



## Custom Functions

Configure in Platform > Function Definition

Version: 20200729

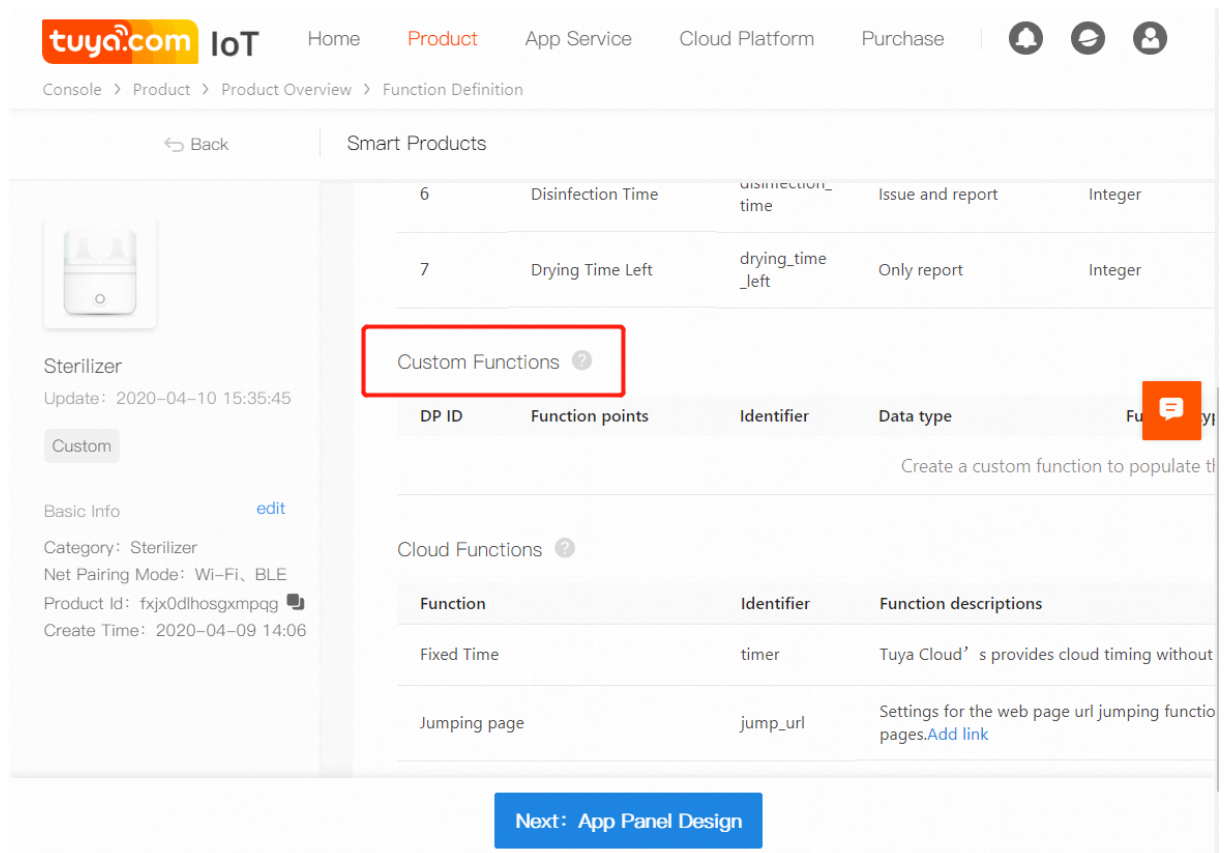
## Contents

<b>1 Overview</b>	<b>1</b>
<b>2 Basic concepts</b>	<b>2</b>
<b>3 Description</b>	<b>3</b>
3.1 Boolean . . . . .	3
3.2 Integer . . . . .	3
3.3 Enum . . . . .	4
3.4 Fault . . . . .	6
3.5 Char type . . . . .	6
3.6 RAW type . . . . .	7

# 1 Overview

Custom function: allows you to create functions that are not supported by standard function.

Data point (DP) is the abstract representation of function, and each function can be defined by different function types. Six data types are available now, namely **Boolean, Integer, Enum, Fault, Char type** and **RAW type**.



- **Boolean, Integer** and **Enum** are the three basic function types, and most of the functions can be defined by them.
- **Fault** is specified for fault statistics in the backend.
- **Char type** and **RAW type** are the advanced types for defining complicated functions. These two types are recommended when the other four types cannot meet your needs.

More details about defining data points are described as follows.

## 2 Basic concepts

**Data point:** represents functions of smart devices and parameters.

**DP ID:** indicates code of a data point. Subsequent function data transfer between the device and the cloud is through data point ID.

**Name:** indicates name of a custom function.

**Identifier:** indicates code value of a data point, displaying function name in multi-language on the app. Support letters, digits and underlines, starting with a letter.

**Data type:**

- **Boolean:** indicates binary variable value (true or false). For instance: switching function (On or Off).
- **Integer:** represents data that can be linearly adjusted. For instance: temperature adjustment, ranging from 20 to 40 °C.
- **Enum:** indicates custom finite set value. For instance: working level (low, middle and high).
- **Fault:** specified for fault report and statistics. Support **Only report** when multiple faults occur.
- **Char type:** indicates data transfer in char type.
- **RAW type:** indicates data transfer in a raw binary format. Generally, it is recommended to use raw data transfer when the other five types cannot cope with the complicated product functions.

**Data transfer type:**

- **Issue and report:** indicates two-way data transfer between the device and the cloud.
- **Only report:** indicates one-way data transfer. Data can only be reported from the device.
- **Only issue:** indicates one-way data transfer. Data can only be issued by the cloud.

## 3 Description

### 3.1 Boolean

Application: represents binary variable value (true or false).

The screenshot shows a web interface for creating a custom function. The dialog box is titled "Newly create custom function" and contains the following information:

- A warning message: "The custom function does not support third-party voice and cloud interconnection. [Submit a work order](#), TuYa will create a standard function as soon as possible. [How Do I Customize Functions?](#)"
- \* DP ID: 101
- \* name: Switch
- \* identifier: Power1
- \* Data type:  Boolean  Integer  Enum  Fault  Char type  RAW type  ?
- \* Data transfer type:  Issue and report  Only report  Only issue  ?
- Notes: (empty text area)
- Buttons: Cancel, Confirm

Instance: device switch, child lock on heating device, fan head, resetting filter cartridge for air purifier and so on.

### 3.2 Integer

Application: represents data that can be linearly adjusted.

The custom function does not support third-party voice and cloud interconnection. [Submit a work order](#), TuYa will create a standard function as soon as possible. [How Do I Customize Functions?](#)

\* DP ID:

\* name:

\* identifier:

\* Data type:  Boolean  Integer  Enum  Fault  
 Char type  RAW type 

\* Values range:  -

\* Pitch:

\* Scale:  

Unit:

\* Data transfer type:  Issue and report  Only report  Only issue 

Notes:

#### Instance:

- For the scenario that temperature ranges from 20 to 40 °C with each change of 1 °C, it indicates the adjustable temperature is between 20 to 40 °C, and each operation triggers change of 1 °C.
- For the scenario that a device with a 12-hour timer is set to turn on or off automatically at a specified time, you can set the reservation from 1 hour to 12 hours with the interval of 1 hour.

If you have any further questions, refer to [Integer FAQ](#).

### 3.3 Enum

Application: represents custom finite set value.

Newly create custom function

The custom function does not support third-party voice and cloud interconnection. [Submit a work order](#), TuYa will create a standard function as soon as possible. [How Do I Customize Functions?](#)

\* DP ID:

\* name:

\* identifier:

\* Data type:  Boolean  Integer  Enum  Fault  
 Char type  RAW type

\* Enumerated values:

\* Data transfer type:  Issue and report  Only report  Only issue

Notes:

#### Instance:

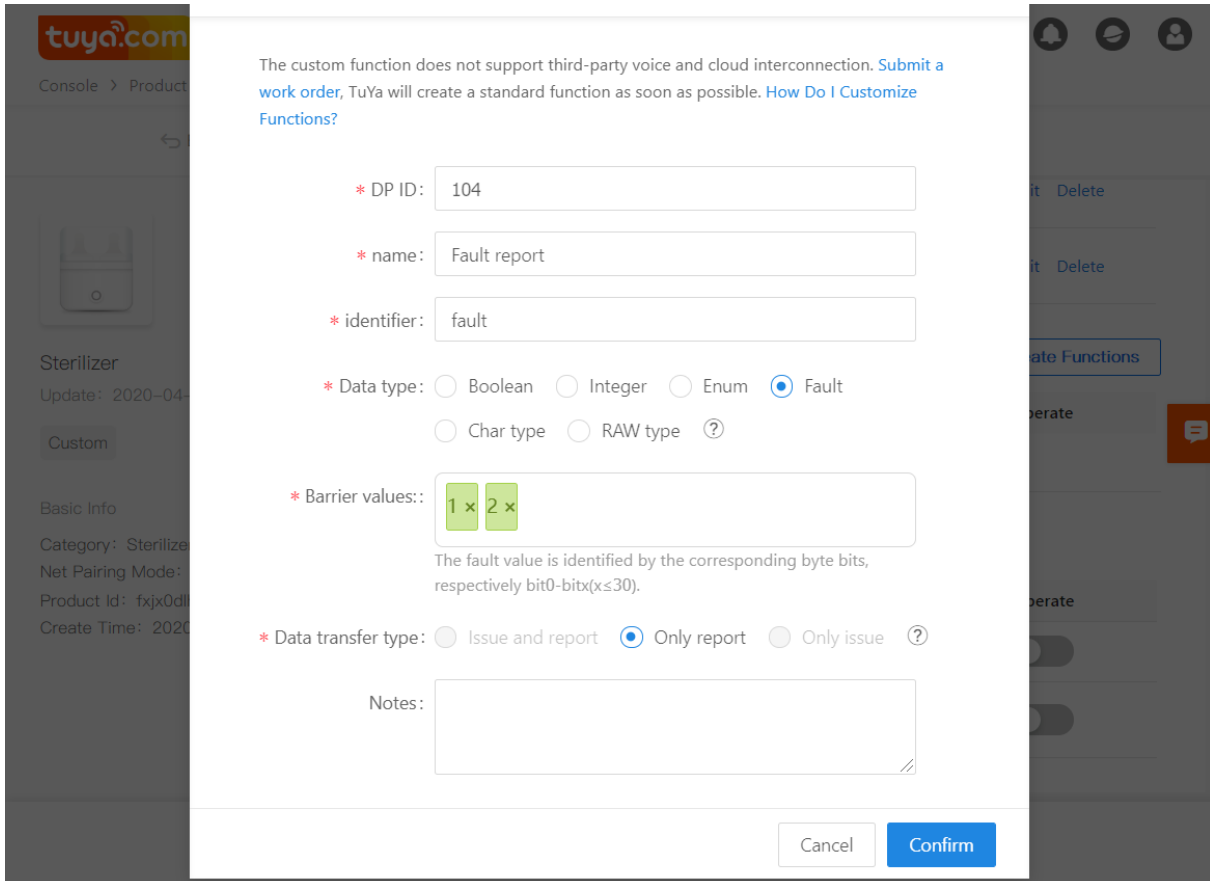
- Working level: low, middle and high.
- Modes: comfort, sleep, smart, and power save.
- Colors: red, blue, green and black.

Note: when editing function points, enum value supports letters, digits and underlines. After you press the Enter key, the enum value will be automatically generated. Code of enum value starts from 0, and data is transferred in enum code during hardware integration testing. The length of each enum value cannot exceed 15 characters, and a maximum of 10 enum values can be defined. Each time you enter an enum value, press the Enter key to generate a value.

If you have any further questions, refer to [Enum FAQ](#).

### 3.4 Fault

Application: specified for fault report and statistics. Later, this function will be used for statistics.



Instance:

- Temperature sensor fault, motor fault and high-temperature fault.

Note: fault data is in the format of bitmap, so defining multiple faults is available. The fault only supports data report, so just **Only report** is available in data transfer type.

If any further questions, refer to [Fault FAQ](#).

### 3.5 Char type

Application: indicates data transfer in char type. If the product definition cannot be completed by above four function types, you can choose char type as the data



transfer type.

Newly create custom function

The custom function does not support third-party voice and cloud interconnection. [Submit a work order](#), TuYa will create a standard function as soon as possible. [How Do I Customize Functions?](#)

\* DP ID:

\* name:

\* identifier:

\* Data type:  Boolean  Integer  Enum  Fault  
 Char type  RAW type 

Maximum length: No over 255 bytes

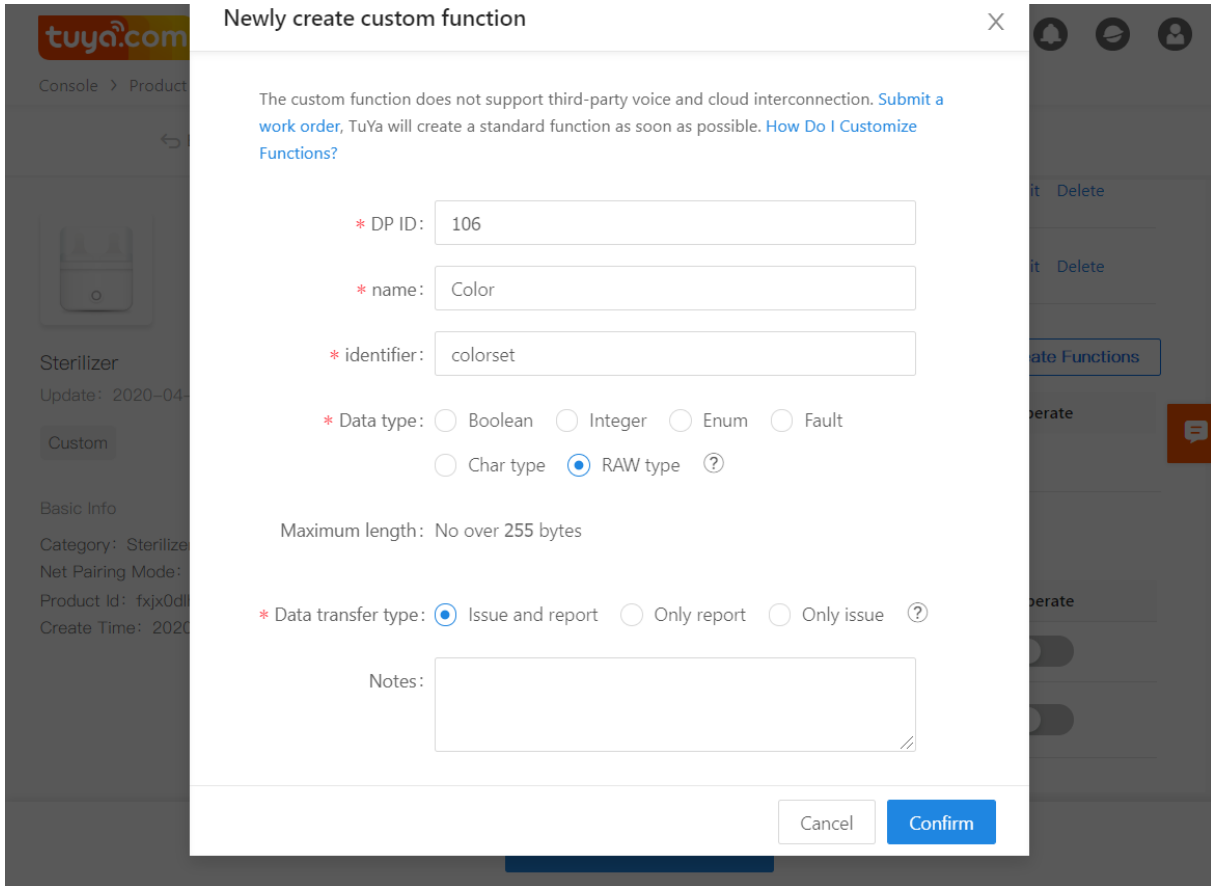
\* Data transfer type:  Issue and report  Only report  Only issue 

Notes:

Note: the length of a single char type cannot exceed 255 bytes.

### 3.6 RAW type

Application: indicates data transfer in a raw binary format. Generally, it is recommended to use raw data transfer when the other five types cannot cope with the complicated product functions.



Note: the length of raw data cannot exceed 255 bytes.

At present, it is recommended that the number of standard data points and custom data points for each product should not exceed 40. If the total number exceeds the maximum, test the product stability.

For more FAQ about function definitions, refer to [Function Definitions FAQ](#).