# 概述

Node是基于Arduino的物联网节点平台。该平台配合SeeedStudio的Grove电子模块系列和Xbee等通讯模块，用户可以快速地组建智能家居网络，并实现一定逻辑控制。使用Node平台来实现一个智能家居系统，例如远程人体监测报警、远程测距等，将变得非常方便和容易。

Node分为Atom.Node和Cloud.Node。

Atom主要用于局域网的通信。Atom通过Xbee或RFbee等无线模块进行相互通信，通信可靠，通讯距离高达100m；在配置上，用户使用SeeedStudio公司开发的基于Android平台的App，使用可见光通信的方式对Atom进行配置，配置过程简单直观；在使用上，Atom可以被单独使用来实现简单的IFTTT功能，也可以配对使用以实现远程无线控制和监控功能。

Cloud配备Xbee接口以及Wi-Fi功能，所以Cloud除了可以与Atom通信之外，也可以与互联网进行通信。配合Cloud，用户可以很方便的将Atom采集到的信号数据等上传至云端，或从云端对Atom进行简单控制。这将是一个很方便易用的智能物联网解决方案。

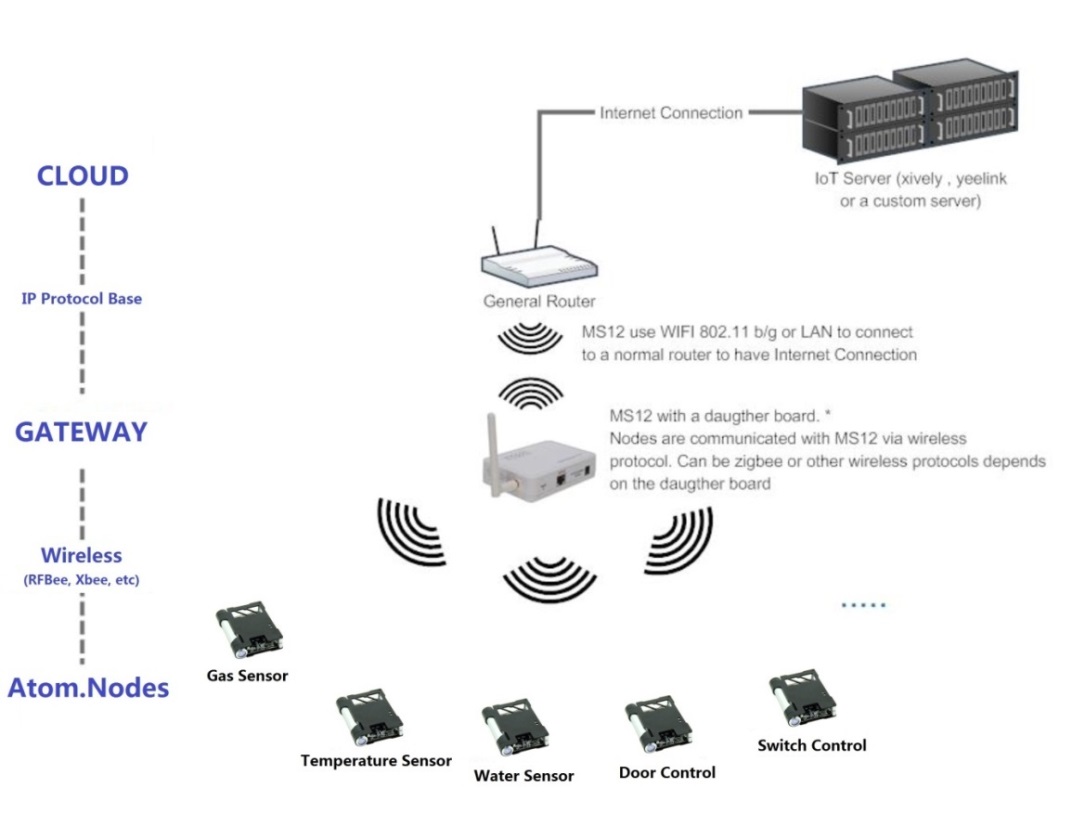


图 1‑1 Node工作示意图

通过阅读本手册，你可以：

* 了解Cloud.Node及其工作流程
* 使用Cloud.Node与Yeelink进行连接，可以远程查看Node网络的传感器数值

下面将会逐步的说明使用的方法，最后有一个Example，帮你更加深刻的理解如何让Cloud.Node运转起来。

# 配置Cloud.Node

电脑通过网线连接Dragino，上电，等待Dragino中间的提示灯（上面有个地球的）熄灭。

注：这个过程大概需要2分钟，请耐心等待

网页输入<192.168.255.1>，登录到Dragino配置页面。需要设置Sensor及Wi-Fi里面的内容即可让Cloud.Node运转起来。熟悉其他选项的内容可以为你带来方便，但不是必须的。

## 传感器设置

点击Sensor选项：

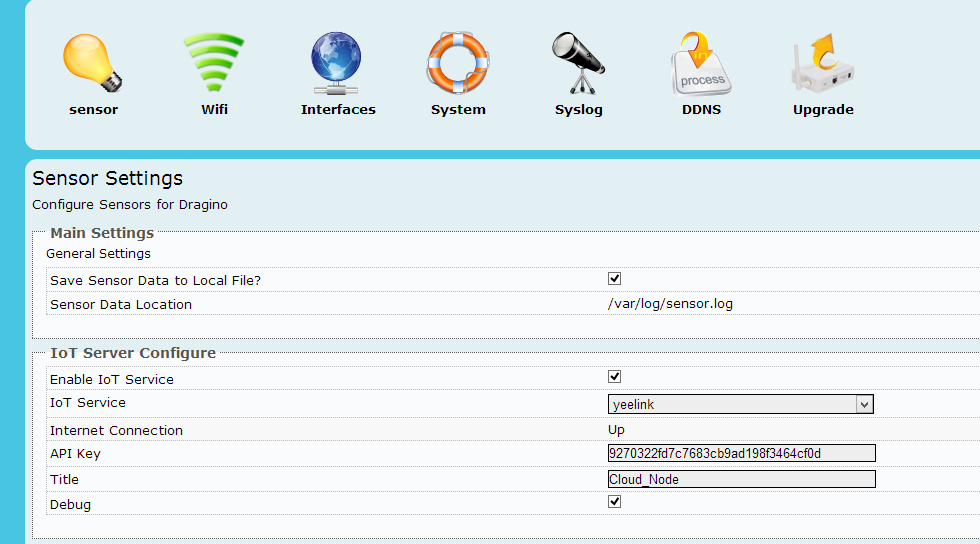


图 2‑1 配置传感器

**Main Settings**:

* **Save Sensor Data to Local File**:是否把传感器数值保存到本地文件
* **Sensor Data Location**: 如果勾选上面的选择，则传感器数据保存到/var/log/sensor.log

**IoT Server Configure:**

* **Enable IoT Service:** 是否开启云端服务
* **IoT Service:** 选择云端服务器，Yeelink
* **API Key:** 第一章提到的APIKEY，注册帐号时生成
* **Title:** 上传到云端的设备名称前缀

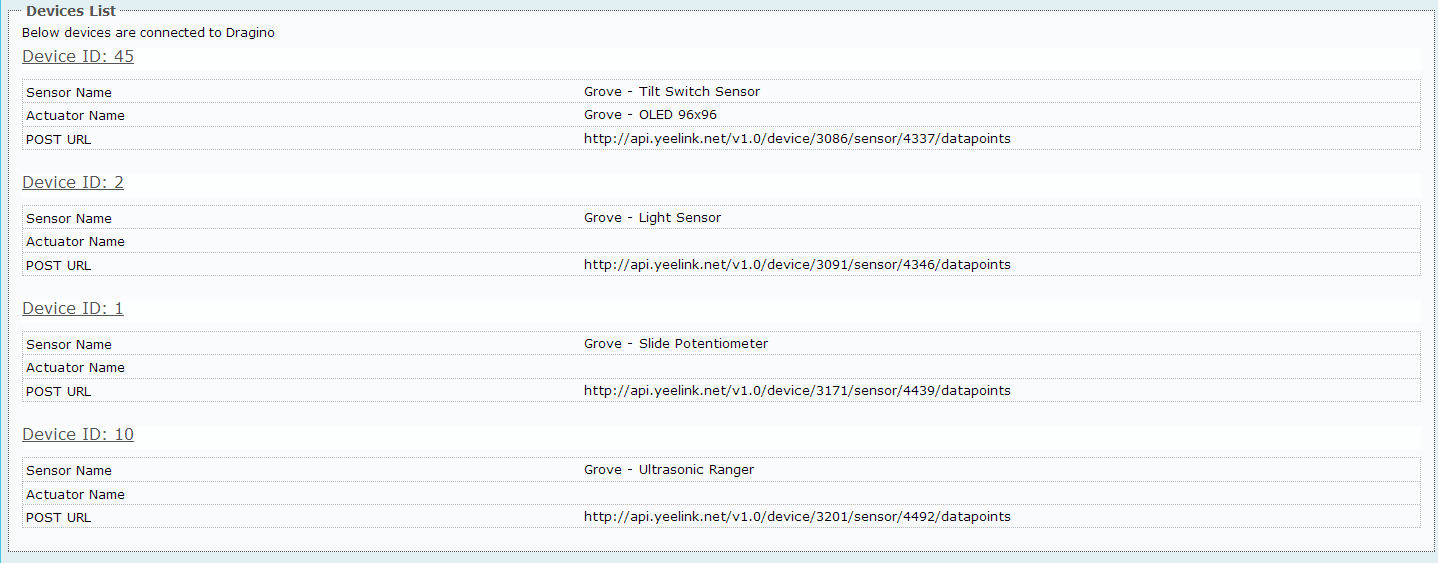


图 2‑2 Devices List

**Devices List:**

列出了已经添加的Atom.Node的列表，从列表可以看到，该Cloud已经添加了4个设备，可以看到每个设备的Device Id, Sensor Name, Actuator Name已经Post地址。Post地址是Yeelink返回的，我们不需要对它进行操作。

点击右下角的Save按钮保存，之后进行Wi-Fi的配置。

## Wi-Fi Settings

首先，假设Cloud.Node工作的环境有Wi-Fi网络存在，并且你知道Wi-Fi的密码。点击Wi-Fi选项，弹出以下界面：

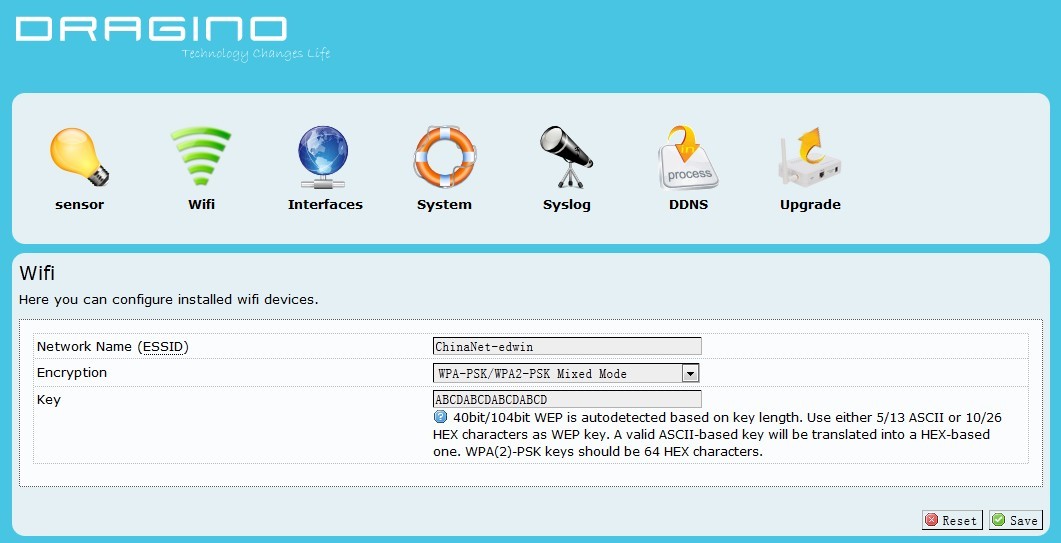
****

图 2‑3 Wi-Fi配置页面

**Wi-Fi Settings:**

* **Network Name(ESSID):** 输入Wi-Fi的SSID
* **Encryption:** 加密方式
* **Key:** 密码

注：点击右下角的保存后，等待页面刷新，这个时间可能要几十秒到1分钟,请耐心等待。

保存完成后，可以看到最外侧的Wi-Fi led闪烁，表示已经成功连上Wi-Fi。

至此，Cloud.Node的配置已经完成。

# 注册Yeelink账号

登录Yeelink网站，<www.yeelink.net> ，如果没有账户，点击右上角新用户注册，完成注册。如果已经有Yeelink账户，直接登录。

登录后看到以下页面：



点击账户->我的账户设置，弹出：



可以看到一些基本的账户信息，其中APIKEY在Dragino网页配置的时候需要用到。

# 配置使用Atom

Cloud.Node已经配置完毕，可以让Atom出场了，如果你不熟悉Atom的使用，可以参考：<http://www.seeedstudio.com/wiki/Atom_Node_V1.0>

Atom工作起来后，Cloud会自动接收Atom的传感器数值，并且在网页配置端以及yeelink上创建一个设备，并且把数据推送到Yeelink。

# 登陆Yeelink查看数据

如果你都顺利完成了以上几步，那么你应该可以到Yeelink上看到数据了。

# 一个使用例子

为了让你更容易地使用Cloud.Node，这里将会从头开始一步一步进行详细的讲解。

我想知道我家阳台一天24小时的光线情况，这样我就可以大概了解什么时候天亮，什么时候开始天黑了。这是个很有趣的过程，你将会发现，大自然的变化虽然很缓慢，但是变化却是持续的；黑夜虽然是漫长的，但光明终将到来。

假设你已经会使用Atom.Node，并且你还有一块Grove-Light Sensor。你上周到SeeedStudio买了一块Cloud.Node，今天上午终于到了，你满怀希望地打开包装，深深的为包装的精美所吸引，实在太漂亮了！于是你迫不及待地想让它立马工作起来。那么，开始吧。

**Step 0: 配置Cloud.Node**

参考第二章的内容

**Step 1: 注册Yeelink帐号**

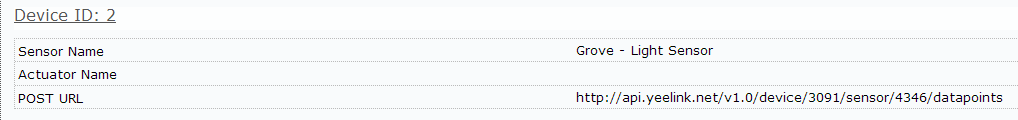
参考第三章内容

**Step 3：配置Atom.Node**

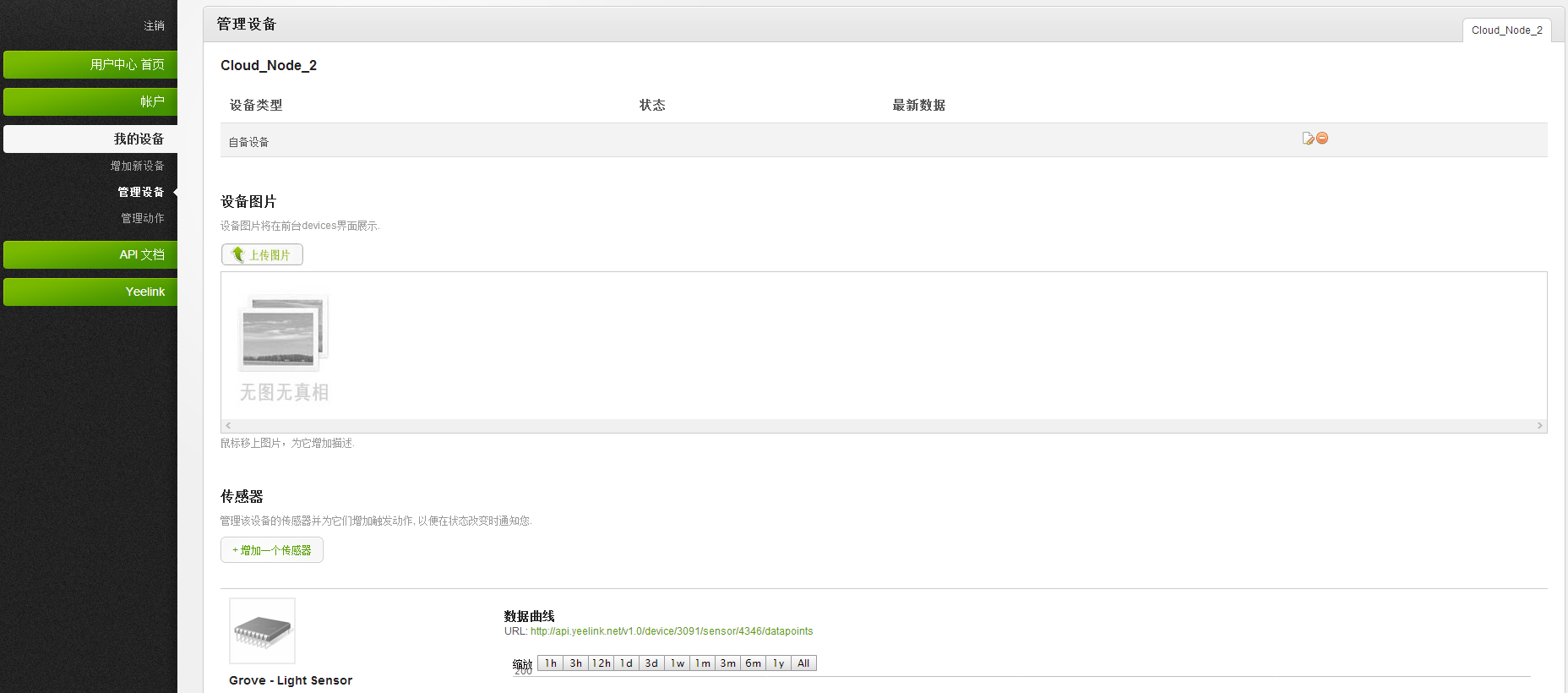
把Grove-Light Sensor接入Atom.Node, 对Atom.Node进行配置，每1s广播一次。配置完毕，Atom.Node开始广播数据，Cloud.Node内置Xbee接收到广播数据后，自动创建了一个设备，并且接收到传感器值后，自动把数据推送到Yeelink.

**Step 4：查看设备列表**

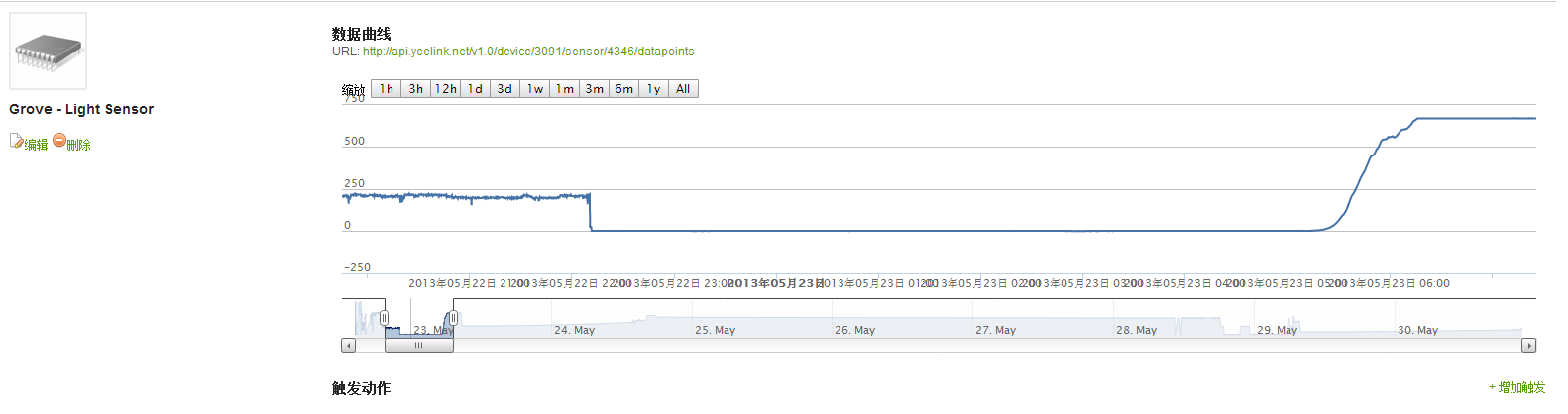
进入网页配置端，点sensor，你会看到，Cloud的Device List里面已经出现了Light Sensor的列表：



同时Yeelink上也添加了该设备，点击管理设备就可以看到多了一个Cloud\_Node\_2的设备，设备上有个Grove-Light Sensor，并且你可以看到一些刚才推送上来的数值了。



这标志着你的Cloud.Node已经初步工作起来了。回到最初的目的，我怎么知道太阳几点升起来？通常，夜里很暗，所以传感器的数字会比较小，太阳慢慢出来天慢慢的变量，传感器的数值也会逐渐变大，看数据吧：



这段数据是我放置在阳台的设备从2013-5-22晚上8点到2013-5-23早上7点的Light Sensor数值，可以看到晚上10点左右，我关灯睡觉了，传感器数值一下子变得很小，早上5点15之后，传感器数值慢慢变大，到了6点18分，升到了最大值。所以我猜想，日出在6点钟左右。

# 参考链接

* [www.seeedstudio.com](http://www.seeedstudio.com) : Dragrove vendor, more info about Dragrove and its development kit can be found here.
* [www.openwrt.org](http://www.openwrt.org): Embedded linux used in Dragino.
* [wiki.dragino.com](http://wiki.dragino.com/index.php?title=Main_Page): General software/hardware design info for Dragino MS12
* [www.xively.com](http://www.xively.com): A public IoT RESTful server.
* [www.yeelink.com](http://www.yeelink.com): A public IoT RESTful server used in China.
* <https://github.com/reeedstudio/Cloud_Dragrove>: Daughter board firmware code
* <https://github.com/reeedstudio/Cloud_Dragino_Firmware>： Dragino Firmware