

中华人民共和国通信行业标准

YD/T XXXXX—XXXX

物联网物模型总体技术要求

General technical requirement of IoT thing model

(报批稿)

XXXX - XX - XX 发布

XXXX - XX - XX 实施

中华人民共和国工业和信息化部 发布

目 次

前 言..... II

1 范围..... 1

2 规范性引用文件..... 1

3 术语与定义..... 1

4 缩略语..... 1

5 物模型总体架构技术要求..... 2

 5.1 总体架构..... 2

6 功能技术要求..... 4

 6.1 物的状态信息（Status）..... 4

 6.2 物的档案信息（Profile）..... 5

 6.3 属性（Property）..... 7

 6.4 行为（Action）..... 9

 6.5 事件（Event）..... 11

 6.6 输入参数和输出参数..... 12

 6.7 组件（Component）..... 13

 6.8 物模板（Thing template）..... 14

7 物模型数据模式..... 16

 7.1 数据模式..... 16

 7.2 数据格式..... 16

 7.3 列表数据格式..... 19

 7.4 非列表数据格式..... 20

附录 A..... 22

附录 B..... 38

附录 C..... 39

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本标准由中国通信标准化协会提出并归口。

本标准起草单位：中国电信集团有限公司、阿里巴巴（中国）有限公司、中国移动通信集团有限公司、华为技术有限公司、新华三技术有限公司、腾讯云计算（北京）有限责任公司、中兴通讯股份有限公司、亚信科技（中国）有限公司、中国信息通信研究院、中国联合网络通信集团有限公司、高通无线通信技术(中国)有限公司、中国信息通信科技集团有限公司、信大捷安信息技术股份有限公司、东土科技（宜昌）有限公司、新讯数字科技（杭州）有限公司。

本标准起草人：于小博、龙容、聂永丰、郑智、邵伟翔、于翔、李培、王永霞、李建慧、郭建超、周佳鑫、冀辰、莫若、邱利昌、刘大鹏、代博为、朱红儒、张育雄、高艳丽、余思聪、刘献伦、孙大勇、邓德政、黎陈、曹予飞、金程。

物联网物模型总体技术要求

1 范围

本文件规定了物联网物模型的总体架构、功能技术要求、物模型数据模式等。
本文件适用于物联网设备与平台、平台与平台之间进行物模型相关的交互与通信。

2 规范性引用文件

本文件没有规范性引用文件。

3 术语与定义

下列术语和定义适用于本文件。

3.1

物模型 thing model

对一个物体的数字化描述。包括三层结构，分别是元素（包括属性、行为以及事件）、组件以及物模板。

3.2

元素 element

主要包括属性（Property）、行为（又称服务或者命令Action）以及事件（Event）。

3.3

组件 component

基于物模型元素的一个表示物组成部分的抽象层。组件可以表示一个包含多个同类型或者多个不同类型元素的集合。

3.4

物模板 thing template

基于组件以及元素（Property, Action, Event）的一个表示物的抽象层。物模板可表示一个包含有组件、元素、以及自定义元素的集合。

4 缩略语

下列缩略语适用于本文件。

Id: 标识（Identifier）

desc: 描述（description）

HTTP: 超文本传输协议（HyperText Transfer Protocol）

LwM2M: 轻量级机器传输协议 (Lightweight Machine to Machine)
LoRa: 远距离无线电 (Long Range Radio)

5 物模型总体架构技术要求

5.1 总体架构

在物联网中，“物”指物理世界（物理装置）或信息世界（虚拟事务）中的对象，可以被标识并整合入通信网。物模型是一种对物理实体进行数字化语义描述的方法。物联网平台可以与不同种类智能设备进行交互，为用户提供智能服务。由于涉及多种设备，且设备来自不同的设备厂商。因此需要定义一套统一的智能设备物模型规范，消除业务中智能设备数据间的差异，使得数据在设备与平台、平台与平台间传输都是统一的，并提供设备间互联互通互懂的能力。物联网物模型架构如图1所示。

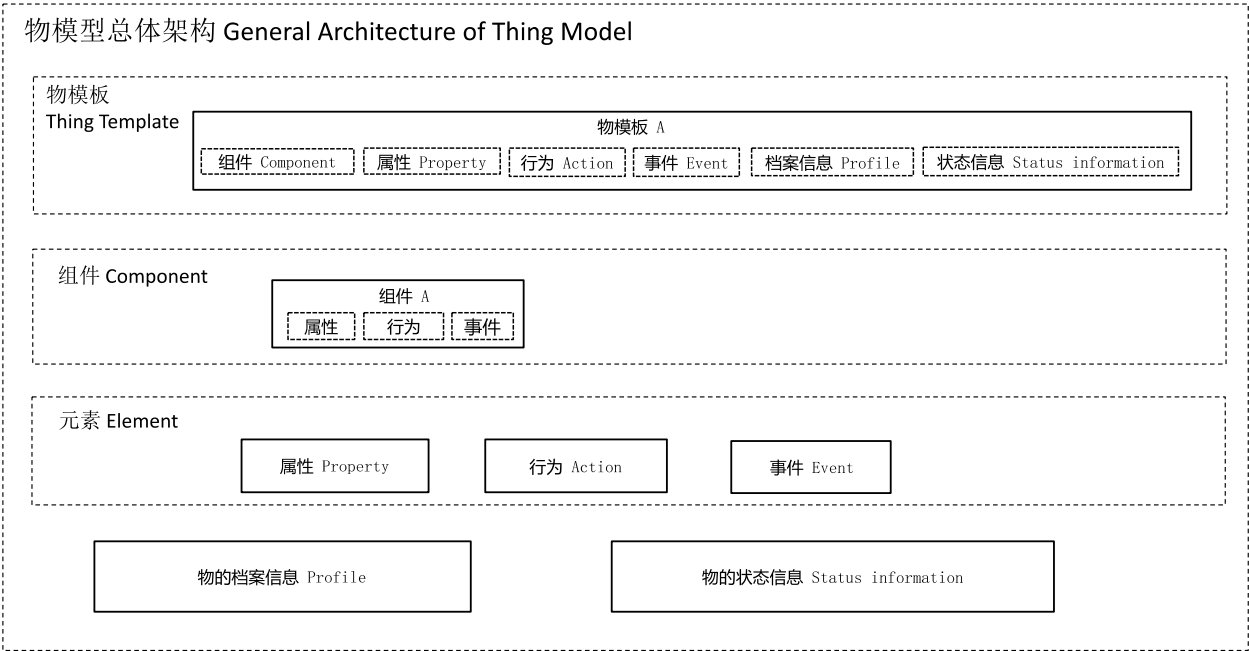


图 1 物模型总体架构

本文件定义了物联网智能设备在交互过程中的数据描述规范。图一给出了物模型总体架构，物模型架构中包含元素（Element）、组件（Component）以及物模板（Thing Template）。此外，物模型结构中还包括档案（Profile）以及状态信息（Status information）。档案以及状态信息可以包含在物模板中，也可以独立存在。物模板可以存在一个或者多个。

物模型三层结构从上到下为包含关系。物模板可以包含组件和/或元素，组件可以包含元素。三层结构中，元素包括属性（Property）、行为（Action）、事件（Event）。

组件是基于物模型元素的一个表示物的抽象层关系。物模板是基于组件以及元素的一个表示复杂物的抽象层关系。图2给出了物模板结构中各个部分的关系。物模板中可以包括状态、档案、属性、行为、事件以及组件中的一个或多个。上述要素（状态、档案、属性、行为、事件）在物模板中也可存在多个。

智能设备的物模型组成部分还可以归类为必选（Required）以及可选（Optional），方便行业用户根据不同的需求对智能设备做出相应的控制，也有助于设备之间的互通。

物模型各个部分的组成元素中给出了关键字命名，方便行业用户根据不同的需求进行元素的命名，有助于实现不同平台之间智能设备的互通。

物模型结构中各个部分（不包括状态）中还引入了自定义元素（Custom），方便行业用户基于规范中各个做定制化或者兼容一些特殊的物模型需求。

本文件中，大括号指开区间，即取值范围不包括边界值。中括号表示闭区间，即取值范围包括边界值。

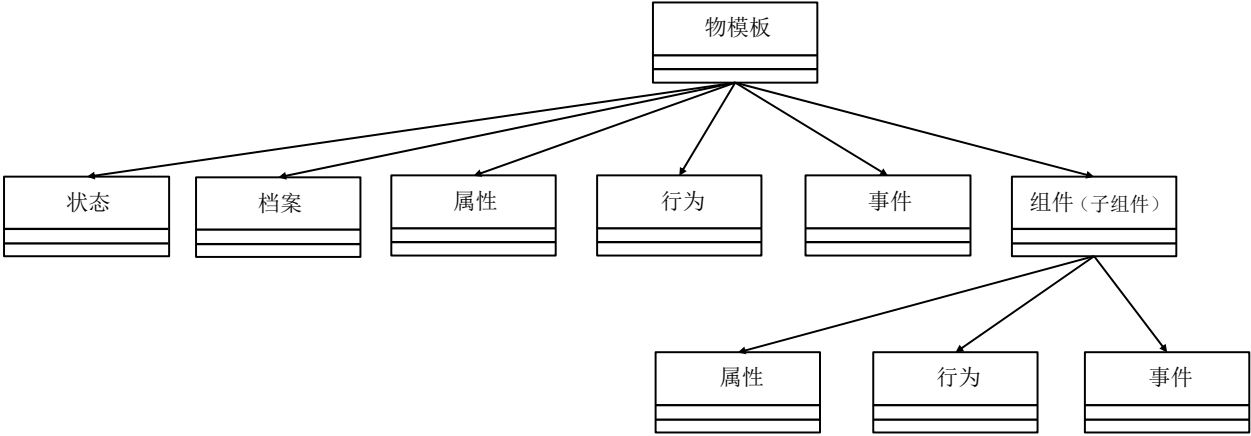


图 2 物模型各个组成部分的关系示意图

物模型描述要素与物联网设备的抽象关系如表1所示。

表 31 物模型架构定义

元素	名称	类型	是否必选	说明	备注
Status	状态信息	Object	否	设备在线/离线、激活/未激活等状态。	
Profile	档案信息	Object	否	设备身份详情的静态描述，包含设备身份标识和设备描述信息。	
Property	属性	Object	否	描述设备静态或动态数据。	
Action	行为	Object	否	代表设备可对外提供的能力，用于描述设备可执行的动作或任务。	
Event	事件	Object	否	设备运行时产生的各类事件信息的数据，包含告警、故障、信息等类型。	
Component	组件	Object	否	是基于元素的一个表示物的抽象层关系。组件可描述一个包含有多种元素（例如属性，行为，事件）的集合。	
Template	模板	Object	否	是基于组件以及物模型要素的一个表示复杂物的抽象层关系。物模型模板（物模板）可表示一个包含有	

				至少一个组件、以及至少一个基础要素（例如功能要素的其中一个或多个）的集合。	
--	--	--	--	---------------------------------------	--

6 功能技术要求

6.1 物的状态信息（Status）

6.1.1 定义

物的状态是指设备在线/离线、激活/未激活等状态。

6.1.2 组成元素

状态信息的组成元素如表2所示：

表 2 状态信息组成元素

名称	类型	是否必选	长度	取值	说明	关键字（可选）	备注
设备名称	String	是	{1, 50}	N/A	设备名称	deviceName, DeviceName	
设备标识	String	是	{1, 100}	N/A	设备标识	deviceId, deviceUID, DeviceUDID, DeviceId	
设备状态	String	是	{1, 50}	N/A	online: 上线。 offline: 离线。	status, Status	
发送通知的时间	String	否	{1, 100}	N/A	发送通知的时间	time, Time	
发送通知的UTC时间	String	否	{1, 100}	N/A	发送通知的UTC时间	utcTime, UtcTime	
状态变更前最后一次通信的时间	String	否	{1, 100}	N/A	状态变更前最后一次通信的时间	lastTime, LastTime	设备状态变更前最后一次通信的时间。 收到消息的时间不是实际设备上下线时间。 设备上下线时间，请按照lastTime判断。 例如，您依次收到3条消息： 上线：

							2018-08-31 10:02:28.195 下线： 2018-08-31 10:01:28.195 下线： 2018-08-31 10:03:28.195 这3条消息展示了，设备先下线，再上线，最后下线的过程。
状态变更前最后一次通信的UTC时间	String	否	{1,100}	N/A	状态变更前最后一次通信的UTC时间	utcLastTime, UtcLastTime	
设备公网出口IP	String	否	{1,100}	N/A	设备公网出口IP	clientIp ,ClientId	

6.2 物的档案信息（Profile）

6.2.1 定义

物的档案信息是指设备身份详情的静态描述，包含设备身份标识和设备描述信息。

6.2.2 组成元素

档案信息的组成元素如表3所示：

表3 档案信息组成元素

名称	类型	是否必选	长度	取值	说明	关键字（可选）	备注
设备描述	String	否	{1,100}	N/A	产品描述，支持中文、大小写字母、数字、短划线和下划线	description, Description	
设备标识	String	有条件必选	{1,100}	N/A	设备标识	deviceId, deviceUID, DeviceUDID, DeviceId	
设备品牌	String	否	{1,100}	N/A	设备品牌	brand, Brand	
设备型号	String	否	{1,100}	N/A	设备型号	model, Model, deviceModel, DeviceModel	设备型号是指生产厂家对生产的不

							同规格同一类型产品，用数字或字母分别标记，以便区分的一种代码，比如不同内存大小，cpu配置的摄像头产品可以用不同的型号进行标识，一般由英文字母和数字表示。
设备类型	Enum	否	否	Int8/UInt8 Int16/UInt16 Int32/UInt32 Int64/UInt64 Bool/ Enum/ String/ Double/ Array/ Struct/ Bitmap	0-直连设备；1-网关设备；2-网关子设备。	type, Type	
协议类型	Enum	否	否	Int8/UInt8 Int16/UInt16 Int32/UInt32 Int64/UInt64 Bool/ Enum/ String/ Double/ Array/ Struct/	0-消息队列遥测传输协议（Message Queuing Telemetry Transport, MQTT）；1-LwM2M；2-HTTP；3：其他	protocolType, ProtocolType	

				Bitmap			
网络类型	Enum	否	否	Int8/UInt8 Int16/UInt16 Int32/UInt32 Int64/UInt64 Bool/ Enum/ String/ Double/ Array/ Struct/ Bitmap	0-WIFI; 1-2G; 2-3G; 3-4G; 4-Cat1; 5-5G; 6-NB-IoT; 7-LoRa; 8-其他	netType, NetType	
设备MAC地址	String	否	{1, 50}	N/A	设备MAC地址	deviceMAC, DeviceMAC	
设备序列号	String	否	{1, 100}	N/A	设备序列号	device Serial number, Device Serial number	
物模型版本号	String	否	{1, 100}	N/A	物模型版本号	version, Version	
设备软件版本	String	否	{1, 20}	N/A	设备软件版本号	firmware	固件版本号，用户可根据需要自行定义格式，如 10.0.20
通信单元（模组）供应商标识	String	否	{1, 100}	N/A	通信单元（模组）供应商识别	deviceCU, DeviceCU	
产品类别	String	否	{1, 100}	N/A	产品类别	productClass, ProductClass	

6.3 属性（Property）

6.3.1 定义

属性描述设备静态或动态数据。属性值是设备采集到的数据值，或者设备处于某种状态的值。用户可以根据实际情况选择名称和/或标识符作为唯一标识。

Json对象（Json object）是一种抽象数据类型，该类型兼容Json语法并可以包含INT、FLOAT、DOUBLE、TEXT、DATE等数据类型。

6.3.2 组成元素

属性的组成元素如表 4 所示：

表 4 属性（Property）组成元素

名称	类型	是否必选	长度	取值	关键字（可选）	备注
标识符	String	条件下必选	{1,100}	N/A	id, Identifier, identifier	用户根据需要可采用标识符作为属性唯一标识，也可以与名称一起作为属性的唯一标识。
英文名称	String	条件下必选	{1,50}	N/A	name, displayName, propertyName, Name	用户根据需要可采用名称作为属性的唯一标识，也可以与标识符一起作为属性的唯一标识。
中文名称	String	条件下必选	{1,50}	N/A	name, displayName, propertyName, Name	用户根据需要可采用名称作为属性的唯一标识，也可以与标识符一起作为属性的唯一标识。
描述	String	否	{1,50}	N/A	description, desc, Description	属性的描述信息
必选标志	Bool	否	N/A	True/False	required, Mandatory, tmRequired, Required	该属性是否必选属性，True为必选，Flase为可选。
数据类型	Enum	否	N/A	Int8/UInt8 Int16/UInt16 Int32/UInt32 Int64/UInt64 Bool/ Enum/ String/ Double/ Array/ Struct/ Bitmap	dataType, Type, DataType	属性数据的类型
取值最小值	String	否	N/A	INT(Int8/UInt8 Int16/UInt16 Int32/UInt32 Int64/UInt64), DOUBLE, FLOAT	min, Min	仅数据类型为数值型时有效
取值最大	String	否	N/A	INT(Int8/UInt	Max, max	仅数据类型为数值型时有

值				8 Int16/UInt16 Int32/UInt32 Int64/UInt64 , DOUBLE, FLOAT		效
精度	String	否	N/A	FLOAT/Double	precise, Precise	仅数据类型为FLOAT、Double时有效
读写标识	Enum	否	N/A	READ/WRITE /READWRITE	rwMode, accessMode, rwACL, rwFlag, mode, RwMode	READ等同于READ_ONLY, WRITE等同于WRITE_ONLY, 根据厂家实现来决定
单位	String	否	{1, 50}	N/A	unit, Unit	数据单位描述
数据模式	Json Object	否	N/A	INT, FLOAT, DOUBLE, TEXT, DATE, ARRAY	schema, Schema, dataSchema, DataSchema, tmData	有嵌套关系的情况下，对于结构型元数据的一种表述。 数据模式主要描述物模型各个元素所使用的不同类型的数据格式，数据模式可以包括数据格式，非列表数据格式以及列表数据格式。 数据模式的定义以及具体内容参见7.1。
自定义标志	Bool	否	N/A	True/False	custom, Custom	是否是自定义功能： ture:是 false:否

6.4 行为（Action）

6.4.1 定义

行为代表设备可对外提供的能力，用于描述设备可执行的动作或任务，行为执行时应携带输入参数，执行完成后的消息可携带输出参数。

行为也可命名为服务或命令。

用户可以根据实际情况选择名称和/或标识符作为唯一标识。

6.4.2 组成元素

行为的组成元素如表 5 所示：

表 5 行为（Action）组成元素

名称	类型	是否必选	长度	取值	关键字（可选）	备注
标识符	String	有条件必选	{1, 100}	N/A	identifier, id, Identifier	行为的标识。用户根据需要可采用标识符作为行为唯一标识，也可以与名称一起作为行为的唯一标识。
英文名称	String	有条件必选	{1, 50}	N/A	name, ServiceName, Servicename, Name	行为的名称。用户根据需要可采用标识符作为行为唯一标识，也可以与名称一起作为行为的唯一标识。
中文名称	String	有条件必选	{1, 50}	N/A	name, ServiceName, Servicename, Name	行为的名称。用户根据需要可采用标识符作为行为唯一标识，也可以与名称一起作为行为的唯一标识。
标题	String	否	{1, 100}	N/A	title, Title	用于显示的行为标题
描述	String	否	{1, 50}	N/A	description, desc, Description	行为的描述
必选标志	Bool	否	N/A	True/False	required, tmRequired, Required	行为是否可选，True为必选，False为可选。
输入参数	Array	是	[1, 100]	N/A	input, inputParams, inputs, Input, inputData	行为应携带的参数。参见 6.6
输出参数	Array	否	[1, 100]	N/A	output, outputParams, outputs, Output, outputData	行为执行结果可通过输出参数携带。参见6.6
调用类型	Enum	否	N/A	SYNC/ASYNC	callType, ActionType, CallType	行为的调用方式。 ASYNC：异步调用。 SYNC：同步调用。
自定义标志	Bool	否	N/A	True/False	custom, Custom	是否是自定义功能： ture:是 false:否

6.5 事件（Event）

6.5.1 定义

事件是设备运行时产生的各类事件信息的数据，包含告警、故障、信息等类型。事件可设置多个参数，并且可以基于标识不同的事件来发送多个事件。如设备故障时的上报的错误码，设备超过工作电压的告警等。

用户可以根据实际情况选择名称和/或标识符作为唯一标识。

6.5.2 组成元素

事件的组成元素如表 6 所示：

表 6 事件（Event）组成元素

名称	类型	是否必选	长度	取值	关键字（可选）	备注
标识符	String	有条件必选	{1, 100}	N/A	identifier, id, Identifier	用户根据需要可采用标识符作为事件唯一标识，也可以与名称一起作为事件的唯一标识。
英文名称	String	有条件必选	{1, 50}	N/A	name, eventName, Name	用户根据需要可采用名称作为事件的唯一标识，也可以与标识符一起作为事件的唯一标识。
中文名称	String	有条件必选	{1, 50}	N/A	name, eventName, Name	用户根据需要可采用名称作为事件的唯一标识，也可以与标识符一起作为事件的唯一标识。
标题	String	否	{1, 100}	N/A	title, Title	用于显示的事件标题
描述	String	否	{1, 50}	N/A	description, desc, Description	事件的描述信息
必选标志	Bool	否	N/A	True/False	required, Required	事件是否可选，True为必选，False为可选。
输出参数	Array/List	否	[1, 100]	N/A	output, outputParams, outputs, Output, outputdata, outputData	事件执行结果可通过输出参数携带。参加6.6。
类型	Enum	否	N/A	INFO_EVENT_TYPE/ALERT_EVENT_TYPE/ERROR_EVENT_TYPE	eventType, EventType	
自定义标志	Bool	否	N/A	True/False	custom, Custom	

设备事件上报后，可收到相应的响应码，例如响应码为 200 时表示处理成功。

6.6 输入参数和输出参数

6.6.1 定义

输入参数（Input）、输出（Output）参数用于行为与事件中。在行为中，输入、输出参数用于协助行为完成可执行的动作或者从动作之后获取反馈信息。在事件中，输出参数用于协助事件完成事件信息的录入或者反馈。

用户可以根据实际情况选择名称和/或标识符作为唯一标识。

6.6.2 组成元素

输入参数、输出参数的组成元素如表 7 所示：

表 7 输入（Input）、输出（Output）参数组成元素

名称	类型	是否必选	长度	取值	关键字（可选）	备注
标识符	String	有条件必选	{1, 100}	N/A	id, Identifier, identifier	参数的标识
名称	String	有条件必选	{1, 50}	N/A	displayName, Name, name	参数的名称
描述	String	否	{1, 50}	N/A	description, desc, Description	输入输出参数的描述信息
方向	String	否	PARAM_INPUT: 输入参数。 PARAM_OUTPUT: 输出参数		direction, Direction	
序号	Integer	否	[1, 100]		paraOrder, ParaOrder	
数据类型	String	否	N/A	ARRAY, STRUCT, INT, FLOAT, DOUBLE	datatype, type, DataType	
取值最小值	String	否	N/A	INT(Int8/UInt8 Int16/UInt16 Int32/UInt32 Int64/UInt64), DOUBLE, FLOAT	min, Min	仅数据类型为数值型时有效
取值最大值	String	否	N/A	INT(Int8/UInt8 Int16/UInt16 Int32/UInt32 Int64/UInt64), DOUBLE, FLOAT	Max, max	仅数据类型为数值型时有效
精度	String	否	N/A	Single/Double	precise, Precise	可参考属性中数据格式定义
单位	String	否	N/A	N/A	unit, Unit	可参考属性中数据格式定义
自定义标志	Bool	否	N/A	True/False	custom	是否是自定义功能： ture:是 false:否
数据模式	Json Object	否	N/A	INT, FLOAT, DOUBLE, TEXT, DATE, ARRAY	schema, Schema, dataSchema, DataSchema, tmData	可参考属性中数据模式定义，参见7.1

6.7 组件（Component）

6.7.1 定义

组件（Component）是基于元素的一个表示物的抽象层关系。组件可描述一个包含有多种元素（例如属性，行为，事件）的集合。也可以表示一个包含至少一个基础设备的抽象设备集合。基础设备包括元素中的多个同类型元素或者多个不同类型元素。例如一个WiFi连接模块可以描述为一个组件，该WiFi组件可由多个属性、行为和事件组成，如包括信噪比、信号强度、信道、BSSID以及频段等多个属性元素，开关WiFi电源、重置WiFi密码等多个行为元素；WiFi掉线、网络拥塞等多个事件元素。

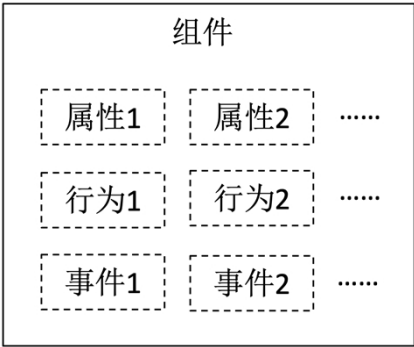


图 3 组件示意图

图3为组件的示意图。该组件可以是由属性1以及属性2组成，也可以是由行为1以及行为2组成，也可以由事件1以及事件2组成。该组件也可以是由属性1，行为1以及事件1组成。

组件又可以命名为功能模块或者组合关系。
用户可以根据实际情况选择名称和/或标识符作为唯一标识。

6.7.2 组成元素

组件的组成元素如表 8 所示：

表 8 组件（Component）组成元素

名称	类型	是否必选	长度	取值	关键字（可选）	备注
标识符	String	有条件必选	{1, 100}	N/A	id, Id, identifier	组件的标识符
英文名称	String	有条件必选	{1, 50}	N/A	name, Name	组件的英文名称
中文名称	String	有条件必选	{1, 50}	N/A	name, Name	组件的中文名称
标题	String	否	{1, 50}	N/A	displayName, DisplayName	组件可显示的标题
描述	String	否	{1, 200}	N/A	description, Description	
属性列表	List	否	0...N	属性复合对象（见 6.3）	properties	组件可包含的属性列表，可选用全

						部/部分元素信息 (见6.3)
行为列表	List	否	0...N	行为复合对象 (见6.4)	ActionList, Actions, actions	组件可包含的行为列表, 可选用全部/部分元素信息 (见6.4)
事件列表	List	否	0...N	事件复合对象 (见6.5)	eventList, events	组件可包含的事件列表, 可选用全部/部分元素信息。 (见6.5)
子组件列表	List	否	0...N	组件复合对象 (见6.5)	subFunctionBlock	组件可包含子组件。
数据模式	Json Object	否	N/A	复合对象	schema, Schema, dataSchema, DataSchema, tmData	用于校验组件的信息描述

其中，组件的组成元素中可以嵌套子组件列表。子组件列表元素可以嵌套包含属性、行为以及事件中的一种或者多种。

6.8 物模板 (Thing template)

6.8.1 定义

物模板 (Thing template) 是基于组件以及物模型要素的一个表示复杂物的抽象层关系。物模型模板 (物模板) 可表示一个包含有至少一个组件、以及至少一个基础要素 (例如功能要素的其中一个或多个) 的集合。可选的，物模板中也可以包含自定义要素，例如自定义的行为等。物模板可以由至多一个档案信息、至多一个状态信息、0个或者若干个组件，0个或者若干个属性、0个或者若干个行为以及0个或者若干个事件列表构成。

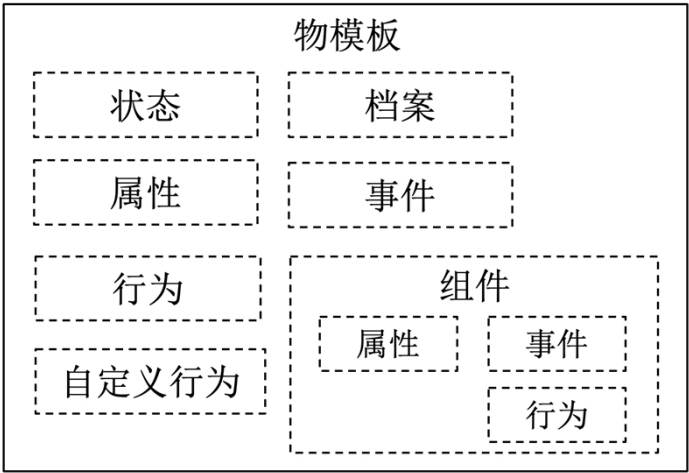


图 4 物模板示意图

图4为物模板的示意图。该物模板由一个组件，一个属性，一个事件以及一个自定义行为构成。

6.8.2 组成元素

物模板的组成元素如表 9 所示：

表 9 物模板（Thing template）组成元素

名称	类型	是否必选	长度	取值	关键字（可选）	备注
标识符	String	否	{1, 100}	N/A	id, Id, identifier	物模板的标识符
物模板名称	String	是	{1, 50}	N/A	name, Name	物模板的名称，物模板的名称可以包含行业信息。
标题	String	否	{1, 50}	N/A	displayName, DisplayName	物模板可显示的标题
描述	String	否	{1, 200}	N/A	description, Description	物模板的描述
属性列表	List	否	0...N	属性复合对象（见6.3）	properties	物模板可包含的属性列表，可选用全部/部分元素信息（见6.3）
行为列表	List	否	0...N	行为复合对象（见6.4）	ActionList, Actions, actions	物模板可包含的行为列表，可选用全部/部分元素信息（见6.4）
事件列表	List	否	0...N	事件复合对象（见6.5）	eventList, events	物模板可包含的事件列表，可选用全部/部分元素信息。（见6.5）
组件列表	List	否	0...N	组件复合对象（见6.5）	FunctionBlock, functionBlocks	物模板可包含组件列表，可选用全部/部分元素信息。（见6.7）
数据模式	Json Object	否	N/A	复合对象	schema, Schema, dataSchema, DataSchema, tmData	用于校验组件的信息描述

7 物模型数据模式

7.1 数据模式

7.1.1 定义

数据模式主要描述物模型各个元素所使用的不同类型的数据格式，数据模式可以包括数据格式，非列表数据格式以及列表数据格式。

7.2 数据格式

7.2.1 定义

数据格式主要用来描述数据的类型、取值范围等约束性的属性（元数据）。物模型中的数据格式可用于对运行时设备的数据实例进行语法校验。

用户可以根据实际情况选择名称和/或标识符作为唯一标识。

7.2.2 组成元素

数据格式的组成元素分为基本数据类型和其它复合数据类型的元素。基本数据类型（例如 int, float, double, bool, string）的元素如表 10 所示：

表 10 基本数据类型数据格式组成元素

名称	类型	是否必选	说明	关键字（可选）
数据类型	String	是	可支持数据类型：int、float、double、string、bitMap	dataType, type, DataTye
最大值	Value type	否	仅 dataType 为 int、float、double 的三种数据类型时有效	max, maximum, Max
最小值	Value type	否	仅 dataType 为 int、float、double 的三种数据类型时有效	min, minimum, Min
步长	String	否	变化幅度, 数据每次变化的增量, 仅 dataType 为 int、float 或 double 三种数据类型时有效	step, scale, Step
单位	String	否	数据的单位	unit, Unit
单位名称	String	否	数据的单位名称	unitName
默认值	Accrding to Data type definition	否	根据数据类型定义	default, defaultValue, Default
精度	String	是	仅 dataType 取值为 float 或 double 时	precise, Precise
最小长度	Integer	否	String 的最小长度, 仅 dataType 为 string 数据类型时有效	minLength, min, MinLength
最大长度	Integer	否	String 的最大长度, 仅 dataType 为 string 数据类型时有效	maxLength, max, MaxLength
自定义标志	Boolean	否	ture: 是。false: 否。	custom, Custom

DATE 以及 TEXT 数据类型的元素如表 11 所示：

表 11 DATE 以及 TEXT 数据类型数据格式组成元素

名称	类型	是否必选	说明	关键字（可选）
数据类型	String	是	可支持数据类型：DATE、TEXT	dataType, type, DataType
默认值	Accrding to Data type definition	否	根据数据类型定义	default, defaultValue, Default
数据长度	Uint64	否	数据长度，取值不超过 2048，单位：字节。仅 dataType 取值为 TEXT 时使用。	length, dataLength, Length
时间格式	String	否	时间格式，仅 dataType 为 date 时有效	timestamp, Timestamp
自定义标志	Boolean	否	ture：是。false：否。	custom, Custom

ARRAY 数据类型的元素如表 12 所示：

表 12 ARRAY 数据类型数据格式组成元素

名称	类型	是否必选	说明	关键字（可选）
数据类型	String	是	可支持数据类型：ARRAY	dataType, type, DataType
数组长度	Uint64	否	数组长度，仅 dataType 为 array 时有效	size, Size
子数据类型	String	否	数组中的元素的数据类型。可选值：STRUCT、INT、FLOAT、DOUBLE 或 TEXT。	childDataType
数组项说明	Json object	否	仅 dataType 为 Array 时有效。数组项说明。 格式为 {元素1, 元素2, 元素3}，每个元素可以是基本类型。	arrayDesc, ArrayDesc
自定义标志	Boolean	否	ture：是。false：否。	custom, Custom

ENUM 和 BOOL 数据类型的元素如表 13 所示：

表 13 ENUM 和 BOOL 数据类型数据格式组成元素

名称	类型	是否必选	说明	关键字（可选）
数据类型	String	是	可支持数据类型：ENUM，BOOL	dataType, type, DataType
枚举项说明	Json object	否	枚举项说明。 格式为 〈枚举值1〉：〈枚举值名称1〉 〈枚举值2〉：〈枚举值名称2〉 ... 〈枚举值n〉：〈枚举值名称n〉	value, enumDesc, mapping value, Value
枚举值	String/enum/Integer	否	枚举值	value, enumDesc, mapping value, Value
自定义标志	Boolean	否	ture：是。false：否。	custom, Custom

STRUCT 数据类型的元素如表 14 所示：

表 14 STRUCT 数据类型数据格式组成元素

名称	类型	是否必选	说明	关键字（可选）
数据类型	String	是	可支持数据类型：STRUCT	dataType, type, DataType
标识符	String	否	结构体中的子参数的标识符。	identifier
名称	String	否	结构体中的子参数名称。	name
子参数数据类型	String	否	结构体中的子参数数据类型。可选值：INT、FLOAT、DOUBLE、TEXT、DATE、ENUM、BOOL。	
结构体项说明	Json object	否	仅dataType为Struct时有效。结构体说明，由2个或者2个以上属性参数组合而成，用来表示多个属性相互紧密关联，同时存在才有意义的数据类型，例如颜色RGB，HSV等。 格式为：属性参数（2..N）。 属性参数可以对应于不同的数据类型，但不能是结构体类型。结构体类型数据不支持嵌套。	structDesc
自定义标	Boolean	否	ture：是。false：否。	custom, Custom

志				
---	--	--	--	--

BITMAP 数据类型的元素如表 15 所示：

表 15 BITMAP 数据类型数据格式组成元素

名称	类型	是否必选	说明	关键字（可选）
数据类型	String	是	可支持数据类型：BITMAP	dataType, type, DataType
位图项说明	Json object	否	仅dataType为Bitmap时有效。位图项说明，每位表示真、假两种状态变量 格式： 0（False）:<状态描述> 1（True）:<状态描述>	bitmapDesc
自定义标志	Boolean	否	ture：是。false：否。	custom, Custom

7.3 列表数据格式

7.3.1 定义

列表数据格式主要用于校验数据类型（dataType）为列表型（ENUM、BOOL、STRUCT）的输入参数。其可以作为属性，输入输出参数，数据格式的可选组成元素。

7.3.2 组成元素

列表数据格式的组成元素如表 16 所示：

表 16 列表数据格式组成元素

名称	类型	是否必选	长度	取值	关键字（可选）	备注
列表数据格式	List	否	N/A	非列表数据格式的列表复合结构	dataSpecsList, DataSpecsList	列表数据格式主要用于校验制定数据类型的输入参数。 数据类型（dataType）为列表型（ENUM、BOOL、STRUCT）的数据规范存储在 dataSpecsList 中。 说明： - dataSpecs 和

						dataSpecsList 之中必须传入且只能传入一个，请根据实际数据类型传入。
--	--	--	--	--	--	--

7.4 非列表数据格式

7.4.1 定义

列表数据格式和非列表数据格式主要用于校验数据类型（dataType）为非列表型（INT，FLOAT, DOUBLE, TEXT, DATE, ARRAY）的输入参数。
其可以作为属性，输入输出参数，数据格式的可选组成元素。

7.4.2 组成元素

非列表数据格式的组成元素如表 17 所示：

表 17 列表数据格式和非列表数据格式组成元素

名称	类型	是否必选	长度	取值	关键字（可选）	备注
非列表数据格式	Json Object	否	N/A	INT, FLOAT, DOUBLE, TEXT, DATE, ARRAY	dataSpecs, DataSpecs	非列表数据格式主要用于校验指定数据类型的输入参数。数据类型（dataType）为非列表型（INT，FLOAT, DOUBLE, TEXT, DATE, ARRAY）的数据规范存储在 dataSpecs 中。 说明： - dataSpecs 和 dataSpecsList 之中必须传入且只能传入一个，请根据实际数据类型传入。

附录 A

(资料性)

实例化实施建议

参考第5章，给出数控机床物模型的实例和人脸识别门禁的实例化示例，并且给出了数据模式的实例化示例。

A.1 数控机床实例化示例

表A.1给出了数控机床属性的参考实例。由于长度限制，本表只列出了非空字段。

表 A.1.1 数控机床属性参考实例

属性	标识符	名称	描述	是否可选	数据类型	单位	取值	读写标识	数据格式
Device_ID	dtml:CNCMInformation:DeviceID;1	设备ID	设备的唯一标识码，例如数控机床的制造商ID和生产SN号码，系统ID不应超过255个字节	是	String		"maxLength": 10, "minLength": 1	READ	"dataSchema": { "@type": "string", "maxLength": 10, "minLength": 1, "pattern": "[a-zA-z]" }
Name	dtml:CNCMInformation:Name;1	名称	设备名称应保证唯一性，能够区分同品牌、同型号的不同数控机床	是	String			READWRITE	dataSchema: "string"
Description	dtml:CNCMInformation:Description;1	描述	设备描述	是	String		maxLength: 5	READ	"dataSchema": { "@type": "string", "maxLength": 5 }
Vendor_Name	dtml:CNCMInformation:VendorName;1	"zh-Hans": "制造 商名"	"zh-Hans": "制造 商名称中文描述", "en": "manu"	是	String		"maxLength": 10, "minLength": 1	READ	"dataSchema": { "@type": "string", "maxLength": 10, "minLength": 1 }

		称", "en": "manufacturer"	facturer descriptio n"						"minLength": 1 }
Device _Lengt h	dtml:CNCMInf ormation:Dev iceLength;1	设备 长度	数控机床的 长度	是	Lengt hUnit	Metre	0-100	READ	"dataSchema": { "@type": "float", "max": 100, "min": 0, "step": 0.1 }
Electr ic_Val ue	dtml:CNCMInf ormation:Ele ctricValue;1	电量 值	数控机床的 电量	是	int		0-50	READWRIT E	"dataSchema": { "@type": "int", "max": 50, "min": 0 }
Close_ Clock_ Time	dtml:Close_C lock_Time;1	关机 时间	数控机床的 最近关机时 间	是	dataT ime		例: 2021-07-23 16:31:25	READWRIT E	"dataSchema": { "@type": "dateTime", "format": "YYYY-MM-DD hh:mm:ss" }

表A.1.2给出了数控机床服务的参考实例。

表 A.1.2 数控机床行为参考实例

服务	标识符	名称	描述	是否 可选	输入参数	输出 参数
ncProgra mLoad	dtml:ncProgr amLoad;1	NC 程序加 载操作	加载操 作演示	是	"MACHINE_ID", "PROGRAM_ID", "PROGRAM"	"SUCC ESS"

A.2 人脸识别门禁的实例化示例

表 A. 2. 1 给出了人脸识别门禁的实例化

表 A.2.1 人脸识别门禁参考实例

功能类型	功能名称	标识符	数据类型	数据值定义	读写类型
属性	人脸库MD5签名	FaceSetMD5Sign	string(字符串)	数据长度：255	读
属性	人脸比对成功阈值	OnMatchThreshOld	double(双精度浮点型)	取值范围：0-100； 步长：0.01； 单位：百分比 / %	读写
属性	人脸库ID	FaceSetID	string(字符串)	数据长度：255	读写
属性	人脸库算法版本	FaceSetAlgorithmVersion	string(字符串)	数据长度：255	读写
属性	当前人脸库大小	FaceSetSize	int32(整数型)	取值范围：0-200000； 步长：1	读写
属性	人脸库图片存储能力	FaceSetPicStoreAbility	int32(整数型)	取值范围：0-200000； 步长：1	读写
属性	地理位置	GeoLocation	struct(结构体)	-	读写
事件	人脸算法SDK需授权	NeedAuthVerifySDK	-	事件类型：信息	-
事件	人脸检测事件上报	OnDetect	-	事件类型：信息	-
事件	人脸比对事件上报	OnMatched	-	事件类型：信息	-
服务	设备授权文件下发	AuthVerifySDK	-	调用方式：同步	-
服务	同步人脸库图片	SyncFacePictures	-	调用方式：同步	-
服务	查询人脸布控进度	QuerySyncPicSchedule-	-	调用方式：同步	-
服务	查询布控成功	QueryAddedUserInfo	-	调用方式：同步	-

功能类型	功能名称	标识符	数据类型	数据值定义	读写类型
	的人脸图				

对应表A.2.1的人脸识别门禁描述文件如下：

```
{
  "version": "1.0",
  "profile": {
    "industryId": "1",
    "sceneId": "14",
    "categoryId": "300",
    "productId": "TqbMjRCaXR"
  },
  "properties": [
    {
      "identifier": "FaceSetMD5Sign",
      "name": "人脸库 MD5 签名",
      "functionType": "st",
      "accessMode": "r",
      "desc": "",
      "dataType": {
        "type": "string",
        "specs": {
          "length": 255
        }
      }
    },
    {
      "identifier": "OnMatchThreshOld",
      "name": "人脸比对成功阈值",
      "functionType": "st",
      "accessMode": "rw",
      "desc": "",
      "dataType": {
        "type": "double",
        "specs": {
          "min": "0",
          "max": "100",
          "unit": "百分比 / %",
          "step": "0.01"
        }
      }
    }
  ]
}
```

```

},
{
  "identifier": "FaceSetID",
  "name": "人脸库 ID",
  "functionType": "st",
  "accessMode": "rw",
  "desc": "",
  "dataType": {
    "type": "string",
    "specs": {
      "length": 255
    }
  }
},
{
  "identifier": "FaceSetAlgorithmVersion",
  "name": "人脸库算法版本",
  "functionType": "st",
  "accessMode": "rw",
  "desc": "",
  "dataType": {
    "type": "string",
    "specs": {
      "length": 255
    }
  }
},
{
  "identifier": "FaceSetSize",
  "name": "当前人脸库大小",
  "functionType": "st",
  "accessMode": "rw",
  "desc": "",
  "dataType": {
    "type": "int32",
    "specs": {
      "min": "0",
      "max": "200000",
      "unit": "",
      "step": "1"
    }
  }
},
},

```

```

{
  "identifier": "FaceSetPicStoreAbility",
  "name": "人脸库图片存储能力",
  "functionType": "st",
  "accessMode": "rw",
  "desc": "",
  "dataType": {
    "type": "int32",
    "specs": {
      "min": "0",
      "max": "200000",
      "unit": "",
      "step": "1"
    }
  }
},
{
  "identifier": "GeoLocation",
  "name": "地理位置",
  "functionType": "st",
  "accessMode": "rw",
  "desc": "",
  "dataType": {
    "type": "struct",
    "specs": [
      {
        "identifier": "Longitude",
        "name": "经度",
        "dataType": {
          "type": "double",
          "specs": {
            "min": "-180",
            "max": "180",
            "unit": "度 / °",
            "step": "0.01"
          }
        }
      }
    ]
  },
  {
    "identifier": "Latitude",
    "name": "纬度",
    "dataType": {
      "type": "double",

```

```

        "specs": {
            "min": "-90",
            "max": "90",
            "unit": "度 / ° ",
            "step": "0.01"
        }
    },
    {
        "identifier": "Altitude",
        "name": "海拔",
        "dataType": {
            "type": "double",
            "specs": {
                "min": "0",
                "max": "9999",
                "unit": "米 / m",
                "step": "0.01"
            }
        }
    },
    {
        "identifier": "CoordinateSystem",
        "name": "坐标系统",
        "dataType": {
            "type": "enum",
            "specs": {
                "1": "WGS_84",
                "2": "GCJ_02"
            }
        }
    }
]
}
],
"events": [
    {
        "identifier": "NeedAuthVerifySDK",
        "name": "人脸算法 SDK 需授权",
        "functionType": "st",
        "eventType": "info",
        "desc": "",

```

```

"outputData": [
  {
    "identifier": "PackageName",
    "name": "包名",
    "dataType": {
      "type": "string",
      "specs": {
        "length": 255
      }
    }
  },
  {
    "identifier": "ClientID",
    "name": "设备唯一标识",
    "dataType": {
      "type": "string",
      "specs": {
        "length": 255
      }
    }
  },
  {
    "identifier": "PublicKey",
    "name": "包公钥信息",
    "dataType": {
      "type": "string",
      "specs": {
        "length": 255
      }
    }
  },
  {
    "identifier": "Token",
    "name": "设备 Token 值",
    "dataType": {
      "type": "string",
      "specs": {
        "length": 255
      }
    }
  },
  {
    "identifier": "DeviceEncrypt",

```

```

    "name": "设备加密信息唯一标识",
    "dataType": {
      "type": "string",
      "specs": {
        "length": 255
      }
    }
  ]
},
{
  "identifier": "OnDetect",
  "name": "人脸检测事件上报",
  "functionType": "st",
  "eventType": "info",
  "desc": "",
  "outputData": [
    {
      "identifier": "StoreID",
      "name": "人脸检测图 StoreID",
      "dataType": {
        "type": "string",
        "specs": {
          "length": 255
        }
      }
    },
    {
      "identifier": "GroupID",
      "name": "人脸组 ID",
      "dataType": {
        "type": "string",
        "specs": {
          "length": 255
        }
      }
    }
  ]
},
{
  "identifier": "OnMatched",
  "name": "人脸比对事件上报",
  "functionType": "st",

```



```

"eventType": "info",
"desc": "",
"outputData": [
  {
    "identifier": "UserPicID",
    "name": "用户人脸图 ID",
    "dataType": {
      "type": "string",
      "specs": {
        "length": 255
      }
    }
  },
  {
    "identifier": "FaceMatchedPicStorID",
    "name": "人脸比对图片 StorID",
    "dataType": {
      "type": "string",
      "specs": {
        "length": 255
      }
    }
  },
  {
    "identifier": "Similarity",
    "name": "人脸比对相似度结果",
    "dataType": {
      "type": "float",
      "specs": {
        "min": "0",
        "max": "100",
        "unit": "",
        "step": "0.1"
      }
    }
  },
  {
    "identifier": "UserInfo",
    "name": "人脸 ID 相关属性",
    "dataType": {
      "type": "string",
      "specs": {
        "length": 255
      }
    }
  }
]

```

```

    }
  }
},
{
  "identifier": "GroupID",
  "name": "人脸组 ID",
  "dataType": {
    "type": "string",
    "specs": {
      "length": 255
    }
  }
}
]
}
],
"services": [
  {
    "identifier": "AuthVerifySDK",
    "name": "设备授权文件下发",
    "functionType": "st",
    "desc": "",
    "callType": "sync",
    "input": [
      {
        "identifier": "LicenseData",
        "name": "授权文件",
        "dataType": {
          "type": "string",
          "specs": {
            "length": 255
          }
        }
      }
    ],
    {
      "identifier": "StatusCode",
      "name": "状态码",
      "dataType": {
        "type": "int32",
        "specs": {
          "min": "0",
          "max": "65535",
          "unit": "",

```

```

        "step": "1"
      }
    }
  },
  {
    "identifier": "StatusCodeDescription",
    "name": "状态码描述",
    "dataType": {
      "type": "string",
      "specs": {
        "length": 255
      }
    }
  }
],
"output": [
  {
    "identifier": "DoAuthorized",
    "name": "设备授权是否成功",
    "dataType": {
      "type": "bool",
      "specs": {
        "true": "授权失败",
        "false": "授权成功"
      }
    }
  }
]
},
{
  "identifier": "SyncFacePictures",
  "name": "同步人脸库图片",
  "functionType": "st",
  "desc": "",
  "callType": "sync",
  "input": [
    {
      "identifier": "FacePicURL",
      "name": "同步文件 URL 地址",
      "dataType": {
        "type": "string",
        "specs": {
          "length": 255
        }
      }
    }
  ]
}

```

```

        }
    }
}
],
"output": [
    {
        "identifier": "SyncPicStatus",
        "name": "设备同步图片状态值",
        "dataType": {
            "type": "enum",
            "specs": {
                "0": "成功",
                "1": "布控中",
                "2": "下载文件失败",
                "3": "解析文件失败"
            }
        }
    }
]
},
{
    "identifier": "QuerySyncPicSchedule",
    "name": "查询人脸布控进度",
    "functionType": "st",
    "desc": "",
    "callType": "sync",
    "input": [
        {
            "identifier": "GroupID",
            "name": "人脸库组 ID",
            "dataType": {
                "type": "string",
                "specs": {
                    "length": 255
                }
            }
        }
    ]
},
"output": [
    {
        "identifier": "Rate",
        "name": "人脸图布控百分比",
        "dataType": {

```

```

        "type": "double",
        "specs": {
            "min": "0",
            "max": "100",
            "unit": "百分比 / %",
            "step": "0.01"
        }
    }
}
]
},
{
    "identifier": "QueryAddedUserInfo",
    "name": "查询布控成功的人脸图",
    "functionType": "st",
    "desc": "",
    "callType": "sync",
    "input": [
        {
            "identifier": "GroupID",
            "name": "人脸库组 ID",
            "dataType": {
                "type": "string",
                "specs": {
                    "length": 255
                }
            }
        }
    ],
    "output": [
        {
            "identifier": "StoreID",
            "name": "人脸布控文件 storeID",
            "dataType": {
                "type": "string",
                "specs": {
                    "length": 255
                }
            }
        }
    ],
    {
        "identifier": "Type",
        "name": "类型",

```

```

    "dataType": {
      "type": "enum",
      "specs": {
        "0": "StoreID",
        "1": "FileName",
        "2": "URL"
      }
    },
    {
      "identifier": "SyncPicStatus",
      "name": "查询添加用户信息的布控状态",
      "dataType": {
        "type": "enum",
        "specs": {
          "0": "成功",
          "1": "布控中"
        }
      }
    }
  ]
}

```

A.3 数据模式中非列表、列表数据格式实例化示例

A.3给出了数据模式中，非列表、列表数据格式实例化示例

dataType 为 INT 的 **dataSpecs** 示例：

```

{
  "dataSpecs": {
    "custom": true,
    "dataType": "INT",
    "defaultValue": "30",
    "max": "1440",
    "min": "0",
    "step": "10",
    "unit": "min"
  }
}

```

dataType 为 ENUM 的 **dataSpecsList** 示例：

```

{
  "dataSpecsList": [

```

```
{
  "custom": false,
  "dataType": "ENUM",
  "defaultValue": "true",
  "name": "打开",
  "value": 1
},
{
  "custom": false,
  "dataType": "ENUM",
  "defaultValue": "false",
  "name": "关闭",
  "value": 0
}
]
```

附录 B

(资料性)

元素、组件以及物模板之间关系

图B. 1展示了元素、组件、以及物模板三者之间的关系。元素池内包括各种属性、行为以及事件。而Wi-Fi连接模块可以视为一个组件，该模块由多个属性（例如，信噪比，信号强度，信道，BSSID以及频段等）组合而成。而灯可以视为一个物模板，该物模板由多个属性（例如颜色，开关，明暗度）以及事件（例如故障上报）以及若干个组件（例如Wi-Fi连接组件，LoRa连接组件等）。组件和物模板可以从不同的抽象层的角度来诠释更复杂的物。



图 B.1 元素、组件以及物模板关系示意图

附录 C

(资料性)
物模型使用场景

多种物联网场景（如智慧社区等）均可使用标准物模型，目前物联网标准物模型应用场景如下表C.1所示：

表C.1 物联网标准物模型应用场景

序号	行业大类	所属场景	品类名称
1	智慧水利	水利设备	水位计
2	智慧城市	智能环卫	智能垃圾桶
3	智慧城市	智能楼宇	电子桌牌
4	智慧城市	种植养殖	喷灌智能终端
5	智慧城市	消防安全	声光报警设备
6	智慧城市	消防安全	烟雾探测器
7	智慧城市	消防安全	燃气泄漏报警器
8	智慧城市	消防安全	井盖
9	智慧城市	消防安全	用户信息传输装置
10	智慧城市	消防安全	灭火器
11	智慧城市	消防安全	烟温复合探测器
12	智慧城市	消防安全	消防栓监测
13	智慧城市	环境感知	空调
14	智慧城市	环境感知	水质监测
15	智慧城市	环境感知	环境监测设备
16	智慧城市	环境感知	大气监测设备
17	智慧城市	环境感知	温湿度检测
18	智慧城市	环境感知	压力检测设备
19	智慧城市	环境感知	土壤监测
20	智慧城市	环境感知	管网监测
21	智慧城市	环境感知	海底探测器
22	智慧城市	能源管理	水表
23	智慧城市	能源管理	燃气表
24	智慧城市	能源管理	热量表
25	智慧城市	能源管理	电表
26	智慧城市	公共服务	门磁
27	智慧城市	公共服务	智能小音箱
28	智慧城市	公共服务	无人机
29	智慧城市	公共服务	话筒
30	智慧城市	公共服务	音柱及收扩器
31	智慧城市	公共服务	红外体征探测器
32	智慧城市	公共服务	摄像头

33	智慧城市	公共服务	电动自行车充电桩
34	智慧城市	公共服务	消防栓帽
35	智慧城市	公共服务	智能用电
36	智慧城市	公共服务	智能广播
37	智慧城市	公共服务	智能垃圾亭
38	智慧城市	公共服务	智能广告牌
39	智慧城市	公共服务	无线远程开关
40	智慧城市	公共服务	电气火灾探测器
41	智慧城市	公共服务	定位器
42	智慧城市	公共服务	执法记录仪
43	智慧城市	公共服务	客流统计
44	智慧城市	公共服务	智慧光交箱锁
45	智慧城市	公共服务	地磁探测器
46	智慧城市	公共服务	电梯探测器
47	智慧城市	公共服务	路灯照明
48	零售	pos 机	A 型 pos 机
49	智能生活	个护健康	血压计
50	智能生活	个护健康	手环
51	智能生活	个护健康	皮肤检测设备
52	智能生活	个护健康	人体体征探测器
53	智能生活	个护健康	体脂秤
54	智能生活	大家电	空调机
55	智能生活	电工照明	插座
56	智能生活	环境电器	音频播放器
57	智能生活	环境电器	新风机
58	智能生活	环境电器	净水器
59	智能生活	环境电器	空气净化器
60	智能生活	家居安防	红外探测器
61	智能生活	家居安防	智能门锁
62	智能生活	家居安防	烟雾报警器
63	智能生活	家居安防	门磁传感器
64	智能生活	家居安防	燃气报警器
65	智能生活	家居安防	温感报警器
66	智能生活	家居安防	报警开关
67	智能停车	车位锁	车位锁
68	智能家居	智能家居	路由器
69	智能家居	智能家居	空调

其他垂直行业（例如智慧社区以及行业现场网）均基于本文件的物模型制定相应规范（物模型技术标准部分）。

参考文献

- [1] GB/T 4754-2017 国民经济行业分类
 - [2] GB/T 33474-2016 物联网 参考体系结构
 - [3] GB/T 38606-2020 物联网标识体系 数据内容标识符
 - [4] GB/T 38662-2020 物联网标识体系 Ecode标识应用指南
 - [5] 2020-1167T-YD 物联网云平台技术要求
 - [6] W3C Web of Things Description 1.1,
URL:<https://www.w3.org/TR/wot-thing-description11/>
 - [7] ITU-T Y.2060 Global Information Infrastructure, Internet Protocol Aspects and Next-Generation Network, Next Generation Networks – Frameworks and functional architecture models
 - [8] OMA LightweightM2M (LwM2M) object,
URL: <https://technical.openmobilealliance.org/OMNA/LwM2M/LwM2MRegistry.html>
 - [9] ISO/IEC 30118-4:2018 Information technology – Open Connectivity Foundation (OCF) Specification – Part 4: Resource Type Specification
 - [10] ISO/IEC 30118-5:2021 Information technology – Open Connectivity Foundation(OCF) Specification – Part 5: OCF Device Specification
 - [11] Bluetooth Mesh Model specification,
URL:<https://www.bluetooth.com/specifications/specs/mesh-model-1-0-1/>
-