|  |
| --- |
| ICS 35.240  CCS L73 |

Community standard

SQL/SCIE XX-XXXX, T/SZIOT XXXXX-XXXX

|  |
| --- |
|  |

City perception system - Thing model requirements

|  |
| --- |
|  |
| (Exposure draft) |

Issued date: XXXX-XX-XX

Implementation date: XXXX-XX-XX implementation

Smart City Industry Ecosystem

Released by Shenzhen Internet of Things Industry Association

Contents

[Foreword II](#_Toc191281772)

[1 Scope 1](#_Toc191281774)

[2 Normative references 1](#_Toc191281775)

[3 Terms and definitions 1](#_Toc191281776)

[4 Overall structure 1](#_Toc191281784)

[5 Thing model requirements 2](#_Toc191281785)

[5.1 Field definition 2](#_Toc191281786)

[5.2 Feature list 4](#_Toc191281787)

[5.3 Scalability 7](#_Toc191281788)

Foreword

This standard is drafted in accordance with the rules given in GB/T 1.1-2020.

This standard was proposed by the Smart City Industry Eosystem.

This standard was prepared by the Smart City Industry Ecosystem and Shenzhen IoT Industry Association.

This standard is drafted by Huawei Technologies Co., Ltd., Shenzhen Institute of Standards and Technology, Chengdu Smart City Information Technology Co., Ltd., China Institute of Electronic Technology Standardization, Hangzhou Xujian Technology Co., Ltd., Guangzhou Xinwei Smart Security Technology Co., Ltd., and Shenzhen Starnet ICT Co., Ltd.

The drafters of this standard are XXX and XXXX.

City perception system - Thing model requirements

1. Scope

This document establishes the overall architecture of the thing model of the urban perception system, and specifies the elements and feature list of the thing model.

This document is applicable to the design, development and application of the thing model of the urban perception system.

1. Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition (including any amendments) applies.

GB/T 38637.2-2020 Internet of things—Access of sensing and controlling device—Part 2:Data management requirements

GB/T 41780.2-2024 Internet of things—Edge computing—Part 2：Data management requirements

GB/T 43697-2024 Data security technology—Rules for data classification and grading

1. Terms and definitions

For the purpose of this document, the following terms and definitions apply.



thing model

the expression form of IoT terminal devices and their perceived information in the digital space after abstract modeling, defines device information, services, and events



service

a logical or physical unit that represents a component of a device and consists of a set of feature elements



event

a specific type of information reported by a terminal device under a certain condition during the running of the terminal device. Generally, notifications, such as faults and alarms, need to be perceived and processed by external systems

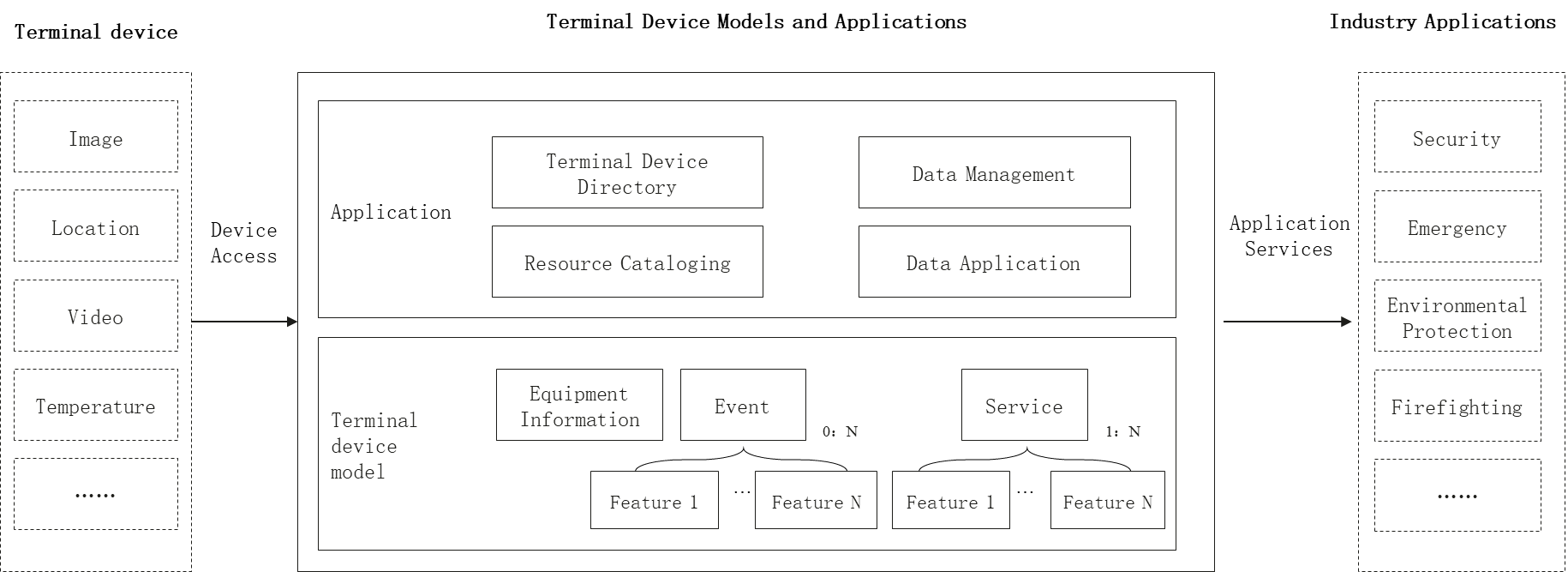


fature characteristics

a capability or status of a terminal device.

1. Overall structure

Figure 1 shows the overall architecture of the city multi-level linkage command platform.



1. Overall structure of thing model and application

Note: Terminal devices and industry applications (contents in dotted boxes) are related and are not restricted in this document.

In the overall architecture of the thing model:

1. A terminal device is a modeling object of a model.
2. Thing models and applications abstract and virtualize terminal devices to form applications. The external system may read data of a feature to obtain the status of the terminal device, or can drive the device to perform an action by modifying the feature value. At least one type of service sahll be provided and at least one feature should be included. Multiple events and multiple features may be provided, or no event should be provided.
3. Integrates and collaborates data resources of different types of devices based on thing models to provide standardized terminal device description for industry applications.
4. Thing model requirements
   1. Field definition
      1. Device information

Table 1 Components of the device information

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | English name | Data type | Maximum length/fixed length | Mandatory or not | Description |
| Device model ID | prodId | String | Fixed length: 5 bytes | Mandatory | The ID of the same device shall be unique. The value is uniformly allocated by the operating entity of the equipment model. The value contains 5 characters. The default first character is 1, that is, 1XXXX. For the other four characters, the value range of each character is [0-9] or [A-Z]. Theoretically, 1679616 value options may be provided. |
| Device name | deviceName | String | Maximum length: 255 bytes | Mandatory | It is represented by a character string. The value contains a maximum of 255 bytes. |
| Certified model | deviceModel | String | Maximum length: 32 bytes | Mandatory | A character string used by the device vendor to identify the device model. The maximum length is 32 bytes. |
| Device category ID | deviceTypeId | String | Fixed length: 4 bytes | Mandatory | The ID indicates the type of the device. The value is uniformly allocated by the operating entity of the equipment model. The value contains four characters. The first character is 0 by default, that is, 0XXX. For the other three characters, the value range of each character is [0-9] or [A-Z]. Theoretically,46655 value options may be provided. |
| Device category name | deviceTypeName | String | Maximum length: 255 bytes | Mandatory | Corresponds to the “Device Category ID” in the previous line, which indicates the device category name. The value contains a maximum of 255 bytes. |
| Device manufacturer ID. | manufacturerId | String | Fixed length: 3 bytes | Mandatory | The manufacturer ID of the same device shall be unique. The length is fixed to 3 bytes. |
| Device manufacturer name | manufacturerName | String | Maximum length: 255 bytes | Mandatory | Corresponds to the “Device Manufacturer ID” in the previous line, which indicates the name of the device manufacturer. The value contains a maximum of 255 bytes. |

* + 1. Event field

An event is a specific type of information that is actively pushed or reported under certain conditions during the running of a terminal device. It mainly describes the information fields that need to be covered by an event. For the definitions of the event fields, see Table 2.

Table 2 Event field definition

| Field | English name | Mandatory or not | Data type | Maximum length | Description |
| --- | --- | --- | --- | --- | --- |
| Event type ID. | eventType | Mandatory | string | 64 | Uniquely identifies an event type in the city awareness system. |
| Event instance ID. | eventId | Mandatory | string | 32 | This parameter uniquely identifies an event instance within a specific device object. This identifier enables external systems to identify the meaning or source of the event after receiving the event. |
| Event name | / | Mandatory | string | / | The value can contain a maximum of 30 characters, including uppercase letters, lowercase letters, digits, hyphens (-), underscores (\_), and dots (.). |
| Event description | / | Optional | string | / | User-defined description of the attribute function, with a maximum of 100 characters. |
| Event title | / | Optional | string | / | Phrase summarizing the event content |
| Output parameter | / | Optional | array | / | Parameters returned after an event is generated. The number of parameters cannot exceed 50. |
| Event type | / | Optional | enum | / | Analyze information, alarms, and faults. The information is a general notification reported by the terminal device. Alarms and faults are sudden or abnormal events reported during the running of the equipment. |
| Event feature list | characteristics | Optional | / | / | Lists all the features supported by the event type. It contains mandatory features and optional features. For the technical requirements for describing a feature, see the definitions in Table 2. |

* + 1. Service field

A service is a logical or physical unit that represents a component of a terminal device and consists of a set of features. It mainly describes the information fields that need to be covered by a type of service. For the definitions of the service fields, see Table 3.

Table 3 Service field definition

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | English name | Mandatory or not | Data type | Maximum length | Description |
| Service type ID. | serviceType | Mandatory | String | 64 | Uniquely identifies the service type in the city awareness system. |
| Service instance ID. | serviceId | Mandatory | String | 32 | This parameter uniquely identifies a service instance within a specific device object. When accessing the device, the external system may specify the service instance to which the feature information to be read or modified belongs in the packet. |
| Service feature list | characteristics | Mandatory | / | / | Lists all the features supported by the service type. It contains mandatory and optional features. |

* + 1. Feature field

A service or event is a basic bearer unit that presents information or reflects specific capabilities. See Table 4 for the definition of the feature field.

Table 4 Feature field definition

|  |  |  |  |
| --- | --- | --- | --- |
| Field | English name | Mandatory or not | Description |
| Feature name | characteristicName | Mandatory | This field is represented by a character string. The maximum length is 128 bytes. Unique identifier of the feature in the entire thing model feature set. |
| Feature data type | characteristicType | Mandatory | This data type is represented by a string. The value contains a maximum of 32 bytes. For example, an integer value is represented by int, and an enumerated value is represented by enum. A list of existing data types is shown in Table 3. |
| Read/Write flag | method | Mandatory | This field represents whether the feature value can be read or modified during device interoperability. Currently, the value range is as follows: "R": The feature value is read-only. "W": The eigenvalue is writable only. RW: readable and writable. |
| Maximum | max | Optional | This field is valid only when the feature data type is numeric, for example, int or float. |
| Minimum value | min | Optional | This field is valid only when the feature data type is numeric, for example, int or float. |
| Step value | Step | Optional | This field is valid only when the feature data type is numeric, for example, int or float. Indicates the minimum change amount when the eigenvalue changes. |
| Number of decimal places | decimalDigits | Optional | This field is valid only when the feature data type is float. Indicates the number of decimal places, that is, precision. |
| Maximum length | maxLength | Optional | This field is valid only when the feature data type is string. Indicates the maximum length of a character string. |
| Enumerated value list | enumList | Optional | This field is valid only when the feature data type is enumerated or numeric. Enumerate the possible values. Table 4 describes the structure of the Enumlist element. |
| Unit | unit | Optional | Description of numerical units, such as ℃, cm, kg, etc. This parameter is valid only when the data type is numeric. See Table 5 for the current selectable values. |

* 1. Feature list
     1. Base feature list

A list of basic features is shown in Table 5.

Table 5 List of basic features

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Feature name | Feature data type | Read/Write flag | Maximum | Minimum value | Step value | Number of decimal places | Maximum length | Enumerated value list | Value unit | Feature description |
| productionDate | string | R | / | / | / | / | 128 | / | / | Equipment manufacturing date |
| brand | enum | R | / | / | / | / | / | To be coordinated and allocated by operation entities |  | Device brand name |
| deviceId | string | R | / | / | / | / | 128 | / | / | Unique ID allocated by the device after the device is registered. |
| ethMac | string | R/W | / | / | / | / | 32 | / | / | MAC address of the ethernet adapter |
| btMac | string | R/W | / | / | / | / | 32 | / | / | Bluetooth MAC address |
| WLANMac | string | R/W | / | / | / | / | 32 | / | / | WLAN MAC address |
| os.version | string | R | / | / | / | / | 64 | / | / | Version number of the operating system software |
| os.type | enum | R | / | / | / | / | / | To be coordinated and allocated by the operation entity | / | Operating system type |
| hardware.version | string | R | / | / | / | / | 64 | / | / | Hardware version (version number of the main board) |
| firmware.version | string | R | / | / | / | / | 64 | / | / | Firmware version number (system software version of the embedded device) |
| SerialNumber | string | R | / | / | / | / | 128 | / | / | Device SN |
| on | enum | R/W | / | / | / | / | / | 0: disabled  1: enabled | / | Indicates the opposite action or state of "on" and "off". It can refer to the on and off of a device, or the power-on and power-off status of a switch component on the device |
| restart | enum | R/W | / | / | / | / | / | 0: restart; 1: hibernate | / | Indicates that a device or system is restarted or hibernated |
| peerConnectStatus | enum | R | / | / | / | / | / | 0: The device is online; 1: The device is offline. | / | Indicates the current network status of the device. Check whether the device is online |
| latitude | string | R | / | / | / | / | 20 | / | / | Dimension |
| longitude | string | R | / | / | / | / | 20 | / | / | Longitude |
| RSSI | int32 | R | / | / | / | / | / | / | dbm | Wireless signal strength |
| SSID | string | R | / | / | / | / | 50 | / | / | Network access point name |
| ipv4Addr | string | R | / | / | / | / | 20 | / | / | IPv4 address of the device |
| ipv6Addr | string | R | / | / | / | / | 20 | / | / | IPv6 address of the device |
| fault.code | Enum | R | / | / | / | / | / | 0: normalXX: to be determinedCurrently, the value can be expanded | / | Device abnormal status code |
| fault.message | String | R | / | / | / | / | 512 | / | / | Device exception description |
| cpu.model | string | R | / | / | / | / | 128 | / | / | CPU model |
| cpu.manufacturer | string | R | / | / | / | / | 64 | / | / | CPU manufacturer |
| cpu.currentUsage | float | R | 100 | 0 | / | 1 | / | / | % | CPU usage |
| cpu.usageThreshold | float | R/W | 100 | 0 | / | 1 | / | / | % | Maximum CPU usage |

* + 1. Service feature list

A list of service characteristics is shown in Table 6.

Table 6 Service feature list

| Service type ID. | Type description | Feature list | Feature mandatory or not |
| --- | --- | --- | --- |
| devInfo | Device information | productionDate | Optional |
| brand | Optional |
| deviceID | Optional |
| serialNumber | Optional |
| peerConnectStatus | Optional |
| ethInfo | Ethernet service information | ethMac | Mandatory |
| ipv4Addr | Optional |
| ipv6Addr | Optional |
| WLANInfo | WLAN service information | WLANMac | Mandatory |
| ipv4Addr | Optional |
| ipv6Addr | Optional |
| RSSI | Optional |
| SSID | Optional |
| btInfo | Information about the bluetooth service | btMac | Mandatory |
| OS | Basic OS information | version | Mandatory |
| type | Optional |
| CPU | Basic CPU information | manufacturer | Mandatory |
| model | Mandatory |
| currentUsage | Optional |
| usageThreshold | Optional |
| GPS | GPS service information | latitude | Mandatory |
| longitude | Mandatory |
| switch | Switch service | on | Mandatory |
| restart | Device restart service | restart | Mandatory |

* + 1. Event feature list

A list of event characteristics is provided in Table 7.

Table 7 Event characteristics

|  |  |  |  |
| --- | --- | --- | --- |
| Event type ID. | Type description | Feature list | Feature mandatory or not |
| faultEvt | Fault reporting event | fault.code | Mandatory |
| fault.message | Optional |

* 1. Scalability

Extended capability is a function description of the characteristic or differentiated capability of the terminal. For example, for devices with high-precision data collection capabilities, the extended description mechanism may specify key information such as precision indicators and application scopes. For the equipment that works in special environments, the environment adaptability parameters can be accurately described. For the differentiated capabilities of terminal devices, the device manufacturer needs to add the prefix character string "cust." before the feature name, service type ID, or event type ID defined by the device manufacturer for explicit declaration, regardless of the feature, service, or event type ID.

Using the terminal device identification nameplate as an example, "file sharing" of the terminal device belongs to the extended capability. A service list for the extended capability is shown in Table 8, and a feature list is shown in Table 9.

Table 8 File sharing extended capability service list

|  |  |  |  |
| --- | --- | --- | --- |
| Service type ID. | Type description | Feature list | Feature mandatory or not |
| cust.shareFile | Describes the file sharing capability | cust.shareFileList | Mandatory |
| cust.sendFile | Mandatory |

Table 9 Features of file sharing extension capabilities

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Feature name | Feature data type | Read/Write flag | Maximum | Minimum value | Step value | Number of decimal places | Maximum length | Enumerated value list | Value unit | Feature description |
| cust.shareFileList | string | R/W | / | / | / | / | 1000 | / | / | Shared file list |
| cust.sendFile | enum | R/W | / | / | / | / | / | 0: Enable sharing  1: stop sharing | / | Instructions for controlling the file sharing process |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_