Data Rule Introduce

DIVOOM Pixoo64 is an intelligent device used for displaying user-related information such as nickname, likes count, followers count, level, and score. To allow users to customize the displayed content, we have designed a rule system to extract data from server responses and display it on the DIVOOM Pixoo64 device based on user configurations.There are two types of parsing rules, Normal and Custom, which are introduced separately:

1： Normal:

It will use a fixed data return method, and the JSON data returned by network requests is:
Eg:https://app.divoom-gz.com/Device/ReturnDispData?UserId=300000001&Flag=1


```
{
   "AppName": "Divoom",
   "DispData": [{
         "AppTitle": "FANS",
         "AppData": "56789",
      },
      {
         "AppTitle": "SCORE",
         "AppData": "89K",
      },
      {
         "AppTitle": "WORKS",
         "AppData": "89K",
      }
   ]
}
```


It will be composed of string type AppName and array type DispData, where each member of DispData must include string type AppTitle and AppData; but AppTitle only need with the display item named “the network first 'Title ' data from url”.

二：Custom

1. Introduction

This document provides a detailed explanation of how to use the DIVOOM Pixoo64 rule system, including rule format, object hierarchy, data types, and the correct interpretation of "Display Format." We will illustrate these concepts using the following example response data:
Eg:https://app.divoom-gz.com/Device/ReturnDispData?UserId=300000001

```json
{
    "ReturnCode": 0,
    "ReturnMessage": "",
    "Nickname": "Best Game",
    "HeadId": "group1/M00/05/2A/L1ghbmGdopOEP45jAAAAAIB05eo4227145",
    "DispData": {
        "LikeCnt": 519947,
        "FansCnt": 14654,
        "UserInfo": {
            "Level": 21,
            "Score": 603141
        },
        "IncreaseScoreList":[
            {
            "Date": "2023-02-10",
            "AddScore": 300
            },
            {
            "Date": "2023-02-11",
            "AddScore": 400
            },
            {
            "Date": "2023-02-12",
            "AddScore": 200
            },
            {
            "Date": "2023-02-13",
            "AddScore": 280
            },
            {
            "Date": "2023-02-14",
            "AddScore": 513
            },
            {
```

```
                "Date": "2023-02-15",
                "AddScore": 218
                },
                {
                "Date": "2023-02-16",
                "AddScore": 141
                }
            ]

    }
}```

This is some basic data information for the user.
Nickname is user’name;
DispData will include some important information.
LikeCnt is the total number of likes;
FansCnt is the total number of friends;
Level is user’s level;
Score is the total points of the user;
IncreaseScoreList is the user's score breakdown for the past seven days.

# # 2. Rule Format

A rule is a string composed of several parts separated by commas. The format for each rule is as follows:

```
Object Hierarchy, Data Type:Display Format
```

- `Object Hierarchy`: Represents the hierarchy of the desired data within the response. Each level is separated by a comma. If arrays are involved, they can be represented as `[IncreaseScoreList:x-y]`, indicating that all elements from the Xth to Yth position in the array should be accumulated.
- `Data Type`: Specifies the data type, including numeric ("n") , float(“f”) and string ("s").
- `Display Format`: Specifies the final data element to be displayed on the device from the extracted response data.

3. Object Hierarchy and Array Handling
```

The object hierarchy is the first part of the rule, used to specify the path for extracting data from the response. Each level is separated by a comma, representing the hierarchy. For arrays, `[IncreaseScoreList:x-y]` can be used to indicate that all elements from the Xth to Yth position should be accumulated. For example:

```
We want to obtain the user's points for the past three days
DispData,[IncreaseScoreList:0-2],n:AddScore will get 900(300+400+200)

We want to obtain the user's points for the last day
DispData,[IncreaseScoreList:0-0],n:AddScore will get 300

We want to obtain the user's points from 2023-02-13 to 2023-02-15
DispData,[IncreaseScoreList:3-5],n:AddScore will get 1011(280+513+218)

```

In this example, the rule accumulates all data elements from the 0th to 2nd positions within the dispData object in the response and displays them on the device.

4. Data Types

Data types are the second part of the rule, specifying the type of data to be extracted. Currently, two data types are supported:

- `"n"`: Numeric data type, e.g., likes count, level, score, etc.
- `"s"`: String data type, e.g., nickname.
- `"f"`: float data type;

5. Display Format

The display format is the third part of the rule, specifying where the data should be displayed on the device. For example:

```
n:LikeCnt
```

In this example, the rule specifies that the numeric data extracted from the response ("LikeCnt") should be displayed as the final data element.

6. Example Rules

Here are some example rules for better understanding:

- `"Nickname": "s:Nickname"`: Specifies that the response's nickname field should be displayed as the final data element.
- `"LikeCnt": "DispData,n:LikeCnt"`: Specifies that the likes count field within the dispData object should be displayed as the final data element.
- `"Level": "DispData,UserInfo,n:Level"`: Specifies that the level field within the UserInfo object of dispData should be displayed as the final data element.
- `"LastScore": "DispData,[IncreaseScoreList:0-2],n:AddScore"`: Specifies that all data elements from the 1st to 3rd positions within the DispData object should be displayed as the final data element.

7. Conclusion

DIVOOM Pixoo64 rules offer a powerful feature for users to customize displayed content based on their needs. By flexibly configuring rules, users can showcase personalized information and customize the device's style and content. We hope this document helps you understand and utilize the DIVOOM Pixoo64 rule system. If you have any questions or concerns, please feel free to contact our technical support team.

If you have any questions, please contact developer@divoom.com