

Internet de las Cosas (IoT)



UNIVERSIDAD
DE GRANADA

Ingeniería de Sistemas de Información

Daniel Bolaños Martínez

ÍNDICE

- Introducción.
- Historia
- Un Mundo Programable.
- Arquitecturas.
 - IoT-A e IIRA.
 - End-to-End.
- Problemas de Despliegue.
- Hoja de Ruta.
- IoT VS APPs.
- Seguridad.
- Programación WSN y Depuración.
- Referencias.



INTRODUCCIÓN

Para una empresa, es importante conocer las necesidades típicas de los usuarios y organizarlas en tiempo real para crear riqueza de valor.

- Flexibilidad.
- Usabilidad.
- Productividad.

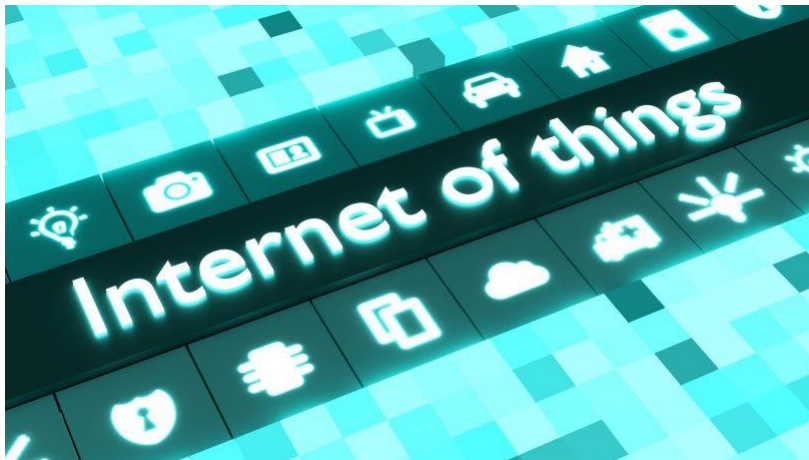


El Internet de las cosas (IoT) impulsará una enorme cantidad de innovación, eficiencia y calidad.

HISTORIA

Concepto utilizado por primera vez por Kevin Ashton en 1999.

Se desarrolla a finales de los 80 con la aparición de la computación ubicua y el desarrollo de sistemas centrados en etiquetas y redes de sensores, middleware y tecnologías en la nube y redes de comunicación.



UN MUNDO PROGRAMABLE

El mundo, programable en un sentido literal, planteará nuevos desafíos para los desarrolladores de software.

Nuestros objetos cotidianos se conectarán y se podrán programar directamente.



Los métodos de desarrollo, los lenguajes y las herramientas actuales no son adecuados para el surgimiento de millones de cosas programables en nuestro entorno.

ARQUITECTURAS DE REFERENCIA

Requisitos indispensables para una arquitectura basada en IoT:

- Conectividad y comunicaciones.
- Recopilación de datos.
- Escalabilidad.
- Administración de dispositivos.
- Seguridad.



Una arquitectura de referencia IoT maneja estos requisitos y forma un superconjunto de funcionalidades, estructuras de información y mecanismos.

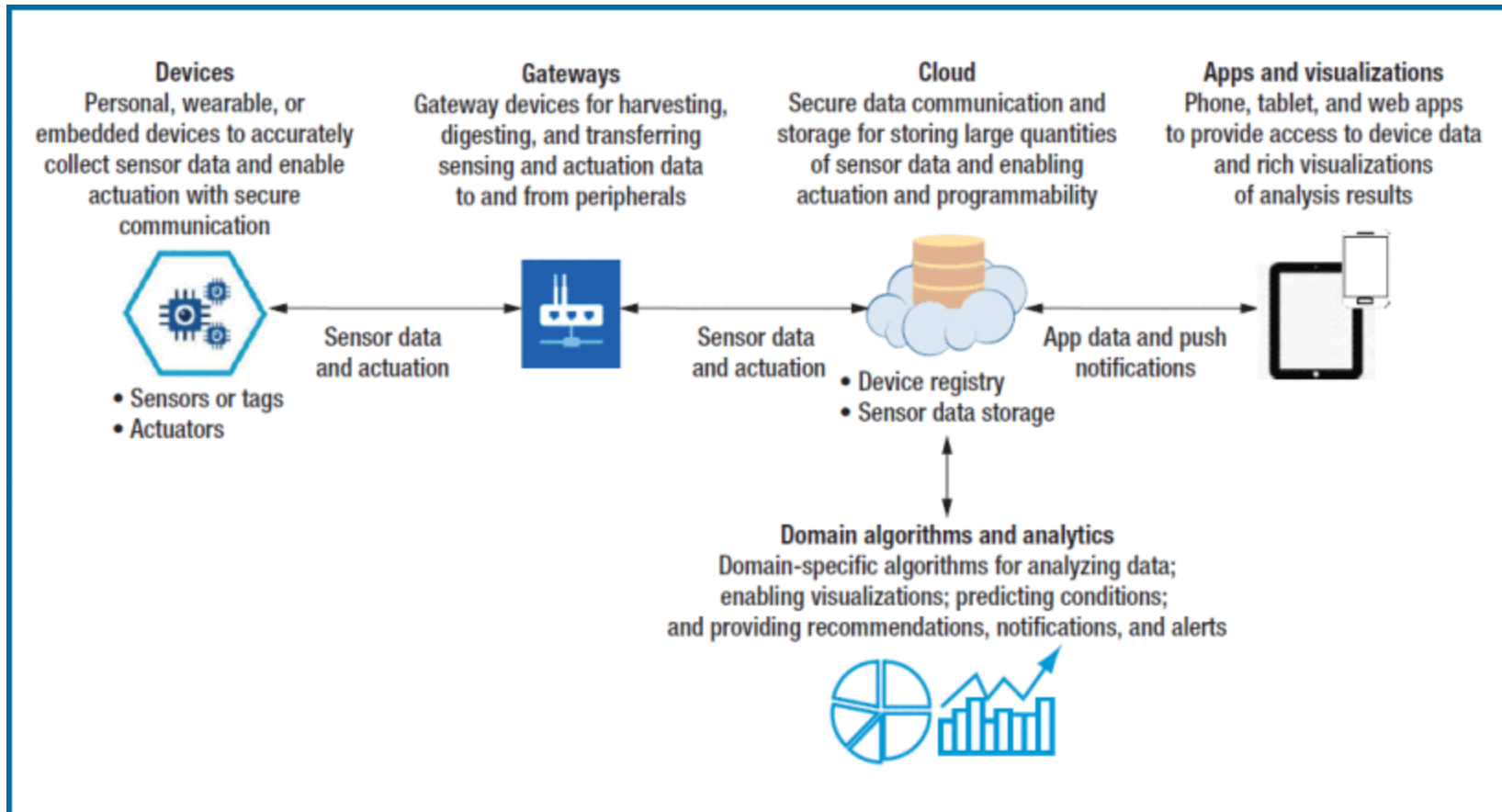
Tanto la arquitectura como el modelo ayudan a describir y asignar tecnologías a casos de negocios.

IoT-A E IIRA

Compararemos ambas arquitecturas basándonos en 3 aspectos:

Perspective	Application
Semantic oriented	Service protocols such as OPC UA (OPC Unified Architecture), UPnP (Universal Plug and Play), DPWS (Devices Profile for Web Services), CoAP (Constrained Application Protocol), and EXI (Efficient XML Interchange).
Internet oriented	Interconnectivity and protocol conversion based on UDP (User Datagram Protocol) vs. TCP with HTTP or MQTT. Support for IPv4 or IPv6.
Things oriented	A physical layer and data link layer with low-level communication protocols suitable for easy installation and maintenance.

ARQUITECTURA END-TO-END



Ingeniería y Sistemas de Información.

- [illegible]

HOJA DE RUTA

Para 2020 (Edge IoT Era):

- Avances de Hardware ==> Mundo Programable.
- Evolución en el Edge Computing.
- Actuación estandarizada de las APIs.
- Dispositivos de control de dominio específico.
- Máquinas virtuales en dispositivos IoT.

Para 2025 (Universal IoT Era):

- Adquisición de datos completamente automatizada.
- Colaboración universal M2M permitida.
- Controles remotos universales.
- Desarrollo de la Programación Dinámica en APIs.

Year	Data viewpoint	Programmability viewpoint
2015	<ul style="list-style-type: none"> • Data acquisition on a massive scale • Monitoring, tracking, routing, command and control, and mining for trends and behaviors • Cloud-centric data analytics, including both real-time and offline analytics • A focus on data visualization and simple If This Then That (IFTTT) alerts • Open source technologies available and widely used for implementing data acquisition and analytics features 	<ul style="list-style-type: none"> • Most serious computation performed in the cloud • Device support for basic actuation only; actuation implemented mostly natively using device- or manufacturer-specific, proprietary APIs and apps • Device- or manufacturer-specific device control applications available in app stores • Visual notations (for example, Node-RED) emerging for implementing device control applications more portably • Standards emerging (for example, OMA Lightweight M2M, and IPSO Smart Objects) but not yet widely adopted
2020 (Edge IoT Era)	<ul style="list-style-type: none"> • Edge computing APIs and mechanisms leveraged extensively in data processing and analytics; more intelligent filtering and denoising of uploaded data • Increasingly autonomous operation of data acquisition and analytics systems based on direct machine-to-machine communication, using local connectivity • Adapting, enhancing, and extending; automatic determination and selection of computing and analytics resources (cloud versus edge) 	<ul style="list-style-type: none"> • Edge computing capabilities and APIs available for provisioning computing flexibly between the cloud and edge devices • More advanced actuation capabilities; standardized actuation APIs • Domain-specific device control applications available—for example, for controlling lighting systems or home security equipment from different manufacturers • Virtual machines commonly available in IoT devices, enabling cross-manufacturer IoT application development and flexible migration of computation between the cloud and edge devices
2025 (Universal IoT Era)	<ul style="list-style-type: none"> • Fully automated, context-aware data acquisition, analytics, and decision optimization based on pervasive use of AI techniques such as machine learning • Universal machine-to-machine collaboration enabled by common cross-manufacturer, cross-industry APIs; social use of data among machines 	<ul style="list-style-type: none"> • A universal, containerized application-deployment-and-execution model supported across multiple manufacturers and industries • Industry-wide, cross-manufacturer programming APIs allowing generalized device discovery, data acquisition, remote device programming, and device management • Universal-remote-control applications and universal device consoles possible • Dynamic remote programming and reprogramming of devices widely supported, enabled by the widespread use of virtual machines, containerization, and common developer APIs

DIFERENCIAS DESARROLLO IOT Y APPS MÓVILES

- Independencia del dispositivo.
- Estado permanentemente activo.
- Cantidad.
- Fuertemente Integrados en el entorno.
- Heterogeneidad.
- Tolerancia a fallos.



Todas estas características deben ser consideradas por los desarrolladores de IoT.

SEGURIDAD

Las capacidades de activación y programación remotas pueden presentar altos riesgos de seguridad.

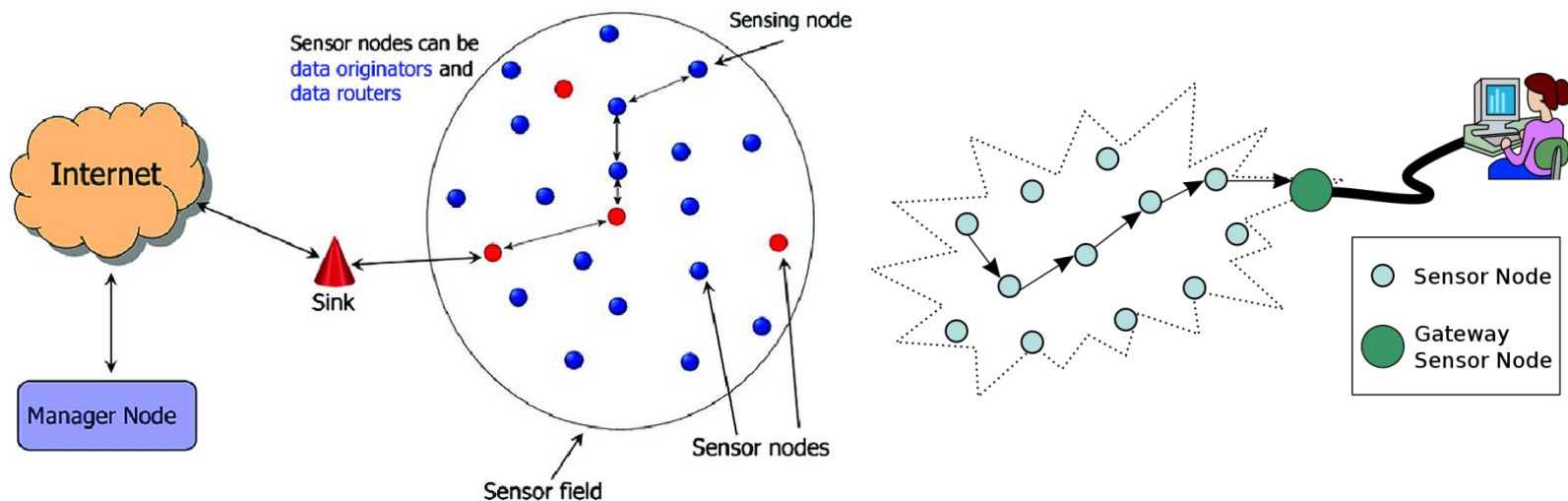
Los protocolos criptográficos de la capa de transporte, certificados de seguridad, aislamiento... desempeñan un papel fundamental en esta área, pero persisten varios desafíos técnicos interesantes.

También es indispensable depurar las aplicaciones y sistemas IoT, aunque no es tarea fácil.



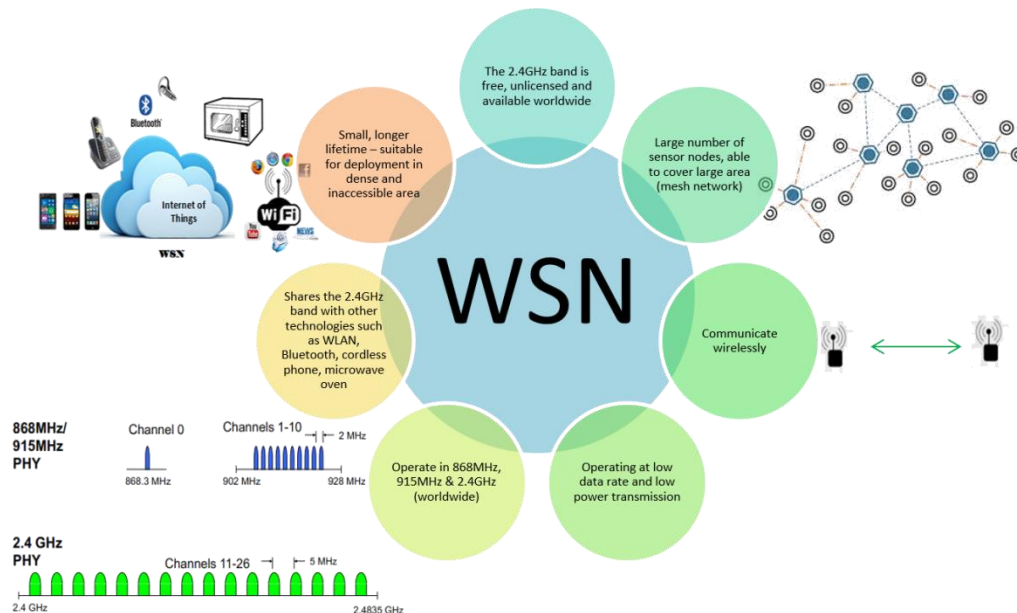
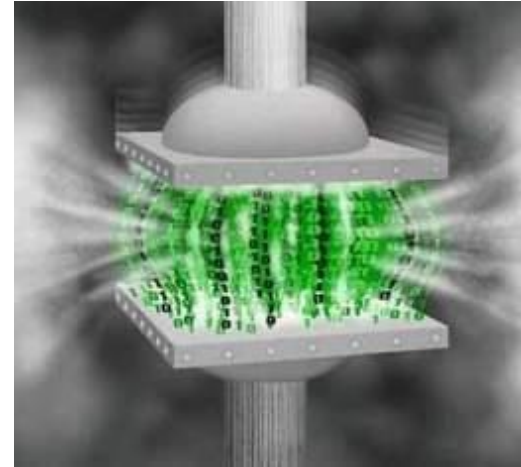
PROGRAMACIÓN WSN

- Problemas específicos (hardware, recursos, interacciones dinámicas)
- Uso de TinyTracer para el rastreo del flujo de control. (Eficacia y Eficiencia)
- Uso de TinyOS.



DEPURACIÓN Y COMPRESIÓN WSN

- Depuración a través de los nodos.
 - FIFO.
 - Orden Local.
- Compresión en trazas.
 - Prius.
 - Algoritmos de compresión.



DUDAS Y PREGUNTAS

REFERENCIAS

- Michael Weyrich, Christ of Ebert: Reference Architectures for the Internet of Things. IEEE Software 33(1):112-116, January/February 2016. DOI.
<http://dx.doi.org/10.1109/MS.2016.20>
- Antero Taivalsaari, Tommi Mikkonen: A Roadmap to the Programmable World: Software Challenges in the IoT Era. IEEE Software 34(1):72-80, January/February 2017. DOI .
<http://dx.doi.org/10.1109/MS.2017.26>
- Patrick Eugster, Vinaitheerthan Sundaram, Xiangyu Zhang: Debugging the Internet of Things: The Case of Wireless Sensor Networks. IEEE Software 32(1):38-49, January/February 2015. DOI .
<http://dx.doi.org/10.1109/MS.2014.132>