



Cloud Node User Manual

BySeeedStudio

VERSION: 0.9

2013,May31

Content

1. Overview.....	3
2. Hardware.....	4
3. Sign-up a Xively Account	4
4. Configure Cloud Node	5
4.1 Sensor Settings	5
4.2 Wi-Fi Settings.....	6
5. Configure Atom's Usage.....	7
6. Example.....	8
6.1 Step 0 Sign-up Xively	9
6.2 Step 1: Configure Cloud Node	12
6.3 Step 2 Configure Atom Node.....	15
6.4 Step 3 Check the Device List.....	17
7. Reference Links.....	18

1. Overview

CloudNode is an Internet of Things platform which is based on the Arduino, it includes sensors, wireless communications, gateways and internet services, to gives you the whole solution about IoT home and helps the user quickly build smart home network and achieve a certain logic control. It will be convenient to create a smart home system such as remote human monitoring, remote distance and temperature measurement, etc. by using the Cloud.Node platform.

Cloud Node communicate with Atom Node via Xbee compatible module, and communicate with internet via Wi-Fi, it will be very convenient to upload the signal data that collected by Atom to the cloud, or even take simple control to the Atom. This will be a convenient and easy-to –use intelligent Internet of Things solution.

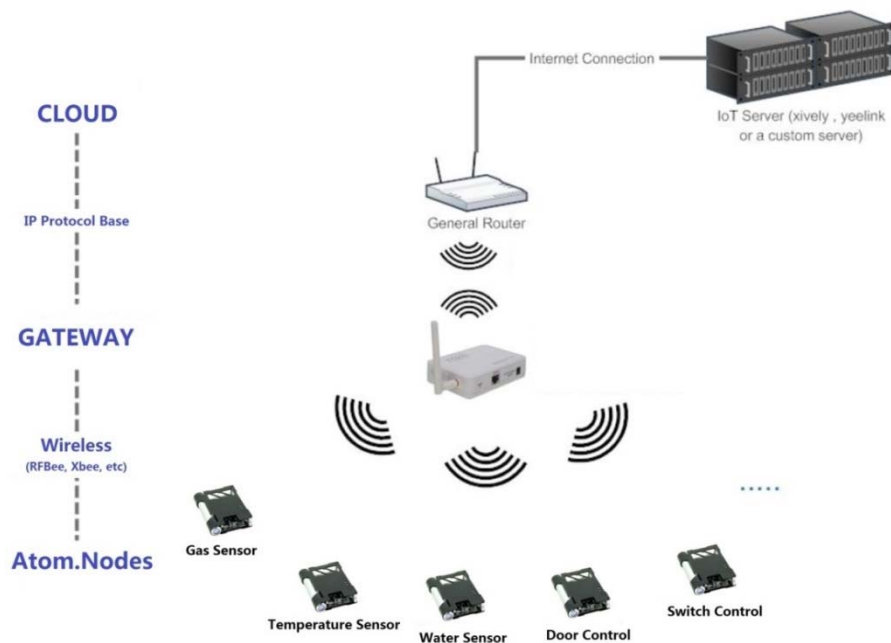


Figure 1-1 Node's working diagram

After reading the Cloud.Node user manual, you can:

- Get more information of Cloud Node and its working process;
- Use Cloud Node with Xively to remotely check the Node network's sensor values;

Next, we'll gradually introduce the instructions. After that, there is an example for you to get deeper understanding about how to make CloudNode up and running. **We recommended that you first read the first five chapters, and then follow the example in chapter six step by step to operate.**

2. Hardware



- Ethernet: Ethernet port, use for connect to PC
- Power: Power Input(9-12V)
- Wi-FiLED: Wi-Fi indicator
- Wan LED: Light indicate connected to PC
- Init LED: Initialize indicator, on indicate initialzing, off indicate initialized
- User: user led, had not been defined yet
- Pwr LED: power

3. Sign-up a Xively Account

Xively is a generic Internet of Things platform which mainly provides sensor data access, storage and display service. It provides a platform of Internet of Things project for all lovers of open source software and hardware, manufacturing enterprises, making them be able to deliver the electronic projects without caring about the implementation details, operation and maintenance of the server. So far, Xively is the major cloud data storage platform used in the Cloud Node.

4. Configure Cloud Node

Connect PC to Dragino with wire, power on it and wait for the Dragino's middle LED (which has a globe icon) going out.

Notice: This process needs about 2 minutes. Please wait!

Input *192.168.255.1*, and log on to Dragino configuration page. After setting the sensor and the inside content of Wi-Fi, the Cloud Node will up and running. If you are familiar with the content of the other options, it will bring you more convenience. But it's not necessary.

4.1 Sensor Settings

Click the option of Sensor:

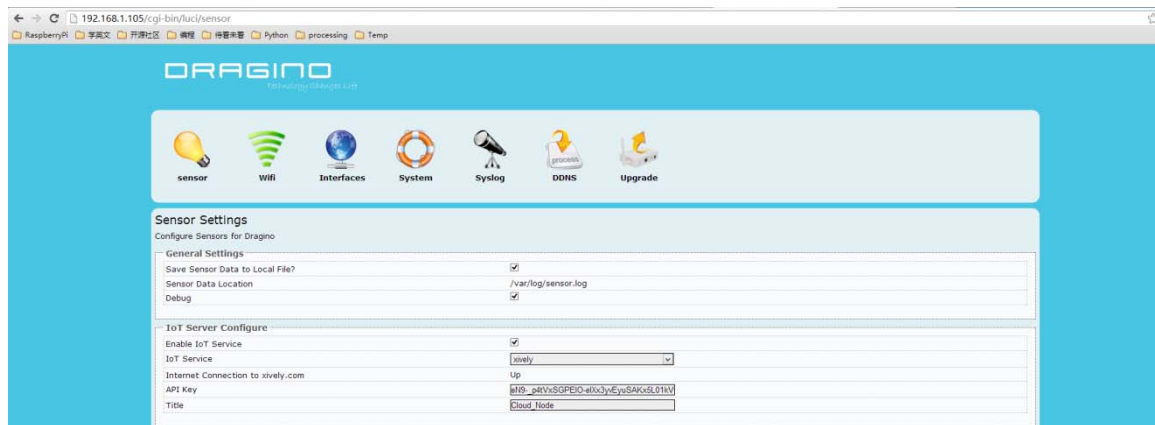


Figure 2-1 Configure Sensor

Main Settings:

- **Save Sensor Data to Local File:** whether save the sensor data to local file or not
- **Sensor Data Location:** choose this option means saving sensor data to */var/log/sensor.log*

IoT Server Configure:

- **Enable IoT Service:** whether open cloud service
- **IoT Service:** choose cloud server, Xively
- **API Key:** created while sign in that mentioned in the first chapter
- **Title:** prefix of the device name uploaded to cloud

Devices List	
Below devices are connected to Dragino	
<u>Device ID: 45</u>	
Sensor Name	Grove - Tilt Switch Sensor
Actuator Name	Grove - OLED 96x96
POST URL	http://api.yeelink.net/v1.0/device/3086/sensor/4337/datapoints
<u>Device ID: 2</u>	
Sensor Name	Grove - Light Sensor
Actuator Name	
POST URL	http://api.yeelink.net/v1.0/device/3091/sensor/4346/datapoints
<u>Device ID: 1</u>	
Sensor Name	Grove - Slide Potentiometer
Actuator Name	
POST URL	http://api.yeelink.net/v1.0/device/3171/sensor/4439/datapoints
<u>Device ID: 10</u>	
Sensor Name	Grove - Ultrasonic Ranger
Actuator Name	
POST URL	http://api.yeelink.net/v1.0/device/3201/sensor/4492/datapoints

Figure 2-2 Device List

Devices List:

Here is a list of the devices which has been added to AtomNode. From the list, we can see clearly that this Cloud has added 4 devices. And also, each device's Device ID, Sensor Name, Actuator Name and Post URL are clear. Post URL is Xively. So it doesn't need any operation.

Click the Save button for saving at the lower right corner, then, configure Wi-Fi.

Notice: All of the content in the Devices List do not require the users to edit. They are automatically added when Atom Node added.

4.2 Wi-Fi Settings

First of all, let's assume that the Cloud Node is working in a Wi-Fi environment, and, of course, you have got the Wi-Fi password. Click the Wi-Fi option, the following screen will pop up:

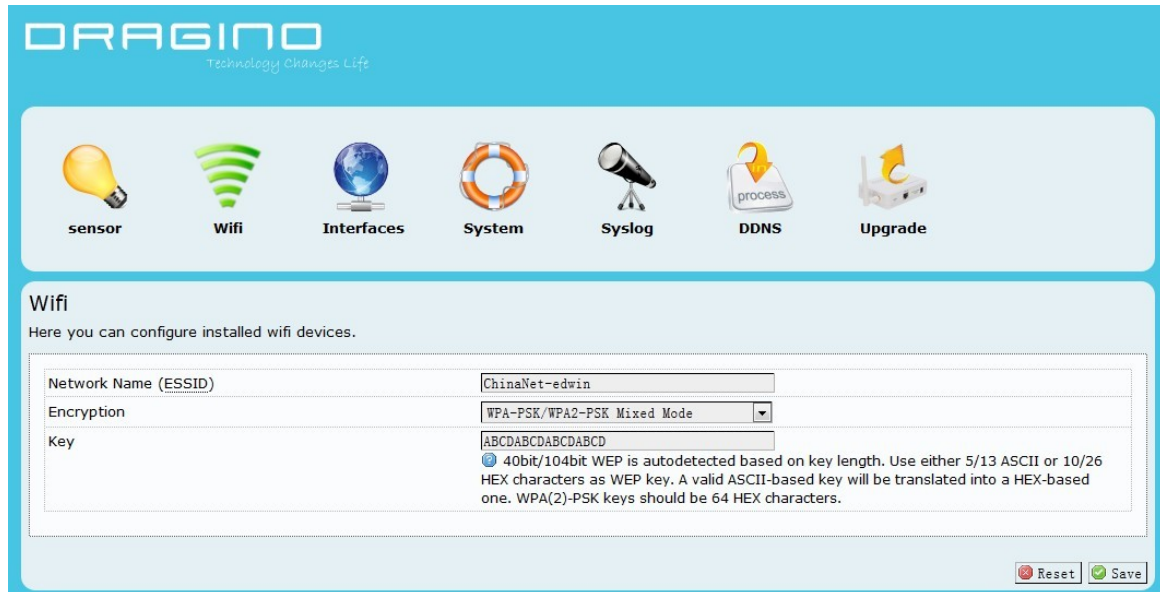


Figure 2-3 Wi-Fi Setting Page

Wi-Fi Settings:

- **Network Name(ESSID):** Input the SSID of Wi-Fi
- **Encryption:** Way of encryption
- **Key:** Password

Notice: After clicking the bottom right corner for saving, please wait for the page to refresh. It will be tens of seconds to 1 minute, please be patient.

After saving, you can see the outermost Wi-Fi led flashing. That is to say, you have successfully connected to Wi-Fi. Thus, Cloud Node configuration has completed.

5. Configure Atom's Usage

Now that you have completed the configuration of Cloud Node, it's the time for Atom. If you don't know how to use it, please click

here<http://www.seeedstudio.com/wiki/Atom_Node_V1.0>

When Atom is in work state, Cloud will automatically receive the sensor values of Atom and add a devices list in the web configuration page and Xively. And also, if there is data broadcasting, Cloud will push the data to Xively.

6. Example

To make it easier for you to use Cloud Node, here we will give you the detailed explanation from the beginning.

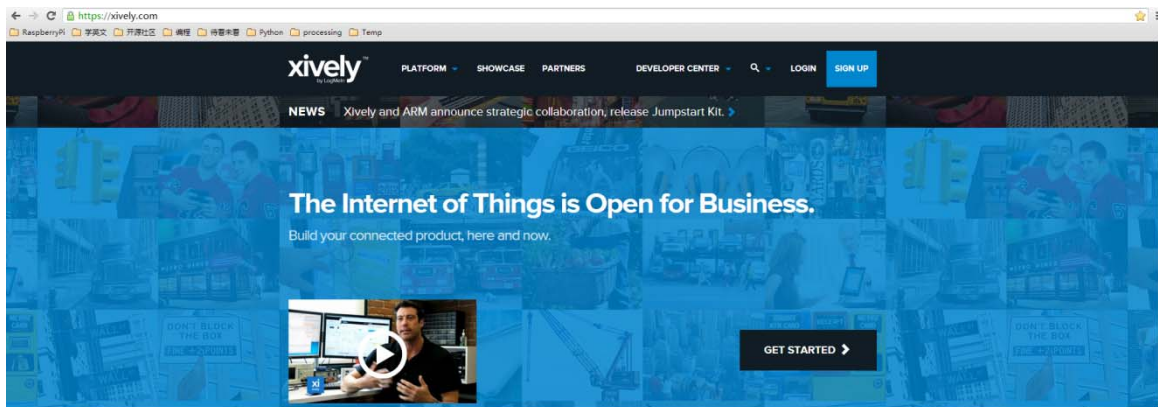
I want to know the outdoor light changes 24 hours a day so that I can probably get what time dawn and when to start getting dark. And you know, this is a very interesting process. You'll find that the change of nature is slow but sustained. And no matter how long the dark night is, the light will come.

Now, suppose that you have already been able to use the Atom Node, and already owned a Grove-Light Sensor. This morning, the CloudNode that you bought at Seeedstudio last week finally arrived. You open the package with full of hope, and deeply attracted by the packaging of fine. It is awesome! Therefore, you can't wait to make it work. Well, let's begin!

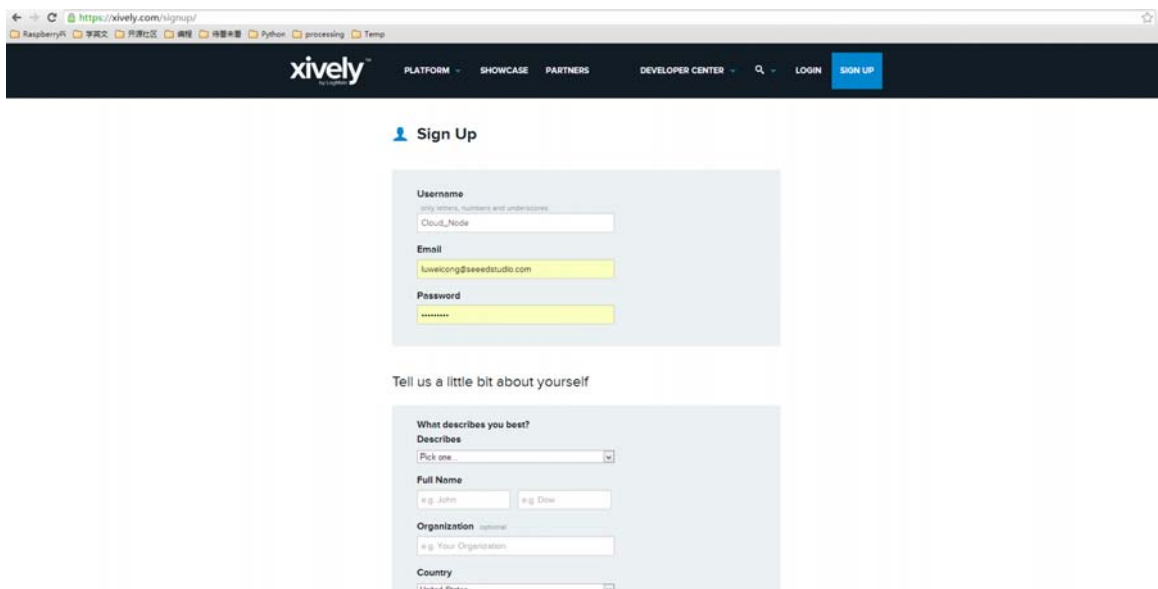


6.1 Step 0 Sign-up Xively

Open a web browser, type www.xively.com



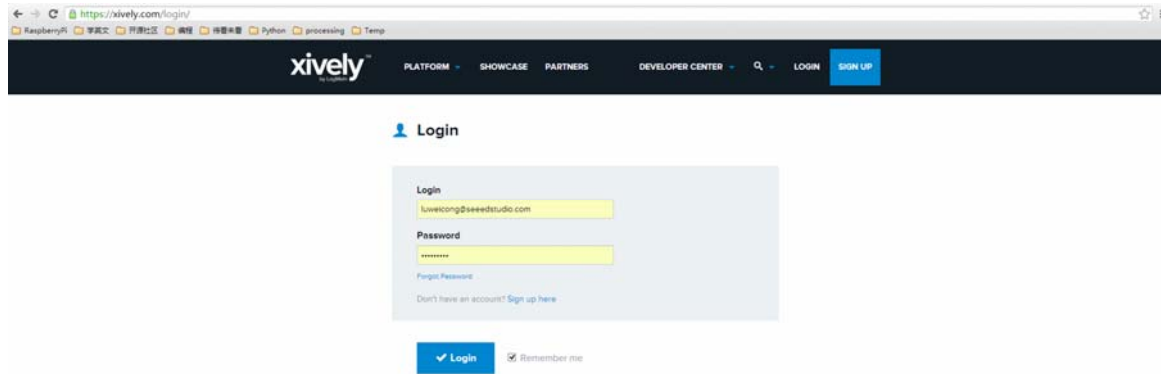
Click SIGN UP at the top right corner:



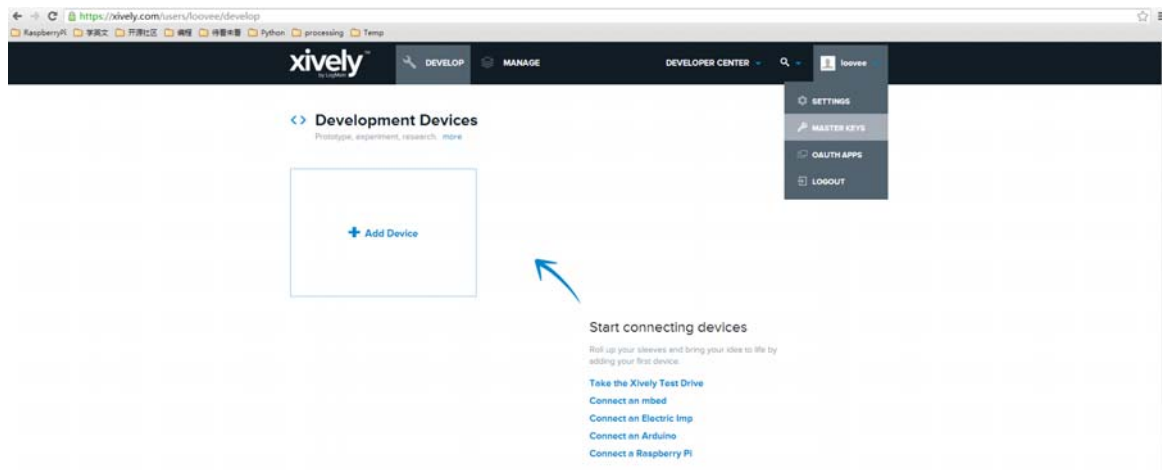
Fill in the relevant information to complete the registration.

After that, enter the email that you chose while registering. There is a letter from Xively. Click on the link inside to finish the registration.

Then, you need to log in www.xively.com again. Click the Login button, input the information you typed just now. Click Login again. Thus, you are on the page of Xively.



In this way, you have already completed the registration of Xively. Next, here we need an APIKey. Just move your mouse cursor to the top right corner of the user name, there will pop-up a drop-down box. What you need to do is click MASTER KEYS button.



Here will pop-up this:



Master API Keys

Master API Keys are account-level that can access any object.

Warning Master API Keys with private access permission can list and edit other keys as well as change their permissions (including giving them greater permission).

You can create device and channel specific keys in the Developer Workbench.

[+ Add Master Key](#)

Click [+ Add Master Key](#)



Master API Keys

Master API Keys are account-level that can access any object.

Warning Master API Keys with private access permission can list and edit other keys as well as change their permissions (including giving them greater permission).

You can create device and channel specific keys in the Developer Workbench.

Add Key

Label required

Permissions required

☒ Read☒ Create☒ Update☒ Delete

Private Feeds

☐ Access private feeds

[Add advanced restrictions](#)

✓ Add Key

Cancel

Fill in a name that you like, choose any options you like under Permissions. And finally, click Add Key.



Master API Keys

Master API Keys are account-level that can access any object.

Warning Master API Keys with private access permission can list and edit other keys as well as change their permissions (including giving them greater permission).

You can create device and channel specific keys in the Developer Workbench.

Cloud_test

J7IkJHpr5m1Rqws7qMRTVjWBjQhKNwVIZaTAwqQf4AVakvyU

permissions READ,UPDATE,CREATE,DELETE

[+ Add Master Key](#)

Now, in this box, you can see that an API Key has been successfully added. Actually, it is a long string of weird characters J7IKJ..... It will be used later.

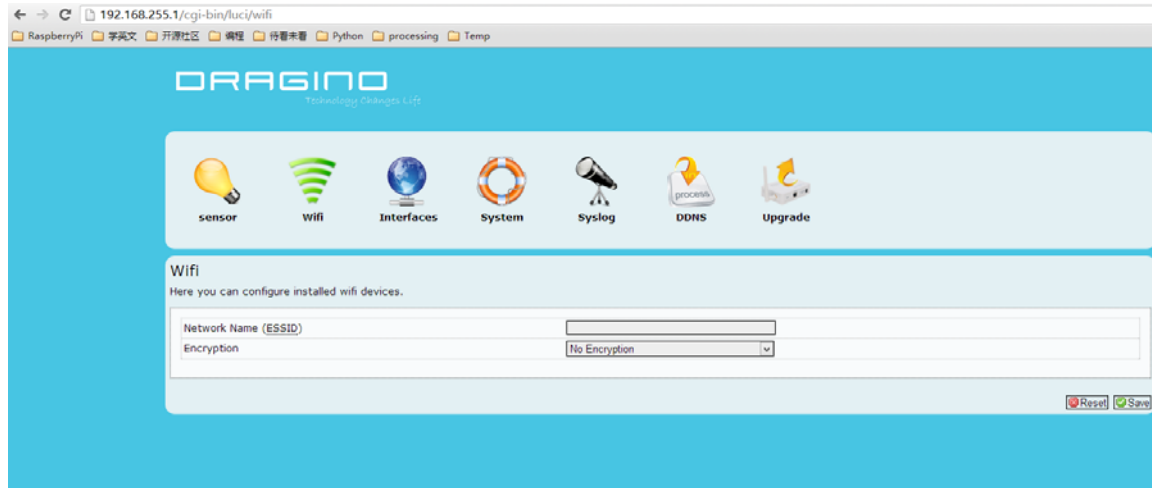
6.2 Step 1: Configure CloudNode

Connect Cloud Node to PC with a cable: just plug one end the cable into the computer's Ethernet port and the other end into the Cloud's Ethernet.

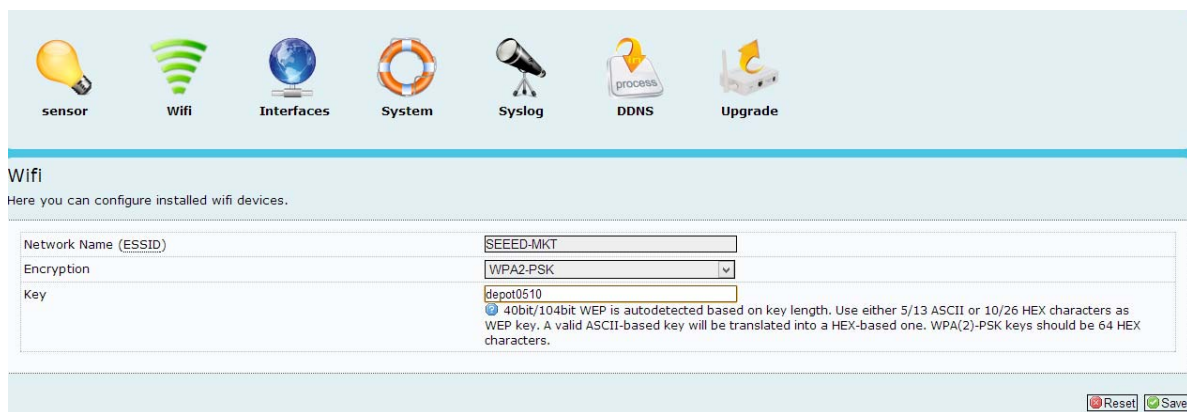
Power up Cloud Node and you can see the power LED to light. About 5 seconds later, the init LED will light. Then, about 1 minute later, the init LED will go out. That is to say, the initialization is completed. Now, you can proceed to the next step.

Note: When you are waiting for the init light go out, please leave out the other lights' flashing state.

After starting up the CloudNode, open the web browser and type 192.168.255.1, click enter button. You can log in to CloudNode configuration page.

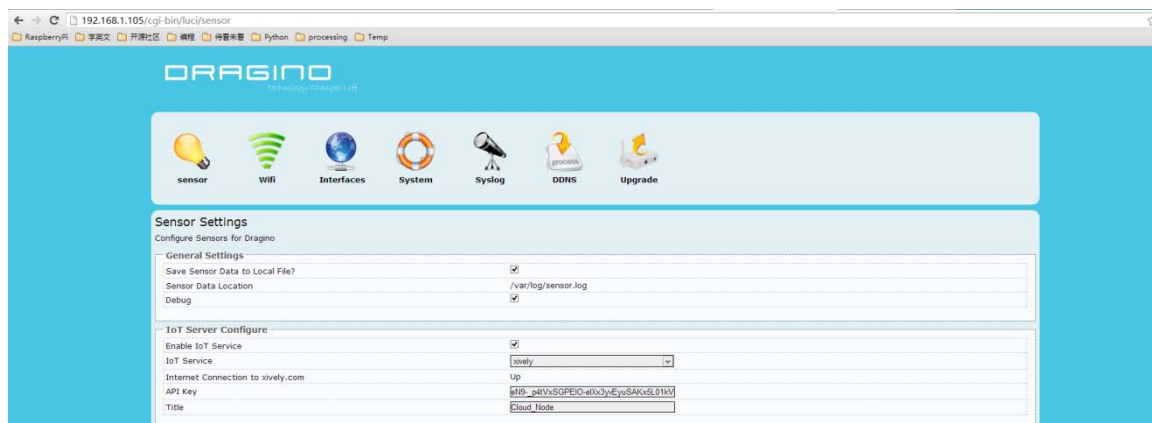


Type the Wi-Fi SSID in the column of Network Name(ESSID), select the encryption method and input Key. Finally, click Save.



Wait for the page to refresh. After that, try to log on www.google.com. Successfully login means that your configuration is success. Otherwise, please check whether you have inputted the information right or not.

Next, let's set Sensor. Please click the Sensor button.





You need to fill in the API Key and Title. Just copy the APIKEY that we have


mentioned in Step 0 to this page.


Fill in a name as you like to be the Title. Here we choose Cloud_Node. Later, you will use the name.


Click Save at the bottom right corner to complete the configuration, and wait for completing page refresh.



sensor



Wifi


Interfaces


System


Syslog


DDNS


Upgrade

Sensor Settings

Configure Sensors for Dragino

General Settings

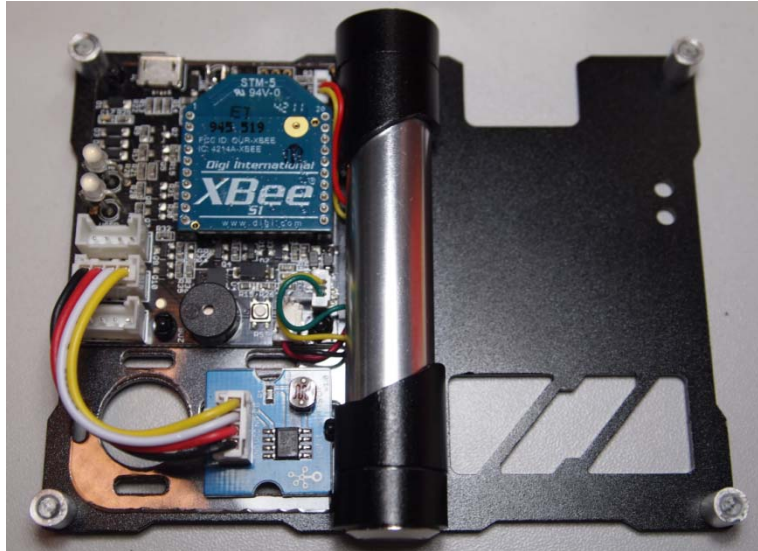
Save Sensor Data to Local File?	<input checked="" type="checkbox"/>
Sensor Data Location	/var/log/sensor.log
Debug	<input checked="" type="checkbox"/>

IoT Server Configure

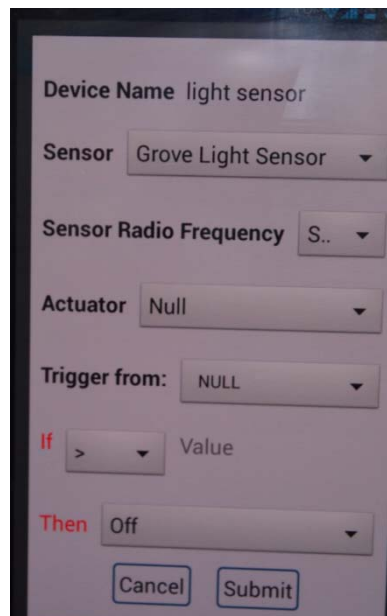
Enable IoT Service	<input checked="" type="checkbox"/>
IoT Service	xively
Internet Connection to xively.com	Up
API Key	eN9-_p4tVxSGPEIO-eIXx3yvEyuSAKx5L01kV
Title	Cloud_Node

6.3 Step 2 Configure AtomNode

Connect Grove-Light Sensor to Atom Node.



Open your mobile phone's APP to configure Atom. After the configuration, The Atom will broadcast the data every seconds.



After completing configuration, Atom Node started broadcasting data. While the Cloud Node receives the data via the Xbee compatible module, it will automatically create a device after receiving the data, and then, automatically push the data to Xively.

Put the configured Atom Node onto the balcony or window sill, and keep it under the outdoor light. That's all!

6.4 Step 3 Check the Device List

Enter the web page configuration end, click Sensor. And you will see that the Light Sensor's List has already existed in Cloud's Device List.

Sensor Settings

Configure Sensors for Dragino

General Settings

Save Sensor Data to Local File?	<input checked="" type="checkbox"/>
Sensor Data Location	/var/log/sensor.log
Debug	<input checked="" type="checkbox"/>

IoT Server Configure

Enable IoT Service	<input checked="" type="checkbox"/>
IoT Service	xively
Internet Connection to xively.com	Up
API Key	efN9-_p4tVxSGPEIO-elXx3yEyuSAKx5L01kV
Title	Cloud_Node

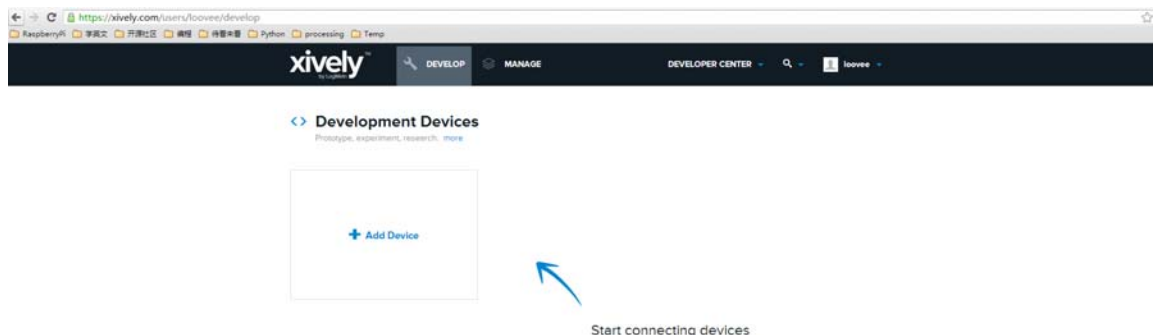
Devices List

Below devices are connected to Dragino

Device ID: 3

Sensor Name	Grove - Light Sensor
Actuator Name	
POST URL	https://api.xively.com/v2/feeds/938135853/datastreams/Grove-LightSensor/datapoints

At the same time, Xively has been added to this device. Now, switch to Xively interface. Then, click the DEVELOP button for starting connecting devices and adding device.




You can see this at the bottom of the page.

Legacy Feeds

You have a total of 2 feeds.

Title	Last Updated
Cloud_Node_3	2013-06-04 13:00:13

It means that Cloud.Node has already added this device to Xively. If you want to get the data inside, please click Cloud_Node_3.



DEVELOP
MANAGE

DEVELOPER CENTER
loovee

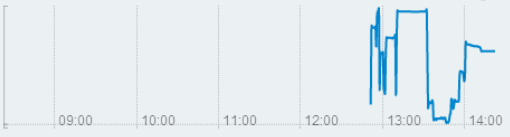
Cloud_Node_3
Public Feed

Feed ID: 938135853
Feed URL: <https://xively.com/feeds/938135853>
API Endpoint: <https://api.xively.com/v2/feeds/938135853>

Channels
Last updated an hour ago

Grove-LightSensor
432.00 Unit

Last updated 11 minutes ago



Edit Delete

+ Add Channel

Request Log
Pause

Waiting for requests
Your requests will appear here as soon as we get them, you can debug by clicking each individual request.

API Keys

+ Add Key

Thus, you can get the data that we uploaded just now. That is to say, your Cloud Node has begun working already. Here, we just give you a brief introduction about the usage of Xively. Actually, Xively itself function is very powerful. You can get a deeper understanding if you try to do something by yourself.

7. Reference Links

- www.seeedstudio.com : Dragrove vendor, more info about Dragrove and its development kit can be found here.
- www.openwrt.org: Embedded linux used in Dragino.
- wiki.dragino.com: General software/hardware design info for Dragino MS12
- www.xively.com: A public IoT RESTful server.
- www.yeelink.com: A public IoT RESTful server used in China.
- https://github.com/seeedstudio/Cloud_Dragrove: Daughter board firmware code
- https://github.com/seeedstudio/Cloud_Dragino_Firmware: Dragino Firmware