | | | |

### 26.1查询事件状态等信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口url | GET /v1/event/metric | | | |
| 调用逻辑 | 前端应用→后端监控 | | | |
| 功能描述 | 查询有哪些事件状态等信息 | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | **备注** |
| **监控项** | **metric** | **string** | **监控项名称** |
| **起始值** | **start** | **string** | **事件起始值** |
| **结束值** | **end** | **string** | **事件的结束值** |
| **级别** | **level** | **string** | **事件级别** |
| **状态** | **eventstatus** | **string** | **事件状态** |
| **状态** | **metricstatus** | **string** | **监控项状态** |
| **告警** | **isalarmon** | **string** | **是否告警** |
| 举例 | **[**  **{**  **"metric": "host.cpu\_usage ",**  **"start": "90",**  **“end“:”100”,**  **“level”:”critical”,**  **“eventstatus”:”on”,**  **“metricstatus”:”on”,**  **“isalarmon”:”on”**  **},**  **{**  **"metric": "host.cpu\_usage ",**  **"start": "79",**  **“end”:”89”,**  **“level”:”warning”,**  **“eventstatus”:”off”,**  **“metricstatus”:”on”,**  **“isalarmon”:”off”**  **}**  **]** | | |
| 是否同步 | true | | | |

### 27.1查询horus-server版本号

|  |  |  |
| --- | --- | --- |
| 接口url | GET /v1/version | |
| 调用逻辑 | 前端应用→后端监控 | |
| 功能描述 | 查询horus-server版本号 | |
|  | 举例 | **{**  **"version": "2.0.0"**  **}** |
| 是否同步 | true | |

### 28.1服务组件状态(node为单位)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/monitor/node/component | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 查询服务组件状态 | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 主机id | endpoint | []string | 主机id | |
| 举例 | {  "endpoint":[“123456”,”565767”]  } | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 举例 | [  {  “node\_id”:”123456”,  “component\_name”:”ConsulAgent”,  “component\_status”:”passing”  },  {  “node\_id”:”123456”,  “component\_name”:”Docker”,  “component\_status”:”passing”  },  {  “node\_id”:”123456”,  “component\_name”:”DockerPlugin”,  “component\_status”:”passing”  },  {  “node\_id”:” 565767”,  “component\_name”:”ConsulAgent”,  “component\_status”:”passing”  },  {  “node\_id”:” 565767”,  “component\_name”:”Docker”,  “component\_status”:”passing”  },  {  “node\_id”:” 565767”,  “component\_name”:”DockerPlugin”,  “component\_status”:”passing”  }  ] | | | |
| 是否同步 | true | | | | |

### 29.1事件查询

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/event /list | | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | | |
| 功能描述 | 此物理机一段时间内cpu使用率的监控数据 | | | | | |
| 输入参数 | **字段名** | **参数名称** | | **参数类型** | | **备注** |
| 起始时间 | startTime | | int64 | | 查询开始时间 |
| 截止时间 | endTime | | int64 | | 查询截止时间 |
| 举例 | {  "startTime": 1449061299,  "endTime": 1449061310  } | | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** | |
| 事件id | eventid | string | | 事件id | |
| 主机或实例id | endPoint | string | | 主机或实例id | |
| 产生事件的监控项 | eventMetric | string | | 产生事件的监控项 | |
| 事件创建时间 | createStamp | Int64 | | 事件创建时间 | |
| 事件等级 | level | string | | 事件等级 | |
| 事件值 | value | string | | 事件值 | |
| 举例 | [  {  " eventid ": 1449061299,  " endPoint ": er456y7u7u8i998,  “eventMetric”:”host.cpu\_usage”,  “createStamp”: 1449061299,  “level”:””critical,  “value “：”96“  },  {  " eventid ": 1449061299,  " endPoint ": er456y7u7u8i998,  “eventMetric”:”host.mem\_usage”,  “createStamp”: 1449061299,  “level”:””critical,  “value “：”98“  },  {  " eventid ": 1449061299,  " endPoint ": gitu56y7u7u8i998,  “eventMetric”:”upproxy,service\_status”,  “createStamp”: 1449061299,  “level”:””critical,  “value “：”critical“  }  ] | | | | |
| 是否同步 | True | | | | | |

## 配置接口

### 1.1监控项类别启动(所有endpoint的操作)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/metric /allstart | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 对某一个监控项类别的启动(所有endpoint的操作) | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 监控项类别 | metricclass | string | 监控项类别 | |
| 举例 | [  {  "metricclass":”hostcpu”  }  ]  说明:数组里面目前只支持一个metricclass,不能多个同时配置 | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 错误信息 | Err | string | | 错误信息 |
| 举例 | {  " Err ":””  }  说明:接收到启动请求,修改模板库,将修改结果返回给前端,然后多线程发送配置给下面的horus-agenr | | | |
| 是否同步 | true | | | | |

### 2.1监控项类别停止(所有endpoint操作)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/metric /allstop | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 对某一个监控项类别的停止(所有endpoint的操作) | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 监控项类别 | metricclass | string | 监控项类别 | |
| 举例 | [  {  "metricclass":”hostcpu”  }  ]  说明:数组里面目前只支持一个metricclass,不能多个同时配置 | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 错误信息 | Err | string | | 错误信息 |
| 举例 | {  " Err ":””  }  说明:接收到停止请求,修改模板库,将修改结果返回给前端,然后多线程发送配置给下面的horus-agenr | | | |
| 是否同步 | true | | | | |

### 3.1监控项类别修改(所有endpoint操作)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/metric /allupdate | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 对某一个监控项类别的更新(所有endpoint的操作) | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 监控项类别 | metricclass | string | 监控项类别 | |
| 间隔时间 | metrictime | int | 间隔时间 | |
| 举例 | [  {  "metricclass":”hostcpu”,  “metrictime”:120  }  ]  说明:数组里面目前只支持一个metricclass,不能多个同时配置 | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 错误信息 | Err | string | | 错误信息 |
| 举例 | {  " Err ":””  }  说明:接收到更新请求,修改模板库,将修改结果返回给前端,然后多线程发送配置给下面的horus-agenr | | | |
| 是否同步 | true | | | | |

### 4.1事件启动(所有endpoint操作)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/event /allstart | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 对某一个事件的启动(所有endpoint的操作) | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 事件 | metric | string | 事件 | |
| 级别 | level |  | 间隔时间 | |
| 举例 | [  {  "metric":”host.cpu\_usage”,  “level”:”critical”  }  ]  说明:数组里面目前只支持一个metric不能多个同时配置  level字段为空表示将所有的级别启动,不为空启动指定级别 | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 错误信息 | Err | string | | 错误信息 |
| 举例 | {  " Err ":””  }  说明:接收到启动请求,修改模板库,将修改结果返回给前端,然后多线程发送配置给下面的horus-agenr | | | |
| 是否同步 | True | | | | |

### 5.1事件停止(所有endpoint操作)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/event /allstop | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 对某一个事件的停止(所有endpoint的操作) | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 事件 | metric | string | 事件 | |
| 级别 | level |  | 间隔时间 | |
| 举例 | [  {  "metric":”host.cpu\_usage”,  “level”:”critical”  }  ]  说明:数组里面目前只支持一个metric不能多个同时配置  level字段为空表示将所有的级别停止,不为空启动指定级别 | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 错误信息 | Err | string | | 错误信息 |
| 举例 | {  " Err ":””  }  说明:接收到停止请求,修改模板库,将修改结果返回给前端,然后多线程发送配置给下面的horus-agenr | | | |
| 是否同步 | True | | | | |

### 6.1 事件更新(所有endpoint操作)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口url | POST /v1/event /allupdate | | | | |
| 调用逻辑 | 前端应用→后端监控 | | | | |
| 功能描述 | 对某一个事件的更新(所有endpoint的操作) | | | | |
| 输入参数 | **字段名** | **参数名称** | **参数类型** | **备注** | |
| 事件 | metric | string | 事件 | |
| 级别 | level | object | 间隔时间 | |
| 级别名称 | name | string | 级别名称 | |
| 级别值 | ranges | string | 级别值 | |
| 事件是否开启 | iseventon | bool | 事件是否开启 | |
| 告警是否开启 | isalarmon | bool | 告警是否开启 | |
| 举例 | [  {  "metric": "host.cpu\_usage",  "level": {  "name": "warning",  "ranges": "60-80"  },  "iseventon": true,  "isalarmon": true  }  ]说明:数组里面目前只支持一个metric不能多个同时配置 | | | |
| 200  返回参数 | **字段名** | **参数名称** | **参数类型** | | **备注** |
| 错误信息 | Err | string | | 错误信息 |
| 举例 | {  " Err ":””  }  说明:接收到更新请求,修改模板库,将修改结果返回给前端,然后多线程发送配置给下面的horus-agenr | | | |
| 是否同步 | True | | | | |