# IOTC概要设计文档

设备定义

设备是不确定的，可能有灯，也可能有空调等，甚至是现在没有出现的智能硬件设备，那么怎么对接这些设备就是一个需要设计的问题。

首先，我们将每个设备可以进行的操作抽象成一种服务，那么设备在加入到IOTC的网络中时，就需要将自身支持哪些服务上报给服务器，这样就可以将每个设备都抽象出来，考虑到有些设备还具有两种服务，如一个传感器，它可能带有好几种传感器，既可以采集温度，又可以采集湿度，包括光照强度等，那么设备的结构体在设计时就需要考虑可能支持多种服务。

那么这样抽象出来后发现，这种架构设计和zigbee的协议非常类似，因此我们可以借鉴zigbee的实现机制。

每个设备下有端点，端点可以有好几个，每个代表一种不同的类型，如温度传感器是一个端点，湿度是一个端点，端点下提供服务，温度可以提供温度值，湿度可以提供湿度值。

协议定义

客户端和服务端的数据通信采用json格式。

**统一的发送格式如下：**

{

"sequence\_no": 1,

"event\_type": 1,

"message\_type": 1,

"description": {…}

}

序列号，事件类型，命令字以及描述是必须加入的字段，用户的信息都放在描述字段中。

下面的数据都是在描述字段中的：

设备加入：

[

{

"device\_name": "light",

"device\_id": 1,

"device\_index": 12345678

},

{

"device\_name": "dimmerlight",

"device\_id": 2,

"device\_index": 87654321

}

]

其中device\_id是预定的，如果此设备是未知的设备那么需要用下面的字段：

{

"device\_name": "light",

"device\_id": 0,

"device\_index": 12345678,

"device\_services": [

{

"onoff": 0

},

{

"level": 255

},

{

"temperature": 50

}

]

}

设备ID必须为0。