

# 数据工会

# (HTTP://DATAGUILD.ORG/)

通过数据观察世界

首页 (HTTP://DATAGUILD.ORG/) | 技术 (HTTP://DATAGUILD.ORG/?PAGE\_ID=3681) | 商务 (HTTP://DATAGUILD.ORG/?PAGE\_ID=6888) | 关于 (HTTP://DATAGUILD.ORG/?PAGE\_ID=4671)

## MQTT入门篇 5 (http://dataguild.org/?p=6817#comments)

Posted on 2015年12月11日 (http://dataguild.org/?p=6817) by 张琪 (http://dataguild.org/?author=1)

物联网（Internet of Things, IoT）最近曝光率越来越高。虽然HTTP是网页的事实标准，不过机器之间（Machine-to-Machine, M2M）的大规模沟通需要不同的模式：之前的请求/回答（Request/Response）模式不再合适，取而代之的是发布/订阅（Publish/Subscribe）模式。这就是轻量级、可扩展的MQTT（Message Queuing Telemetry Transport）可以施展拳脚的舞台。

## MQTT简介

MQTT是基于二进制消息的发布/订阅编程模式的消息协议，最早由IBM提出的，如今已经成为OASIS规范。由于规范很简单，非常适合需要低功耗和网络带宽有限的IoT场景，比如：

- 遥感数据
- 汽车
- 智能家居
- 智慧城市
- 医疗医护

由于物联网的环境是非常特别的，所以MQTT遵循以下设计原则：

1. 精简，不添加可有可无的功能。
2. 发布/订阅（Pub/Sub）模式，方便消息在传感器之间传递。
3. 允许用户动态创建主题，零运维成本。
4. 把传输量降到最低以提高传输效率。
5. 把低带宽、高延迟、不稳定的网络等因素考虑在内。
6. 支持连续的会话控制。
7. 理解客户端计算能力可能很低。
8. 提供服务质量管理。
9. 假设数据不可知，不强求传输数据的类型与格式，保持灵活性。

运用MQTT协议，设备可以很方便地连接到物联网云服务，管理设备并处理数据，最后应用到各种业务场景，如下图所示：

关键词...

搜索

## 近期文章

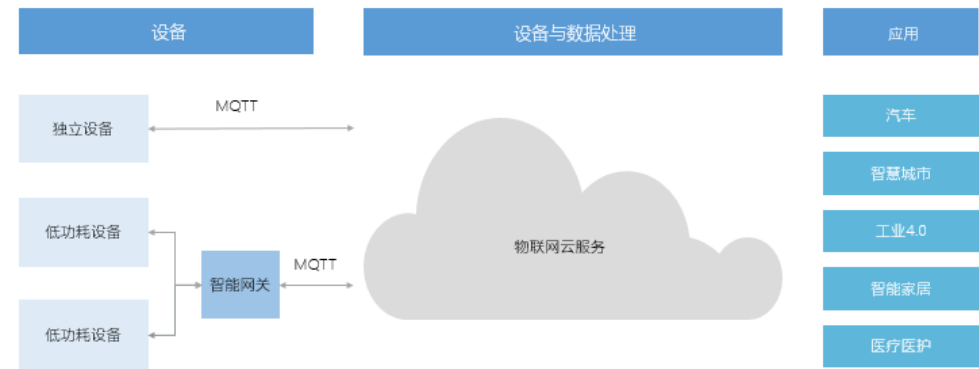
- 魔兽世界大数据分析 (http://dataguild.org/?p=7206)
- 为什么开源开放的大数据平台才能成功 (http://dataguild.org/?p=7190)
- WordPress日志分析 (http://dataguild.org/?p=7169)
- 讲故事的艺术 (http://dataguild.org/?p=7129)
- 谈谈OAuth（下） (http://dataguild.org/?p=7090)

## 近期评论

- 流行的爆碎点穴 – 数据工会 (http://dataguild.org/?p=6934)发表在《《从0到1》读书笔记 (http://dataguild.org/?p=6461#comment-320)》
- 魔兽世界大数据分析 – 数据工会 (http://dataguild.org/?p=7206)发表在《使用百度开放云分析网站日志 (http://dataguild.org/?p=6739#comment-275)》
- WordPress日志分析 – 数据工会 (http://dataguild.org/?p=7169)发表在《使用百度开放云分析网站日志 (http://dataguild.org/?p=6739#comment-259)》
- miumiuforyou发表在《MQTT实战篇 (http://dataguild.org/?p=6957#comment-251)》
- 春泥面包 (http://huntinux.github.com)发表在《MQTT实战篇 (http://dataguild.org/?p=6957#comment-250)》

## 文章归档

- 2016年九月 (http://dataguild.org/?m=201609)
- 2016年八月 (http://dataguild.org/?m=201608)
- 2016年七月 (http://dataguild.org/?m=201607)
- 2016年五月 (http://dataguild.org/?m=201605)
- 2016年四月 (http://dataguild.org/?m=201604)
- 2016年三月 (http://dataguild.org/?m=201603)
- 2016年二月 (http://dataguild.org/?m=201602)



(<http://dataguild.org/wp-content/uploads/2015/12/iot-mqtt-tutorial-01.png>)

## 发布/订阅模式

与请求/回答这种同步模式不同，发布/订阅模式解耦了发布消息的客户（发布者）与订阅消息的客户（订阅者）之间的关系，这意味着发布者和订阅者之间并不需要直接建立联系。打个比方，你打电话给朋友，一直要等到朋友接电话了才能够开始交流，是一个典型的同步请求/回答的场景；而给一个好友邮件列表发电子邮件就不一样，你发好电子邮件该干嘛干嘛，好友们到有空了去查看邮件就是了，是一个典型的异步发布/订阅的场景。

熟悉编程的同学一定非常熟悉这种设计模式了，因为它带来了这些好处：

- 发布者与订阅者不必了解彼此，只要认识同一个消息代理即可。
- 发布者和订阅者不需要交互，发布者无需等待订阅者确认而导致锁定。
- 发布者和订阅者不需要同时在线，可以自由选择时间来消费消息。

## 主题

MQTT是通过主题对消息进行分类的，本质上就是一个UTF-8的字符串，不过可以通过反斜杠表示多个层级关系。主题并不需要创建，直接使用就是了。

主题还可以通过通配符进行过滤。其中，+可以过滤一个层级，而#只能出现在主题最后表示过滤任意级别的层级。

举个例子：

- building-b/floor-5：代表B楼5层的设备。
- +/floor-5：代表任何一个楼的5层的设备。
- building-b/#：代表B楼所有的设备。

注意，MQTT允许使用通配符订阅主题，但是并不允许使用通配符广播。

## 服务质量

为了满足不同的场景，MQTT支持三种不同级别的服务质量（Quality of Service，QoS）为不同场景提供消息可靠性：

- 级别0：尽力而为。消息发送者会想尽办法发送消息，但是遇到意外并不会重试。
- 级别1：至少一次。消息接收者如果没有知会或者知会本身丢失，消息发送者会再次发送以保证消息接收者至少会收到一次，当然可能造成重复消息。
- 级别2：恰好一次。保证这种语义肯定会减少并发或者增加延时，不过丢失或者重复消息是不可接受的时候，级别2是最合适的。

服务质量是个老话题了。级别2所提供的不重不丢很多情况下是最理想的，不过往返多次的确认一定对并发和延迟带来影响。级别1提供的至少一次语义在日志处理这种场景下是完全OK的，所以像Kafka这类的系统利用这一特点减少确认从而大大提高了并发。级别0适合鸡肋数据场景，食之无味弃之可惜，就这么着吧。

## 消息类型

2016年一月 ( <a href="http://dataguild.org/?m=201601">http://dataguild.org/?m=201601</a> )
2015年十二月 ( <a href="http://dataguild.org/?m=201512">http://dataguild.org/?m=201512</a> )
2015年十一月 ( <a href="http://dataguild.org/?m=201511">http://dataguild.org/?m=201511</a> )
2015年八月 ( <a href="http://dataguild.org/?m=201508">http://dataguild.org/?m=201508</a> )
2015年七月 ( <a href="http://dataguild.org/?m=201507">http://dataguild.org/?m=201507</a> )
2015年六月 ( <a href="http://dataguild.org/?m=201506">http://dataguild.org/?m=201506</a> )
2015年五月 ( <a href="http://dataguild.org/?m=201505">http://dataguild.org/?m=201505</a> )
2015年二月 ( <a href="http://dataguild.org/?m=201502">http://dataguild.org/?m=201502</a> )
2015年一月 ( <a href="http://dataguild.org/?m=201501">http://dataguild.org/?m=201501</a> )
2014年十二月 ( <a href="http://dataguild.org/?m=201412">http://dataguild.org/?m=201412</a> )
2014年十一月 ( <a href="http://dataguild.org/?m=201411">http://dataguild.org/?m=201411</a> )
2014年九月 ( <a href="http://dataguild.org/?m=201409">http://dataguild.org/?m=201409</a> )
2014年八月 ( <a href="http://dataguild.org/?m=201408">http://dataguild.org/?m=201408</a> )
2014年七月 ( <a href="http://dataguild.org/?m=201407">http://dataguild.org/?m=201407</a> )
2014年六月 ( <a href="http://dataguild.org/?m=201406">http://dataguild.org/?m=201406</a> )
2014年四月 ( <a href="http://dataguild.org/?m=201404">http://dataguild.org/?m=201404</a> )
2013年十二月 ( <a href="http://dataguild.org/?m=201312">http://dataguild.org/?m=201312</a> )
2013年十一月 ( <a href="http://dataguild.org/?m=201311">http://dataguild.org/?m=201311</a> )
2013年十月 ( <a href="http://dataguild.org/?m=201310">http://dataguild.org/?m=201310</a> )
2013年九月 ( <a href="http://dataguild.org/?m=201309">http://dataguild.org/?m=201309</a> )
2013年八月 ( <a href="http://dataguild.org/?m=201308">http://dataguild.org/?m=201308</a> )
2013年六月 ( <a href="http://dataguild.org/?m=201306">http://dataguild.org/?m=201306</a> )
2013年五月 ( <a href="http://dataguild.org/?m=201305">http://dataguild.org/?m=201305</a> )

## 分类目录

云计算 ( <a href="http://dataguild.org/?cat=2">http://dataguild.org/?cat=2</a> )
产品经理 ( <a href="http://dataguild.org/?cat=8">http://dataguild.org/?cat=8</a> )
大数据 ( <a href="http://dataguild.org/?cat=3">http://dataguild.org/?cat=3</a> )
数据分析 ( <a href="http://dataguild.org/?cat=5">http://dataguild.org/?cat=5</a> )
物联网 ( <a href="http://dataguild.org/?cat=6">http://dataguild.org/?cat=6</a> )
软件工程 ( <a href="http://dataguild.org/?cat=7">http://dataguild.org/?cat=7</a> )

## 功能

登录 ( <a href="http://dataguild.org/wp-login.php">http://dataguild.org/wp-login.php</a> )
文章RSS (Really Simple Syndication) ( <a href="http://dataguild.org/?feed=rss2">http://dataguild.org/?feed=rss2</a> )
评论RSS (Really Simple Syndication) ( <a href="http://dataguild.org/?feed=comments-rss2">http://dataguild.org/?feed=comments-rss2</a> )
WordPress.org ( <a href="https://cn.wordpress.org/">https://cn.wordpress.org/</a> )

MQTT拥有14种不同的消息类型：

1. CONNECT：客户端连接到MQTT代理
2. CONNACK：连接确认
3. PUBLISH：新发布消息
4. PUBACK：新发布消息确认，是QoS 1给PUBLISH消息的回复
5. PUBREC：QoS 2消息流的第一部分，表示消息发布已记录
6. PUBREL：QoS 2消息流的第二部分，表示消息发布已释放
7. PUBCOMP：QoS 2消息流的第三部分，表示消息发布完成
8. SUBSCRIBE：客户端订阅某个主题
9. SUBACK：对于SUBSCRIBE消息的确认
10. UNSUBSCRIBE：客户端终止订阅的消息
11. UNSUBACK：对于UNSUBSCRIBE消息的确认
12. PINGREQ：心跳
13. PINGRESP：确认心跳
14. DISCONNECT：客户端终止连接前优雅地通知MQTT代理

后面我们会给出具体的例子。

## MQTT代理

市面上有相当多的高质量MQTT代理，其中mosquitto是一个开源的轻量级的C实现，完全兼容了MQTT 3.1和MQTT 3.1.1。下面我们就以mosquitto为例演示一下MQTT的使用。环境是百度开放云的云服务器以及Ubuntu 14.04.1 LTS，简单起见MQTT代理和客户端都安装在同一台云服务器上了。

首先SSH到云服务器，安装mosquitto以及搭配的客户：

```
1 apt-get install mosquitto
2 apt-get install mosquitto-clients
```

现在在云端模拟云服务，订阅某办公楼5层的温度作为主题：

```
1 mosquitto_sub -d -t 'floor-5/temperature'
2 Received CONNACK
3 Received SUBACK
4 Subscribed (mid: 1): 0
```

然后另外打开一个SSH连接，模拟温度计发送温度消息：

```
1 mosquitto_pub -d -t 'floor-5/temperature' -m '15'
2 Received CONNACK
3 Sending PUBLISH (d0, q0, r0, m1, 'floor-5/temperature', ... (2 bytes))
```

此时回到第一个SSH客户端可以看到信息已经接收到了，之后便是心跳消息：

```
1 Received PUBLISH (d0, q0, r0, m0, 'floor-5/temperature', ... (2 bytes))
2 15
3 Sending PINGREQ
4 Received PINGRESP
```

需要注意的是mosquitto客户端默认使用QoS 0，下面我们使用QoS 2订阅这个主题：

```
1 mosquitto_sub -d -q 2 -t 'floor-5/temperature'
2 Received CONNACK
3 Received SUBACK
4 Subscribed (mid: 1): 2
```

切换到另外SSH连接然后在这个主题里面发送温度消息：

```
1 mosquitto_pub -d -q 2 -t 'floor-5/temperature' -m '15'
2 Received CONNACK
3 Sending PUBLISH (d0, q2, r0, m1, 'floor-5/temperature', ... (2 bytes))
4 Received PUBREC (Mid: 1)
5 Sending PUBREL (Mid: 1)
6 Received PUBCOMP (Mid: 1)
```

此时回到第一个SSH客户端可以看到信息已经接收到了，以及相应的多次握手消息：

```
1 Received PUBLISH (d0, q2, r0, m1, 'floor-5/temperature', ... (2 bytes))
2 Sending PUBREC (Mid: 1)
3 Received PUBREL (Mid: 1)
```

至此我们初步了解了MQTT的基本知识，祝大家在物联网的世界里面玩得开心！

MQTT系列索引：

1. MQTT入门篇 (<http://dataguild.org/?p=6817>)
2. MQTT进阶篇 (<http://dataguild.org/?p=6846>)
3. MQTT安全篇 (<http://dataguild.org/?p=6866>)
4. MQTT实战篇 (<http://dataguild.org/?p=6957>)

 物联网 (<http://dataguild.org/?cat=6>)

 MQTT (<http://dataguild.org/?tag=mqtt>)



张琪

MORE POSTS ([HTTP://DATAGUILD.ORG/?AUTHOR=1](http://dataguild.org/?author=1))

< 说说参观的微软数据中心

([HTTP://DATAGUILD.ORG/?P=6768](http://dataguild.org/?p=6768))

MQTT进阶篇 >

([HTTP://DATAGUILD.ORG/?P=6846](http://dataguild.org/?p=6846))

## 5 COMMENTS

Pingback: 物联网技术周报第27期：自己动手制作智能手表-IT大道 (<http://www.itdadao.com/article/130300/>)

Pingback: 物联网技术周报第27期：自己动手制作智能手表-慢慢码(ManManMa) (<http://www.manmanma.com/archives/2125>)

Pingback: MQTT实战篇 – 数据工会 (<http://dataguild.org/?p=6957>)

Pingback: MQTT安全篇 – 数据工会 (<http://dataguild.org/?p=6866>)



2016年7月20 回复 ([HTTP://DATAGUILD.ORG/?P=6817&REPLYTOCOM=238#RESPOND](http://dataguild.org/?p=6817&replytocom=238#respond))

日

(<http://dataguild.org/?p=6817#comment-238>)

LJY2010A

关于通配符,貌似错了

/ is used to separate names in a path（分割路径）

+ is used to match any name in a path（通配地址任何字符）

# is used to recursively match path names（递归通配）

## 发表评论

电子邮件地址不会被公开。 必填项已用\*标注

评论

姓名 \*

电子邮件 \*

站 点

发表评论