

## Introducción al paradigma de Internet de las Cosas

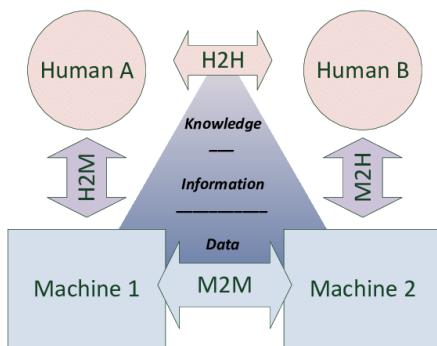
### Internet de las cosas en el contexto de Big Data

Máster Interuniversitario en Big Data: Tecnologías de Análisis de Datos Masivos  
Universidade de Santiago de Compostela (USC)

## Motivación

- Revolución tecnológica que está promoviendo una investigación multidisciplinar: Robótica, Biología, Informática, Física, Química, Psicología, Legal, ...
- IoT es un nuevo paradigma
  - Extensión de conectividad y red a todo dispositivos (sensores, RFIDs, etc.)
- Objetivo de esta sesión
  - Proporcionar una visión general de las tendencias y posibilidades.

## Tendencias de Internet



© 2008-2023 ResearchGate GmbH

M2M vs. IoT: What's the difference?	
<b>M2M</b>	<b>IoT</b>
Machines	Sensors
Hardware-based	Software-based
Vertical applications	Horizontal applications
Deployed in a closed system	Connects to a larger network
Machines communicating with machines	Machines communicating with machines, humans with machines, machines with humans
Uses non-IP protocol	Uses IP protocols
Can use the cloud, but not required to	Uses the cloud
Machines use point-to-point communication, usually embedded in hardware	Devices use IP networks to communicate
Often one-way communication	Back and forth communication
Main purpose is to monitor and control	Multiple applications; multilevel communication
Operates via triggered responses based on an action	Can, but does not have to, operate on triggered responses
Limited integration options, devices must have complementary communication standards	Unlimited integration options, but requires software that manages communications/protocols
Structured data	Structured and unstructured data

- **Internet del Futuro**
- Internet de las Cosas (IoT)
- IoT y Big Data
- Conclusión

## Tendencias de Internet

- El **dispositivo móvil** es clave en la carrera de servicios: redes sociales (Web 2.0) - Facebook, Twitter, LinkedIn, etc.
  - Servicios habilitados para ubicación
  - Aplicaciones combinadas: p.e. red social + ubicación
  - Recuperación de contenido de video – YouTube
  - Apps, Apps, miles de Apps
  - Humano a máquina (H2M) vs. Máquina a máquina (M2M)

## M2M

- Extensión de los sistemas SCADA con redes de acceso público. Monitorización remota.
- Sensores, RFID, Wi-Fi y computación automática
- IA y ML como tecnologías facilitadoras
- Comunicación punto a punto (a diferencia de la comunicación basada en IP de IoT)
- Bajo consumo, detección de *eventos*, envío/recepción continuo de datos

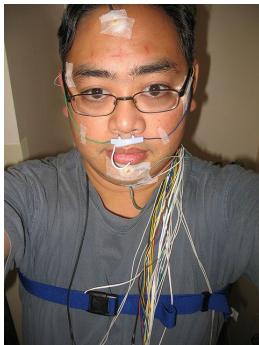
## Tendencias de Internet: Retos

- **Industria de la salud**
  - Envejecimiento activo, vida asistida.
  - Seguimiento de la salud en tiempo real
  - Cirugía robótica
- **Gobierno y ciudad**
  - Optimización de la operaciones incluyendo el consumo de energía.
  - Emisiones, residuos y otros temas ecológicos.
- **Industria automotriz**
  - Fabricantes de automóviles, seguridad, etc.

# Tendencias de Internet: dispositivos

- El [Foro Mundial de Investigaciones Inalámbricas](#) (WWRF) predijo:
  - habrá entre 5 y 7 **billones** de dispositivos inalámbricos
  - que darán servicio a entre 5 y 7 mil millones de personas → mil dispositivos por persona para 2020
- Cada objeto se integrará en la red y se considerará como un dispositivo de red que genera, transmite y / o absorbe datos.

## Sensores por todas partes



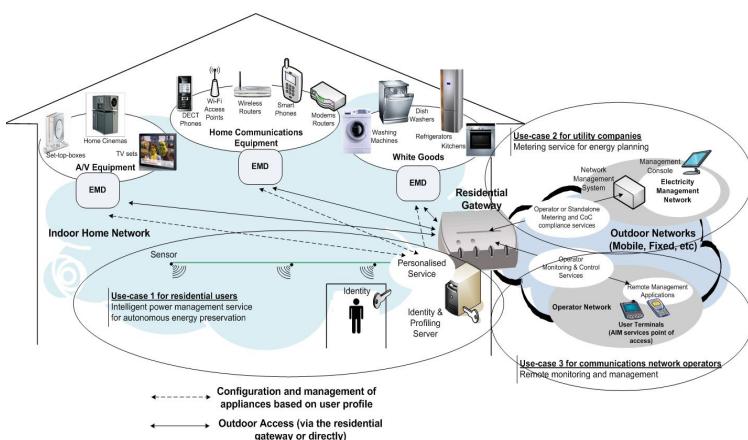
# Tendencias de Internet: dispositivos

- ¿**Tipos** de dispositivos?
  - **Dispositivos personales**, dispositivos portátiles, en el hogar y en el automóvil
  - **Dispositivos autónomos** como robots con capacidades de comunicación.
  - **Dispositivos de medios específicos** como sensores acústicos inalámbricos bajo el agua o biosensores
  - **Dispositivos voladores** como vehículos terrestres tripulados / no tripulados

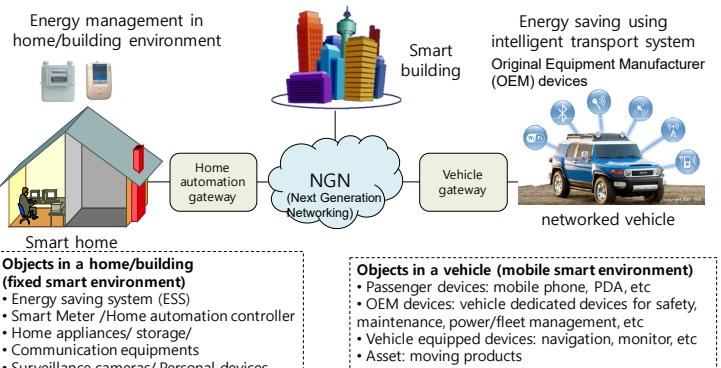
## Internet del futuro

- **Dispositivos inteligentes heterogéneos** omnipresentes, que se comunican de forma **inalámbrica** a través de redes híbridas y ad hoc de dispositivos, sensores y actuadores, ...
- que trabajan en sinergia para **mejorar** la calidad de **nuestras vidas**, optimizando el consumo de energía y reduciendo constantemente el impacto ecológico de la humanidad.

## Internet del futuro



## Internet del futuro



## Caso de uso



<https://www.youtube.com/watch?v=d55rBuB9D7s>

## Índice

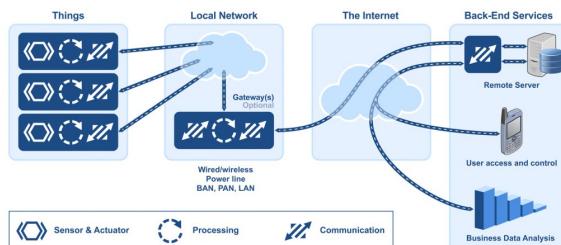
- Internet del Futuro
- **Internet de las Cosas (IoT)**
- Internet en el borde (*edge computing*)
- IoT y Big Data
- Conclusión

# Internet de las Cosas

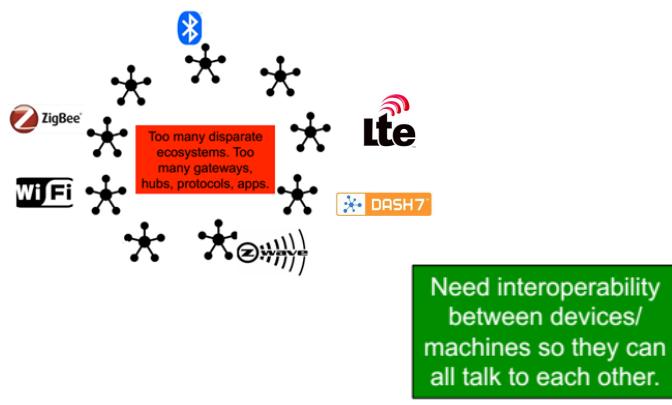
- "El Internet de las cosas" fue acuñado por Kevin Ashton de Procter & Gamble en una presentación en 1999.
- Inicialmente ligado a RFID
- Ashton se mudó al Instituto de Tecnología de Massachusetts, donde cofundó y dirigió Auto-ID Center: el laboratorio de investigación que ayudó a construir la base del Internet de las cosas

## IoT como paradigma de comunicaciones

- La naturaleza del entorno de IoT requiere **protocolos, diseños de red y arquitecturas** de servicio que puedan hacer frente a miles de millones de entidades de IoT, y conectar a los proveedores de datos con los consumidores.



## Redes inalámbricas como tecnologías facilitadora



## Claves del mercado

- Algunas cifras:
  - Mercado global de USD 308,97 billones en 2020.
  - Crecimiento del 23,1% en 2020 con respecto al promedio del período 2017-19.
- Para el período 2021-28 la perspectiva de crecimiento es de \$381.30 billones USD en 2021 a \$1,854.76 billones USD en 2028 con una tasa de crecimiento anual compuesto (CAGR) del 25.4% en dicho período.

# Internet de las Cosas

- IoT consiste en conjuntos de **dispositivos y estrategias de comunicación heterogéneos**
- IoT proporciona un **conjunto de soluciones** en diferentes niveles e instancias en las que las *cosas* (por ejemplo, objetos cotidianos, ubicaciones, vehículos, medidores, etc.) se extienden con **sensores, RFID, actuadores o procesadores**.

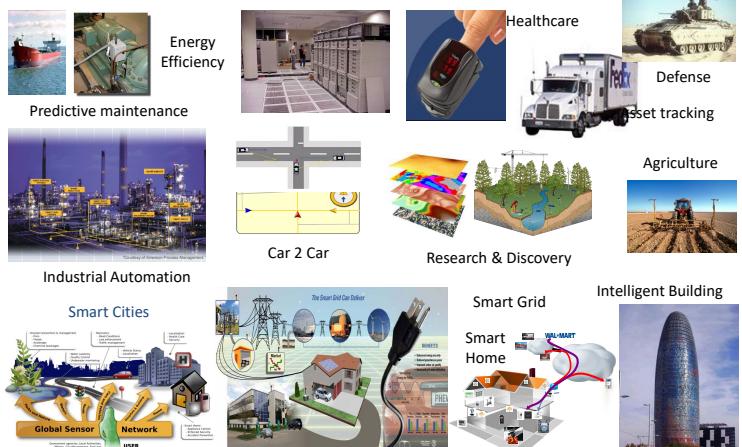


## Redes inalámbricas como tecnologías facilitadora

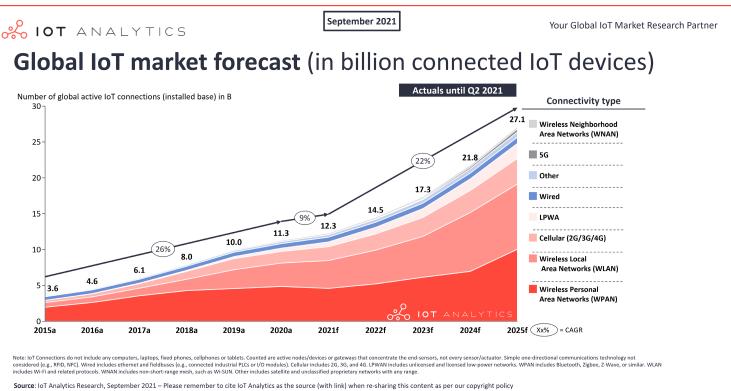
- Accesos multipunto económicos:
- Nuevos tipos de dispositivos
- Nuevos usos
  - Baratos de instalar: cablear es lento, tedioso y caro
  - Cobertura Global
    - Desde Near Field (NFC) a Satélite o 4G/5G
    - Cualquier lugar



## Monitorización y automatización



## Claves del mercado



## Proyección actual

- 27 billones de nodos IoT@2025: El esquema tradicional de medir → transmitir → procesar no es escalable ni sostenible en términos de ancho de banda y energía consumida.
- Esta limitación es más relevante en el caso de flujos densos de datos tales como la información visual.

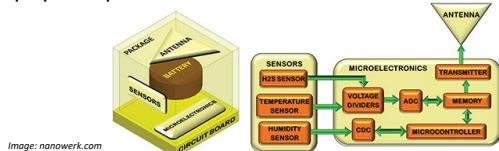
## Solución

- Información *versus* datos
- Procesar la información localmente y con bajo coste energético:
  - Nuevas modalidades de sensado
  - Nuevos algoritmos
  - Hardware de propósito específico

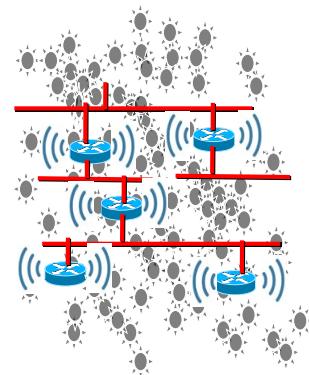


## Smart Dust

- Distribución geográfica arbitraria
- Numerosos sensores con **capacidades limitadas** (batería...)
- Redes inalámbricas de sensores de escala sub-milimétrica **autónomos**
- Programa [SHIELD](#) del DARPA para autenticar la cadena de suministro en chips para aplicaciones de defensa

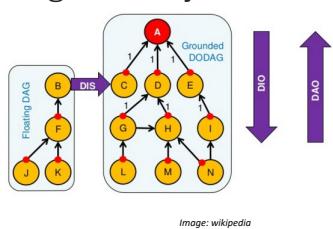


- Shortest Geopath Routing (e.g. protocolo RPL, *Routing Protocol for Low-Power and Lossy Networks*)
- Comunicación multipunto a punto (MP2P), punto a multipunto(P2MP) y punto a punto (P2P)
- RPL puede ser usado con los sistemas operativos de redes de sensores inalámbricos más difundidos, TinyOS y ContikiOS



## Smart Dust - RPL

- Redes inalámbricas de bajo consumo
- Topología tipo DAG (grafo acíclico dirigido)
  - Cada nodo tiene un *rango* que aumenta al alejarse del nodo raíz (DODAG). Criterio de selección de ruta → rango más bajo:
  - Mensaje DIS
  - Mensajes DIO
  - Mensajes DAO



## Smart Dust

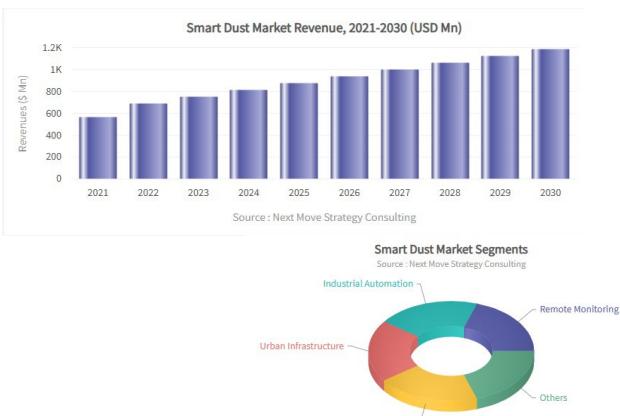
### Aplicaciones:

- Agricultura
- Industria
- Medioambiente
- Infraestructura urbana
- Gestión de inventario
- Médica – [Neural Smart Dust](#) and [Iota](#)
- Militar
- Exploración espacial

### Retos:

- Privacidad
- Control
- Coste
- Contaminación
- Salud
- Legal

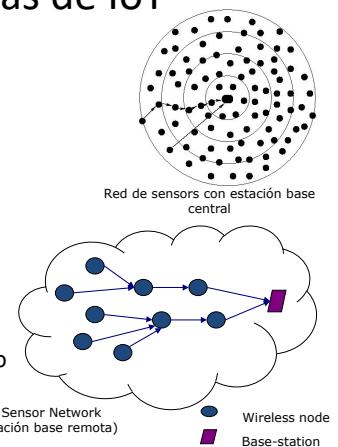
## Smart Dust Market



Source: Smart Dust Market Size, Share, Forecast, Industry Analysis Report | 2022 - 2030 (nextmsc.com)

## Características de IoT

- Red muy **densa** (densidad espacial): ¿qué nivel de direccionamiento?
- Puede **monitorear** "de cerca" y con una escala de tiempo variable (de  $\mu$ s a días)
- Posible despliegue **aleatorio** debido a terreno inaccesible → necesidad de capacidades de auto-organización
- **Movilidad** típicamente **baja**, pero la topología podría ser dinámica



# Retos en IoT

- Gran escala
- Interconexión de dispositivos heterogéneos, sistemas operativos y subredes
- Apertura y seguridad
  - ¿Es posible conectar cualquier cosa a Internet?
  - ¿Queremos?
    - Protección empresarial
    - Seguridad y privacidad
    - Confianza



## NAT y Gateways no escalan

- Los recursos privados se protegen del acceso público detrás de un dispositivo Network Address Translation (NAT), y es a través de este recurso desde el cual las instancias privadas pueden acceder a internet.
- Las *gateways* como NAT solo soportan un conjunto limitado de protocolos
- IP es independiente de los medios de comunicación y también lo es IPv6
- IPv6 proporciona 340 sextillones de direcciones

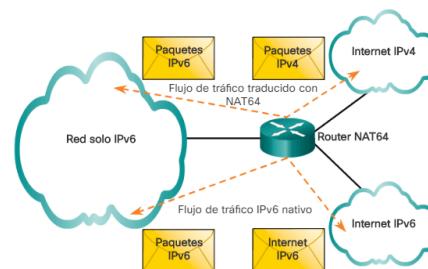
## ¿Por qué una arquitectura de comunicación basada en IP?

- IP proporciona **una capa de unificación común** a las tecnologías subyacentes
  - 2G/3G, 802.11 a / b / g, 802.11p, 802.16, satélite, ...
  - Cualquier aplicación que se ejecute sobre IP es independiente del medio
- IP asegura la **interoperabilidad**
  - IPs en todas partes: ITS, educación, salud, ejército.
  - Sin estar limitado a una aplicación concreta

# Dirección IP en los sensores

- ¿Por qué IP?
- Estándares abiertos vs. propietarios
  - Los proveedores de COTS (commercial off-the-shelf) reducen los costes
  - Fiabilidad, disponibilidad y seguridad
- Abstracción de IP vs. MAC
  - 802.11, 802.15.4 (e), Sat, 3G, UWB
- Sin gateways intermedios, túneles, middle-boxes y otros “trucos”

## IPv6 y NAT



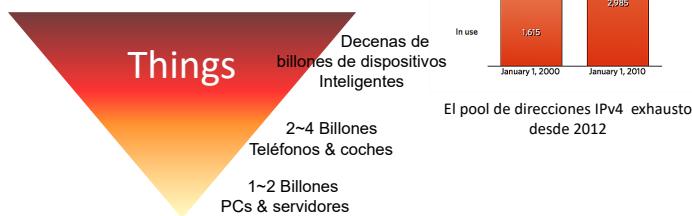
<https://ccnadesdecero.es/nat-para-ipv6/>

## ¿Por qué una arquitectura de comunicación basada en IP?

- IP asegura la **portabilidad**
  - Los usos ordinarios de Internet se pueden llevar al dispositivo (navegación web, video streaming, ...)
- IP asegura un **despliegue** más amplio
  - Los equipos IP son más baratos de desarrollar.
  - Los productos se pueden actualizar constantemente (agujeros de seguridad, nuevas características)

## ¿Qué versión IP?

- Internet comprende varios miles de millones de dispositivos.
- Los objetos inteligentes agregarán decenas de miles de millones de dispositivos adicionales
- **IPv6 es el único camino viable para avanzar.**



## IPv6 características avanzadas

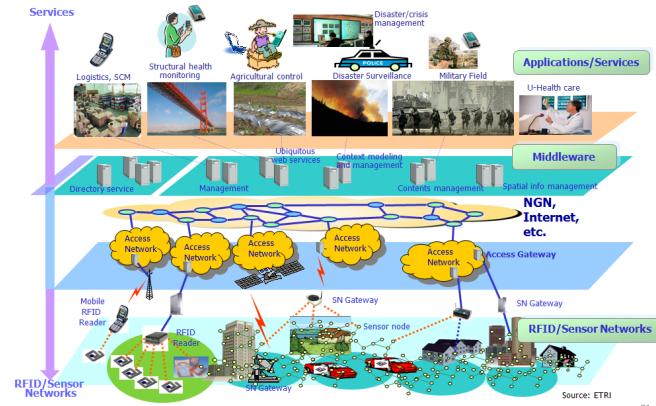
- QoS\* para flujos de tráfico prioritarios
- Compatibilidad Anycast y Multicast (múltiples interfaces de red)
- Enrutamiento inalámbrico avanzado para redes multihop
- Movilidad y seguridad
  - Mobile IPv6 (evita triangulación MIPv4)
  - IPSec obligatorio (cifrado y autenticación IP)

\*QoS: La calidad de servicio (quality of service o QoS) se refiere a cualquier tecnología que gestiona el tráfico de datos para reducir la pérdida de paquetes, la latencia y el jitter, o fluctuación, en una red

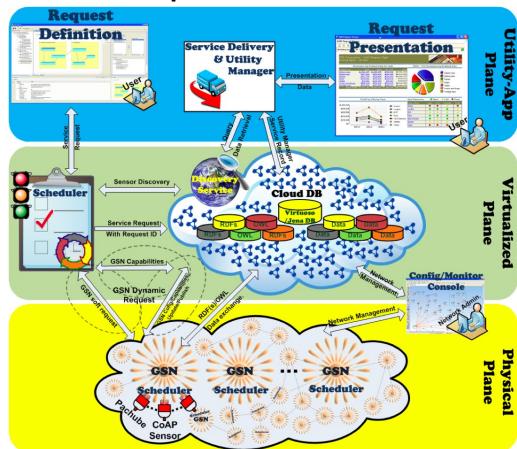
# Índice

- Internet del Futuro
- Internet de las Cosas (IoT)
- IoT y Big Data
- Conclusión

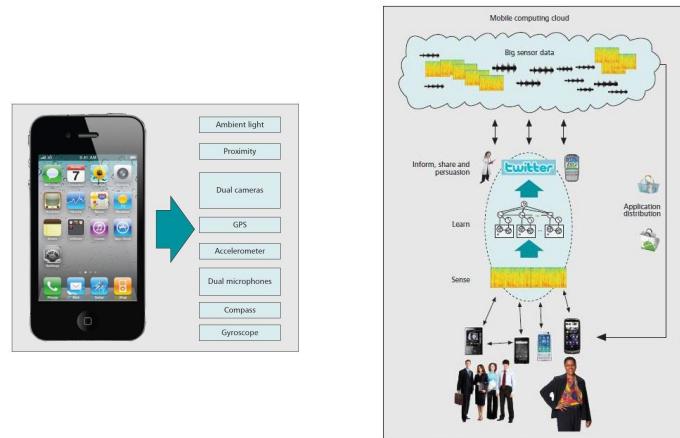
# Servicios globales para ciudadanos



## Principios de diseño



## Arquitectura de sensorización para teléfonos móviles



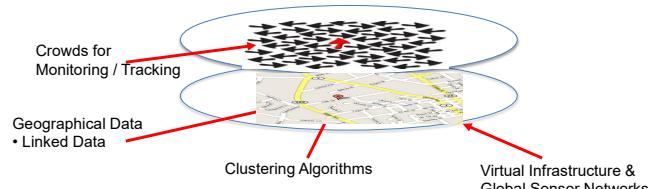
## Smart cities

- Problemas:
  - Sostenibilidad del medio ambiente
  - Creciente población urbana
- Servicios potenciales:
  - Automatización (riego)
  - Seguimiento (contaminación)
  - Ahorro de energía / agua
  - Regulación del tráfico
  - ...



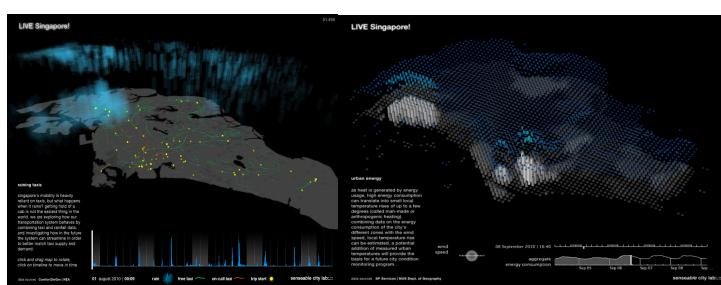
## Smart cities

- Monitorización de "smart crowds"



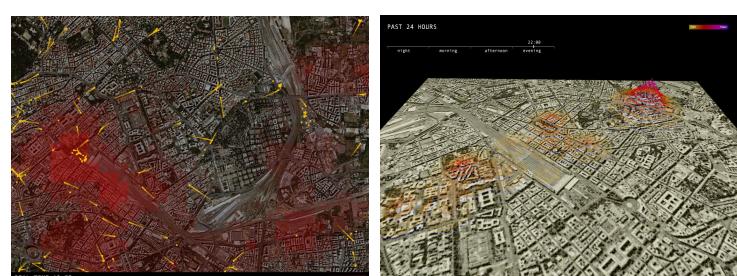
## Smart cities: caso de uso

"Live Singapore", the power of "Crowd-sourcing"



## Smart cities: caso de uso

"Real Time Rome", the power of "Crowd-sourcing"



# Índice

- Internet del Futuro
- Internet de las Cosas (IoT)
- IoT y Big Data
- **Conclusión**

# Conclusión

- IoT es un elemento clave de Internet
- Potencial para nuevas aplicaciones y servicios
  - SmartCities como área clave para explotar servicios ciudadanos
  - Smartphones como plataforma sensorial.
- Nuevas oportunidades de explotación y definición de servicios centrados en el usuario.
- Desafíos: seguridad y privacidad, integración heterogénea de sistemas heredados

# Índice

## Herramientas y plataformas para Internet de las Cosas

### Internet de las cosas en el contexto de Big Data

Máster Interuniversitario en Big Data: Tecnologías de Análisis de Datos Masivos  
Universidade de Santiago de Compostela (USC)

- BLOQUE I: Plataformas para gestión de la información
- BLOQUE II: Dispositivos y soluciones para recogida de información de sensores
- BLOQUE III: Los dispositivos móviles como sensores de datos

## Plataformas IoT

- Conjunto de **hardware y software** sobre la que otras aplicaciones pueden funcionar
- La plataforma permite desplegar y ejecutar **aplicaciones**
- Las plataformas de aplicaciones IoT proporcionan un amplio conjunto **herramientas** (funcionalidades independientes que se pueden utilizar para construir aplicaciones IoT)
- Plataforma en la **nube**

### BLOQUE I: Plataformas para gestión de la información

## Cloud computing: Introducción

- Paradigma basado en el concepto de **provisión dinámica y bajo demanda** de:
  - Computación
  - Almacenamiento
  - Servicios de red
  - Infraestructuras IT en general
- Recursos disponibles a través de una red de acceso (normalmente Internet)
- Recursos disponibles sólo cuando se necesitan

## Cloud computing: perspectiva de usuario

- Acceso a **infraestructuras IT de altas capacidades** (computación, almacenamiento) sin grandes inversiones
- **Accesible** fácilmente
- Usuarios **sólo pagan** por lo que realmente **necesitan**
  - Cantidad de recursos
  - Tiempo de uso
- Acceso a los servicios de forma **ubicua**

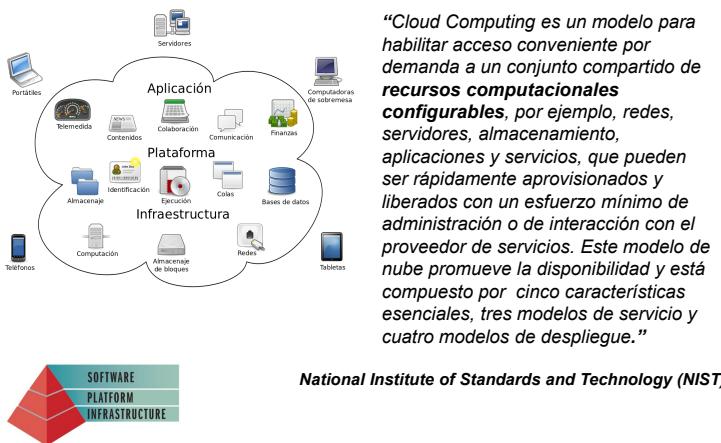
# Cloud computing: perspectiva empresarial

- **Empresas grandes** pueden **delegar** actividades no relacionadas con su actividad principal
- **Empresas pequeñas** pueden abordar nuevos proyectos sin grandes inversiones económicas
- Los **desarrolladores** pueden centrarse en la lógica de negocio y en sus tareas de desarrollo, dejando de lado las tareas de despliegue y gestión de la arquitectura IT

# Cloud computing

- **Virtualización** de infraestructuras y dispositivos
- **Modelos de servicios** flexibles y dinámicos:
  - IaaS (Infrastructure as a Service)
    - Virtualización de Hardware, computación o almacenamiento
  - PaaS (Platform as a Service)
    - Herramientas e interfaces para desarrollo software
  - SaaS (Software as a Service)
    - Aplicaciones finales

## Cloud computing



## Cloud computing: características

4. **Acceso a través de Internet**: los servicios son entregados a través de internet utilizando mecanismos y protocolos estándar.
5. **Modelo de pago por uso**: los servicios son monitorizados a través de métricas que permiten el establecimiento de diferentes modelos de pago.

## Cloud computing: características

1. **Auto-servicio bajo demanda**: sin tener que comunicarse personalmente con el proveedor de servicios.
2. **Escalabilidad y elasticidad**: añadir o eliminar recursos de cómputo.
3. **Conjunto de recursos compartidos**: los recursos sirven a múltiples consumidores. El consumidor no posee control o conocimiento sobre la ubicación exacta de los recursos.

## Cloud computing: pilares

1. **Sistemas distribuidos**: permiten la gestión de grandes despliegues descentralizados, aumentando la escalabilidad de los sistemas
2. **Virtualización**: concepto clave que ha permitido el desarrollo flexible de plataformas software sobre un hardware común:
  - Mejora en la utilización de los recursos hardware
  - Ahorro de espacio físico
  - Ahorro energético y de costes de gestión
  - Necesidad de arquitecturas flexibles
3. **WEB 2.0**: habilita el acceso fácil a estos servicios para los usuarios finales. La mayoría de estos servicios puedes ser gestionados vía web

## Cloud computing: virtualización

- Simulación de **máquinas virtuales**.
- Se lleva a cabo en una plataforma de hardware mediante un **software "host"** (que es un programa de control) que simula un entorno computacional (máquina virtual) para su **software "guest"**
- Este *guest*, que generalmente es un SO completo, corre como si estuviera instalado en una plataforma HW autónoma.

## Cloud computing: virtualización

- Muchas maquinas virtuales son simuladas en una maquina física dada.
- Para que el sistema operativo guest funcione, la simulación debe ser lo suficientemente grande como para soportar todas las interfaces externas de los sistemas guest, las cuales pueden incluir los drivers de hardware.

# Cloud computing: virtualización

- **Host:** anfitrión en un entorno virtualizado. Como ejemplo concreto podemos pensar en un **equipo físico** con un sistema operativo donde se crean máquinas virtuales.
- **Guest:** el recurso virtual acogido en el host. Típicamente la maquina virtual.
- **Hipervisor o VMM (Virtual Machine Monitor/Manager):** SW donde reside la emulación y gestión de las maquinas virtuales.

## Tipos de hipervisor

- **Hipervisor Tipo 1 o Hipervisor Nativo:**
  - Se ejecuta directamente sobre el HW del host
  - No necesita un SO de base
  - Acceso directo a los recursos HW
  - Ejemplos: VMware ESXi, Citrix XenServer and Microsoft Hyper-V hypervisor

## Tipos de hipervisor

- **Hipervisor Tipo 2 o *hosted hypervisor*:**
  - Se ejecuta en el SO del host
  - Estos hipervisores no se ejecutan directamente sobre el HW sino como una *aplicación* en un determinado host (máquina física)
  - El hipervisor solicita al SO que haga las llamadas al HW.
  - Ejemplos: PCs y analistas de seguridad

## Tipos de hipervisor

Criteria	Type 1 hypervisor	Type 2 hypervisor
AKA	Bare-metal or Native	Hosted
Definition	Runs directly on the system with VMs running on them	Runs on a conventional Operating System
Virtualization	Hardware Virtualization	OS Virtualization
Operation	Guest OS and applications run on the hypervisor	Runs as an application on the host OS
Scalability	Better Scalability	Not so much, because of its reliance on the underlying OS.
Setup/installation	Simple, as long as you have the necessary hardware support	Lot simpler setup, as you already have an Operating System.
System Independence	Has direct access to hardware along with virtual machines it hosts	Are not allowed to directly access the host hardware and its resources
Speed	Faster	Slower because of the system's dependency
Performance	Higher-performance as there's no middle layer	Comparatively has reduced performance rate as it runs with extra overhead
Security	More Secure	Less Secure, as any problem in the base operating system affects the entire system including the protected Hypervisor
Examples	<ul style="list-style-type: none"><li>▪ VMware ESXi</li><li>▪ Microsoft Hyper-V</li><li>▪ Citrix XenServer</li></ul>	<ul style="list-style-type: none"><li>▪ VMware Workstation Player</li><li>▪ Microsoft Virtual PC</li><li>▪ Sun's VirtualBox</li></ul>

Source: <https://www.hitechnectar.com/blogs/hypervisor-type-1-vs-type-2/>

# Cloud computing: virtualización

- **Hipervisor o VMM (virtual machine monitor):**
  - Capa entre el SO y el HW. Proporciona los servicios necesarios para que puedan ejecutarse varios SO sin conflictos.
  - Gestiona las colas y las instrucciones de/hacia el HW.
  - Se implemente como código embebido en el firmware del sistema o como una capa software.

## Tipos de hipervisor

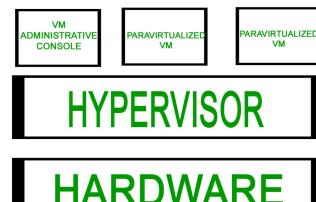
- **Hipervisor Tipo 1 (Pros & Cons):**
  - + Muy eficientes (acceso directo a la CPU, memoria, red, almacenamiento)
  - + Seguros
  - Suelen necesitar una máquina dedicada para gestionar varias MV y controlar el acceso a los recursos HW del host

## Tipos de hipervisor

- **Hipervisor Tipo 2 (Pros & Cons):**
  - + Acceso rápido y sencillo al SO del guest
  - + Incluyen herramientas adicionales para el guest
  - No hay acceso directo a los recursos HW, por lo que son menos eficientes que los Tipo 1.
  - Menor seguridad ya que si se accede al SO del host, también se puede hacerlo al del guest.

## Tipos de virtualización

### • Paravirtualización:

- El HW no es simulado
  - El SO del guest **se modifica y recompila** antes de su instalación en la MV, lo que mejora su rendimiento ya que de esta forma se comunica directamente con el hipervisor.
- 

Source: <https://www.geeksforgeeks.org/types-of-server-virtualization-in-computer-network/>

## Tipos de virtualización

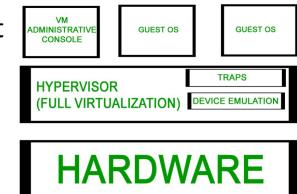
- **Para-virtualización (Pros & Cons):**

- + Sencillo
- + Buen rendimiento
- Requiere la modificación del SO del guest

## Tipos de virtualización

- **Virtualización completa**

- Similar a la paravirtualización
- Puede emular el HW subyacente de ser necesario
- El hipervisor gestiona las operaciones máquinas, las emula en SW y devuelve los códigos de status de forma consistente a como lo haría el HW real.
- No se modifica el SO del guest

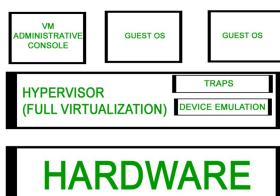


Source: <https://www.geeksforgeeks.org/types-of-server-virtualization-in-computer-network/>

## Tipos de virtualización

- **Virtualización completa (Pros & Cons)**

- + No se modifica el SO
- Complejo
- Lento (emulación)

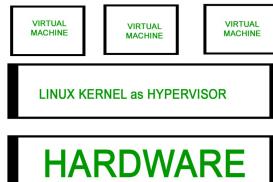


Source: <https://www.geeksforgeeks.org/types-of-server-virtualization-in-computer-network/>

## Tipos de virtualización

- **Virtualización a nivel de núcleo:**

- En vez de un hipervisor, ejecuta un *kernel* Linux y trata a la MV asociada como un proceso de usuario del host
- Necesita ayuda del procesador
- Driver específico para la comunicación kernel/MV



Source: <https://www.geeksforgeeks.org/types-of-server-virtualization-in-computer-network/>

## Modelos de servicio: IaaS

- El proveedor ofrece una infraestructura escalable y compartida a diferentes clientes
- Los clientes pueden obtener servicios de infraestructura computacional bajo demanda.
- El cliente puede ajustar la infraestructura a sus necesidades.
- Ejs: web hosting, computación alto rendimiento, infraestructura de pruebas, desarrollo y producción, almacenamiento en la nube

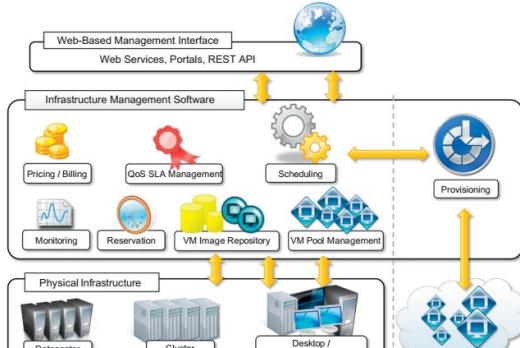
## Modelos de servicio: IaaS

- **IaaS (infraestructura como servicio)**

Proporciona la *infraestructura* informática, física u otros recursos como máquinas virtuales, imágenes de disco, almacenamiento basado en archivos, firewalls,平衡adores de carga, direcciones IP, redes de área local virtuales, etc.



## Modelos de servicio: IaaS



Ejemplos: Amazon EC2, OpenSource: OpenNebula, OpenStack, Vmware cloud, etc.

## Modelos de servicio: PaaS

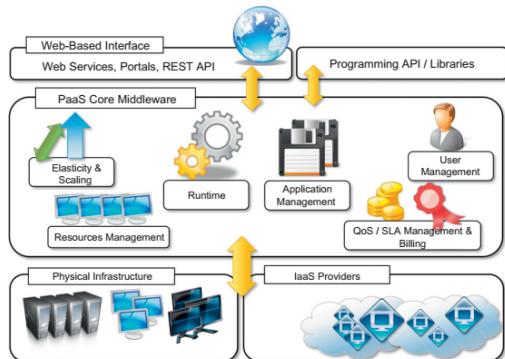
- PaaS (plataforma como servicio)

Proporciona las *plataformas* de cómputo que normalmente incluye sistema operativo, lenguaje de programación del entorno de ejecución, base de datos, servidor web, etc.

En este caso los desarrolladores se encargan de desarrollar las propias aplicaciones que se ejecutan en la nube.



## Modelos de servicio (PaaS)



Ejemplos: Google AppEngine, Microsoft Azure, IBM Cloud (Bluemix)

## Modelos de servicio: SaaS

- El software es desarrollado y gestionado remotamente por un proveedor de servicios.
  - Todo está en la nube. Los clientes no tienen que realizar operaciones en su infraestructura.
  - El pago se realiza por suscripción (mensual, anual, etc.) o a través de métricas de uso.
  - Ejemplos: redes sociales, editores on-line, aplicaciones científicas, etc.

## Modelos de servicio: PaaS

- El proveedor de servicios ofrece el middleware requerido para el desarrollo y la ejecución de aplicaciones.
  - No hay que preocuparse de la infraestructura.
  - Se proveen APIs y Frameworks para el desarrollo de sistemas
  - Ejemplos de uso: plataformas de desarrollo en la nube, entornos de programación y compilación on-line

## Modelos de servicio: SaaS

- SaaS (software como servicio)

Se trata de cualquier *servicio* basado en la web. En este tipo de servicios se accede a través del navegador sin atender al software. Todo el desarrollo, mantenimiento, actualizaciones, copias de seguridad es responsabilidad del proveedor. El usuario paga por el uso, por la infraestructura necesaria.

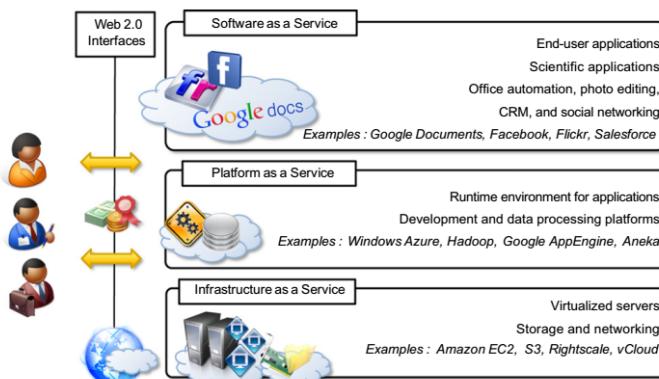
## Modelos de servicio: SaaS



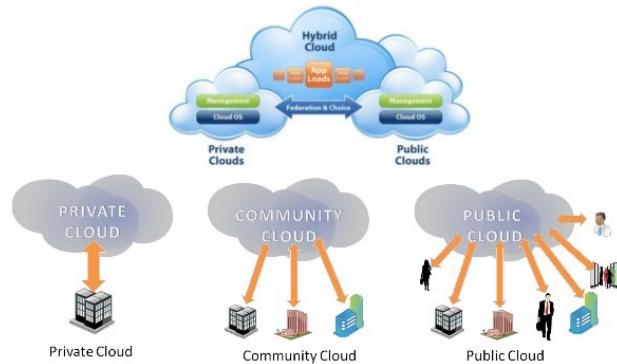
Ejemplos:  
Google apps  
Web-mail  
Overleaf  
Office 365



## Modelos de servicio



## Modelos de despliegue



## Nube pública

- Grandes infraestructuras para soportar un gran número de usuarios.
- **Disponible al público** en cualquier momento.
- Cualquier organización puede hacer uso de los servicios de forma **rápida**.
- Formadas por multitud de servidores geográficamente distribuidos
- **Multi-usuario:** necesario el aislamiento de recursos

## Nube comunitaria

- La infraestructura soporta los requerimientos de varias organizaciones.
- Los servicios están disponibles a los usuarios de las organizaciones participantes.
- Sólo las organizaciones participantes pueden hacer uso de los servicios de una manera rápida (coste medio de configuración inicial).
- Las organizaciones participantes pueden aprovechar los recursos de las demás.

## Cloud computing: ventajas

- ✓ **Bajo coste.** No hay necesidad de adquirir un hardware y un software determinado
- ✓ **Bajos requerimientos en los dispositivos de acceso,** ya que se accede a los servicios a través de interfaz web.
- ✓ **Mejora del rendimiento.** Los ordenadores que utilicen aplicaciones en la nube estarán menos sobrecargados porque tienen un menor procesado a nivel local.
- ✓ **Flexibilidad:** El servicio de nube se paga de acuerdo a la demanda.
- ✓ **Movilidad:** La información al quedar alojada en la nube pueden ser consultada por el usuario desde cualquier lugar

## Cloud computing: desventajas

- ✓ **Recursos compartidos:** QoS (Quality of Service)
- ✓ **Falta de estándares entre plataformas:** dificultad de migración
- ✓ **Software limitado al ofrecido por la plataforma**
- ✓ Muy **dependientes** del acceso a la **red**
- ✓ Riesgos de **seguridad y privacidad** de los datos
- ✓ **Altas latencias** en la recuperación de los datos
- ✓ **Servicio centralizado** en la nube

## Nube privada

- Privacidad de los datos
- La infraestructura es operada por una sola organización
- Servicios están disponibles a **usuarios internos** de la organización.
- Sólo departamentos internos pueden hacer uso de los servicios de una manera rápida
- Necesidad de desarrollar estándares de utilización

## Nube híbrida

- Combinación de los modelos anteriores.
- Infraestructura operada por organizaciones externas (Public Cloud) e internas (Private Cloud).
- Las organizaciones pueden mantener su infraestructura privada y, de requerir más recursos, utilizan infraestructuras públicas.
- Los datos sensibles pueden mantenerse en la red privada.

## Cloud computing: ventajas

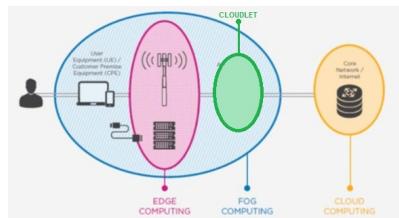
- ✓ **Mejor utilización de recursos hardware.** Los elementos que forman la nube son explotados de forma más exhaustiva.
- ✓ **Reducción del consumo energético** al haber menos equipamiento hardware por separado
- ✓ **Menor tiempo de despliegue** de aplicaciones
- ✓ **Facilidad para agregar y eliminar servicios**, al estar todos estos virtualizados
- ✓ Acceso a toda la **información en tiempo real**
- ✓ Las **actualizaciones** de software son **instantáneas y transparentes** al usuario final
- ✓ Capacidad de **almacenamiento casi ilimitada**

## Cloud computing: alternativas

- Nuevas aplicaciones con nuevos requisitos:
  - Baja latencia
  - Alto ancho de banda
  - Servicios contextualizados y dependientes de la localización
- Los servicios cloud presentan problemas para manejar estos requerimientos:
  - Camino muy largo desde el usuario final hasta los servidores (**alta latencia**)
  - Posible colapso del backbone
  - Dificultad para proveer servicios contextualizados por el gran volumen y precisión de la información necesaria para esta contextualización

# Cloud computing: alternativas

- Tres alternativas:
  - Cloudlets
  - Fog Computing
  - Mobile Edge Computing (MEC)
- Aproximan la computación y el almacenamiento al usuario



## Fog Computing

- Propone situar la nube más cerca de los usuarios finales
- En un principio, el fog se situaba en los dispositivos finales (móviles, portátiles, dispositivos IoT con capacidad de procesado, etc.)
- Actualmente, el fog se entiende como toda la infraestructura comprendida entre los dispositivos finales y la red troncal

## Cloudlets

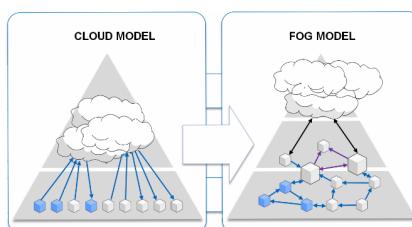
- Propone instalar **datacenters dedicados** más próximos al usuario final
- Capa de procesado entre los dispositivos finales y el cloud: requiere de hardware y software específico: por ejemplo OpenStack++
- No ha tenido gran repercusión por la complejidad del despliegue

## Fog Computing

- Los datos, procesamiento y aplicaciones se concentran en los dispositivos de usuario en lugar de existir casi en su totalidad en la nube.
- Computación distribuida de proximidad. Cada uno de los dispositivos conectados a la red puede procesar los datos y sólo transmitir un resumen al siguiente nivel, o hacerlo sólo en determinados casos.

## Fog Computing

- ✓ Baja latencia y conciencia de la ubicación
- ✓ Heterogeneidad
- ✓ Movilidad
- ✓ Alta cantidad de nodos
- ✓ Acceso inalámbrico predominante
- ✓ Amplia distribución geográfica
- ✓ Fuerte presencia de streaming y aplicaciones en tiempo real



## Mobile Edge Computing (MEC)

- Características de funcionamiento similares a la de *fog computing* con una menor heterogeneidad de los dispositivos involucrados
- Acceso a las métricas de nivel físico de los nodos conectados a la estación base, lo que permite tomar decisiones en la configuración de las comunicaciones
- Otra acepción: Multi-Access Edge Computing
  - En 5G se consideran múltiples tecnologías de acceso simultáneas
  - 5G-RAN. Nodos RAN - desde torretas o antenas a pequeñas infraestructuras. Cloud-RAN (C-RAN): Centros de datos locales que sirven a varios nodos

## Mobile Edge Computing (MEC)

- Procesamiento en quasi-tiempo real de grandes volúmenes de datos
- Elementos de procesado y almacenamiento situados **exclusivamente** en las estaciones base
- Ligado al desarrollo de 5G, por la baja latencia necesaria (<1ms)



## Mobile Edge Computing (MEC)

- Seguridad y videovigilancia
- Realidad virtual y aumentada, gaming
- Vehículo autónomos
- Redes 5G

**MEC = Baja latencia + alto ancho de banda**  
**MEC = Σ (movilidad + servicios cloud + edge computing)**

# Fog & MEC vs Cloud

# Plataformas IoT

FOG & MEC	CLOUD
Almacenamiento y procesamiento local	Almacenamiento y procesamiento lejano
Servicios alojados en los dispositivos cercanos al borde de la red	Los servicios se alojan en servidores virtuales
Compatible con aplicaciones de IoT que demandan tiempo real o latencia predecible	Compatible con los servidores de hardware, aplicaciones y casi cualquier tipo de datos
Se basa en el principio de aislamiento de datos del usuario que se alojan en el borde de la red	Todos los datos se centralizan en uno o más centros de datos

Google Cloud Platform



**Yaler**



**ThingSpeak**



**ThingWorx**  
A PTC Business

**zetta**

**Axedá**

## Google Cloud Platform



- Capacidades de aprendizaje automático para cualquier necesidad de IoT.
- Información empresarial en tiempo real para dispositivos dispersos por todo el mundo.
- Capacidades de IA.
- Brinda soporte para una amplia gama de sistemas operativos integrados.
- Inteligencia de ubicación.

## Salesforce IoT Cloud



- Gestión de relaciones con los clientes para todo tipo de empresas
- Proporciona datos reales sobre el uso y el rendimiento del producto.
- Datos de cualquier dispositivo.
- Con la API RESTful, puede importar datos de cualquier fuente.
- Vista de tráfico en tiempo real.

## ThingWorx



- Industrial IoT (IIoT)
- Análisis predictivo de datos
- Supervisión y mantenimiento remotos
- Acceso a los datos de aplicaciones desde servidores web locales
- Aplicaciones en la nube y entornos híbridos.

## IBM Watson IoT

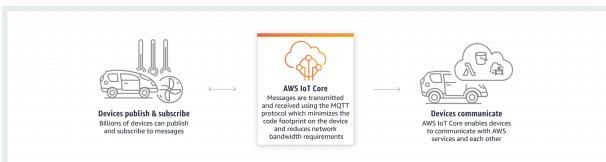


- Analítica basada en IA
- REST y API en tiempo real
- Conexión MQTT y HTTP
- Soluciones flexibles
- Seguridad
- Captura datos en tiempo real

## Amazon AWS IoT Core



- Procesa una gran cantidad de mensajes
- Plataforma confiable
- Compatible con otros servicios de AWS
- Acceso seguro a los dispositivos



Source: <https://aws.amazon.com/es/iot-core/>

## Microsoft Azure IoT



- Plataforma abierta
- Principiantes y expertos
- Azure IoT Hub:
  - comunicación entre la aplicación de IoT y los dispositivos que ésta administra.
- Azure Digital Twins
  - modelos digitales para la mejora de productos y optimización de procesos



## Oracle IoT

- Permite conectar un dispositivo a JavaScript, Android, iOS, Java y C POSIX.
- Virtualización de dispositivos, mensajería de alta velocidad y administración de terminales
- Análisis, procesamiento y enriquecimiento de datos.
- API REST para la integración con aplicaciones de Oracle y dispositivos IoT.

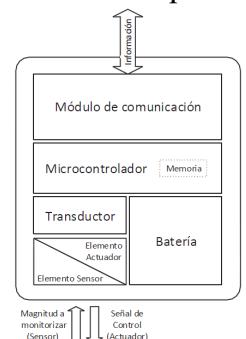
## Plataformas IoT

	Google	IBM	ThingWorx	Azure	Thing Speak	Zetta	Yaler	Amazon	Axedea
Escalabilidad	✓	✓	✓	✓	✓	✗	✓	✓	✓
Seguridad y privacidad	✓	✓	✓	✓	✗	✗	✗	✓	✓
Plug&Play	✓	✓	✓	✓	✗	✗	✗	✓	✓
Soporte millones de dispositivos	✓	✓	✓	✓	✗	✗	✗	✓	✓
Datos en tiempo real	✓	✓	✓	✓	✓	✗	✗	✓	✓
Almacenamiento de datos	✓	✓	✓	✓	✗	✗	✗	✓	✓
Soporte técnico	✓	✓	✓	✓	✓	✗	✓	✓	✓
Tipo de solución	PaaS	PaaS	Complete IoT	PaaS	Data Analytics	API to devices	-	IaaS	SaaS

Fuente: Bhumi Nakhuva and Prof. Tushar Champaneria, Department of Computer Science and Engineering, L. D. College of Engineering, India.  
International Journal of Computer Science & Engineering Survey (IJCSSES) Vol.6, No.6, December 2015

## Arquitectura funcional de un nodo sensor/actuador

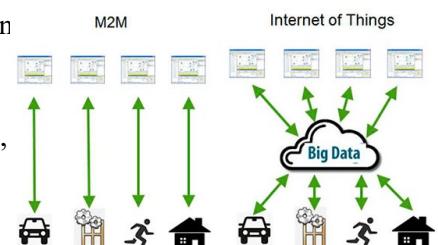
- La arquitectura de un elemento sensor o actuador se puede dividir en tres bloques funcionales:
  - Bloque sensor/actuador
  - Bloque de procesado
  - Bloque de comunicación



65

## M2M vs IoT

- En M2M los dispositivos solo envían la información que recolectan
- En IoT envían, reciben, **procesan** información y toman acciones con (o sin) la ayuda de soluciones Big Data.



	M2M	IoT
Se aplica en	Máquinas	Sensores
Basado en	Hardware	Software
Aplicaciones	Verticales	Horizontales
Tipo de instalación	Sistemas cerrados	Grandes redes
Tipo de comunicación	Máquina-a-máquina <small>Máquina-a-máquina, humano-a-máquina y máquina-a-humano</small>	No IP
Tipo de protocolo	No IP	IP
¿Usa la nube?	Puede, pero no es imprescindible	Si
Singularidad	Comunicación punto-a-punto con hardware embocado	Redes IP
Dirección de la comunicación	Normalmente, unidireccional	Bidireccional
Propósito principal	Monitorización y control	Múltiples aplicaciones
Modo de operación	Desencadenar respuestas basadas en una acción	Puede desencadenar respuestas, pero no es imprescindible
Capacidad de integración	Opciones limitadas (los dispositivos deben tener estandares complementarios)	Opciones ilimitadas vía software
Tipos de datos	Estructurados	Estructurados y no estructurados
Necesita conexión a internet	No	Si

## Gateways en una arquitectura IoT (I)

- Los sensores y dispositivos tienen **capacidades de interconexión limitadas**, utilizando protocolos de bajo consumo (Bluetooth Low Energy, Zigbee, LoRaWAN, Sigfox...) para conectarse a redes de área local (LAN) o residencial (HAN). Es decir, no tienen acceso directo a Internet

Es necesario un **Gateway** para coordinar la red de sensores y actuar de proxy con la red donde la plataforma IoT expone sus interfaces para recibir y proporcionar información.

## Gateways en una arquitectura IoT (II)

- El tratamiento de la información en bruto procedente de sensores y dispositivos, directamente en la plataforma IoT, puede ser en ocasiones ineficiente en términos de rendimiento y ancho de banda.

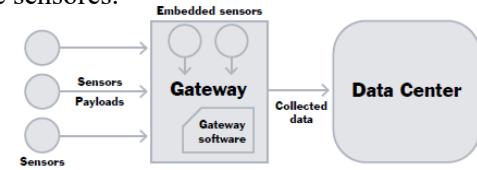
Es necesario disponer de Gateways con capacidad de almacenamiento temporal de la información (*caching*) y procesamiento de la misma. De este modo es posible filtrar y agregar la información antes de enviarla a la plataforma, así como garantizar que no se pierde información en caso de interrupción temporal de las comunicaciones.

## Diferencia entre Gateway y rooter

- Un **router** es un dispositivo de red que **integra dos o más redes**, a la vez que controla el tráfico de datos sobre la red externa global (Internet), permitiendo, por ejemplo, el control sobre los puertos de entrada y salida, y asegurando que los paquetes de datos viajen de manera correcta. En términos de redes, opera en la capa 3 (capa de red).
- Un **gateway** es cualquier **punto de conexión o nodo en una red** que provee acceso a otra a través de él mismo. Aunque puede utilizarse de la misma forma que el router para conducir el tráfico de una red, lo más habitual es usarlo como conexión de salida externa para comunicar entornos, protocolos y arquitecturas diferentes.
- La diferencia fundamental radica en que se emplean los gateways para gestionar el tráfico entre redes diferentes en cuanto a protocolos y arquitecturas, a diferencia de los routers que gestionan el tráfico entre redes similares.

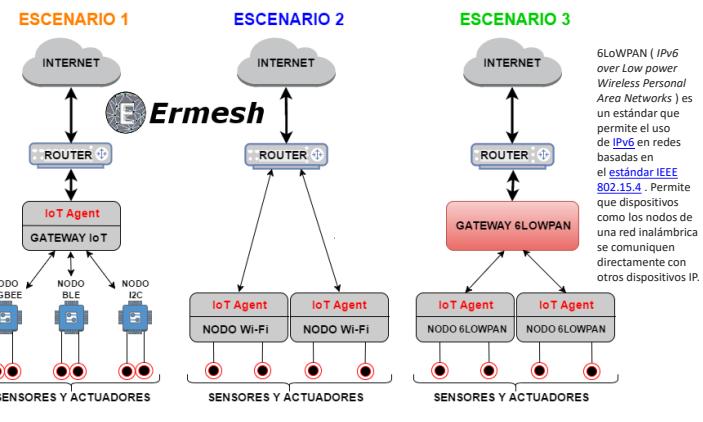
## Gateways en una arquitectura IoT (III)

- El Gateway centraliza las operaciones de monitorización de todos los dispositivos conectados a su red de sensores.

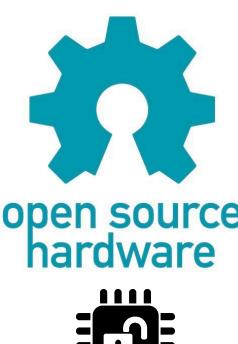


- Recuperación ante fallos: de la alimentación, de comunicaciones, en sensores o dispositivos...
- Soporte para actualizaciones automáticas
- Soporte para configuración

## Gateways: Tipos de conexión



## Dispositivos para IoT: Arduino



- Plataforma de creación de prototipos electrónicos de código abierto.
- El **hardware** consiste en una placa con un **microcontrolador y puertos de entrada/salida**.
- El **software** consiste en un entorno de desarrollo que implementa el lenguaje de programación **Processing/Wiring**.
- Ejecución de proyectos sin necesidad de conectar a un computador.

## Arduino

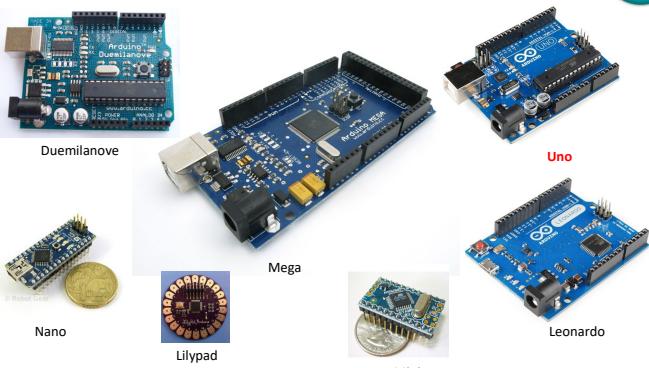


Al ser **Open-Hardware**, tanto su diseño, como su distribución es libre.

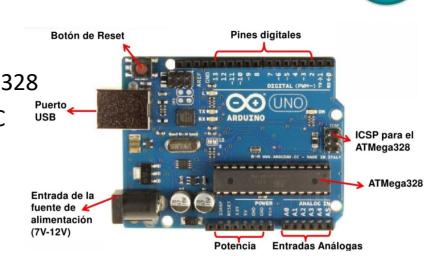
Es decir, puede utilizarse libremente para el desarrollo de cualquier tipo de proyecto sin necesidad de adquirir ninguna licencia

## Arduino

From 35€



## Arduino Uno



- Microcontrolador: ATmega328
- Voltaje de operación: 5V DC
- Alimentación: 7 – 12V DC
- Pines digitales I/O: 14
- Entradas analógicas: 6
- Bootloader: SW alojado en la memoria flash que nos permite programar Arduino a través del puerto serie (USB) sin necesidad de usar un programador externo.
- Entrada ICSP (In Chip Serial Programmer): graba desde el PC al microcontrolador sin usar el puerto USB.

## Arduino Uno

- ATmega328:
  - Microcontrolador de 8-bits de bajo consumo
  - Arquitectura RISC: menos instrucciones (131), más rápido
  - 32Kbytes de memoria flash de programa
  - Bajo consumo. 6 modos en reposo: Idle, ADC noise reduction, power-save, power-down, standby, and extended standby
    - Active mode: 1.5mA at 3V - 4MHz
    - Power-down mode: 1µA at 3V



## Arduino para IoT



- **Arduino MKR1000**, diseñado específicamente para IoT.
- Microcontrolador **SAMD21 + ARM Cortex M0 de 32-bits** ( $9\mu\text{A}/\text{MHz}$ ) con 256Kb de memoria FLASH y 32Kb de SRAM.
- **Chip Wifi WINC1500** integrado de bajo consumo con el stack TCP/IP e IEEE 802.11 así como el modo cliente o servidor, DNS y DHCP, así como un **chip de encriptación por hardware ECC508** que permite realizar comunicaciones seguras.
- Conexión para batería LiPo de 3.7V con cargador USB integrado → dispositivo totalmente autónomo



## Raspberry Pi



- Raspberry Pi comenzó a fabricarse en 2012 como ordenador de bajo coste para facilitar la enseñanza de la informática en los colegios.
- Placa base de pequeñas dimensiones en la que se aloja un chip Broadcom BCM2835 con procesador ARM hasta a 1 GHz de velocidad, GPU VideoCore IV y hasta 512 MB de memoria RAM.
- Se puede añadir teclado, ratón o cualquier dispositivo USB y conectarlo a un monitor. Disco duro conectado de forma externa, bien por USB o con una tarjeta SD.



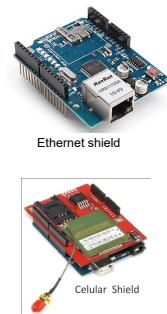
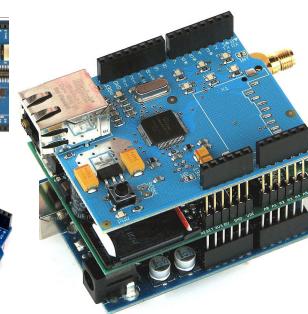
## Raspberry Pi: conexión



## Arduino Shields



- Expansión de funcionalidad



## Familia MKR Arduino para IoT

- Basada en microcontrolador SMAD21
- Menor factor de forma.
- Productos:
  - [Arduino MKR1000 WIFI](#)
  - [Arduino MKR WiFi 1010](#): módulo ESP32 de U-BLOX para acelerar aplicaciones
  - [Arduino MKR FOX 1200](#): conectividad Sigfox.
  - [Arduino MKR WAN 1300](#): conectividad LoRA
  - [Arduino MKR NB 1500](#): Narrow Band IoT



## Raspberry Pi: conexión

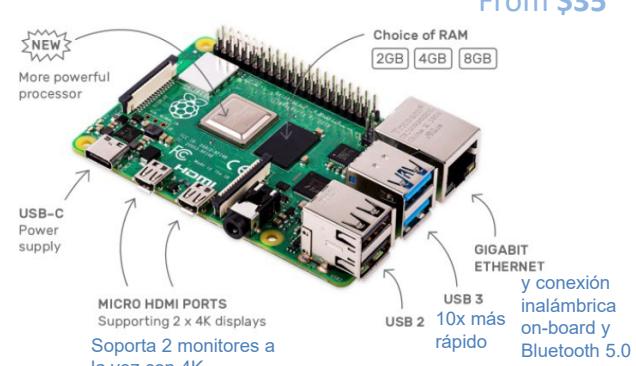


- **microUSB**: sistema de alimentación de la Raspberry.
- **GPIO**: 40 puertos, 26 se pueden usar como E/S.
- **HDMI**: conexión a TV u otro monitor. También se puede acceder en remoto desde otro PC, smartphone, etc.
- **USB**: La versión 2 B cuenta con 4 puertos USB
- **microSD**.
- **Ethernet**: para conexión de red.
- **Display DSI**: pequeñas pantallas táctiles que podemos acoplar a la Raspberry.
- **Audio 3,5 mm**.

## Raspberry Pi 4



From \$35



Además es más silenciosa y tiene menor consumo

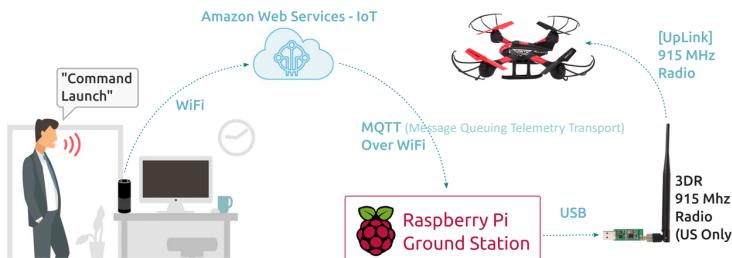
# Raspberry Pi 4: procesador

- ARM Cortex A72
- Arquitectura ARM big.LITTLE: combina núcleos potentes y lentos: 4 núcleos @ 2.5 GHz por núcleo + 4 núcleos A53 de bajo consumo
- Proceso de fabricación en 16 nm FinFet+ → 64b con un consumo 75% menor que en 32b
- Caché CoreLink CCI-500
- Tarjeta gráfica Mali-T880

# Raspberry Pi: Casos de uso



- Dron controlado por voz (Amazon Alexa)



\* <https://www.hackster.io/veggiebenz/voice-controlled-drone-with-raspi-amazon-echo-and-3dr-iris-c9fd2a>



# Arduino, Raspberry Pi, BeagleBone

Name	Arduino Uno	Raspberry Pi	BeagleBone
Model Tested	R3	Model B	Rev A5
Price	\$29.95	\$35	\$89
Size	2.95"x2.10"	3.37"x2.125"	3.4"x2.1"
Processor	ATMega 328	ARM11	ARM Cortex-A8
Clock Speed	16MHz	700MHz	700MHz
RAM	2KB	256MB	256MB
Flash	32KB	(SD Card)	4GB(microSD)
EEPROM	1KB		
Input Voltage	7-12v	5v	5v
Min Power	42mA (.3W)	700mA (3.5W)	170mA (.85W)
Digital GPIO	14	8	66
Analog Input	6 10-bit	N/A	7 12-bit
PWM	6		8
TWI/I2C	2	1	2
SPI	1	1	1
UART	1	1	5
Dev IDE	Arduino Tool	IDLE, Scratch, Squeak, Cloud9, Linux 10/100	Python, Scratch, Squeak, Cloud9, Linux 10/100
Ethernet	N/A	2 USB 2.0	1 USB 2.0
USB Master	N/A	HDMI, Composite	N/A
Video Out	N/A	HDMI, Analog	Analog
Audio Output	N/A		

# Netduino



- Potentes y flexibles.
- Implementación de robots y pequeños autómatas con lenguajes de alto nivel. El SO es el Microsoft .NET\* Micro Framework.
- Programables en C# (versión C++ de Microsoft para .Net)
- Netduino Plus incluye 22 GPIO con SPI, I2C, 4 UART (1 RTS / CTS), 6 PWM y 6 canales de ADC para interactuar con los interruptores, sensores, LEDs, dispositivos de serie, y mucho más.

**DISCONTINUED**

# Raspberry Pi: Casos de uso



- Mini PC de escritorio
- Servidor de impresión WiFi, servidor web o FTP
- Sistema de almacenamiento NAS (Network-attached storage)
- Convertir TV en Smart TV
- Sistema de música en streaming casero
- Controlador para robótica
- Hogar domótico
- Vigilancia y videoseguridad



# BeagleBone

From 115€

- **BeagleBone** es un **ordenador** de placa reducida basado en **Linux**.
- Diseñado por Texas Instruments con una filosofía **open-source y de bajo coste**.
- Utilizada para aplicaciones de visión artificial
- **BeagleBone AI**: TI C66x digital-signal-processor (DSP) cores y embedded-vision-engine (EVE) cores. OpenCL API para machine learning con herramientas pre-instaladas.

# Libelium WaspMote



- Plataforma modular Open Source, optimizada en cuanto a consumo y aplicaciones. Pensada para la creación de grandes redes de sensores, con gran autonomía energética.
- Mismo entorno de desarrollo que Arduino.
- Protocolo de comunicación Zigbee
- Módulos para dotarle de diferentes medios de comunicación adicionales (Bluetooth, GPS, y GPRS).

<https://www.libelium.com/>



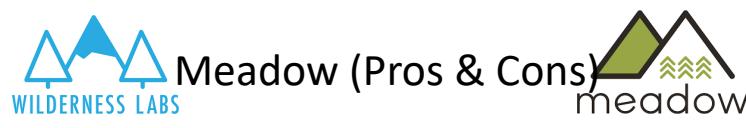
meadow



- Tiene el poder de RaspberryPi, el factor de computación de un Arduino y la capacidad de administración de una aplicación móvil.
- Diseñado para ejecutarse en una variedad de microcontroladores. La primera placa se basa en la MCU STM32F7 (STMicroelectronics), 2xpotente, 0,5xconsumo.
- WiFi, BLE, 32 MB de RAM, 32 MB de Flash. 25 puertos GPIO, PWM, I2C, SPI, CAN, UART y cargador de batería LiPo integrado, todo ello en el factor de forma Adafruit Feather.



meadow



- Incluye soporte para actualizaciones seguras por aire (OTA), lo que permite la gestión remota de las instalaciones de campo de IoT → clave para las implementaciones empresariales de IoT.
- IA en IoT (AloT): Meadow permite ejecutar visión artificial a través de TensorFlow y otros paquetes de inteligencia artificial de alto nivel localmente en chip.
- **Código abierto** (como el de Netduino). Desde el código nativo, a las bibliotecas de clases hasta la extensión de Visual Studio.

## Jetson Nano



- *"Bringing the Power of Modern AI to Millions of Devices"*
- Aplicaciones en visión por computador y procesado de voz

Developer Kit Technical Specifications	
GPU	128-core NVIDIA Maxwell™
CPU	Quad-core ARM® A57 @ 1.43 GHz
Memory	2 GB 64-bit LPDDR4 25.6 GB/s
Storage	microSD [Card not included]
Video Encoder	4Kp30   4x 1080p30   9x 720p30 [H.264/H.265]
Video Decoder	4Kp60   2x 4Kp30   8x 1080p30   18x 720p30 [H.264/H.265]
Connectivity	Gigabit Ethernet, 802.11ac wireless*
Camera	1x MIPI CSI-2 connector
Display	HDMI
USB	1x USB 3.0 Type A, 2x USB 2.0 Type A, 1x USB 2.0 Micro-B
Others	40-pin header [GPIO, I2C, I2S, SPI, UART] 12-pin header [Power and related signals, UART] 4-pin fan header*
Mechanical	100 mm x 80 mm x 29 mm

## Jetson Nano

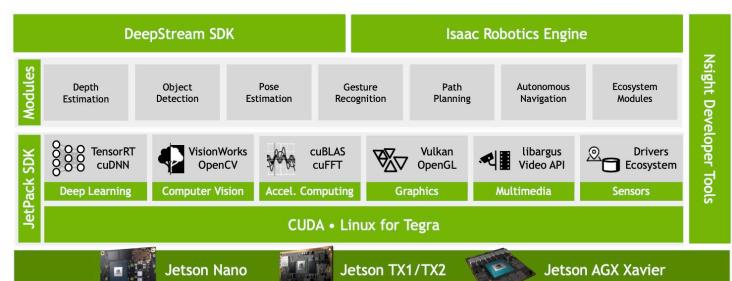
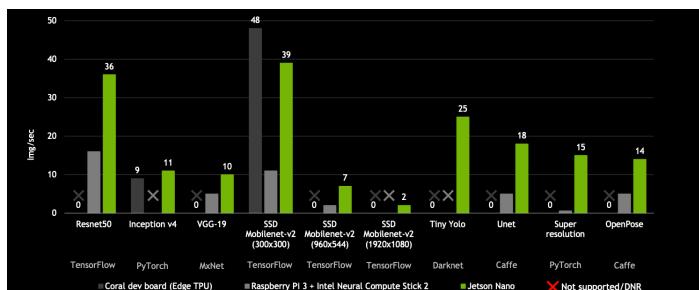


- Permite ejecutar múltiples redes neuronales en paralelo.
- Incluye soporte NVIDIA JetPack con software CUDA-X
- Soporte para tecnologías cloud
- Frameworks específicos para análisis de video, salud, genómica y robótica.

## Jetson Nano



## Otros productos NVIDIA



## BLOQUE III: Los dispositivos móviles como sensores de datos

## Los dispositivos móviles como sensores de datos

- Características generales de los dispositivos móviles
- Smartphones como sensores de datos
- WEARABLES como sensores de datos
- Código QR (Quick Response)
- Tecnología RFID & NFC



## Características generales

- Aparatos pequeños.
- Capacidad de procesamiento.
- Conexión permanente o intermitente a una red.
- Memoria (RAM, tarjetas MicroSD, flash, etc.).
- Uso individual
- Alta capacidad de interacción mediante la pantalla o el teclado

## Smartphones como sensores de datos



## Smartphones como sensores de datos

- **Giroscopio:** permite girar la pantalla.
- **Acelerómetro:** mide la aceleración con respecto a la fuerza de la gravedad, es decir, detecta el movimiento y la orientación.
- **Magnetómetro:** detecta campos magnéticos. Se usa en las aplicaciones de la brújula para señalar el polo norte o para detectar metales.
- **Podómetro:** mide los pasos que da el usuario

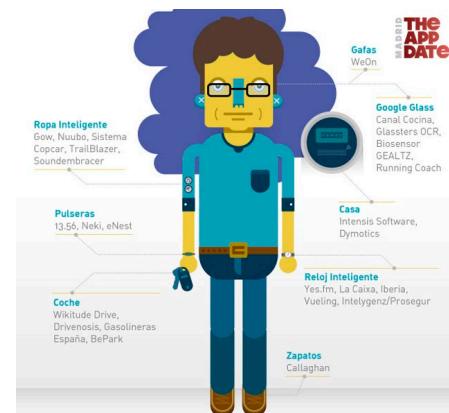
## Smartphones como sensores de datos

- **Sensor de proximidad:** se sitúa cerca del auricular del teléfono para reconocer cuándo el usuario pega la oreja al hablar por el móvil.
- **Sensor de luz:** mide la luz ambiental y ajusta automáticamente el brillo de la pantalla.
- **Termómetro:** controla la temperatura interior del dispositivo y la batería para evitar daños

## Smartphones como sensores de datos

- **Sensor de huellas dactilares:** desbloqueo de pantalla o para pagar con el móvil.
- **Pulsómetro:** mide el ritmo cardíaco a través de los vasos sanguíneos de los dedos.
- **Sensor de humedad del aire:** se utiliza, p.e., en las aplicaciones de salud para saber si el usuario está o no en su 'zona de confort'.

## Wearables como sensores de datos

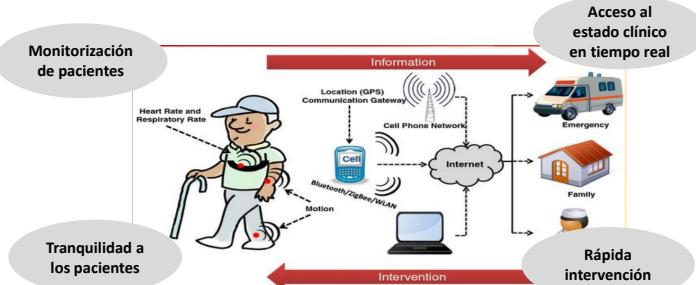


## Wearables como sensores de datos

- Desmonopolizar la atención constante del usuario.
- No debe restringir la movilidad del usuario mientras este realiza otras tareas.
- Observable por el usuario: el medio de salida es perceptible por el portador.
- Controlable por el usuario: Puede tomar control en el momento que lo deseé.

## Wearables como sensores de datos

**Wearables de salud:** *E-health*, alude a la práctica de cuidados sanitarios apoyada en tecnologías de la información y las comunicaciones (TIC). Las estadísticas recogidas por estas prendas electrónicas WEARABLES permiten conocer determinados parámetros sanitarios de la persona que las lleva.



## Smartglasses



- Tecnología inteligente para trabajar con fines correctivos, combatir anomalías oculares y proteger la vista contra los rayos ultravioleta.
- Incluyen cámaras y reproducen imágenes directamente en el campo de visión, siendo capaces de reconocer rostros, interactuar con sitios web, enviar información a redes sociales y mostrar gráficas computarizadas en 3D.

## Smartglasses



- GoogleGlass: se utilizan únicamente con la voz. Permiten estar conectados pudiendo sacar fotos, vídeos o realizar llamadas desde las propias gafas. Aplicaciones en la educación, el turismo, la medicina o el marketing.
- Alternativas: Smart Eyeglass (Sony) que apuesta por la tecnología de realidad aumentada. También Epson y otras marcas han optado por proponer sus propias alternativas.

## Smartwatches



- Incluyen funciones como podómetros, medidores de ritmo cardíaco, GPS, control de los ciclos del sueño y muchas de las funciones de un smartphone.
- Idóneos para **monitorizar** el entrenamiento deportivo, como guías de viaje o para controlar las notificaciones que recibamos sin necesidad de visualizar directamente el Smartphone.

## Fitness trackers



- Su **tamaño pequeño** y su aspecto moderno hacen que sus **funcionalidades** se vean **reducidas**, hasta incluso sólo una funcionalidad.
- Las **pulseras fitness** pueden ser un elemento motivador para gente que lleva un estilo de vida sedentario.

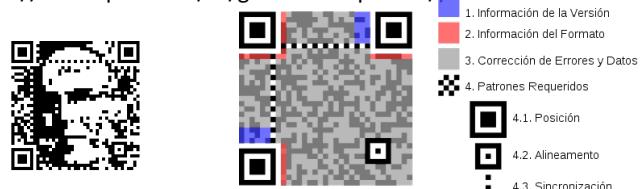
## Código QR

Matriz en dos dimensiones formada por una serie de cuadrados negros sobre fondo blanco. Esta matriz es leída por un lector específico y de forma inmediata nos lleva a una aplicación en Internet ya sea un mapa de localización, un correo electrónico, una página web o un perfil en una red social.

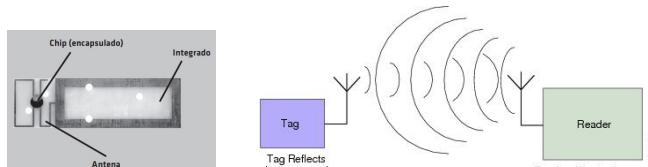


## Código QR

- Compuestos por tres cuadrados en las esquinas que permiten detectar al lector la posición del código QR y una serie de cuadrados dispersos que codifican el alineamiento y la sincronización.
- Se pueden incorporar imágenes personalizada a nuestro código QR dando un aspecto más artístico y personal.
- Son libres y cualquiera puede crear sus propios QR (<https://www.qrcode.es/es/generador-qr-code/>)



## Tecnología RFID



Los **tags** RFID constan de dos elementos: un chip y una antena.



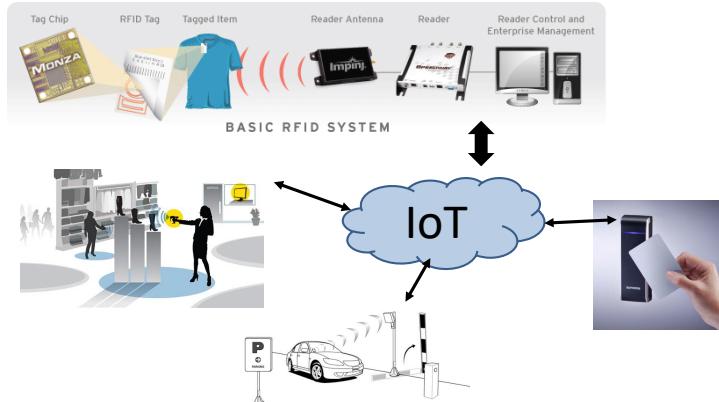
**Definición**  
La identificación por radiofrecuencia o **RFID** (*Radio Frequency Identification*), es una tecnología de identificación remota e inalámbrica en la cual un dispositivo lector o **reader**, se comunica a través de una antena con un **transponder** (también conocido como tag o etiqueta) mediante ondas de radio.

## Tecnología RFID

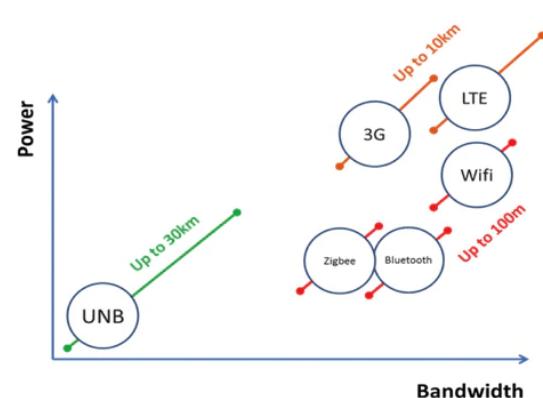
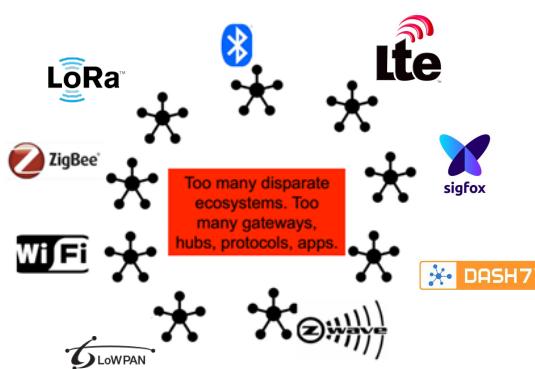
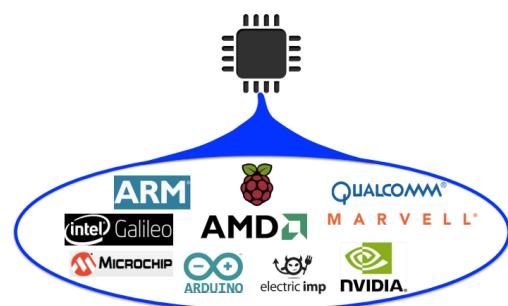
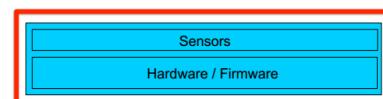
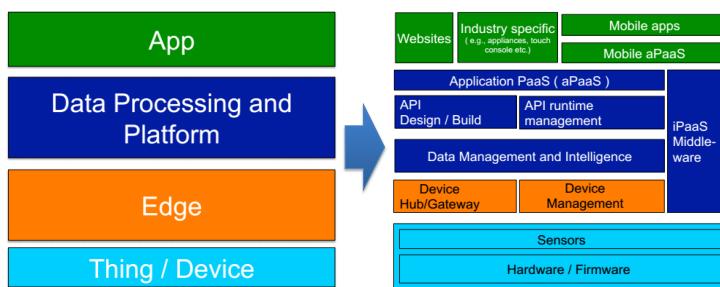


- El alcance de lectura varía de unos cuantos centímetros a decenas de metros, en función de la frecuencia que se utilice, de la potencia y de la sensibilidad direccional de la antena. La tecnología HF tiene un alcance máximo de lectura de unos tres metros. La tecnología UHF proporciona un alcance de lectura de 20 metros o más.
- La presencia de metales y líquidos puede causar interferencias que afecten a la lectura/escritura.
- Los chips RFID son difíciles de hackear.
- Numerosas normas que garantizan la diversidad de frecuencias y aplicaciones.

# Tecnología RFID e IoT

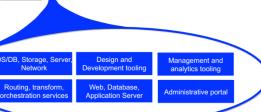


## Arquitectura de referencia en IoT



## Data Processing and Platform

Open Application PaaS (aPaaS) </>



## API Design / Build

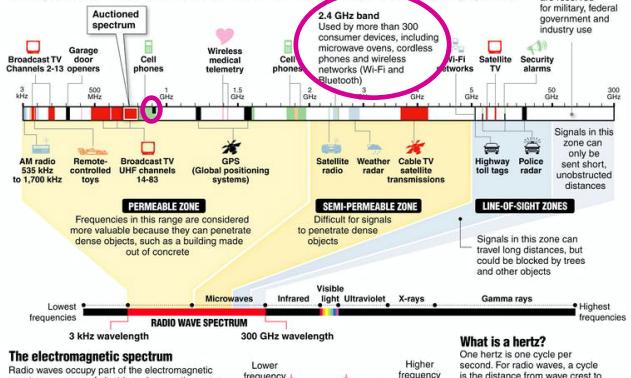


## API runtime management



## Inside the radio wave spectrum

Almost every wireless technology – from cell phones to garage door openers – uses radio waves to communicate. Some services, such as TV and radio broadcasts, have exclusive use of their frequency within a geographic area. But many devices share frequencies, which can cause interference. Examples: radio waves used by everyday devices:



### The electromagnetic spectrum

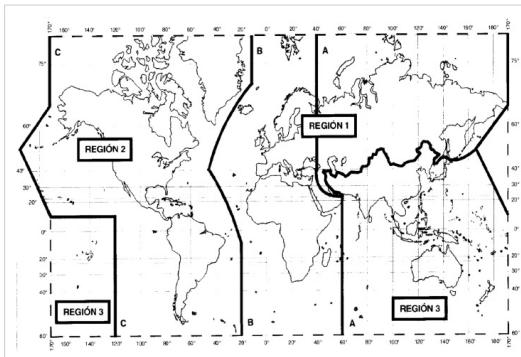
Radio waves occupy part of the electromagnetic spectrum. They travel at the speed of light, and other parts of the spectrum include visible light and x-rays; the shortest wavelengths have the highest frequency, measured in hertz.

Source: New America Foundation, MCT, Howstuffworks.com

Graphic: Nathaniel Lavor, Sacramento Bee

Source: <https://theconversation.com/wireless-spectrum-is-for-sale-but-what-is-it-11794>

## Espectro electromagnético



## Espectro electromagnético

ATRIBUCIÓN A LOS SERVICIOS según el RR de la UIT		
8.3 - 110 kHz		
Región 1	Región 2	Región 3
70 - 72 RADIONAVEGACIÓN 5.60	74 - 76 FUJO MOVIL MARITIMO 5.57 RADIONAVEGACIÓN 5.60	78 - 72 RADIONAVEGACIÓN 5.60
72 - 84 FUJO MOVIL MARITIMO 5.57 RADIONAVEGACIÓN 5.60	72 - 84 FUJO MOVIL MARITIMO 5.57 RADIONAVEGACIÓN 5.60	Móvil Marítimo 5.57 5.59
84 - 86 RADIONAVEGACIÓN 5.60	84 - 86 RADIONAVEGACIÓN 5.60	84 - 86 RADIONAVEGACIÓN 5.60
86 - 90 FUJO MOVIL MARITIMO 5.57 RADIONAVEGACIÓN 5.60	86 - 90 FUJO MOVIL MARITIMO 5.57 RADIONAVEGACIÓN 5.60	Móvil Marítimo 5.57 5.60
90 - 110 RADIONAVEGACIÓN 5.62	90 - 110 RADIONAVEGACIÓN 5.62	90 - 110 RADIONAVEGACIÓN 5.62
Fijo 5.64		Fijo 5.64

ATRIBUCIÓN NACIONAL USOS OBSERVACIONES		
8.3 - 110 kHz		
70 - 72 RADIONAVEGACIÓN	R 5.60	UN-114, UN-117
72 - 84 FUJO MÓVIL MARÍTIMO 5.57 RADIONAVEGACIÓN 5.60	M M R 5.58 5.57 5.60	UN-114, UN-117
84 - 86 RADIONAVEGACIÓN	R 5.60	UN-114, UN-117
86 - 90 FUJO MÓVIL MARÍTIMO 5.57 RADIONAVEGACIÓN 5.60	M M R 5.58 5.57	UN-114, UN-117
90 - 110 RADONAVEGACIÓN Fijo	R 5.62 5.64	UN-114, UN-117

<https://www.boe.es/eli/es/o/2021/12/16/etd1449/dof/spa/pdf>

## Espectro electromagnético

ATRIBUCIÓN A LOS SERVICIOS según el RR de la UIT		
495 - 1800 kHz		
Región 1	Región 2	Región 3
495 - 505 MÓVIL MARÍTIMO 5.82C		
505 - 526.5 MÓVIL MARÍTIMO 5.79 5.79A 5.84	506 - 510 MÓVIL MARÍTIMO 5.79 5.79A 5.84	505 - 526.5 MÓVIL MARÍTIMO 5.79 5.79A 5.84
510 - 526.5 MÓVIL MARÍTIMO 5.79 5.79A 5.84	510 - 526.5 MÓVIL MARÍTIMO 5.79 5.79A 5.84	510 - 526.5 MÓVIL MARÍTIMO 5.79 5.79A 5.84
526.5 - 1606.5 RADIODIFUSIÓN 5.86 RADIONAVEGACIÓN AERONÁUTICA	526.5 - 1606.5 RADIODIFUSIÓN 5.86 RADIONAVEGACIÓN AERONÁUTICA	526.5 - 1606.5 RADIODIFUSIÓN 5.86 RADIONAVEGACIÓN AERONÁUTICA
526.5 - 1606.5 RADIODIFUSIÓN 5.86 RADIONAVEGACIÓN AERONÁUTICA	526.5 - 1606.5 RADIODIFUSIÓN 5.86 RADIONAVEGACIÓN AERONÁUTICA	526.5 - 1606.5 RADIODIFUSIÓN 5.86 RADIONAVEGACIÓN AERONÁUTICA
535 - 1605 RADIODIFUSIÓN	535 - 1606.5 RADIODIFUSIÓN	535 - 1606.5 RADIODIFUSIÓN
5.87 5.87A	5.88	5.88

AM

ATRIBUCIÓN NACIONAL USOS OBSERVACIONES		
495 - 1800 kHz		
495 - 505 MÓVIL MARÍTIMO RADIONAVEGACIÓN AERONÁUTICA	M	UN-114, UN-117 LA FRECUENCIA 500 kHz ES NAL DE SOCORRO Y LLAMADA EN RADOTEX
505 - 526.5 MÓVIL MARÍTIMO RADIONAVEGACIÓN AERONÁUTICA	M R	5.79 5.79A 5.84 UN-114, UN-117 LA FRECUENCIA 515 kHz SE DESTINA PARA TRANSMISIÓN DE AVISOS A NAVEGANTES (NAVTEX)
526.5 - 1606.5 RADIODIFUSIÓN SONORA EN Onda Media (526.5-6106.5 kHz)	P	UN-114, UN-117
526.5 - 1606.5 RADIODIFUSIÓN SONORA EN Onda Media (526.5-6106.5 kHz)	P	UN-114, UN-117

<https://www.boe.es/eli/es/o/2021/12/16/etd1449/dof/spa/pdf>

## Espectro electromagnético

ATRIBUCIÓN A LOS SERVICIOS según el RR de la UIT		
75.2 - 137.175 MHz		
Región 1	Región 2	Región 3
76.2 - 87.6 FUJO MÓVIL, selvo móvil aeronáuticos	75.2 - 78.4 FUJO MÓVIL, selvo móvil aeronáuticos	
87.5 - 100 RADIODIFUSIÓN	87.5 - 100 RADIODIFUSIÓN	87.5 - 100 RADIODIFUSIÓN
100 - 108 RADIODIFUSIÓN	100 - 108 RADIODIFUSIÓN	100 - 108 RADIODIFUSIÓN
108 - 117.975 RADONAVEGACIÓN AERONÁUTICA		108 - 117.975 RADONAVEGACIÓN AERONÁUTICA
Fijo 5.194		Fijo 5.194

ATRIBUCIÓN NACIONAL USOS OBSERVACIONES		
75.2 - 137.175 MHz		
75.2 - 87.5 FUJO MÓVIL, selvo móvil aeronáuticos	*	UN-132, UN-154, UN-156 * Usos M y C (según nota UN)
87.5 - 108 RADIODIFUSIÓN	P	UN-117 Radiodifusión sonora en ondas medias (PN)
108 - 117.975 RADONAVEGACIÓN AERONÁUTICA	R	5.197A

<https://www.boe.es/eli/es/o/2021/12/16/etd1449/dof/spa/pdf>

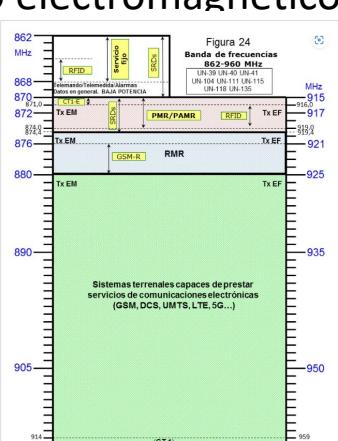
## Espectro electromagnético

ATRIBUCIÓN A LOS SERVICIOS según el RR de la UIT		
460 - 890 MHz		
460 - 489 FUJO MÓVIL, selvo móvil aeronáuticos por satélite (espacio-Terrestre)	S 5.291 5.292	
470 - 494 RADIODIFUSIÓN	S 5.293 5.295	
512 - 608 RADIODIFUSIÓN	S 5.291 5.298	
608 - 614 RADIODIFUSIÓN	S 5.291 5.298	
614 - 694 RADIODIFUSIÓN	S 5.291 5.298	
694 - 790 RADIODIFUSIÓN	S 5.291 5.298	
800 - 810 RADIODIFUSIÓN	S 5.291 5.298	
810 - 890 RADIODIFUSIÓN	S 5.291 5.298	
890 - 914 RADIODIFUSIÓN	S 5.291 5.298	
914 - 954 RADIODIFUSIÓN	S 5.291 5.298	

Televisión Digital Terrestre

<https://www.boe.es/eli/es/o/2021/12/16/etd1449/dof/spa/pdf>

## Espectro electromagnético(España)



# Protocolos de comunicaciones en IoT

## Multitud de protocolos:

### Nivel enlace y red:

- **802.15.4**
  - **6LoWPAN**
  - **Zigbee**
- **LoRaWAN**
- Bluetooth
- WiFi
- Weightless protocol
- Proveedores:
  - Ingenu (M2M)
  - **Sigfox**

### Nivel de aplicación:

- Constrained Application Protocol (**CoAP**)
- MQ Telemetry Transport (**MQTT**)
- Lightweight M2M (LWM2M)
- REST API
- XMPP (Extensible Messaging and Presence Protocol)

## Internet Engineering Task Force (IETF):

- [6LoWPAN Working Group \(IPv6\\_global\)](#)
- [CoRE Working Group](#) (Rest for IoT, CoAP , Resource Directory)

## OMA SpecWorks:

- Lightweight M2M ([LWM2M](#)), basado en CoAP, DTLS, REST

## ETSI: Estandarización en comunicaciones M2M, CoAP, HTTP binding... – NGSI-LD, context management (Fiware)

## W3C : Efficient XML Interexchange (EXI), estándares web

## ZigBee Alliance

## IEEE: IEEE 802.XX.YY

## DASH7 Alliance: Dash7 protocol (bi-directional, sub-Ghz medium range wireless communication)

## OASIS: Message Queuing Telemetry Transport (MQTT)

# IPv6

- [IPv6](#) permite que un dispositivo IoT obtenga de forma sencilla una dirección IP global, facilitando la comunicación peer-to-peer
- Comunicación usando tecnologías inalámbricas que requieran menos consumo energético por parte de los dispositivos.
- Las redes 6LoWPAN proporcionan mecanismos de encapsulado y compresión de cabeceras lo que reduce los tiempos de transmisión y ofrece conectividad a Internet sin un overhead excesivo.

# Diferencias entre IPv4 e IPv6

## Tamaño de la dirección IP:

- IPV4: 32 bits
- IPv6: 128 bits

## Método de direccionamiento:

- IPV4: numérico y bits binarios separados por un punto
- IPv6: alfanumérico y bits separados por dos puntos (:)

## Clases de direcciones IP:

- IPv4: cinco clases de direcciones IP diferentes
- IPv6: número casi ilimitado de direcciones IP. Soporte para rangos privados

# Diferencias entre IPv4 e IPv6

## Configuración:

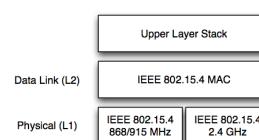
- IPV4: cada sistema debe ser configurado para poder comunicarse con otros. La red también se configura de manera manual o con DHCP.
- IPv6: configuración opcional según las funciones. Soporta la autoconfiguración entre dispositivos IPv6

## Interoperabilidad

- IPV4: topologías de red relativamente restringidas, con capacidad limitada de interoperabilidad y movilidad
- IPv6: capacidad de interoperabilidad y movilidad incluida en los dispositivos de red

# IEEE 802.15.4

- [Estándar](#) muy empleado de redes inalámbricas de área personal (WPAN). 1ª versión: 2003
- Utilizado en redes domésticas, control industrial, automatización edificios, WSN... → corto alcance
- Define control físico y control de acceso al medio (MAC)
- Tres bandas de frecuencias: 868 MHz, 915 MHz, 2.4 GHz
- Tasas bajas de transferencia: 20kbps - 250kbps
- Baja potencia de emisión: 0.5-1 mW
- Rango de entre 10m a 100m
- Modulación: DSSS



# IPv6

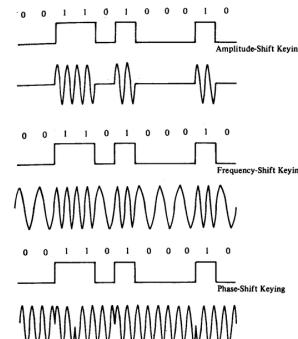
## Otras ventajas

- Jerarquía estructurada para disminuir tamaño de tablas de enrutamiento. Desacopla prefijo (64bits) del identificador del host (64 bits)
- Puede ser mejorado con IPsec (Internet Protocol Security, en inglés) para gestionar la encriptación y autenticación entre hosts
- Soporta el protocolo IPv6 móvil, MIPv6, que permite a los dispositivos móviles cambiar de una red a otra y recibir notificaciones itinerantes

# IEEE 802.15.4

## Modulación:

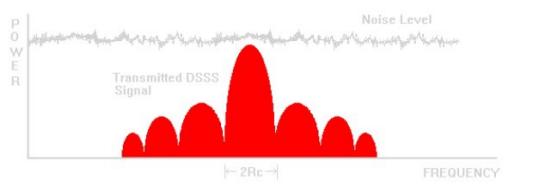
- DSSS: Direct-Sequence Spread Spectrum
- PSK: Phase-Shift-Keying. Modulación por desplazamiento de fase



## IEEE 802.15.4

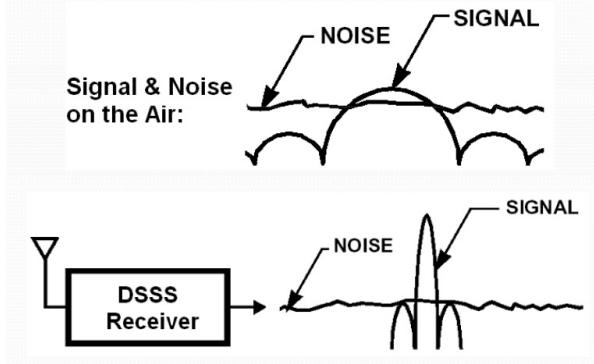
- Modulación DSSS:

- Primero se modula digitalmente una señal de pseudo-ruido (PRN) con la señal de información.
- Después, se multiplica la portadora RF y la señal PN modulada. De esta forma la señal de RF se transforma en una señal con un gran ancho de banda y un espectro equivalente al de una señal de ruido.
- En el receptor, se recupera la señal original y se elimina la interferencia.
- GPS, Galileo, GLONASS, IEEE 802.11b, IEEE 802.15.4, WiFi, ....



## IEEE 802.15.4

- Mayor inmunidad a ruido y/o interferencias:

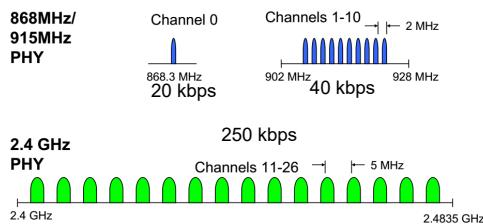


(\*) Spread Spectrum and Wi-Fi Basics by Syed M. Mahmu

## IEEE 802.15.4

- Nivel físico. Funciones:

- Activación/desactivación del módulo radio
- Detección de ocupación del canal para Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)
- Indicador de calidad del enlace en la recepción de paquetes
- Selección de canal de frecuencia (27 canales)

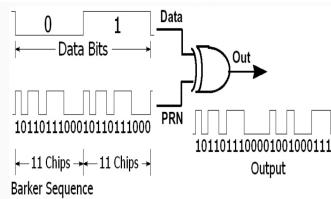


## IEEE 802.15.4

- Nivel de enlace:

- Algoritmo MAC (Media Access Control) basado en el protocolo de acceso a redes CSMA (Carrier Sense Multiple Access)
- Primero espera hasta que el canal esté inactivo.
- Una vez que el canal esté libre, comienza a enviar los data frames.
- El receptor reconoce la correcta recepción de un data frame .
- Si el remitente no recibe un acuse de recibo, se reintenta la transmisión de datos.

## IEEE 802.15.4



La salida de la XOR es modulada en una portadora usando BPSK o QPSK

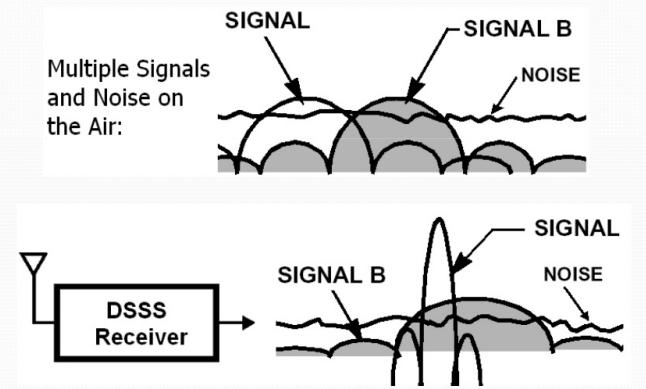
BPSK Encoding	
XOR Output	Phase Change
0	0
1	$\pi$

2-Bit (d0,d1) XOR Output (d0 is the first bit in time)	Phase Change
00	0
01	$\pi/2$
11	$\pi$
10	$3\pi/2$

(\*) Spread Spectrum and Wi-Fi Basics by Syed M. Mahmu

## IEEE 802.15.4

- Acceso múltiple usando diferentes códigos PRN:



(\*) Spread Spectrum and Wi-Fi Basics by Syed M. Mahmu

## IEEE 802.15.4

- Nivel físico. Formato de trama:

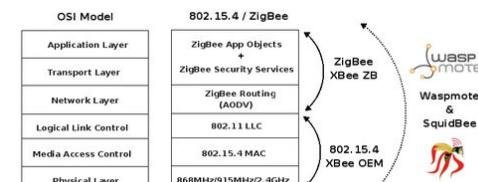
- Preámbulo: sincronización
- Indicador de comienzo de paquete: "11100101"
- Cabecera de nivel físico: tamaño de la trama
- *Payload*: hasta 127 bytes



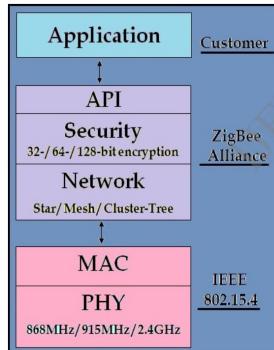
Zigbee



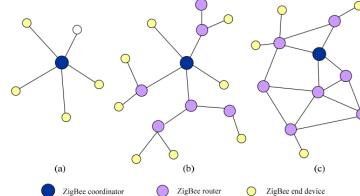
- Protocolo estándar abierto que provee funcionalidad y características adicionales sobre 802.15.4
- Promovido por la Zigbee Alliance desde 2002.



- Características adicionales a IEEE 802.15.4
  - Servicios añadidos de cifrado: a nivel de red y aplicación
  - Asociación y autenticación: durante el proceso de asociación a la red
  - Direccionamiento a nivel de red
  - Enrutamiento: basado en árbol o protocolo reactivo basado en AODV



- Topología en estrella
  - Comunicación a través del coordinador PAN
  - Pueden ser una combinación de FFD (Full Functional Device) y RFD (Reduced Functions Device).
  - Coordinador PAN suele ser un dispositivo confiable conectado a la red eléctrica



Tipos de nodos lógicos

- ▶ Dispositivo final
  - RFD o FFD sin tareas de control
- ▶ Router Zigbee
  - FFD con tareas de gestión y control de la red
- ▶ Coordinador Zigbee
  - Controlador principal de la red. Una red solo puede tener uno

## Zigbee: Topologías

- Topología peer-to-peer (mesh)
  - Extensión de la topología en estrella para comunicación directa entre dispositivos
  - Enrutamiento
- Topología en árbol clusterizada
  - Varios coordinadores conectados entre sí dan servicio a nodos finales
  - Un coordinador es designado coordinador PAN

## 6LoWPAN

- Capa de adaptación para transportar paquetes IPv6 sobre Low-Power Wireless Personal Area Networks (LP-WPAN)
- Definido sobre el estándar IEEE 802.15.4
- Está siendo adaptado también para otros protocolos a nivel de enlace (Bluetooth Smart, Low-power Wi-Fi, Power Line Control (PLC))
- Interoperabilidad con otras tecnologías
- Integración con Internet transparente
  - Permite el uso de API de sockets estándar
- Uso mínimo de código y memoria

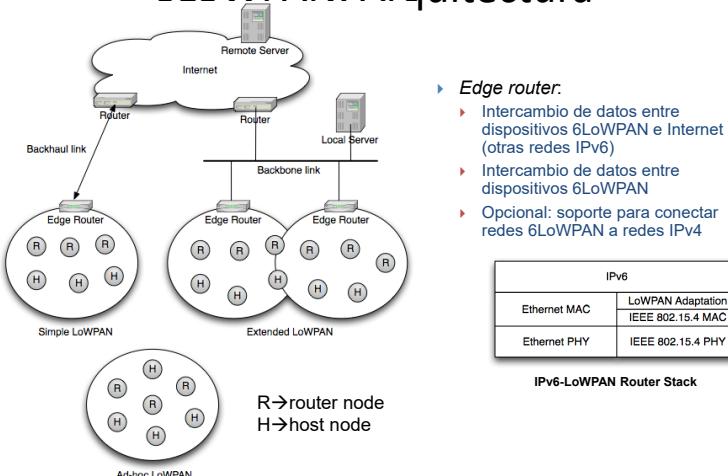
## 6LoWPAN

- Problemas de IPv6 para trabajar en WSN:
  - Direcciones IPv6 muy largas (128 bits): soporte para direcciones de 64 bits y 16 bits 802.15.4
  - Cabecera IPv6 muy larga: compresión eficiente de cabeceras
    - Compresión IPv6
    - Compresión cabeceras de extensión IPv6
    - Compresión cabecera UDP
  - Autoconfiguración de red → uso de *network Discovery* con intensos envíos multicast

## 6LoWPAN

- Los protocolos de enrutamiento ad-hoc suelen introducir mucho overhead → 6LoWPAN emplea su propio protocolo de enrutamiento eficiente (RPL)
- Estándares del IETF sobre 6LoWPAN:
  - RFC 4944 - cabeceras
  - RFC 6282 - formato de compresión
  - RFC 6550 - *routing* (RPL)
  - RFC 6775 - *neighbour discovery*

## 6LoWPAN: Arquitectura

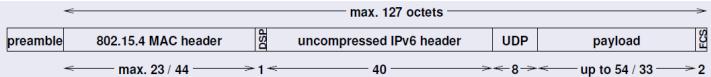


## 6LoWPAN: Resolución de direcciones

- A los nodos IPv6 se le asignan direcciones IP de 128 bits
- Los dispositivos IEEE 802.15.4 pueden usar direcciones IEEE de 64 bits o, después de un evento de asociación, direcciones de 16 bits que son únicas dentro de una Personal Area Network.
- 6LoWPAN comprime las direcciones IPv6
  - Quitando el prefijo de IPv6 (primeros 64 bits)
  - El prefijo global ya es conocido por todos los nodos de la red
  - Comprimiendo el identificador de la interfaz (IID)
  - Se quita para comunicaciones locales
  - Compresión utilizando un “contexto” bien conocido (RFC-6282)
  - Compresión de direcciones multicast

# 6LoWPAN – Adaptación de los tamaños de paquetes

- Peor escenario: sin compresión



- Cálculo de la cabecera.

- Cabecera IPv6 → 40 octetos
- Cabecera UDP → 8 octetos
- Cabecera MAC de 802.15.4
  - 23+2=25 octetos (sin seguridad)
  - 44+2=46 octetos (si se utiliza seguridad AES-CCM-128)

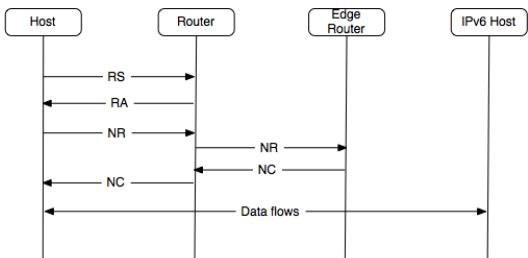
## 6LoWPAN - Fragmentación

- Bajo rendimiento de paquetes grandes IPv6 fragmentados sobre redes LoWPAN
  - La pérdida de fragmentos causa que los paquetes se retransmitan
  - Bajo ancho de banda y alto retardo del canal inalámbrico
  - Los protocolos de aplicación sobre 6LoWPAN deben evitar fragmentación
  - Los protocolos de aplicación deberían aplicar compresión cuando se usan en 6LoWPAN

## 6LoWPAN - Fragmentación

- IPv6 tiene un límite inferior en MTU (*Maximum Transmission Unit*) de 1280 octetos. Es decir, IPv6 no fragmenta los paquetes por debajo de este límite. Para enviar IPv6 a través de un enlace con una MTU de menos de 1280 octetos, la capa de enlace debe fragmentar y desfragmentar con transparencia los paquetes IPv6.
- Tamaño trama 802.15.4 es 127 octetos, queda para datos:
  - 127-25-40-8 = **54 octetos** (sin seguridad)
  - 127-46-40-8 = **33 octetos** (usando AES-CCM-128)
- Se necesita fragmentación y reensamblado para encajar los 1280 octetos IPv6 en los 33 (o 54) octetos de 802.15.4

## 6LoWPAN – Network Discovery



RS → Router Solicitation. Búsqueda de router  
RA → Router Advertisement. Respuesta de router  
NR → Node Registration. Registro de un nodo  
NC → Node Confirmation. Confirmación de registro

## 6LoWPAN - Seguridad

- En redes inalámbricas el canal es muy vulnerable
- Nivel 2: Mecanismos de IEEE 802.15.4
  - Basado en 128-bit Advanced Encryption Standard (AES)
  - Muchos dispositivos incluyen chips para procesamiento AES128
- Nivel 3: IPSec standard [RFC4301] seguridad IP, *end-to-end*
  - Dos formatos de cabecera
    - Authentication Header (AH) in [RFC4302]
      - Sólo integridad y autenticación
    - Encapsulating Security Payload (ESP) [RFC4303] (más usado)
      - También cifra para conseguir confidencialidad

## 6LoWPAN – Neighbor Discovery

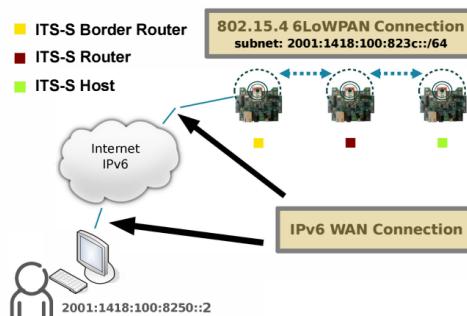
- El standard de *Neighbor Discovery Protocol* (NDP) para IPv6 no es apropiado para 6LoWPAN:
  - Asume que los nodos están operativos
  - Uso intensivo de multicast
- 6LoWPAN *Neighbor Discovery* proporciona:
  - Un enlace adecuado y modelo para comunicaciones inalámbricas de baja potencia
  - Minimiza el tráfico de control requerido por los nodos
  - Node Registration (NR) and Confirmation (NC)
  - Duplicate Address Detection (DAD) and recovery
  - Soporte para infraestructuras *extended Edge Router*
- RFC 6775 → Define el *Neighbor Discovery* para 6LoWPAN

## 6LoWPAN - Routing

- *Routing Over Low power and Lossy networks* (ROLL)
  - Grupo de trabajo del IETF
- Protocolo RPL “Ripple”
  - *IPv6 Routing Protocol for Low-Power and Lossy Networks*
  - RFC 6550
  - Cada router envía información a sus vecinos sobre toda la red, para calcular el camino más corto al destino
  - Utiliza varias métricas. Diferentes función objetivo
  - Detección de inconsistencias: evitar bucles, mantener convergencia, etc.
- Otros protocolos ad-hoc: AODV, OLSR, BATMAN, JOKER...

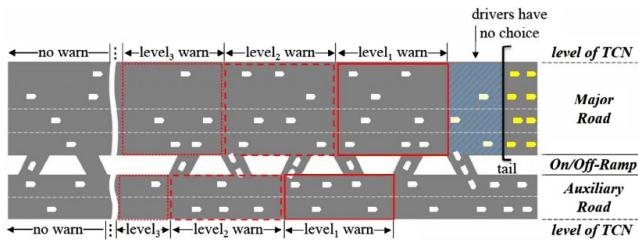
## 6LoWPAN: Caso de uso

- *Intelligent Transportation Systems (ITS)*



# 6LoWPAN: Caso de uso

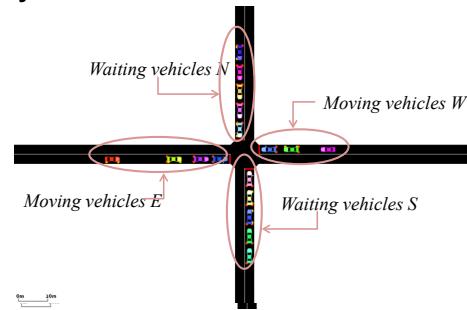
- ITS: redistribución de tráfico



Q. Yuan, Z. Liu, J. Li, J. Zhang, and F. Yang, "A traffic congestion detection and information dissemination scheme for urban expressways using vehicular networks," *Transp. Res. Part C Emerg. Technol.*, vol. 47, pp. 114–127, Oct. 2014.

# 6LoWPAN: Caso de uso

- ITS: ajuste tiempo semáforos

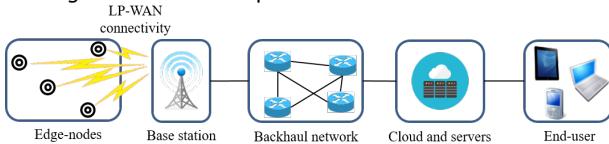


R. Sanchez-Iborra, J. F. Ingles-Romero, G. Domenech-Asensi, J. L. Moreno-Cegarra, and M.-D. Cano, "Proactive intelligent system for optimizing traffic signaling," in Proc. 14th IEEE International Conference on Pervasive Intelligence and Computing (PICoM-2016), 2016, pp. 544–551.

## LP-WAN (Low-Power Wide Area Networks)

- Características

- Largo alcance como las redes celulares (o más)
- Bajo consumo como las redes de sensores
- Alta escalabilidad
- Topología en estrella o estrella de estrellas
- Uso de frecuencias libres ISM sub-GHz (868 MHz o 902 MHz)
- Dispositivos de bajo coste
- Enlaces asimétricos: mayor valor al *uplink*
- *Roaming*: conexión del dispositivo a distintas estaciones base



## LP-WAN: Sigfox



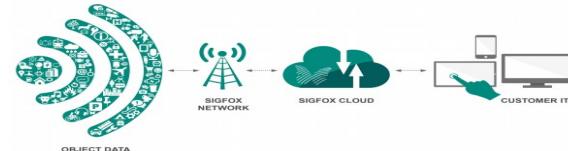
- Limitaciones técnicas:
  - Límite de mensajes: 140 mensajes al día (*duty cycle*)
  - Tamaño máximo de *payload*: 12 bytes
  - Seguridad: no ofrece ni cifrado ni seguridad extremo a extremo. Se asume que sólo el usuario conoce el contenido y significado del *payload*. Saltos en frecuencia
- Éxito: modelo de negocio
  - Muy buena cobertura y escalabilidad
  - Amplio despliegue: varios países europeos cubiertos, mediante convenios con operadores de servicios móviles. España: Cellnex
  - Instalación muy simple
  - Alta eficiencia energética: las baterías duran más de 10 años
  - Ejemplo: Securitas Direct

## LP-WAN: Sigfox



- Protocolo propietario
- Modulación Ultra Narrow Band (200 Hz). Differential Binary Phase Shift Keying (DBPSK )
- Tasa de transmisión muy limitada: 100 bps
- Utilización de bandas libres ISM (Industrial, Scientific and Medical) sub-GHz: 868 MHz (Europa) y 902 MHz (EEUU)
- Largos alcances y penetración (10 km en campo abierto y 2-3 km en zona urbana) y muy bajo consumo de energía

## LP-WAN: Sigfox



Estaciones base Sigfox en España

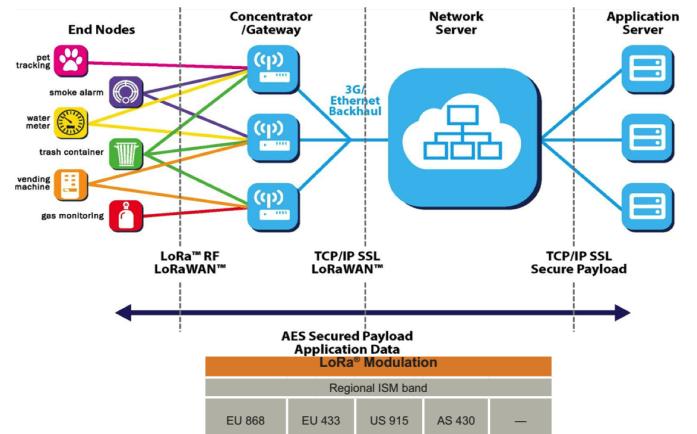


## LP-WAN: LoRaWAN

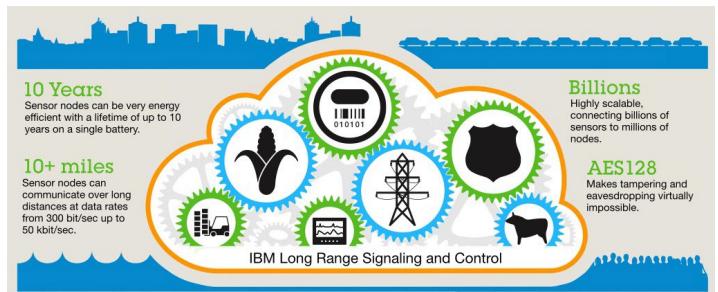


- Conectividad de largo alcance (similar a Sigfox), tanto en uplink como en downlink:
  - Nodos Clase A: después de cada transmisión en uplink, el nodo abre dos ventanas de escucha para recibir: menor consumo energético
  - Nodos Clase B: misma funcionalidad que Clase A, pero además, los nodos abren unas ventanas de escucha de forma programada: consumo energético medio
  - Nodos Clase C: ventana de escucha siempre abierta: dispositivos conectados a alimentación
- LoRa: Long Range.

## LP-WAN: LoRaWAN



	Europe	North America	China	Korea	Japan	India
Frequency band	867-869MHz	902-928MHz	470-510MHz	920-925MHz	920-925MHz	865-867MHz
Channels	10	64 + 8 + 8				
Channel BW Up	125/250kHz	125/500kHz				
Channel BW Dn	125kHz	500kHz				
TX Power Up	+14dBm	+20dBm typ (+30dBm allowed)				
TX Power Dn	+14dBm	+27dBm				
SF Up	7-12	7-10				
Data rate	250bps- 50kbps	980bps-21.9kbps				
Link Budget Up	155dB	154dB				
Link Budget Dn	155dB	157dB				



## LP-WAN: LoRaWAN

### Caso de uso

- E-health



J. Petäjäjärvi, K. Mikhaylov, M. Hämäläinen and J. Iinatti, "Evaluation of LoRa LPWAN technology for remote health and wellbeing monitoring," 2016 10th International Symposium on Medical Information and Communication Technology (ISMICIT), Worcester, MA, 2016, pp. 1-5.

## CoAP: Constrained Application Protocol



- IETF CoRE Working Group: RFC 7252 (2014); RFC 7641: extensión OBSERVE (2015)
- Protocolo eficiente modelo cliente-servidor basado en RESTful
- Soporta operaciones GET / PUT / POST / DELETE (como HTTP)
- Aparentemente sencillo, pero muy potente

## Sigfox vs. LoRaWAN

- LoRaWAN es una solución tecnológicamente más avanzada
- LoRaWAN presenta mejores prestaciones y parámetros de configuración adaptables a los distintos entornos de transmisión
- Sigfox tiene una arquitectura de red más amigable para el usuario final que no quiere invertir en infraestructura ni sabe como gestionar sus datos en la nube

## CoAP: Constrained Application Protocol



- Usuarios CoAP:
  - Usuarios de servicios web: CoAP implementa un protocolo para servicios web
  - Otros dispositivos CoAP (máquinas)
  - Servidores de gestión, p.e., uso de LWM2M, para acceder y gestionar un dispositivo restringido usando CoAP
- CoAP es un rediseño (no una simple compresión) de HTTP. CoAP puede ser traducido a HTTP para interoperabilidad e integración con la WEB
- CoAP se ha definido sobre UDP como protocolo de transporte

## CoAP: Constrained Application Protocol

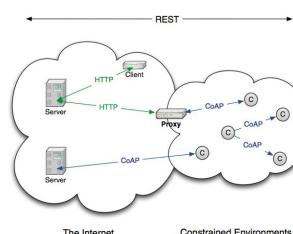


- Emplea recursos limitados:
  - Tamaño mensajes reducido (cabecera 4 bytes)
  - Redes *constrained*, ej. Low-Power and Lossy Networks (LLN)
  - Soporta nodos inactivos, uso de *proxies*
- Reduce las ineficiencias de operaciones REST
  - No codifica en texto plano, y reduce el tamaño de los mensajes
  - No utiliza TCP que añade overhead, sino UDP
- Permite operaciones *Machine to Machine* (M2M)
  - Descubrimiento de recursos
  - publicación / suscripción / notificación
  - Soporte *multicast*

## CoAP: Constrained Application Protocol



The CoAP Architecture



	Traditional IP	IoT protocols
Application protocol	HTTP (and related protocol, eg SMTP)	CoAP
Transport layer	TCP (or UDP)	UDP only
Network layer	IPv4 / IPv6	6LoWPAN
Link layer	802.11n (or ethernet)	802.15.4e

## Mensajes CoAP

CoAP

## Mensajes CoAP

CoAP

- Se utiliza UDP (no orientado a la conexión, no garantía entrega)
- Tipo de mensajes:
  - Request: Confirmable/Non-confirmable
  - Response: Acknowledgement/Reset
- Control de entrega realizado a nivel de aplicación por el protocolo CoAP.
  - Mensajes confirmables: requiere de un ACK
  - Mensajes no-confirmables: *fire and forget*
- Seguridad → DTLS (Datagram Transport Layer Security)

### Formato:

```

0   1   2   3   4   5   6   7   8   9   0   1   2   3   4   5   6   7   8   9   0   1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|Ver| T | TKL |   Code   |       Message ID        |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Token (if any, TKL bytes) ... |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Options (if any) ... |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|1 1 1 1 1 1 1|   Payload (if any) ...
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

- **T (type)** → indica el rol del mensaje como parte de la transacción. CON / ACK / NON / RST
- **TKL** → token length
- **code** → da información adicional sobre el propósito del mensaje:
  - Request o response. GET, POST, PUT, DELETE
- **Message id** → valor único a la transacción
- **token** → para especificar el concepto de "topic"
- **options** → para incluir parámetros y mecanismos de gestión de mensajes

## Mensajes CoAP

CoAP

## CoAP

### Implementaciones

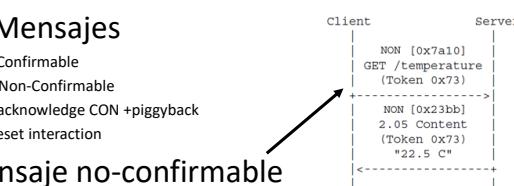
- Contiki-2.6: Erbium. <http://www.contiki-os.org>
- C: libcoap <http://sourceforge.net/projects/libcoap/develop>
- .NET (C#) CoAPSharp <http://www.coapsharp.com>
- Python. <http://sourceforge.net/projects/coapy>
- Java. Californium <https://github.com/mkovatsc/californium>
- Firefox Javascript browser plugin: Cooper <https://github.com/mkovatsc/Copper>

### Proxies:

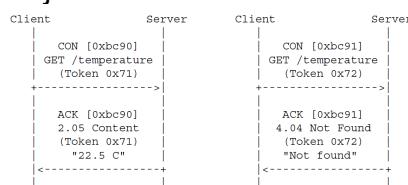
- Squid 3.1.9 with transparent HTTP-CoAP mapping module <http://telecom.dei.unipd.it/pages/read/90/>
- jcoap Proxy <https://code.google.com/p/jcoap/>
- Californium cf-proxy <https://github.com/mkovatsc/Californium>
- CoAPthon <https://github.com/Tanganelli/CoAPthon>

### Tipos Mensajes

- **CON** → Confirmable
- **NON** → Non-Confirmable
- **ACK** → acknowledge CON +piggyback
- **RST**→ reset interaction



- Ej. Mensaje no-confirmable
- Ej. Mensajes confirmables con ACK



## MQTT (Message Queue Telemetry Transport)

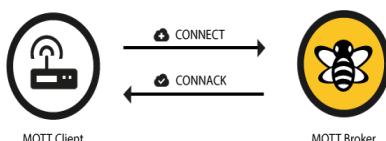


- Protocolo ligero de publicación/subscripción **sobre TCP/IP** para sensores, dispositivos y redes "*constrained*".
- Estándar de OASIS para IoT (2014), pero diseñado por IBM para conectar instalaciones petrolíferas vía satélite (1999)
- Ideal para situaciones con comunicaciones M2M e IoT:
  - Simple de implementar
  - Provee una capa de QoS
  - Requiere de poco ancho de banda
  - Agnóstico a las aplicaciones de capa superior
  - Permite el establecimiento continuo de una conexión (TCP)
- Amplio grado de desarrollo y despliegue: Arduino, Android/iOS, C/C++/C#, Java/JavaScript
- El productor publica el mensaje solo una vez (al servidor/*broker*)
- Múltiples consumidores (aplicaciones/dispositivos) reciben el mensaje a través del *broker*
- Desacopla al productor y al consumidor.
- El productor manda el mensaje con un *topic*
- Los consumidores se han suscrito previamente a ese *topic*
- El servidor/*broker* realiza la asociación entre publicaciones y suscripciones: recibe → filtra → asocia → re-envía
  - Si no hay ninguna asociación, el mensaje se descarta
  - Si uno o más mensajes cumplen la asociación, el mensaje se entrega al consumidor correspondiente

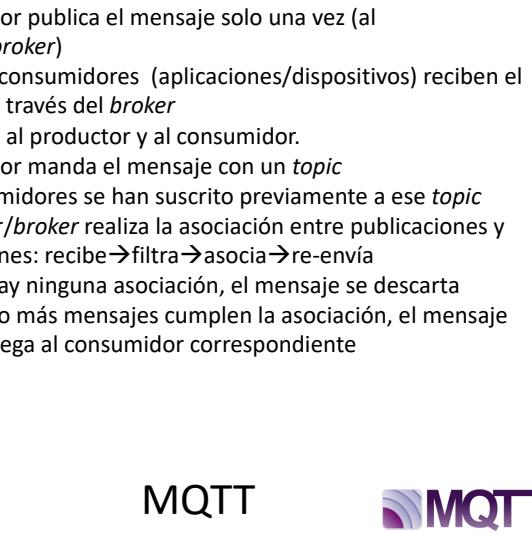
## MQTT



### MQTT Connection



## MQTT



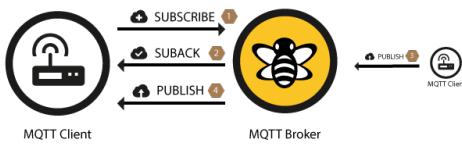
## MQTT



MQTT-Packet: CONNECT	MQTT-Packet: CONNACK
contains: <b>clientId</b> "client-1" <b>cleanSession</b> true <b>username</b> "hans" <b>password</b> "letmein" <b>lastWillTopic</b> "/hans/will" <b>lastWillQos</b> 2 <b>lastWillMessage</b> "unexpected exit" <b>lastWillRetain</b> false <b>keepAlive</b> 60	Example "client-1" true "hans" "letmein" "/hans/will" 2 "unexpected exit" false 60

MQTT-Packet: SUBSCRIBE	MQTT-Packet: SUBACK
contains: <b>packetId</b> 4312 <b>qos1</b> 1 <b>topic1</b> "topic/1" <b>qos2</b> 0 <b>topic2</b> "topic/2" ...	Example 4312 1 "topic/1" 0 "topic/2" ...

## Subscription/publishing

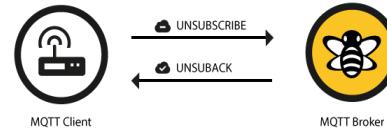


MQTT Client

MQTT Broker

MQTT-Packet: PUBLISH	
contains: packetId (always 0 for qos 0) topicName qos retainFlag payload dupFlag	Example 4314 "topic/1" 1 false "temperature:32.5" false

## Unsubscribe



MQTT Client

MQTT Broker

MQTT-Packet: UNSUBSCRIBE	MQTT-Packet: UNSUBACK
contains: packetId topic1 topic2 ... Example 4315 "topic/1" "topic/2" ...	contains: packetId Example 4316

- *Topics* con estructura jerárquica. Cada jerarquía se separa con '/'. Por ejemplo:
  - “edificio1/planta3/sala1/raspberry0/temperatura”
  - “edificio1/planta1/sala0/arduino0/ruido”
  - Suscripción agregada (no la publicación)  
ej.“edificio1/planta2/#”

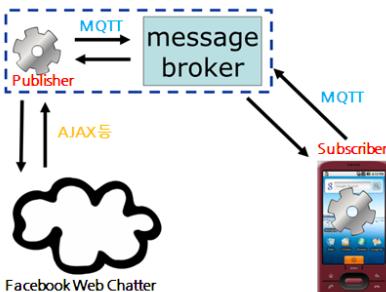
- Ejemplo de *topic*: Una casa que publica información sobre si misma:
  - <country>/<region>/<town>/<postcode>/<house>/energyConsumption
  - <country>/<region>/<town>/<postcode>/<house>/fireAlarm
  - <country>/<region>/<town>/<postcode>/<house>/floodingAlarm
- Un suscriptor se puede suscribir a un *topic* concreto usando un valor absoluto o *wildcards*
  - Single-level wildcards “+” → puede aparecer en cualquier lugar del nombre
  - Multi-level wildcards “#” → deben aparecer al final del namespace
  - Los *wildcards* se deben poner a continuación del separador
  - Ejemplos:
    - Spain/Murcia/Espinardo/30110/1/energyConsumption
      - Consumo de energía para una casa concreta en Espinardo
    - Spain/Murcia/Espinardo/+/-/energyConsumption
      - Consumo de energía para todas las casas de Espinardo
    - Spain/Murcia/Espinardo/30110/#
      - Consumo de energía y alarmas (2) para todas las casas con el código postal:30110

- Diseñado para *dispositivos constrained*:
  - Recursos limitados en cuanto a memoria, batería y CPU
  - Implementaciones de clientes MQTT para diferentes lenguajes
- Diseñado para *redes constrained*:
  - El protocolo comprime las cabeceras y tiene campos variables para reducir tamaño
  - Menor tamaño posible de paquete: 2 bytes
  - Testado en diferentes tipos de redes VSAT, GPRS, 2G....
- Soporta Calidad de servicio QoS para asegurar la entrega de mensajes de forma determinista. Niveles QoS:
  - 0 – mensaje enviado como mucho una vez (*fire and forget*) → entrega garantizada por TCP
  - 1 – mensaje entregado al menos una vez
  - 2 – mensaje entregado exactamente una vez
  - Productor y consumidor pueden tener niveles de QoS diferentes

- MQTT-SN: MQTT for Sensor Networks
- Aunque diseñado para dispositivos muy limitados, MQTT aún puede ser demasiado pesado para ciertos casos específicos:
  - Mantener conexión TCP
  - *Topics* excesivamente largos para algunos protocolos de capas inferiores (ej. 802.15.4)
- MQTT-SN: para dispositivos embebidos, sobre UDP
- Rediseño de algunos mensajes, predefinición (indexado) de algunos *topics*

## MQTT – Caso de uso

- Facebook messenger:



## Arquitectura y uso de Fiware

Internet de las cosas en el contexto de Big Data

Data

Máster Interuniversitario en Big Data: Tecnologías de Análisis de Datos Masivos  
Universidade de Santiago de Compostela (USC)

# Introducción



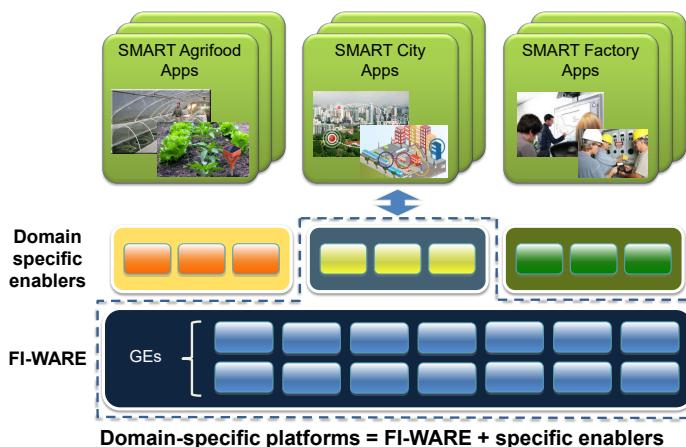
- Impulsada por la Comisión Europea y las principales empresas TIC europeas en 2011.
- Plataforma para el despliegue global de aplicaciones.
- Telefónica lidera el desarrollo de la plataforma FI-WARE.
- Las especificaciones de las APIs (Application Programming Interfaces) ofrecidas por los componentes de esta plataforma son abiertas, públicas y libres de royalties.

# Introducción

- Pilares:

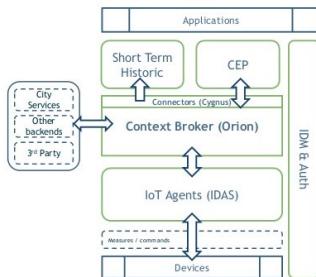
- FIWARE : Ofrece un conjunto amplio de APIs (Interfaces de aplicaciones) que facilitan el **desarrollo de aplicaciones** en varios sectores.
- FIWARE Lab : entorno donde se pueden probar todas las aplicaciones basadas en tecnologías FI-WARE. Es un **entorno de prueba** y no comercial.
- FIWARE Accelerate : un **programa de apoyo** a desarrolladores y emprendedores, con especial atención a las PYMEs y empresas de nueva creación
- FIWARE Ops : herramientas que facilitan el despliegue, configuración y operación de instancias FI-WARE por los proveedores de la plataforma.
- FIWARE Mundus : programa creado para extender la tecnología FI-WARE más allá de Europa incluyendo América Latina, África y Asia.

## Introducción



## Arquitectura básica de Fiware

1. Los **dispositivos físicos** actúan como fuentes de datos en la arquitectura.
2. **IoT Agents** es la puerta de entrada para que los dispositivos puedan comunicarse con el resto de los componentes de la arquitectura. Soporta diversos protocolos específicos (CoAP, MQTT, etc.) para recibir datos de sensores y se conecta al Context Broker el cual los almacena y los gestiona.
3. El **Context Broker** mantiene representaciones virtuales de los dispositivos físicos y mantiene el último dato entregado por los dispositivos.
4. Los componentes de gestión de **identidad y autorización** bridan los mecanismos para garantizar la confiabilidad, seguridad y privacidad en la entrega y uso de servicios.

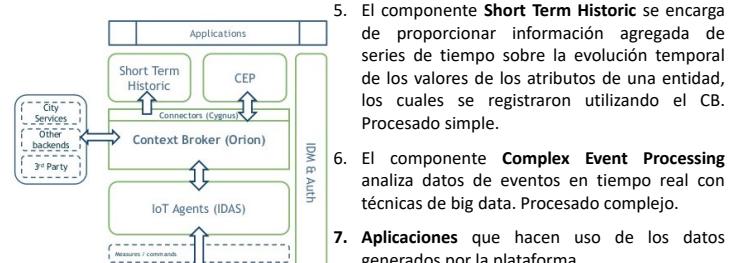


## Generic Enablers (GEs)

- Diferentes servicios disponibles en su catálogo.

1. Cloud Hosting
2. Data/Context Management
3. Interfaces to Network and Devices (I2ND)
4. Advanced Web-based User Interface
5. Security
6. Internet of Things
7. Applications/Services and Data Delivery

## Arquitectura básica de Fiware



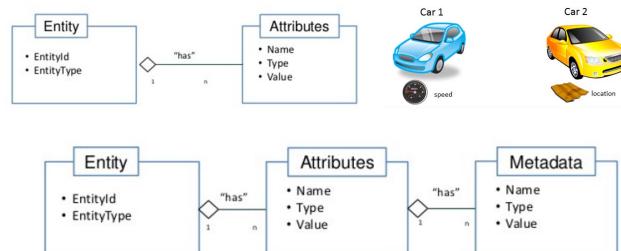
## Fiware Context Broker

- Las aplicaciones de IoT necesitan recoger y gestionar información que describa el "estado" actual de todo lo que está sucediendo a su alrededor (contexto): información de sensores u otros sistemas externos.

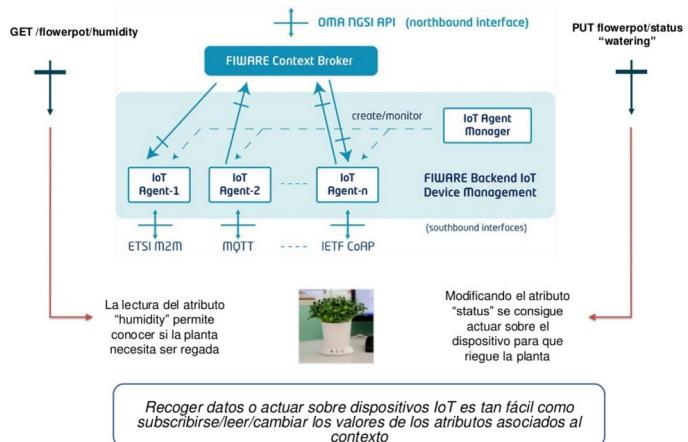


## Fiware Context Broker

- La API FIWARE NGSI de FIWARE Context Broker sirve para la integración de componentes y aplicaciones en la plataforma para actualizar o consumir información.



# Fiware Context Broker



# Fiware Context Management

- **Orion Context Broker**: componente central y obligatorio de cualquier plataforma o solución "Powered by FIWARE".
  - Proporciona **FIWARE NGSIV2 API**, API Restful simple pero potente que permite realizar actualizaciones, consultas o suscribirse a cambios en la información de contexto.
  - Obtiene información sobre el contexto actual.
- **Cygnus** ofrece los medios para administrar el historial de contexto que se crea como una secuencia de datos que se puede inyectar en múltiples sumideros de datos, incluidas algunas bases de datos populares como PostgreSQL, MySQL, MongoDB o AWS DynamoDB, así como plataformas BigData como Hadoop , Tormenta, Chispa o Flink.

## Fiware Context Processing, Analysis and Visualization

- **Cosmos** GE permite un análisis de Big Data más fácil sobre el contexto integrado con las plataformas de Big Data más populares (Orion Flink, Apache Flink, Apache Hadoop).
- **Wirecloud**: plataforma web que hace que sea más fácil desarrollar aplicaciones web altamente personalizables por los usuarios finales capaces de integrar datos heterogéneos.
- **Knowage**: plataforma de Business Intelligence que permite realizar análisis de negocios sobre fuentes tradicionales y sistemas de big data. Compuesto de varios módulos: BD (big data) para el análisis de datos en clústers BD o bases de datos NoSQL; SI (smart intelligence), ER (enterprise reporting); PM (performance management); PA (predictive analysis); EI (embedded intelligence)

## Publish/Subscribe Context Broker

### OPERACIONES:



- **Registrar** aplicaciones productoras de información de contexto, p. ej. un sensor de temperatura en una habitación.
- **Actualizar** el valor de contexto, p. ej. actualizar el valor del sensor de temperatura.
- **Recibir notificaciones** cuando se produce un cambio en la información de contexto o cada cierto tiempo, p. ej. recibir el valor de temperatura cada minuto.
- **Consultar** la información de contexto mediante **queries**.

## Fiware Context Processing, Analysis and Visualization

- **Kurento** (Stream-oriented Generic Enabler): procesamiento en tiempo real de transmisiones en streaming que admiten la transformación de cámaras de video en sensores, así como la incorporación de funciones de aplicación avanzadas (comunicaciones audiovisuales integradas, realidad aumentada, etc.)
- **FogFlow** es un framework distribuido para admitir flujos de procesamiento dinámico sobre el Cloud y el Edge.
- **OpenVidu** es una capa de abstracción para Kurento que facilita la programación.

## Publish/Subscribe Context Broker

Entity	Entity Type		
Bedroom1	Room		
Bedroom2	Room		
Kitchen	Room		
Frontdoor	Door		
Backdoor	Door		
Entity Type	Attr. Name	Attr. Type	Example value
Room	Temperature	float	27.8
	Presence	boolean	true
Door	Locked	boolean	false
	Closed	boolean	false

```
curl localhost:1026/v2/entities -s -S --header
"Content-Type: application/json" -d @ {
  "id": "Bedroom1",
  "type": "Room",
  "timeInstant": {
    "value": "2018-01-01T00:00:00Z",
    "type": "time",
    "metadata": {
      "timeInstantUnit": {
        "value": "ISO8601",
        "type": "string"
      }
    }
  },
  "temperature": {
    "value": 25.0,
    "type": "float",
    "metadata": {
      "temperatureUnit": {
        "value": "celsius",
        "type": "string"
      }
    }
}
```

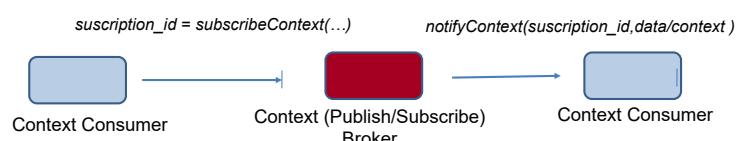
## Entity Creation

## Publish/Subscribe Context Broker

### Interacción básica entre actores del CB:



### Suscripciones:



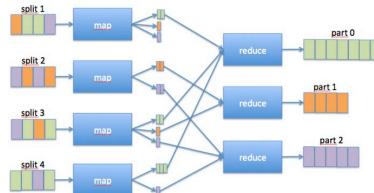
## Big Data Analysis - Cosmos

**Cosmos** es una implementación del GE Big Data de Telefónica que permite el despliegue de clusters de computación privada basados en el ecosistema Hadoop

- Orion únicamente actualiza y guarda el último valor leído.
- Para conseguir la persistencia de datos en Cosmos es necesario utilizar el inyector Cygnus. Éste se encarga de enviar la información recibida a la máquina virtual donde reside Cosmos y a hacer que éstos persistan allí.
- Esta información puede almacenarse en diferentes formatos: HDFS, CKAN y MySQL.
- La forma de hacer que Cygnus reciba información es suscribirlo a la instancia que contiene dicha información, generalmente con una suscripción de tipo ONCHANGE.

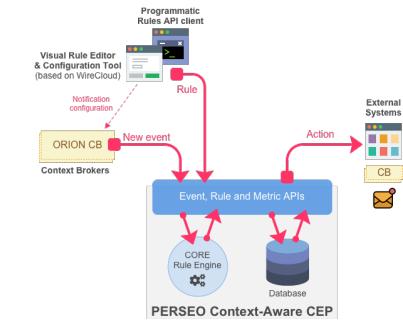
# Cosmos - Apache Hadoop

- Entorno escalable para el procesamiento distribuido de grandes datasets en diferentes clústers
- Hadoop Distributed File System (HDFS)
  - Basado en el sistema de ficheros de Google
  - Los ficheros grandes se distribuyen entre varias máquinas (**datanodes**) dividiéndolos en bloques
  - Sistema escalable añadiendo datanodes adicionales
  - Tolerante a fallos (los bloques se replican en varios nodos, 3 por defecto)



# Perseo - CEP

- Procesador Complex Event Processing (CEP) diseñado para ser *NGSIv2-compliant*
- Escucha eventos que provienen del Context Broker
- Analiza e identifica patrones descritos por reglas
  - Correlación de eventos
  - Agregación, filtrado, unión, ordenación de eventos...
- Reacciona a patrones y aplica acciones.



# CYGNUS

- Basado en Apache Flume
- Permite mandar datos NGSI del context broker para datos de aplicaciones como:
  - HDFS, sistema de ficheros distribuido de Hadoop
  - MySQL, gestor de base de datos relacional
  - CKAN, plataforma de datos abierta
  - MongoDB, base de datos NoSQL orientada a documentos
  - DynamoDB, base de datos NoSQL basada en cloud de Amazon Web Services
  - PostgreSQL, gestor de bases de datos relacionales.
  - Elasticsearch, motor de búsqueda distribuida con documentos JSON

# Wirecloud- Application Mashup

- Plataforma de construcción de *mashups* de explotación de datos
- Foco en la interfaz de usuario
- Catálogo de *widgets* y *operadores*
- Panel de visualización
- Entorno *wiring*. Conexión de widgets mediante eventos

- Widget.** Aplicación web de alcance limitado que facilita el acceso a funciones y provee de una interfaz de usuario.
- Mashup.** Combinación en una página web o app de datos y funcionalidad de múltiples fuentes heterogéneas para crear una experiencia de usuario enriquecida y unificada, sin desarrollos y con *time-to-market* mínimo.
- El **catálogo** permite que un usuario constructor seleccione widgets o mashups de widgets disponibles
- Los **desarrolladores** pueden enriquecer el catálogo extendiendo o con nuevos widgets/mashups.



# Fiware Lab

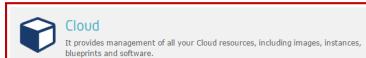


FI-LAB, una instancia de FI-WARE accesible en Internet para experimentar con las tecnologías FI-WARE

# Fiware Lab



## Fiware Lab: Cloud



### Basic functionalities

- Create your account in lab.FI-WARE.org
- Enter in the Cloud Portal
- Create your keypair (private key)
- Deploy your instance
- Add a public IP
- Open ports to the VM

### Networks functionalities

- Create your own network
- Create your subnet associate to the previous network
- Create a router
- Set gateway
- Assign subnet
- Deploy your instance
- Assign public IP to your instance
- Check the new instance.

## Fiware Lab: Cloud – Crear cuenta

### Community Account Request

Fill in the data to request a Community Account. You will receive a confirmation email at each step of the process. If not please check your SPAM inbox or contact the helpdesk: [fware-lab-help@lists.fware.org](mailto:fware-lab-help@lists.fware.org).

User Full Name*	<input type="text"/>
Your full name as by your ID	
User Account Email*	<input type="text"/>
Insert the email associated to the main representative in your project	
Are you already registered in FIWARE Lab?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Confirm that you created a main account for your project in FIWARE Lab. You should be able to register a "basic" account without any issue. In case of problems, the Help Desk will support you in the creation of the account.	
Company*	<input type="text"/>

Create your account in [lab.FI-WARE.org](http://lab.FI-WARE.org)

# Fiware Lab: Cloud – keypair

## SEGURIDAD

- Existe un apartado donde se crean las IPs públicas, a través de las cuales se puede acceder a una máquina virtual y otro, donde se crean los llamados Grupos de Seguridad (*Security group*), en los cuales se permiten administrar los puertos abiertos a una máquina.
- Para el acceso a las máquinas a través de la IP pública se generan claves públicas llamadas *keypair*. Todo ello es assignable a una máquina virtual de forma intuitiva y en el caso de los *keypair* y *security group*, a varias a la vez.

Keypair Name \* Keypairs are ssh credentials which are injected into images when they are launched. Creating a new key pair registers the public key and downloads the private key (as pem file). Protect and use the key as you would any normal ssh private key.

\* Mandatory fields.

# Fiware Lab: Cloud – crear instancia

Instances Name \* Instance Name \* Flavor Name \* Image Disk Count \* Memory

Description: Select the details for launching on instance. The chart below shows the resources used by the project in previous launches of this instance type.

Flavor Details

Project Quota

Success! Retrieved Floating IP 10.209.83.21

# Fiware Lab: Cloud – IP

Associate new IP

Associate IP to Project

Associate Floating IP

Associate Floating IP

Associate Floating IP

# Fiware Lab: Cloud – puertos

Create a Security Group

Edit Security Group Rules

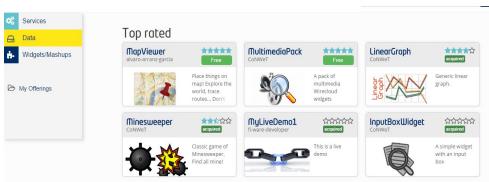
Success! Keypair was deleted.

Success! Floating IP 10.209.83.21 was deleted.

# Fiware Lab: Store



- En este apartado del portal FI-Lab se puede encontrar una **tienda** en la que se publican las **aplicaciones** que los desarrolladores han publicado. Éstas pueden ser **gratuitas o de pago**.
- En la mayoría de los casos es posible descargar el código y observar su composición.



# Fiware Lab: Mashup



La sección **Mashup** es aquella donde se suben los **wIDGETS** o se descargan otros existentes, enlazarlos mediante **wIRING** y finalmente, hacerlos funcionar. Esta sección a su vez se divide en tres:

- En **Marketplace** se pueden ver las aplicaciones disponibles, subir las propias e instalar y desinstalar del espacio de trabajo del usuario.
- Wiring**, espacio para situar los widgets que se desean enlazar y crear las convenientes conexiones. Todo ello se realiza de forma intuitiva y gráfica. Además, se podrán configurar los widgets mediante el menú destinado a ello que cada uno posee.
- Editor**, es el espacio donde ejecutar las aplicaciones y ver su funcionamiento final. Se subdivide en espacios de trabajo o *workspaces* con el fin de manejar diferentes widgets y aplicaciones en escenarios separados.

# Caso de Uso: Smart cities

- En 2007, por primera vez en la historia, la mayoría de la población mundial vivía en ciudades.
- Para 2050, se espera que los habitantes de la ciudad representen hasta el 70%



## Casos de Uso

### Internet de las cosas en el contexto de Big Data

Máster Interuniversitario en Big Data: Tecnologías de Análisis de Datos Masivos  
Universidade de Santiago de Compostela (USC)

# Smart cities



## Problemas:

1. Sostenibilidad del medio ambiente
2. Creciente población urbana

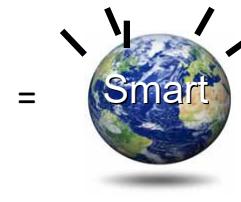
## Observación:

Los gobiernos municipales, regionales y federales recopilan cantidades masivas de datos, los depósitos de datos más grandes del mundo.

## Pregunta:

¿Qué se puede hacer?

# Smart cities



4



- Gestión sostenible e inteligente de residuos sólidos urbanos:**
  - Crear modelo predictivo de compostaje comunitario urbano basado en control biológico respirométrico con IA sobre consumo de oxígeno por masa microbiana.
  - Demostrar la viabilidad de la plataforma de gamificación para PAYT (Pay As You Throw) interactivo y personalizado, con ahorros transformados en moneda social digital aceptada por comercio local.
  - Probar la viabilidad de la sustitución de inspección humana experta de calidad de limpieza viaria por nueva solución smart combinando inteligencia colectiva e IA.



## Movilidad inteligente y sostenible:

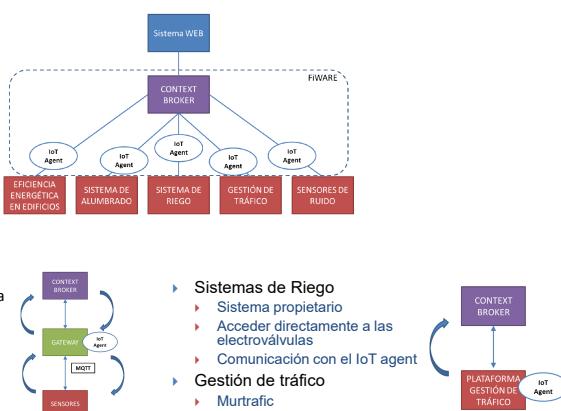
- Plataforma smart logistics de última milla, optimización de rutas y tracking individualizado de bienes desde centro de consolidación urbano.
- Vehículo eléctrico smart con acceso identificado para logística de última milla acomodado a casco histórico: respetuoso con el patrimonio, cero emisiones y multifuncional.
- Acceso y estacionamiento identificado y trazable de vehículos a casco histórico basado en smartphone, eliminando infraestructura tradicional (cámaras, bolardos).



# MiMurcia

- Centro Único de Seguimiento (CEUS) es el cerebro de la ciudad. Superficie de 84,5 m2. Sala crisis.
- 66 monitores por la ciudad y Videosensores.
- Ampliación de la cobertura wifi en plazas, parques y jardines
- Paneles que informan sobre las plazas libres de aparcamiento
- Pasos de peatones inteligentes
- Estaciones de monitorización ambiental para medir la calidad del aire ...

# MiMurcia – Modelos de conexión



# MiMurcia – Ejemplo entidad NGSI

## Atributos

- Posición geográfica (latitud, longitud)
- Energía activa. Total y por fases
- Energía reactiva. Total y por fases
- Factor de potencia

```

"contextElements": [
  {
    "type": "Sensor",
    "isPattern": "false",
    "id": "EnergiaXXX",
    "attributes": [
      {
        "name": "Active Energy",
        "type": "Float",
        "value": "30,2"
      },
      {
        "name": "Active Power Fx",
        "type": "Float",
        "value": "20"
      },
      ...
      {
        "name": "Apparent Energy",
        "type": "Float",
        "value": "30000"
      }
    ],
    "metadata": [
      {
        "name": "location",
        "type": "string",
        "value": "WSGS84"
      }
    ]
  },
  {
    "name": "geoposicion",
    "type": "coords",
    "value": "1.1111111, 1.1111111"
  }
]
  
```

# MiMurcia – Ejemplo entidad NGSI

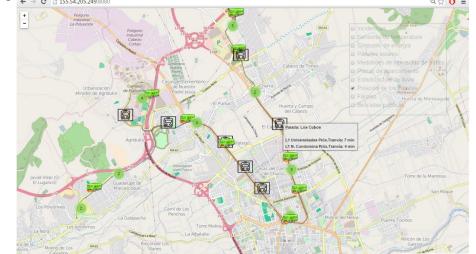
- Ejemplo consulta NGSI

```
(curl localhost: 1026 / v1 / queryContext - s - S--header 'Content-Type: application/json'\n--header 'fiware-service: energia'\n--header 'fiware-servicepath: /murcia/ayuntamiento'\n--header 'Accept: application/json' - d @ - | python - mjson.tool) << EOF{\n"entities": [\n{\n"type": "Sensor",\n"isPattern": "true",\n"id": "T.*"\n}\n]\n}
```

- **Fiware-service** categorizará el tipo de información integrada en la plataforma, para nuestro caso utilizamos el valor **energía**.
- **Fiware-servicepath** permitirá estructurar la información de forma jerárquica agilizando posteriormente la búsqueda de información.

# MiMurcia-Ejemplo: integración tranvía en tiempo real

- Desarrollo de conector para acceder a los servicios web de Tranvía de Murcia y ponerlos en ContextBroker
  - Información de las paradas
  - Tiempo llegada
  - Posición tranvías

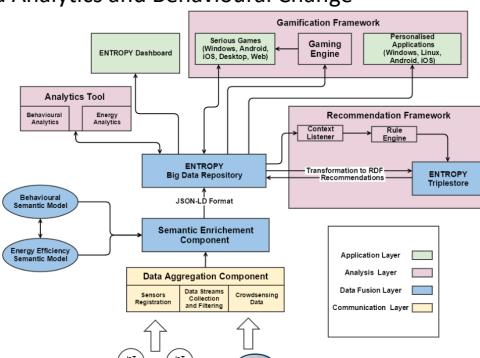


## MiMurcia: Otros ejemplos

- Integración información de tráfico: intensidad, ocupación, velocidad y fluidez de tráfico
- Plazas de bicicletas disponibles
- Monitorización de la temperatura del edificio de policía
- Consumo de energía

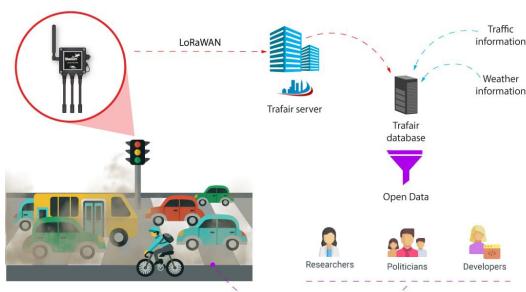
## Proyecto Entropy (UMU)

- ENTROPY: An IoT based Platform for Energy Efficiency based on Data Analytics and Behavioural Change



## Proyecto Trafair (USC)

- Smart Environment PRO Plug&Sense! IoT platform (Libelium) con detectores de contaminación
- Conjunto masivo de datos heterogéneos



## Caso de Uso: Industria 4.0

- **El Lobo, alimentación, turrones e inteligencia artificial**

- Desde el año 2015, se ha ido conectando de forma gradual toda la maquinaria de las dos plantas de producción y se han establecido diferentes sensores en la cadena de suministros y proveedores.
- Estos datos se han analizado de forma continua y **se ponen a disposición de toda la compañía en tres niveles**: uno para la directiva, otro para los mandos medios y otro para cada puesto de trabajo específico.
- Descentralizar la toma de decisiones, dotar de autonomía a los trabajadores y ganar agilidad de respuesta
- Mejora de la productividad

## Caso de Uso: Industria 4.0

- **Mejora de procesos de almacén:** Gracias a sensores y dispositivos portátiles, las empresas pueden mejorar la eficiencia operativa al detectar errores humanos, hacer controles de calidad y mostrar rutas óptimas.
- **Eliminación de cuellos de botella:** Big Data identifica las variables que puedan afectar el rendimiento.
- **Demandas de predicción:** Predicciones más precisas y significativas gracias a la visualización de actividad a través de análisis interno (preferencias de clientes) y externo (tendencias y eventos externos) más allá de datos históricos. **Mantenimiento predictivo:** Sensores nutridos por datos identifican posibles fallos en el funcionamiento de la maquinaria.

## Caso de Uso: Industria 4.0

- **BBVA**

- Data-driven bank: a través de una unidad propia, BBVA Data & Analytics, se coordinan los proyectos en diferentes áreas de negocio, desde el desarrollo de nuevas aplicaciones para el cliente final hasta potentes algoritmos internos que potencian la eficiencia de la compañía.
- RedeX: evaluación de riesgos en las solicitudes de crédito

## Caso de Uso: Industria 4.0

- **Pago de Carraovejas**

- Sensores, estaciones climáticas y un moderno sistema interconectado
- Los datos llegan en tiempo real desde sensores en el suelo y en la vid , que miden el crecimiento o el estrés hídrico de las plantas, del análisis de imágenes tomadas con satélites y drones para definir índices de vigor, del sistema de riego o de la monitorización constante del clima.
- Decisiones rápidas e inteligentes

## Caso de Uso: Industria 4.0

- **Cepsa (Data Lake)**

- Con [Kepler](#) como integrador de las soluciones en la nube de Amazon Web Services (AWS), Cepsa ha construido un **Data Lake en la nube** que centraliza la información procedente de cientos de miles de sensores instalados en sus plantas
- **Dos mil señales por segundo en una primera fase**, así como persistir la información de un histórico de varios años con una proyección de crecimiento a nivel de petabytes.
- AWS IoT como broker central de mensajería MQTT.
- AWS Greengrass para la integración con sensores vía MQTT y OPC-UA.
- Amazon DynamoDB como almacenamiento de metadatos.