


# Demostrador de Internet of Things con la tecnología IEEE 802.15.4e utilizando la plataforma OpenMote, sistema operativo OpenWSN y TheThings.io



Rubén Cabello Chacón

Master Universitario de Ingeniería de Telecomunicación UOC-URL

Enero 2018



# Índice

- ▶ Objetivos y motivación.
- ▶ Internet of Things
- ▶ Campos de aplicación IoT
- ▶ IEEE 802.15.4
- ▶ IEEE 802.15.4e
- ▶ Fog Computing
- ▶ Cloud Computing
- ▶ Cloud vs Fog
- ▶ OpenWSN
- ▶ Plataformas IoT
- ▶ Diseño e implementación

# Objetivos y motivación

- ▶ Simular una red Open-WSN donde se recoge información de distintos sensores y se envían a una Raspberry pi actuando como Gateway.
- ▶ 2 tipos de procesado: fog computing y cloud computing.
- ▶ Datos subidos a Cloud para su visualización.
- ▶ Comparación entre las dos clases de procesado.

# Objetivos y motivación

- ▶ Estudio sobre IoT.
- ▶ Conocimiento de tipos de procesado en IoT.
- ▶ IEEE 802.15.4 y IEEE 802.15.4e.
- ▶ Aprendizaje de OpenWSN.
- ▶ Conocer y utilizar plataformas Cloud.

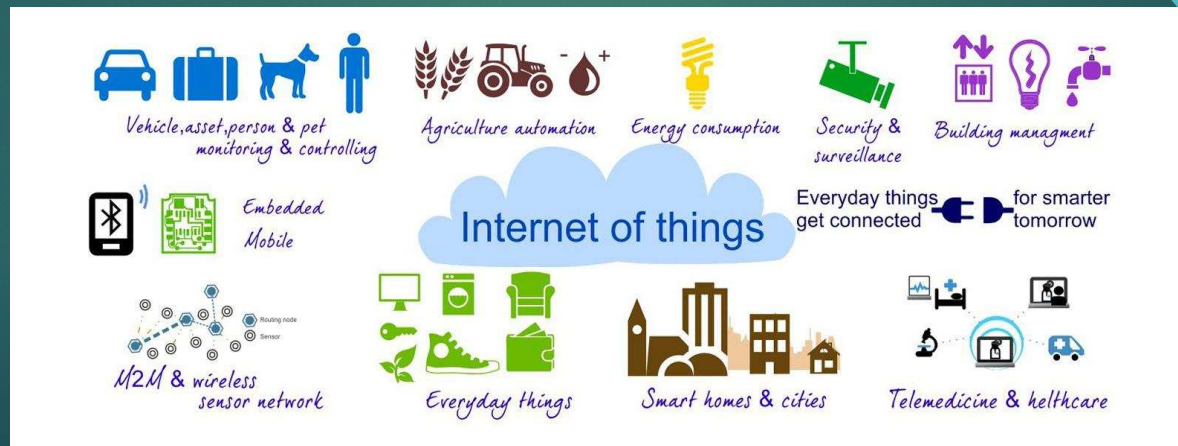
# Internet of Things (IoT)

- ▶ Concepto que se refiere a la interconexión digital de las cosas u objetos cotidianos con Internet.
- ▶ Protocolos: IEEE802.15.4e, IETF 6TiSCH, 6LoWPAN, CoaP, HTTP, etc.



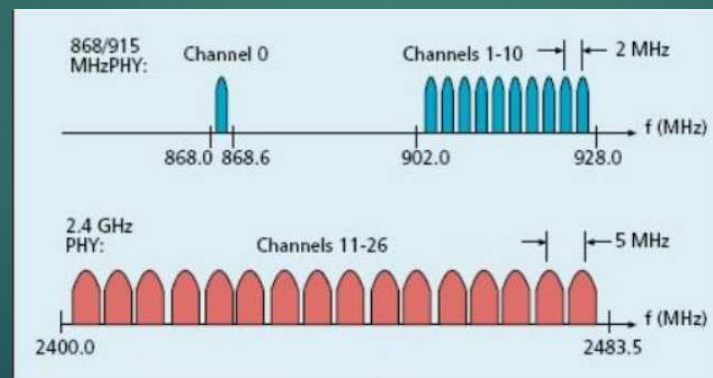
# Campos de aplicación IoT

- ▶ Smart home (sensores temperatura, humedad, automatización, etc.)
- ▶ Smart City (control meteorológico, contaminación, tráfico, etc.)
- ▶ Medicina y salud (marcapasos, control constantes vitales, etc.)
- ▶ Infraestructura (vías férreas, carreteras, parques eólicos, etc.)



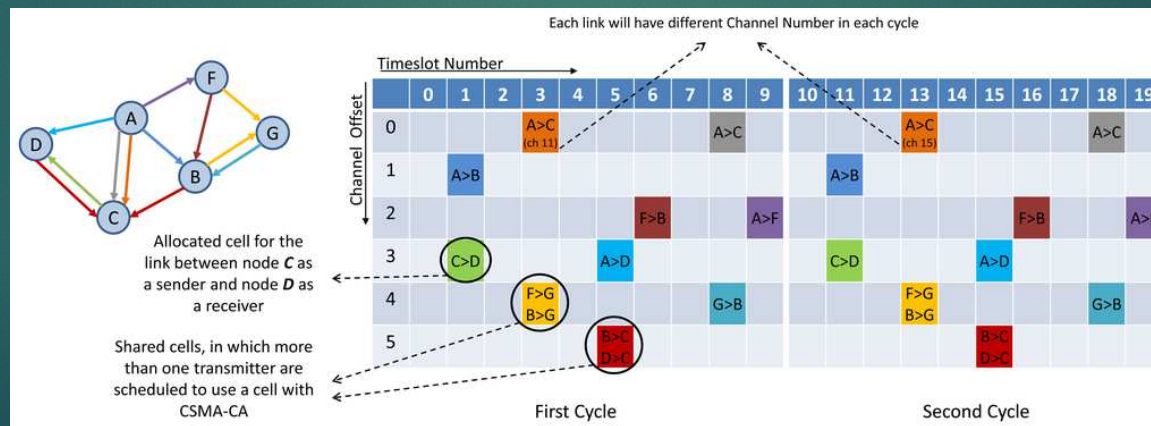
# IEEE 802.15.4

- ▶ Define capa física y capa de control acceso al medio.
- ▶ Bajo consumo de energía.
- ▶ Redes LR-WPAN (Low Rate Wireless Personal Area Network).



# IEEE 802.15.4e

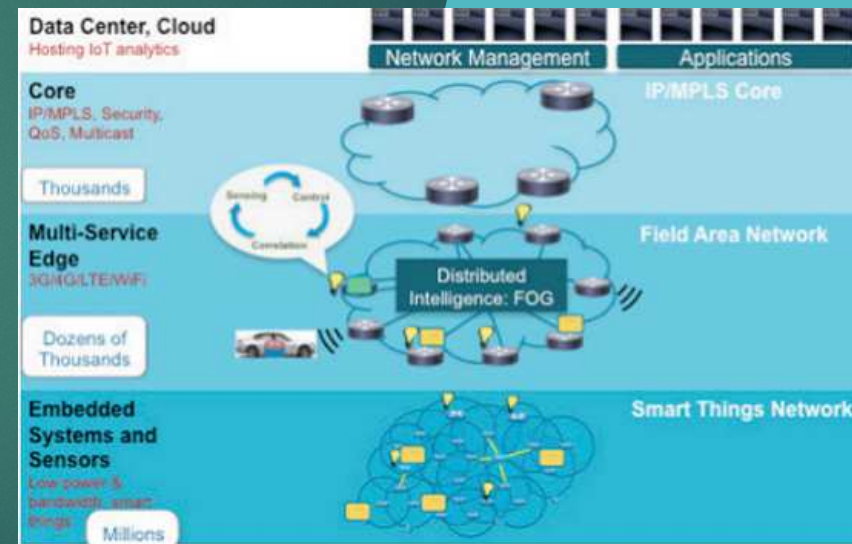
- ▶ Añade mejoras a IEEE 802.15.4.
- ▶ Mejor eficiencia energética, nodos intermedios no siempre activos.
- ▶ TSCH (Time slotted channel hopping), protección multicamino y fading.
- ▶ Protección contra interferencias externas.





# Fog Computing

- ▶ Datos tratados en el lugar mas eficiente entre origen y cloud.
- ▶ Mejorar eficiencia.
- ▶ Reducir consumo de Bw.
- ▶ Aumentar seguridad.
- ▶ Costes reducidos.



# Cloud Computing



- ▶ Servicios a través de Internet.
- ▶ 3 modelos:
  - ▶ SaaS (Software as a Service): aplicaciones en la nube (ofimática, e-mail, etc.)
  - ▶ PaaS (Platform as a Service): application engines, middleware, etc.
  - ▶ IaaS (Infrastructure as a Service): storage, backup, BBDD, etc.
- ▶ HW/SW bajo demanda.
- ▶ Escalabilidad.
- ▶ Flexibilidad.

# Comparativa

Requerimientos	Cloud Computing	Fog Computing
Latencia	Alta	Baja
Jitter	Alta	Muy bajo
Localización del servicio	Internet	Final red local
Distancia client/serve	Múltiples saltos	Uno
Seguridad	Sin definición	Puede ser definida
Ataque datos	Alta probabilidad	Poca probabilidad
Importancia localización	No	Si
Geo-distribución	Centralizada	Distribuida
Nº nodos	Pocos	Amplia
Movilidad	Limitada	Soportada
RTI	Soportado	Soportado
Última milla	Línea arrendada	Wireless

Cloud Computing	Fog Computing
Los datos y aplicaciones son procesados en el cloud, lo cual consume mucho tiempo para largas cadenas de datos.	Opera en el final de la red, tiene menor consumo de tiempo al no tener que ir a través de Internet.
Problema de ancho de banda, cada bit ha de ser enviado al cloud.	Menos demanda de ancho de banda, cada bit va a puntos de acceso.
Bajo tiempo de respuesta y problemas de escalabilidad al depender de servidores remotos.	Desplegando pequeños servidores bajo la visibilidad de los usuarios se evita delay y problemas de escalabilidad.

# OpenWSN



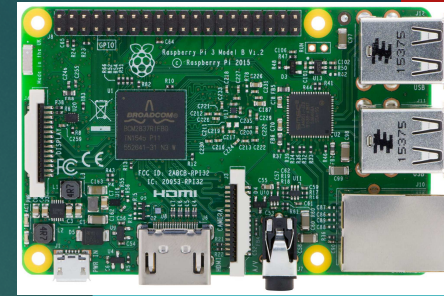
- ▶ Proyecto OpenSource para redes de sensores inalámbricas.
- ▶ IEEE802.15.4e, IETF 6TiSCH, 6LoWPAN, IETF ROLL, CoAP, HTTP, UDP, TCP...
- ▶ Open Visualizer: visualización y depuración (Web y CLI).
- ▶ OpemSim: simula una red ipv6 de motes.
- ▶ Compilación de programas a los motes en lenguaje C.

# Plataformas IoT

- ▶ Distintas clases dependiendo de los servicios aportados.
- ▶ Amazon Web Services (AWS): IaaS, SaaS, PaaS, noSql, HTTP, MQTP.
- ▶ Dweet.io: solo envío de mensajes.
- ▶ Freeboard: consulta dweet.io y realiza gráficas y dashboards.
- ▶ Thethings.io: API Python para conexión, Dashboard, scripting Jobs, Triggers.

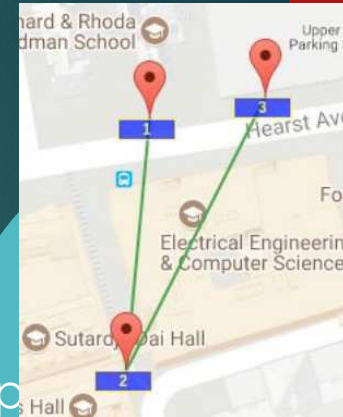
# Diseño e implementación

- ▶ Gateway: Raspberry Pi 3.
- ▶ Open-WSN, OpenSim y OpenVisualizer.
- ▶ SQL Mariadb (fog computing).
- ▶ API en Python para subida datos a Thethings.io.
- ▶ Scripts generación de datos aleatorios.



# Topología

- ▶ Red en estrella.
- ▶ 1 DAGroot, 1 mote interno, 1 mote externo.
- ▶ Ambos envían la información al DAGroot conectado a Raspberry pi.
- ▶ Raspberry pi actúa como Gateway de la red Ipv6 y como interconexión con la red WAN.



# BBDD

- ▶ Solo utilizada en procesamiento fog.
- ▶ Almacena la media del sensor, timestamp, tipo y localización.
- ▶ Guarda los datos históricos generados.

```
[MariaDB [(none)]> show columns from data from IOTDB;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
value	varchar(45)	YES		NULL	
name	varchar(45)	YES		NULL	
timestamp	varchar(45)	YES		NULL	
type	varchar(45)	YES		NULL	

```
5 rows in set (0.01 sec)
```

```
[MariaDB [(none)]> USE IOTDB;
Reading table information for completion of table and column name
You can turn off this feature to get a quicker startup with -A

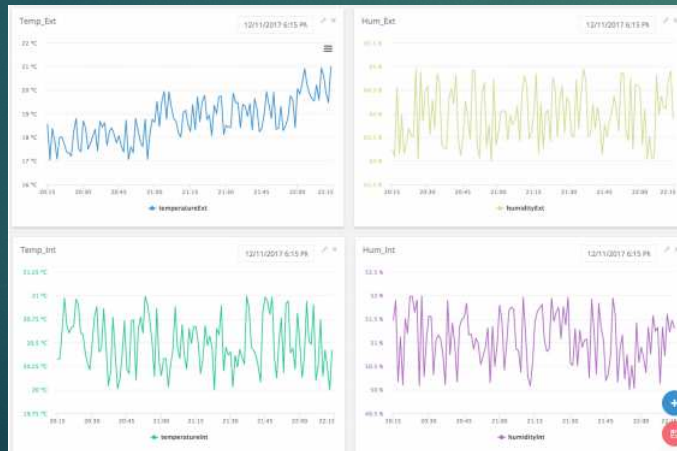
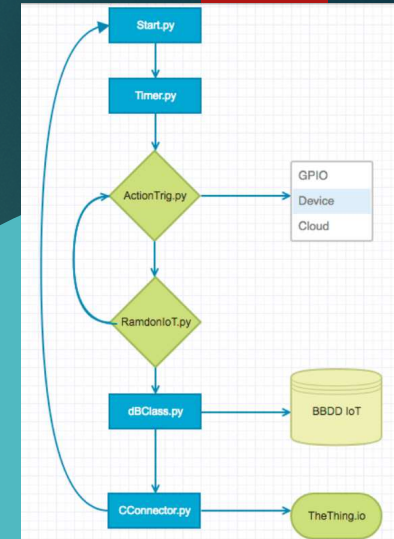
Database changed
[MariaDB [IOTDB]> select * from data;
```

id	value	name	timestamp	type
1	20.08	internal	2017-12-06 18:36:43.660833	temp
2	18.62	external	2017-12-06 18:36:43.928529	temp
3	51.91	internal	2017-12-06 18:36:44.125200	hum
4	63.98	external	2017-12-06 18:36:44.314414	hum
5	20.41	internal	2017-12-06 18:37:43.662053	temp
6	17.7	external	2017-12-06 18:37:43.870064	temp
7	50.68	internal	2017-12-06 18:37:44.074926	hum
8	64.46	external	2017-12-06 18:37:44.278225	hum
9	20.43	internal	2017-12-06 18:38:43.663212	temp
10	17.24	external	2017-12-06 18:38:43.910823	temp
11	51.75	internal	2017-12-06 18:38:44.161455	hum
12	63.76	external	2017-12-06 18:38:44.368459	hum
13	20.04	internal	2017-12-06 18:39:43.663987	temp
14	18.02	external	2017-12-06 18:39:43.842819	temp
15	51.6	internal	2017-12-06 18:39:44.028518	hum
16	63.84	external	2017-12-06 18:39:44.327659	hum
17	20.16	internal	2017-12-06 18:40:43.665232	temp



# Procesado fog

- ▶ Todas las acciones tomadas en Raspberry Pi.
- ▶ Generación datos aleatorios.
- ▶ Toma de decisiones dependiendo de la medida obtenida.
- ▶ Grabación de datos en BBDD.
- ▶ Subida de datos a Cloud.



# Procesado cloud

- ▶ Datos generados aleatoriamente.
- ▶ Se suben todos a Thethings.io.
- ▶ Uso de un job y un trigger para la toma de decisiones.
- ▶ Informe por sms de superación de umbral usando twilio.



```
Trigger
Name: LowTempAlarmSms
Code:
1 function trigger(params, callback){
2   if(params.action != 'write') return callback()
3
4   var values = params.values
5
6   for(var i=0; i<values.length; ++i){
7     if(values[i].key == 'temperatureExt' && values[i].value < 19){
8       console.log('Low Temp Alarm')
9       var telf = '+3466611222'
10
11      var message = 'LOW TEMP ALARM!: ' + values[i].value
12
13      var twilio = new Twilio(
14        'AC4b3e35b39d9ac0d234073adcc8b23a8c',
15        '5b2a33ddb10e8a77bdfdd362081e22d4'
16      )
17
18      twilio.sendMessage({
19        to: telf,
20        from: '+34955160164',
21        body: message
22      },
23      callback
24    )
25  }
26 }
27 callback()*/
28
```

```
Job
Name: TempExtData
Frequency: Hourly
Code:
1
2 function job(params, callback){
3   analytics.events.getValueByName('temperatureExt',function(error, data){
4     analytics.kpis.create('lowTemp',{ low: data });
5     callback();
6   });
7 }
8
9
[SAVE] [Save And Continue Editing] [CANCEL]
```

# Cloud vs fog

- ▶ Mayor latencia Cloud (aprox. 30ms) dependiente de la Wan.
- ▶ Seguridad: fog en área local, se puede implementar seguridad.
- ▶ Cloud no recomendable para aplicaciones críticas, delay.
- ▶ Coste: fog mas barato, no necesita Wan ni plataforma Cloud.
- ▶ Uso de Bw, cloud consumirá ancho de banda dependiendo de los datos recopilados.

# Gracias

Rubén Cabello Chacón  
rcabello@uoc.edu

