1. 插件目录结构（后端）

extensions/

├── base.go

├── base\_interface.go

├── humiture

│   ├── actions

│   │   └── humiture\_chart.go

│   ├── config.yaml

│   └── view

│   ├── HumBrokenlineChart.js

│   ├── humChart.js

│   ├── TempBrokenlineChart.js

│   ├── tempChart.js

│   └── temp\_chart.vue

└── pm25

├── actions

│   └── pm25.go

├── config.yaml

└── view

├── pm25BrokenlineChart.js

└── pm25CurrentChart.js

1. ThingsPanel-Go\extensions\humiture\config.yaml配置文件规则如下图所示
2. 映射到传感器的字段就是vue图表中的字段
3. 插件名保持一致
4. 模板名与ThingsPanel-Go\extensions\humiture\view下的js文件名保持一致，js为驼峰，配置文件里为中划线
5. 一个模板对应一个js图表文件，一个config.yaml文件对应一个插件
6. 其他如图示例



1. 拿到前端插件代码后样例后（推荐vscode\HBuilder X开发）
2. 安装环境node.js 16.13
3. 执行命令 npm install安装依赖
4. 开发后不运行，执行命令npm run build编译为js文件
5. 将js文件放入ThingsPanel-Go\extensions\插件名\view文件夹下
6. 修改js文件和配置文件不用重启系统
7. 后端接口返回数据格式：{

"data": "[{\"device\_id\":\"f1ab2c47-951f-10b8-60c0-c6b33440f189\",\"fields\":[{\"hum\":24,\"systime\":\"2022-01-18 18:59:11\",\"temp\":26},{\"hum\":24,\"systime\":\"2022-01-18 18:59:17\",\"temp\":26},{\"hum\":24,\"systime\":\"2022-01-18 18:59:41\",\"temp\":26}],\"latest\":{\"hum\":24,\"systime\":\"2022-01-18 18:59:41\",\"temp\":26}}]",

"endTs": 1642503540000,

"startTs": 1642503240000

}

7）后端接口第一次返回全量，后面返回增量数据