# 行为树的整体json格式

{

"id": "4afa4af8-3622-4560-8e01-7a74f89597b1",

"title": "A behavior tree",

"description": "",

"root": "719e1d40-7cc3-4fcd-88e4-47f7f1d1e761",

"properties": {},

"nodes": {

"719e1d40-7cc3-4fcd-88e4-47f7f1d1e761": {

"id": "719e1d40-7cc3-4fcd-88e4-47f7f1d1e761",

"name": "Sequence",

"title": "Sequence",

"description": "",

"properties": {},

"children": [

"50e0768e-7325-4516-83c6-e4fe6b60000e",

"13e58360-dc0b-40e1-a509-b508243d8ed7",

"86466bfc-150d-4f43-8e70-5cd8ee2e2095"

]

},

"13e58360-dc0b-40e1-a509-b508243d8ed7": {

"id": "13e58360-dc0b-40e1-a509-b508243d8ed7",

"name": "BTQueryNode",

"title": "BTQueryNode",

"description": "",

"properties": {

"cmd\_id": "test",

"result\_name": "res",

"result\_type": "str",

"device\_id": 0,

"cmd\_data": 0,

"device\_type": "light"

}

},

"86466bfc-150d-4f43-8e70-5cd8ee2e2095": {

"id": "86466bfc-150d-4f43-8e70-5cd8ee2e2095",

"name": "BTIsResultSatisfy",

"title": "BTIsResultSatisfy",

"description": "",

"properties": {

"operator": "eq",

"expect\_result": "test",

"result\_type": "str",

"result\_name": "res"

}

},

"50e0768e-7325-4516-83c6-e4fe6b60000e": {

"id": "50e0768e-7325-4516-83c6-e4fe6b60000e",

"name": "BTOperatorNode",

"title": "BTOperatorNode",

"description": "",

"properties": {

"device\_id": 0,

"cmd\_id": "test",

"cmd\_data": 0,

"device\_type": "light"

}

}

}

}

## Sequence

一个组件，不详细介绍了

{

"id": "719e1d40-7cc3-4fcd-88e4-47f7f1d1e761",

"name": "Sequence",

"title": "Sequence",

"description": "",

"properties": {},

"children": [

"50e0768e-7325-4516-83c6-e4fe6b60000e",

"13e58360-dc0b-40e1-a509-b508243d8ed7",

"86466bfc-150d-4f43-8e70-5cd8ee2e2095"

]

},

## BTQueryNode

查询节点：

cmd\_id表示命令

"result\_name": "res",//表示获取到的结果的名称，用于给后面的节点来获取这个结果。（比如，我把这个结果存在行为树的一个map里，后面的节点如果需要的话，就用这个名字去获取）。

"result\_type": "str",//表示结果的类型，是字符串，还有float

"device\_id": 0,//设备的id，用于指定操作那一个设备

"cmd\_data": 0,//这个命令所需的参数

"device\_type": "light"//这个设备的类型

具体格式：

{

"id": "13e58360-dc0b-40e1-a509-b508243d8ed7",

"name": "BTQueryNode",

"title": "BTQueryNode",

"description": "",

"properties": {

"cmd\_id": "test",

"result\_name": "res",

"result\_type": "str",

"device\_id": 0,

"cmd\_data": 0,

"device\_type": "light"

}

}

## BTIsResultSatisfy

一个条件节点：

"result\_name": "res",//表示获取到的结果的名称，用于给后面的节点来获取这个结果。（例如，这个时候我就可以在行为树的一个map中获取名为res的结果了）

"result\_type": "str",//表示结果的类型，是字符串，还有float

"device\_id": 0,//设备的id，用于指定操作那一个设备

"cmd\_data": 0,//这个命令所需的参数

"device\_type": "light"//这个设备的类型

具体结构如下：

{

"id": "86466bfc-150d-4f43-8e70-5cd8ee2e2095",

"name": "BTIsResultSatisfy",

"title": "BTIsResultSatisfy",

"description": "",

"properties": {

"operator": "eq",

"expect\_result": "test",

"result\_type": "str",

"result\_name": "res"

}

}

## BTOperatorNode

一个操作节点，用去去执行具体的操作。

{

"id": "50e0768e-7325-4516-83c6-e4fe6b60000e",

"name": "BTOperatorNode",

"title": "BTOperatorNode",

"description": "",

"properties": {

"device\_id": 0,

"cmd\_id": "test",

"cmd\_data": 0,

"device\_type": "light"

}

}

}