

UNIVERSIDAD DE OVIEDO
Departamento de Informática



TESIS DOCTORAL

“METAMODELO PARA LA INTEGRACIÓN DEL
INTERNET DE LAS COSAS Y REDES SOCIALES”

Autor

José Ignacio Rodríguez Molano

Directores

Doctor D. Juan Manuel Cueva Lovelle

Doctor D. Carlos Enrique Montenegro Marín

Oviedo, 2017

Agradecimientos

Agradezco a DIOS por permitirme culminar este trabajo doctoral, por brindarme la fuerza para no desistir y por mantener en mí la confianza para lograrlo.

A mi familia que con paciencia ha soportado mi angustia y me ha brindado su apoyo, en especial a mi madre Teresa.

A mis hijas Karol, Valery y Mayra, que han soportado mi ausencia en los momentos más importantes de sus vidas.

A mis Directores Juan Manuel y Carlos, que con su apoyo y su valiosa colaboración permitieron la culminación de este trabajo.

A mi compañera de vida, Jenny.

Resumen

El Internet de las cosas (IoT) está cambiando el mundo, y las aplicaciones desarrolladas en la industria son consideradas como la nueva revolución industrial generando nuevas oportunidades en la economía y la sociedad. La adopción del Internet de las Cosas al desarrollo de la industria 4.0 cambia los procesos y sistemas de fabricación planteando muchos desafíos. La información de estas aplicaciones está relacionada con el estado de la producción, el comportamiento de consumo de energía de máquinas, el movimiento de materiales, órdenes de clientes, datos de proveedores y en general, cualquier tipo de datos asociados a elementos inteligentes implementados en los procesos.

En este documento se hace una descripción de Internet de las cosas y sus aplicaciones a la industria, los principios en los que se basa, los elementos y tecnologías disponibles para lograr la comunicación entre personas, objetos y aplicaciones que se han desarrollado en esta área y demuestra la importancia de su Implementación.

Describe una propuesta de arquitectura de la Internet de Cosas aplicada a la industria, un metamodelo de integración (Internet de las Cosas, Redes Sociales, La Nube e Industria 4.0) para la generación de aplicaciones para la Industria 4.0 y un prototipo de monitoreo implementado a través del microordenador Raspberry Pi, un servidor de almacenamiento en la nube y un dispositivo móvil para el control de un proceso de producción en línea.

Con el aumento del IoT y el desarrollo de aplicaciones Cloud Computing, los problemas tecnológicos relacionados disminuyen y se encuentra una ampliación de servicios en un contexto industrial. En este artículo se expone la necesidad de integrar el internet de las Cosas, los sensores, los actuadores, las redes sociales y la computación en la nube que permitan realizar el Internet de las cosas industrial (IIoT).

La IoT requiere reconocer los objetos inteligentes y mantener un flujo de mensajes constante entre los diferentes objetos. No obstante, cada implementación de diferentes redes de sensores puede presentar diferentes problemas y cada aplicación proporciona una solución diferente. Aunque se han adelantado trabajos para una mejor interconexión en la IoT y una solución en este caso es realizar una arquitectura que soporte el paso de mensajes de los diferentes tipos de dispositivo y sea capaz de responderles. Una de las aproximaciones de este planteamiento es la arquitectura dirigida por modelos “Model-Driven Architecture” MDA [1][1][1][1][1][1][1][1][1][1] la cual establece una serie de tecnologías a utilizar en la construcción de software bajo el esquema de la Ingeniería Dirigida por Modelos (MDE), con la cual se propone la creación de aplicaciones que permitan interconectar objetos heterogéneos, siendo estas la base en la generación de un Metamodelo que integre los elementos requeridos (sensores, Cloud y Web 2.0) para el avance de la IoT.

El documento se realizó en fases, en la primera fase se presentan los aspectos conceptuales de los elementos a tener en cuenta en la propuesta de integración. En la segunda fase se presenta una propuesta de metamodelo de integración y se describe su arquitectura conducente al desarrollo del IIoT.

Palabras Claves: Internet de las cosas, metamodelo, internet industrial de las cosas, redes sociales, industria 4.0

Abstract

Internet of Things (IoT) is changing the world. Software for manufacturing industry is perceived as the new industrial revolution. It is creating new opportunities for both the economies and the society. Deployment of Internet of Things for development of Industry 4.0 changes processes and manufacturing systems while it also changes players in a wide variety of types and shapes. In that sort of systems, information is related to manufacturing status, trends in energy consumption by machinery, movement of materials, customer orders, supply data and all data related to smart devices deployed in the processes.

This document analyzes Internet of Things (IoT), its use into manufacturing industry, its foundation principles, available elements and technologies for the man-things-software communication already developed in this area. And it proves how important its deployment is.

Describes a proposal of architecture of the Internet of things applied to the industry, a metamodel of integration (Internet of Things, Social Networks, Cloud and Industry 4.0) for generation of applications for the Industry 4.0, and the manufacturing monitoring prototype implemented with the Raspberry Pi microcomputer, a cloud storage server and a mobile device for controlling an online production process.

The IoT requires recognizing smart objects and maintain a constant flow of messages between different objects. However, each implementation of different sensor networks may present different problems, and each application provides a different solution. Although works for a better interconnection in the IoT have been advanced, a solution in this case is to make an architecture that supports message interchange between different types of devices and able to respond to them. One of the approaches from this perspective are architectures built under the "Model-Driven Architecture" MDA models which establishes a series of technologies to be used in the construction of software under the scheme of the model driven

engineering (MDE). Within this approach, it is proposed the creation of applications that allows interconnection of heterogeneous objects. Being it based on the generation of a metamodel that integrates the required elements (sensors, Cloud and Web 2.0) for the progress of the IoT.

Due to the enlargement of the IoT and the Cloud Computing application development, technological problems decrease and an expansion of services is created in an industrial context. This paper outlines the need to integrate the IoT, sensors, actuators, social networks and computing in the cloud. It will enable to build the Industrial Internet of Things (IIoT) up.

This document was produced in phases. In the first phase, the conceptual features of the elements to take into account in the proposal of integration are described. In the second phase, a proposal of integration Metamodel is presented, and its architecture, leading to the development of the IoT, is described.

Keywords: Internet of things, metamodel, industrial internet of things, social networks, industry 4.0

Tabla de contenido

PARTE I. PLANTEAMIENTO DEL PROBLEMA	10
CAPÍTULO 1 -INTRODUCCIÓN.....	11
1.1. PLANTEAMIENTO DEL PROBLEMA	12
1.2. HIPÓTESIS.....	14
1.3. OBJETIVOS	15
CAPÍTULO 2 -METODOLOGÍA Y DESARROLLO DE LA INVESTIGACIÓN	16
2.1. FASE 1: RECOPIACIÓN DE INFORMACIÓN.....	17
2.2. FASE II: SELECCIÓN DE HERRAMIENTAS, DESARROLLO Y ELABORACIÓN DE LA PROPUESTA	17
2.3. FASE III: ESTUDIO DE ELABORACIÓN DE METAMODELO DE INTEGRACIÓN	18
2.4. FASE IV: PRUEBAS DEL METAMODELO Y DIFUSIÓN DE RESULTADOS.....	18
2.5. ORGANIZACIÓN DE ESTA TESIS	19
PARTE II MARCO TEÓRICO.....	20
CAPÍTULO 3 – INTERNET DE LAS COSAS	21
3.1. DEFINICIÓN.....	21
3.2. TECNOLOGÍAS RELACIONADAS CON INTERNET DE LAS COSAS.....	29
3.3. PLATAFORMAS PARA INTERNET DE LAS COSAS.....	32
3.4. CAMPOS DE APLICACIÓN DE INTERNET DE LAS COSAS	35
CAPÍTULO 4 – REDES SOCIALES	40
4.1 DEFINICIÓN	41
4.2. REDES SOCIALES E INTERNET DE LAS COSAS	49
4.3. REDES SOCIALES Y CLOUD COMPUTING	50
CAPÍTULO 5 – INGENIERÍA DIRIGIDA POR.....	52
5.1. INGENIERÍA DIRIGIDA POR MODELOS MDE	52
5.2 ARQUITECTURA DIRIGIDA POR MODELOS (MDA).....	58
5.3 LENGUAJES DE DOMINIO ESPECÍFICO (DSL)	61
PARTE III. DESARROLLO DEL PROTOTIPO	63
CAPÍTULO 6 - DESCRIPCIÓN DEL PROTOTIPO	64
6.1. PROPUESTA DE ARQUITECTURA DE IIOT	64
6.2. CONSTRUCCIÓN DE METAMODELO DE INTEGRACIÓN	68
6.3. PROTOTIPO	78
CAPÍTULO 7 - VALIDACIÓN Y PRUEBAS	83
7.1. PARADAS POR FALLAS EN EL FLUJO DE PRODUCCIÓN	83
7.2. PARADAS POR FALLAS ALINEACIÓN Y ESTABILIDAD DEL EQUIPO	84
CAPÍTULO 8 - RESULTADOS	86
8.1. PARADAS POR FALLAS EN EL FLUJO DE PRODUCCIÓN	86
8.2. PARADAS POR FALLAS ALINEACIÓN Y ESTABILIDAD DEL EQUIPO	87
PARTE IV. CONCLUSIONES.....	89
CAPÍTULO 9 - CONCLUSIONES Y TRABAJO FUTURO	91
9.1. VERIFICACIÓN Y CONTRASTE DE LA HIPÓTESIS.....	92
9.2. VERIFICACIÓN, CONTRASTE Y EVALUACIÓN DE OBJETIVOS	92
9.3. ESCRUTINIO PÚBLICO	93
9.4 LÍNEAS DE INVESTIGACIÓN FUTURAS	96
REFERENCIAS.....	98

Índice de Figuras

Figura 1. Metodología de la tesis	17
Figura 2. Dimensiones de las tecnologías de información y comunicaciones. Adaptado de Nomura Research Institute [29]	22
Figura 3. El teléfono inteligente como un mediador entre las personas, cosas y el Internet[32].	25
Figura 4. Arquitectura Básica del Internet de las Cosas.	28
Figura 5. Elementos de un sistema RFID [56].	30
Figura 6. Arquitecturas IoT [90].	40
Figura 7. Capas de la nube	51
Figura 8. Arquitectura de los conceptos de MDE.	53
Figura 9. Arquitectura de cuatro capas [67]	56
Figura 10. Propuesta Arquitectura de integración	65
Figura 11. Bases de datos físicas y virtuales	66
Figura 12. Data response layer	67
Figura 13. Arquitectura de datos para Internet de las cosas.....	68
Figura 14. Ubicación del prototipo dentro de la industria 4.0 y la internet de las cosas.....	79
Figura 15. Diagrama de Conexión del prototipo	81
Figura 16. Metamodelo de integración de internet de las cosas, las redes sociales y la industria 4.0	82
Figura 17. Paradas por fallas en el flujo de producción	84
Figura 18. Paradas por fallas alineación y estabilidad del equipo	84
Figura 19. Paradas por fallas en el flujo de producción	87
Figura 20. Paradas por fallas alineación y estabilidad del equipo	88

***PARTE I. PLANTEAMIENTO
DEL PROBLEMA***

CAPÍTULO 1 -INTRODUCCIÓN

Actualmente, la mayoría de las conexiones a Internet en todo el mundo corresponde a dispositivos utilizados directamente por los seres humanos, tales como computadoras y teléfonos móviles. La forma principal de comunicación es un humano. En un futuro no lejano, cada objeto puede ser conectado. Las cosas pueden intercambiar información por sí mismos y el número de "cosas" conectado a internet será mucho más grande que el número de "personas"[2]. Los usuarios de internet mezclamos el mundo físico y el mundo de la información. Estamos entrando en una nueva era de la ubicuidad, estamos entrando en la era de Internet de las cosas (IoT, por sus siglas en inglés) en que nuevas formas de comunicación entre humanos y las cosas, y las cosas entre sí, ya es una realidad.

Es tal el desarrollo y la importancia que el IoT ha alcanzado, que diferentes informes la consideran como una de las tecnologías de mayor impacto a 2025 [3][4][5] y se prevé que miles de millones de elementos físicos u objetos serán equipados con diferentes tipos de sensores y actuadores conectados a internet a través de las redes de acceso heterogéneas en tiempo real, generando una gran cantidad de flujo de datos[6] que deben ser almacenados, procesados y presentados en forma eficiente y fácilmente interpretable. Es aquí donde la integración del IoT con Cloud Computing permite que esta gran cantidad de datos puedan ser alojados en internet, permitiendo que recursos, servicios y datos estén disponibles para su uso y para la prestación de servicios [7] end-to-end para empresas y usuarios que accedan desde cualquier lugar, proporcionando la infraestructura virtual de integración para dispositivos de almacenamiento, herramientas de análisis, visualización y plataforma.[8]. Hoy la interacción realizada en Internet es del tipo humano-humano (H2H). No obstante, cada vez más objetos pueden conectarse a Internet, sean estos heterogéneos y ubicuos. Se espera que llegue el momento en que haya más objetos conectados que personas[9][10]. Estos podrán interactuar entre ellos, enviar y recibir datos y realizar determinadas acciones según las condiciones de la información.

El Internet de las Cosas se ha catalogado como objetos heterogéneos y ubicuos interconectados entre ellos y comunicándose entre sí a través de Internet[11].

El Internet de las Cosas al tener como objetivo contar con sensores u objetos dispersos para que generen información desde cualquier sitio accesible o bien en el interior de una máquina [12], requiere la interconexión de estos objetos heterogéneos a través de Internet [13][14]. Esto llevará a un futuro en el que no sólo sea usado para la comunicación entre personas, si no, entre humano y máquina, e incluso, entre diferentes máquinas (M2M) [15][2]. Por ello cobran también importancia los Smart Objects: objetos físicos con un sistema embebido que le permite procesar información y comunicarse con otros dispositivos y realizar acciones con base en una acción o evento determinado[16][17]. No obstante, todos estos sistemas complejos presentan un problema al momento de interconectar los Smart Objects debido a las diferencias entre software y hardware utilizado por cada uno en los diferentes procesos[18].

Con el aumento del IoT y el desarrollo de aplicaciones Cloud Computing, los problemas tecnológicos relacionados disminuyen y se encuentra una ampliación de servicios en un contexto industrial[19]. En este documento se expone la necesidad de integrar el internet de las Cosas, los sensores, los actuadores, las redes sociales y la computación en la nube que permitan realizar el Internet de las cosas industrial (Industrial Internet of Things (IIoT)).

El documento en una primera fase presenta los aspectos conceptuales de los elementos a tener en cuenta en la propuesta de integración. En una segunda fase se presenta una propuesta de metamodelo de integración y se describe su arquitectura conducente al desarrollo de la IIoT.

1.1. Planteamiento del problema

Si bien hoy la interacción realizada en Internet es del tipo humano-humano (H2H). Cada vez más objetos pueden conectarse a internet, sean estos heterogéneos o ubicuos. Se espera que un futuro haya más cantidad de objetos conectados a internet que personas [2].

Internet de las Cosas (IoT) surgió en base a la necesidad de las cadenas de suministro y la identificación de objetos, personas y animales mediante el uso de etiquetas inteligentes *Radio Frequency IDentification* (RFID) [11], [20]. Con ellas se consiguió otorgar de un identificador único al objeto deseado. No obstante, para la existencia de Internet de las Cosas, son necesarias tres cosas: inteligencia integrada en los objetos, la conectividad de los objetos a Internet y la interacción entre los propios objetos. Para lograr este fin se necesitan componentes que unan el mundo real con el mundo digital. Entre estos destacan los sistemas informáticos embebidos en objetos, los sistemas avanzados de etiquetado como son *Near Field Communication* (NFC) y RFID, las redes de sensores (WSN) y las redes de sensores y actuadores (WSAN) [21]. Si combinamos varios de estos componentes, podemos hacer que un objeto concreto realice una acción con base a un evento captado por otro, como puede ser un sensor u otro objeto. Para lograr esto, se necesita de una infraestructura que sea capaz de conectarlos y dote a los objetos de la inteligencia necesaria para comunicarse entre ellos, incluso, en algunos casos, indicándoles la decisión que deben tomar. Esto se debe a que muchos objetos no tienen la capacidad suficiente y necesaria para poder tomar decisiones por sí mismos [22].

Si bien, una de las definiciones de Internet de las Cosas, determina que son objetos heterogéneos y ubicuos interconectados entre ellos y comunicándose entre sí a través de Internet, son sus propias características las que dificulta la comunicación entre los objetos. En esta era digital, las personas han tenido contacto con algún objeto inteligente o *Smart Object* aunque no sean conscientes de lo que son. Las actuales tecnologías están compuestas por sensores y/o actuadores que les permiten realizar acciones de manera automática o semiautomática, de acuerdo a las condiciones de su programación.

Sin embargo, el uso de soluciones basadas en el Internet de las Cosas no está todavía muy extendido debido a la necesidad de diversos componentes hardware como sensores, actuadores o incluso alguna plataforma para desplegar las soluciones que interconecten los objetos inteligentes. Dado que el objetivo de Internet de las Cosas es interconectar objetos, se presentan una

serie de dificultades al intentar cumplir con este objetivo. Encontrar o crear aplicaciones que interconecten objetos de internet de las cosas es uno de estos problemas en razón a la heterogeneidad de los objetos existentes y de los diferentes protocolos de comunicación. Solo una minoría de usuarios puede tener acceso a las herramientas necesarias para conectar objetos físicos y por ser internet de las cosas una novedad, los usuarios no poseen el poder adquisitivo suficiente para adquirir objetos que interconecten y los medios que faciliten esa interconexión.

IoT requiere reconocer los objetos inteligentes y mantener un flujo de mensajes constante entre los diferentes objetos. No obstante, cada implementación de diferentes redes de sensores puede presentar diferentes problemas y cada aplicación proporciona una solución diferente. Aunque se han adelantado trabajos para una mejor interconexión en el IoT [23] y una solución en este caso es realizar una arquitectura que soporte el paso de mensajes de los diferentes tipos de dispositivo y sea capaz de responderles [9][18]. Una de las aproximaciones de este planteamiento es la arquitectura Dirigida por modelos “Model-Driven Architecture” MDA[1] la cual establece una serie de tecnologías a utilizar en la construcción de software bajo el esquema de la Ingeniería Dirigida por Modelos (MDE), con la cual se propone la creación de aplicaciones que permitan interconectar objetos heterogéneos[12], siendo estas la base en la generación de un Metamodelo que integra los elementos requeridos (sensores, Cloud y Web 2.0) para el avance del IoT.

1.2. Hipótesis

Una posible solución a los problemas planteados y que permita resolver las situaciones de heterogeneidad y comunicación, surge como propuesta un metamodelo de integración entre el internet de las cosas y las redes sociales, lo que lleva a este trabajo a plantear la siguiente hipótesis:

¿Es posible interconectar objetos heterogéneos a través de un metamodelo que integre los beneficios del internet de las cosas, la nube, los sensores, las redes sociales y la gestión de la información de Big Data?

1.3. Objetivos

A partir de la hipótesis generada se propone como objetivo general de la tesis doctoral:

Proponer y diseñar un metamodelo que integre internet de las cosas, la nube, los sensores y las redes sociales

1.3.1 Objetivos Específicos

A fin de alcanzar el objetivo planteado, se proponen los siguientes objetivos específicos:

- Elaborar el estado de Arte de Internet de las cosas, la nube, los sensores y las redes sociales
- Especificar las herramientas tecnológicas aplicables y los requerimientos para la propuesta de un metamodelo de integración.
- Identificar los Smart Objects y comprobar la interoperabilidad con las plataformas web y los servicios Internet of Things disponibles para el diseño y desarrollo del metamodelo.
- Analizar, diseñar, desarrollar e implementar un prototipo de modelos y metamodelo de integración.
- Proponer y aplicar pruebas de validación para la propuesta.

CAPÍTULO 2 -METODOLOGÍA Y DESARROLLO DE LA INVESTIGACIÓN

Para el desarrollo de esta investigación se optó por un enfoque iterativo e incremental. Cada iteración inicia estableciendo unos componentes teóricos conducentes al desarrollo y construcción de modelos y prototipos que permitan evaluar los objetivos planteados. Los resultados de las pruebas se contrastan con situaciones reales en condiciones óptimas.

El proceso investigativo se llevó a cabo en una fase inicial de recopilación de información y estableciendo la condición actual de la problemática planteada. Una segunda fase en la que se definieron las herramientas y el área específica del internet de las cosas sobre la cual se efectuaría el proceso de integración, para el caso concreto industria 4.0. La tercera fase determinó los requerimientos del metamodelo y se evaluaron diferentes arquitecturas de integración. Una última fase en la que se desarrolla la propuesta, presentando una arquitectura, un metamodelo y un prototipo para el desarrollo de pruebas y validación. En el desarrollo de cada fase se estuvo efectuando la difusión de los avances y resultados obtenidos. La metodología seguida para la elaboración de esta tesis, se describe en la figura 1.

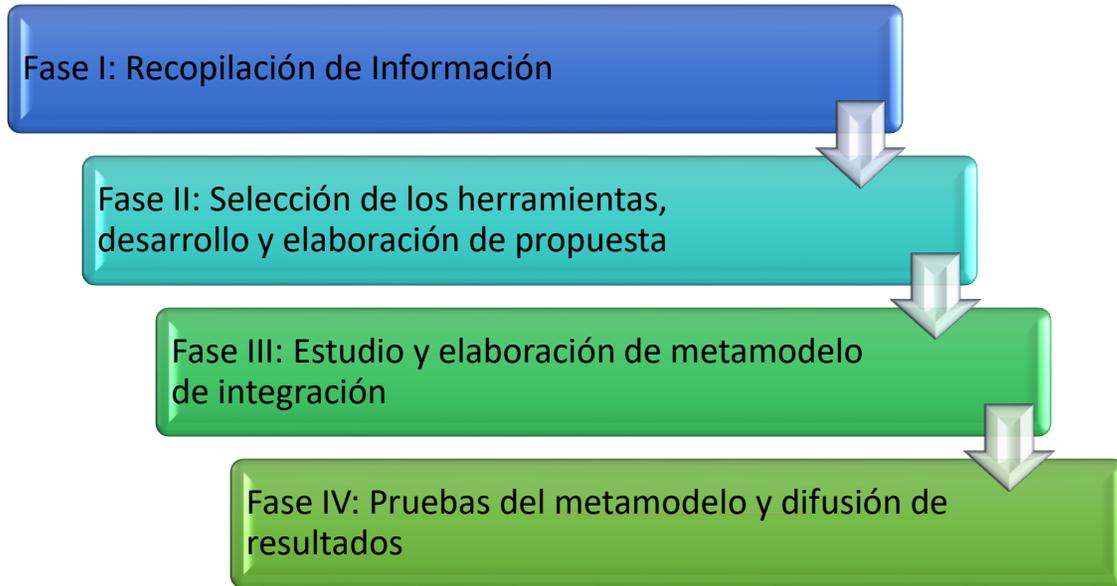


Figura 1. Metodología de la tesis

2.1. Fase 1: Recopilación de información

La recolección de información se realizó mediante la búsqueda de artículos en diversas bases de datos, entre ellas IEEE Explore, Science Direct, Web of Science, ACM Digital Library, Springer, Google Scholar y demás fuentes de información. Las palabras clave de búsqueda fueron combinaciones de los términos Internet of Things, Industrial Internet of Things, Industry 4.0, sensores y actuadores, cloud y redes sociales.

En total, se encontraron 655 documentos. Estos fueron filtrados a partir de las palabras claves y los resúmenes, así se determinaron los relevantes para la investigación. De estos documentos y aunque disponibles, por contener temáticas similares se referencian los utilizados en el desarrollo de este trabajo.

2.2. Fase II: Selección de herramientas, desarrollo y elaboración de la propuesta

En esta fase se realizó un estudio de herramientas y tecnología utilizadas para la generación, adquisición, almacenamiento y análisis de datos y que permitieran la conexión entre ellos, independiente del hardware. De esta

manera se determinó utilizar un microcontrolador Raspberry Pi, con un kit de sensores y actuadores compatibles.

2.3. Fase III: Estudio de elaboración de metamodelo de integración

Se efectuó una revisión de los resultados obtenidos por diferentes arquitecturas y metamodelos de estudios similares. Se evalúan las posibles aplicaciones en el marco de la industria 4.0, y que pueden ser integradas con las redes sociales. Se procedió a elaborar una arquitectura y un metamodelo que permitiera el desarrollo y cumplimiento de los objetivos planteados.

2.4. Fase IV: Pruebas del metamodelo y difusión de resultados

2.4.1. Desarrollo metamodelo

De acuerdo con el hardware y software disponible para el desarrollo de aplicaciones de internet de las cosas, y teniendo como lineamiento aplicaciones a la industria se procedió a la construcción de:

- Una propuesta de arquitectura de integración
- Generación del metamodelo de integración
- Implementación de la propuesta

2.4.2. Pruebas

Para la validación de la propuesta se realizaron pruebas de funcionabilidad y operación en una línea de producción industrial. Evaluando el desempeño del prototipo con los elementos constitutivos del equipo automático.

2.4.3. Difusión de resultados

La difusión de resultados se efectuó de acuerdo a las pruebas realizadas y de manera continua durante el desarrollo de este trabajo doctoral, la participación en congresos y eventos académicos y la publicación de artículos permitió ir evaluando los avances en el cumplimiento del objetivo principal.

2.5. Organización de esta Tesis

El presente documento está estructurado en secciones y capítulos. En este apartado se presenta en forma resumida la organización de esta tesis para facilitar su lectura posterior. La **parte I** titulada “**Planteamiento del Problema**” incluye los capítulos introductorios donde se presentan el problema, la hipótesis y su contexto. En el **capítulo 1**, “**Introducción**”, se presentan la justificación y planteamiento del problema a resolver en esta tesis. Luego se presenta la hipótesis y se establecen los objetivos de la investigación.

En el **capítulo 2** titulado “**Metodología y Desarrollo de la Investigación**” se presenta información metodológica sobre el desarrollo de esta tesis incluyendo la metodología y la estructura de esta memoria. En la **parte II** titulada “**Marco teórico**” se presentan los conceptos y generalidades sobre los que se fundamenta la investigación realizada en esta tesis doctoral. El **capítulo 3** se centra en el estudio del internet de las cosas, el **capítulo 4** en las redes sociales y el **capítulo 5** en la ingeniería dirigida por modelos MDE

En la **parte III** titulada “**Desarrollo del prototipo**” se presenta descripción del prototipo en el **capítulo 6**, la validación y las pruebas en el **capítulo 7**, y en el **capítulo 8** los resultados. En la **parte IV** “**Conclusiones y trabajos futuros**”

PARTE II
MARCO TEÓRICO

CAPÍTULO 3 – INTERNET DE LAS COSAS

3.1. Definición

Si bien una de las definiciones de Internet de las Cosas, determina que son objetos heterogéneos y ubicuos interconectados entre ellos y comunicándose entre sí a través de Internet [3]. Desde que Kevin Ashton en 1999 acuñó el término de Internet de las Cosas (Internet of Things (IoT)) en una presentación para Procter & Gamble (P&G), el IoT se afianzó como un nuevo paradigma en el escenario de las comunicaciones inalámbricas [9]. El concepto de la IoT [24][25] se generaliza alrededor de una variedad de elementos, cosas u objetos de carácter cotidiano, como etiquetas de identificación por radiofrecuencia (RFID)[20], sensores, actuadores, smartphones [26] y demás dispositivos móviles, los cuales son reconocibles, localizables y legibles por medio de esquemas de direccionamiento únicos capaces de interactuar unos con otros y controlables a través de internet [27] [28]. Es tal el impacto de esta tecnología que se ha añadido una nueva dimensión al mundo de las tecnologías de información y comunicación. La Figura. 2 muestra esta nueva dimensión.

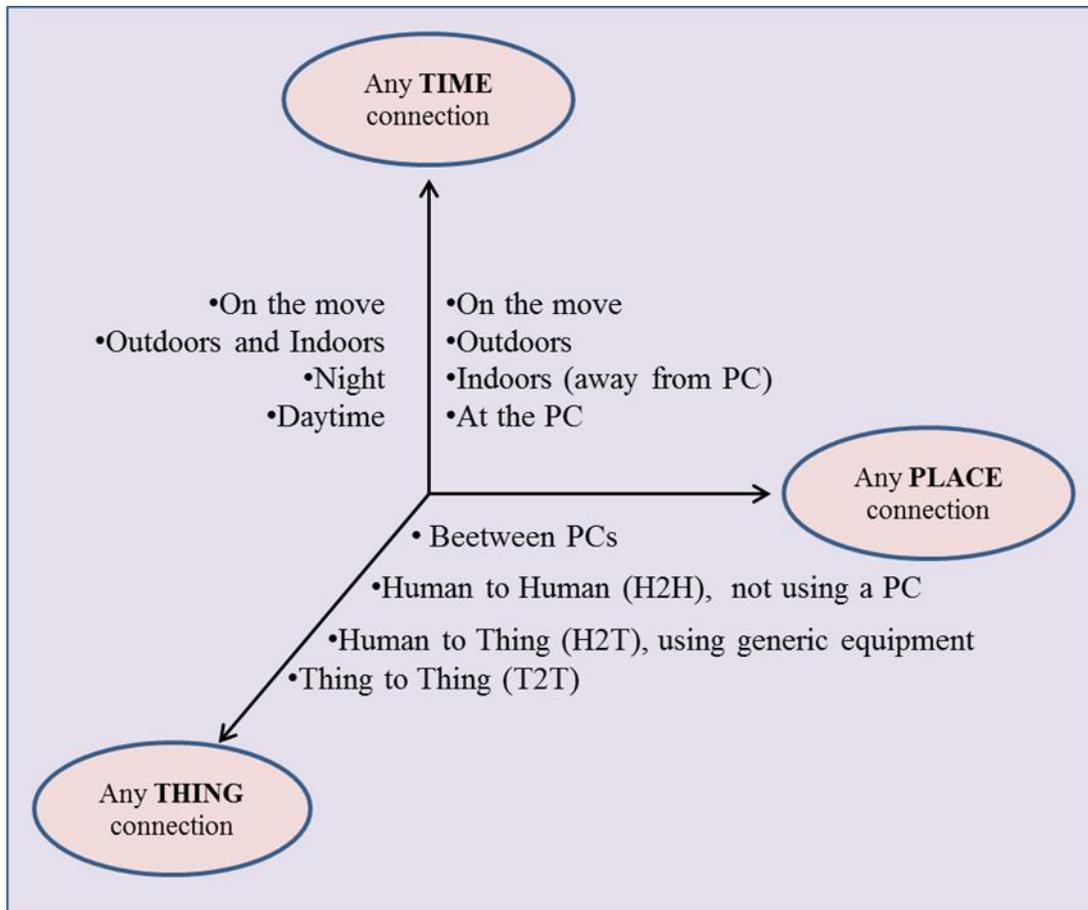


Figura 2. Dimensiones de las tecnologías de información y comunicaciones. Adaptado de Nomura Research Institute [29]

Desde una perspectiva a largo plazo, la tendencia de desarrollo de la Internet de las cosas incluye tres pasos:

1. Inteligencia Embebida: se han integrado inteligencias que pueden llevar a cabo acciones automáticamente, por ejemplo: la etiqueta RFID incrustada en los alimentos puede grabar la información acerca de la comida y se tiene la información mediante el uso de un lector RFID; el controlador de la lavadora puede hacer que complete su trabajo automáticamente; controladores de motor y antibloqueo para automóviles; sistema de guía inercial, hardware y software de control de vuelo y otros sistemas integrados en aviones y misiles; brazos artificiales con manos semi funcionales, etc. Aunque todos esos dispositivos son inteligentes, se puede

observar que trabajan solos y localmente, no hay nada conexión con la red[30].

2. Conectividad: conectar todos los dispositivos inteligentes. Son inteligentes porque están conectados. Las cosas pueden ser conectadas por cable o inalámbrica. En la Internet de las cosas la conexión inalámbrica es la alternativa. Con base en la infraestructura existente, existen las formas de conectar una cosa: RFID, ZigBee, WPAN, WSN, DSL, UMTS, GPRS, WiFi, WiMax, LAN, WAN, 3G, etc[31]. .
3. Interacción. Así que deben crearse nuevas cosas inteligentes que pueden procesar la información, auto-configurarse, auto-mantenerse y auto-repararse. Como aplicación impulsada por la Internet de las cosas, se deben crear nuevas aplicaciones de negocio que puede mejorar la innovación y el desarrollo de la Internet de las cosas[32].

Desde un punto de vista técnico, Internet de las cosas no es el resultado de una única tecnología novedosa; en cambio, varios desarrollos técnicos complementarios proporcionan capacidades que tomados juntos ayudan a cerrar la brecha entre el mundo físico y virtual. Estas capacidades incluyen:

- *Comunicación y cooperación:* los objetos tienen la capacidad de red con los recursos de Internet o incluso mutuamente, para hacer uso de los datos y servicios y actualizar su estado. Tecnologías inalámbricas como GSM y UMTS, Wi-Fi, Bluetooth y otras redes inalámbricas actualmente en desarrollo, particularmente los relativos a redes inalámbricas de área Personal (WPANs), son de primordial importancia aquí.
- *Capacidad de direccionamiento:* dentro de Internet de las cosas, objetos pueden ser localizados y dirigidos por los servicios de descubrimiento, búsqueda o nombre y por lo tanto remotamente interrogados o configurados.
- *Identificación:* los objetos son únicamente identificables. RFID, NFC (Near Field Communication) y ópticamente puede leer códigos de barras son ejemplos de tecnologías con el cual pueden identificarse incluso pasivos objetos que no tienen recursos energéticos incorporado (con la ayuda de un "mediador" como un lector de RFID o teléfono móvil) [33]. Permite la identificación de objetos

vinculados a información relacionada con el objeto concreto y que puede ser obtenido desde un servidor, siempre y cuando el mediador está conectado a la red (ver figura 3).

- *Detección:* objetos recopilar información acerca de su entorno con sensores, grabarlo, reenviarlo o reaccionar directamente a él.
- *Actuación:* objetos contienen actuadores para manipular su entorno (por ejemplo, convertir las señales eléctricas en movimiento mecánico). Estos actuadores pueden utilizarse para controlar remotamente los procesos del mundo real a través de Internet.
- *Procesamiento de información Embebida:* objetos inteligentes cuentan con una capacidad de procesador o microcontrolador, además de almacenamiento. Estos recursos pueden utilizarse, por ejemplo, para procesar e interpretar información del sensor, o para dar productos una "memoria" de cómo se han utilizado
- *Localización:* cosas inteligentes son conscientes de su ubicación física, o pueden ser localizadas. GPS o la red de telefonía móvil son tecnologías adecuadas para lograr esto, así como las medidas de tiempo de ultrasonido, UWB (Ultra-Wide Band), radio balizas (e. g. vecinos lectores RFID con coordenadas conocidas o estaciones base WLAN) y tecnologías ópticas.
- *Interfaces de usuario:* objetos inteligentes pueden comunicarse con las personas de una manera apropiada (directa o indirectamente, por ejemplo a través de un teléfono inteligente)[16].

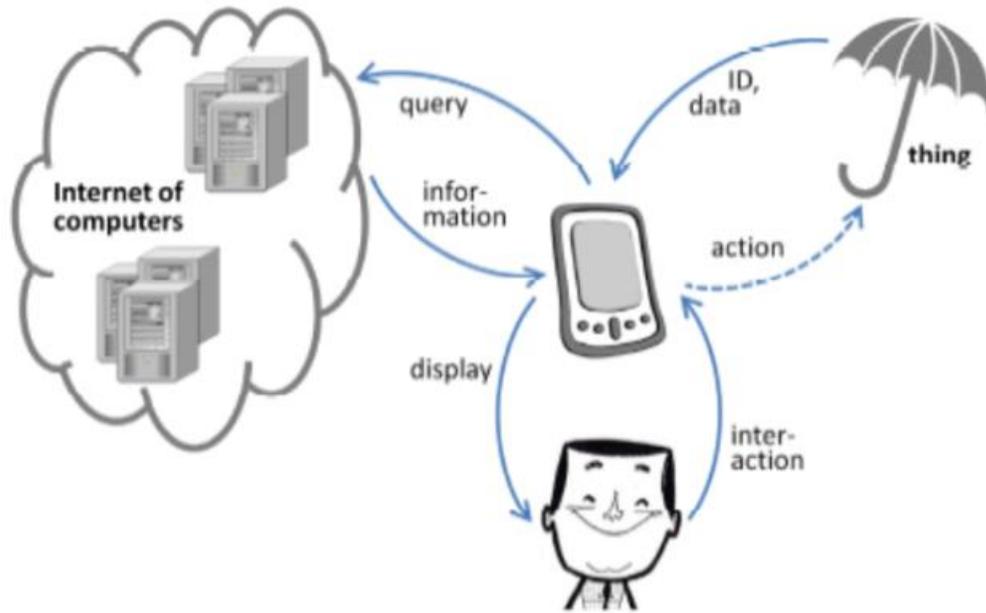


Figura 3. El teléfono inteligente como un mediador entre las personas, cosas y el Internet[32].

Precursores de comunicar los objetos cotidianos ya son evidentes, particularmente en relación con RFID – por ejemplo la comunicación de corto alcance de llaves a las puertas de las habitaciones del hotel, o tarjetas que permiten levantar torniquetes. Escenarios más futuristas incluyen una mesa juego inteligente, donde el curso del juego se controla mediante RFID-equipado jugando a las cartas [8]. Sin embargo, todas estas aplicaciones implican aún sistemas dedicados en un despliegue local; No se refiere a un "Internet" en el sentido de un sistema abierto, escalable y estandarizado.

Pero estos módulos de comunicaciones inalámbricas cada día son cada vez más pequeños, IPv6 está siendo utilizado cada vez más, está aumentando la capacidad de chips de memoria flash, los requisitos energéticos por instrucción de procesadores disminuye y los teléfonos móviles tienen reconocimiento de código de barras integrado, NFC y pantallas táctiles y puede asumir el papel de intermediarios entre la gente, artículos de uso diario e Internet. Todo esto contribuye a la evolución del paradigma de Internet de las cosas: desde la identificación remota de objetos y una conexión a Internet "con" las cosas, se está avanzando hacia un sistema donde los objetos inteligentes se comunican con los usuarios de servicios de Internet e incluso entre los demás. Estas nuevas

capacidades que ofrecen las cosas abren perspectivas y posibilidades interesantes de aplicación; pero también son acompañados por importantes requisitos relativos a la tecnología y la infraestructura subyacente. De hecho, la infraestructura para un Internet de las cosas no sólo debe ser eficiente, escalable, segura y confiable, sino también debe cumplir con las expectativas sociales y políticas generales, aplicarse ampliamente y debe tomar en cuenta consideraciones económicas.

Internet de las Cosas (IoT) surgió con base en la necesidad de la cadena de suministro y la identificación de objetos, personas y animales mediante el uso de etiquetas inteligentes *Radio Frequency IDentification* (RFID) [11], [20]. Con ellas se consiguió otorgar de un identificador único al objeto deseado. No obstante, para la existencia de Internet de las Cosas, son necesarias tres cosas: inteligencia integrada en los objetos, la conectividad de los objetos a Internet y la interacción entre los propios objetos. Para lograr este fin se necesitan componentes que unan el mundo real con el mundo digital. Entre estos destacan los sistemas informáticos embebidos en objetos, los sistemas avanzados de etiquetado como son *Near Field Communication* (NFC) y RFID, las redes de sensores (WSN) y las redes de sensores y actuadores (WSAN) [21]. Si combinamos varios de estos componentes, podemos hacer que un objeto concreto realice una acción con base a un evento captado por otro, como puede ser un sensor u otro objeto. Para lograr esto, se necesita de una infraestructura que sea capaz de conectarlos y dote a los objetos de la inteligencia necesaria para comunicarse entre ellos, incluso, en algunos casos, indicándoles la decisión que deben tomar. Esto se debe a que muchos objetos no tienen la capacidad suficiente y necesaria para poder tomar decisiones por sí mismos [22].

La IoT al tener como objetivo contar con sensores u objetos dispersos para que generen información desde cualquier sitio accesible o bien en el interior de una máquina [12], requiere la interconexión de estos objetos heterogéneos a través de Internet [13][14]. Esto llevará a un futuro en el que no sólo sea usado para la comunicación entre personas, si no, entre humano y máquina, e incluso, entre diferentes máquinas (M2M) [15][2]. Por ello cobran también importancia los Smart Objects: objetos físicos con un sistema embebido que le permite procesar información y comunicarse con otros dispositivos y realizar acciones con base en

una acción o evento determinado[16][17]. No obstante, todos estos sistemas complejos presentan un problema al momento de interconectar los Smart Objects debido a las diferencias entre software y hardware utilizado por cada uno en los diferentes procesos[18].

Internet de las Cosas pretende conectar millones de objetos de cualquier tipo a Internet, lo que repercutirá en posibilidades de grandes cambios en nuestras vidas, proporcionando una mayor productividad, mejorando la salud, mejorando la eficiencia, reduciendo energía y haciendo confortables nuestras casas [34]. Estos pueden ser, como ya se demostró en ciertas investigaciones microcontroladores como los Arduino [35], [36], microordenadores como la Raspberry Pi [37][38], neveras[39], smartphones [40][41], sensores [42], actuadores, Smart TVs, coches, etiquetas inteligentes como RFID [43], NFC y códigos de barra [44], los códigos QR (*Quick Response Code*, QR Code), entre muchos otros tipos de objetos[45][46].

La finalidad de Internet de las Cosas es interconectar todos los objetos, sea cuál sea su finalidad, tipo y emplazamiento, con todos los demás objetos. Esto ayudará a crear una Internet más inteligente y con más datos, que nos ayude con nuestra vida y problemas diarios. Por ello, de lo que se trata es de interconectar a través de Internet, las personas con las máquinas (H2M)[47], mejorar la comunicación entre personas (H2H) [48], o conectar los propios objetos entre ellos bajo lo que se conoce como *Machine-to-Machine* (M2M) [10], [48], [49]. O, dicho de otra forma, conectar todo el mundo físico con el mundo virtual a través de diferentes lugares inteligentes para automatizar, mejorar y facilitar la vida diaria. Además de poder mantener estas interconexiones a cualquier hora en cualquier lugar del mundo, mientras los objetos o las personas se están moviendo, ya estén fuera o dentro de los edificios. El único requisito necesario para estos objetos es tener acceso a Internet [50].

La arquitectura general utilizada para Internet de las cosas es una arquitectura de tres niveles[51][23]. En un primer nivel se encuentran desplegados los diferentes sensores, cada uno de ellos es una fuente de información, y diferentes tipos de sensores capturan diversos contenidos y formatos de información. Los Datos obtenidos del sensor están tiempo real y el sensor recoge la información sobre el entorno en una cierta frecuencia y sigue actualizando los datos.

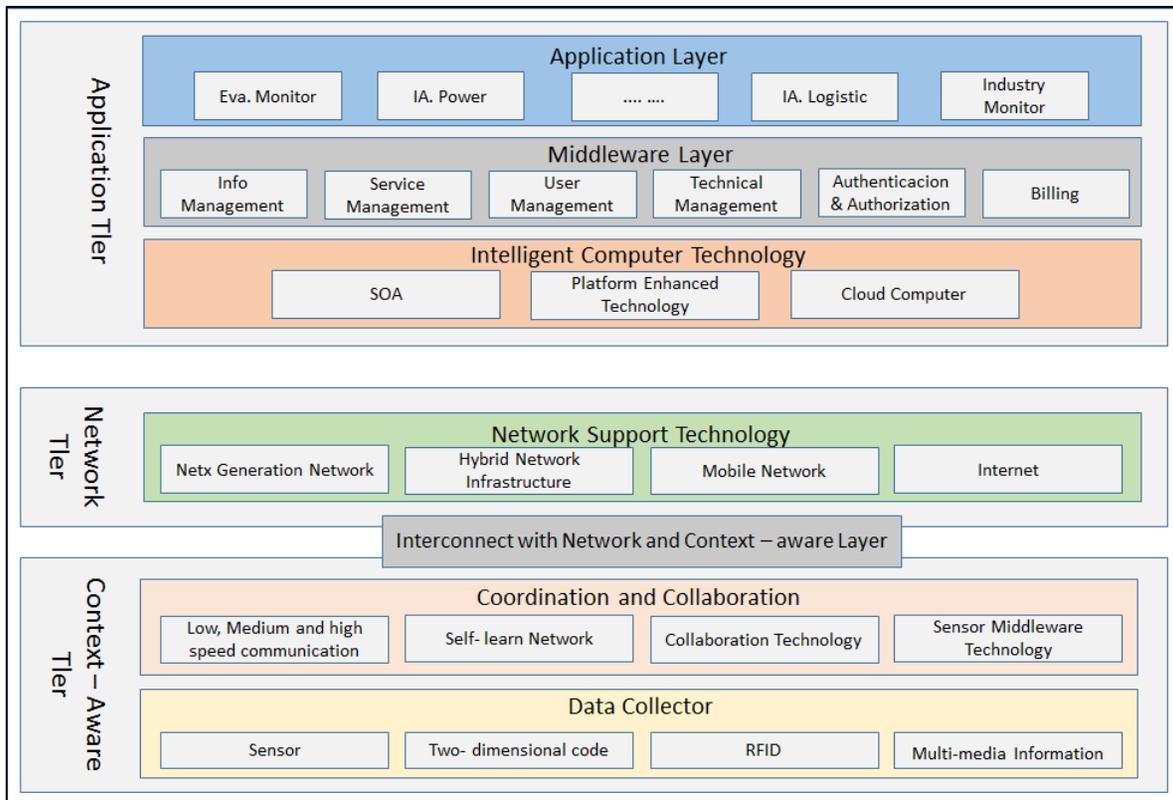


Figura 4. Arquitectura Básica del Internet de las Cosas.

El nivel intermedio integra varias redes cableadas e inalámbricas para transferir la información de las cosas con precisión. Se transfiere información regularmente recopilada por los sensores en Internet de las cosas por la red. El Nivel superior de la arquitectura es el nivel de aplicación el cual consta de tres capas (capa de tecnología, capa de middleware, capa de aplicación. La capa de aplicación se compone de aplicaciones que exportan toda la funcionalidad de los sistemas para el usuario final y explota las funcionalidades de la capa de red. El middleware es una capa de software interpuesta entre la tecnología y los niveles de aplicación. Su característica de esconder los detalles de las diferentes tecnologías es fundamental para eximir al programador de temas que no son directamente pertinentes a su enfoque, que es el desarrollo de la aplicación específica habilitada por la infraestructura[28]. El middleware está ganando cada vez más importancia en los últimos años debido a su importante papel en simplificar el desarrollo de nuevos servicios y la integración de las tecnologías heredadas en los nuevos.

3.2. Tecnologías relacionadas con Internet de las Cosas

Las tecnologías más relevantes asociadas al desarrollo de la internet de las cosas y sus posibles aplicaciones están lideradas por RFID (Identificación por radiofrecuencia), EPC (Código Electrónico de Producto), y NFC (Comunicación en Campo Cercano). A continuación se realiza una presentación de cada una de ellas.

3.2.1. Radio Frequency Identification (RFID)

Radio Frequency Identification (RFID) es una tecnología de punta para la completa identificación de objetos de cualquier tipo que permite una rápida captura de datos de manera automática mediante radio frecuencia[43]. RFID es un método de identificación automática basado en el almacenamiento y recuperación de datos que utilizan ciertos dispositivos llamados etiquetas RFID[52]. La premisa básica detrás de los sistemas de RFID es marcar los artículos con etiquetas. Estas etiquetas contienen transpondedores (dispositivos receptores y transmisores de señales) que emiten mensajes legibles por los lectores RFID[53].

Una de las extensiones más comunes de las técnicas de identificación de objetos es la función de seguimiento. Las personas pueden usar la técnica RFID para transferir los objetos físicos al sistema digital. A medida que estos datos de identificación de entrada en el sistema con información adicional, como la hora, la ubicación o incluso datos de sensores, se les da a los objetos una nueva característica, la trazabilidad [54].

Además de la trazabilidad, con el uso de esta tecnología se obtienen las siguientes ventajas [52]:

- Mayor automatización en el proceso de lectura de las etiquetas puesto que la lectura se puede realizar sin necesidad de tener una línea de visión directa con el dispositivo lector.
- Ahorro en tiempo de lectura de las tarjetas ya que es posible realizar la lectura simultanea de más de una etiqueta.
- Visibilidad completa de toda la información almacenada dado que la información permanece intacta en la etiqueta.

Un sistema RFID está conformado habitualmente por tres elementos (véase Figura 5): etiquetas (Tags), lectores y Middleware para integrar datos con diferentes aplicaciones[55].

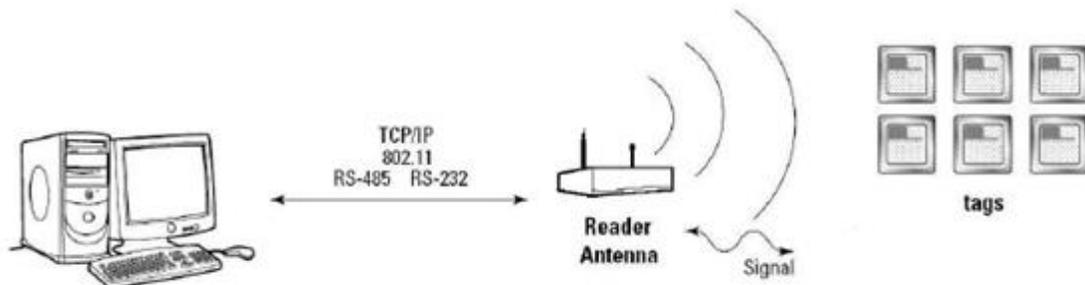


Figura 5. Elementos de un sistema RFID [56].

3.2.1.1. Etiquetas RFID

La tecnología RFID utiliza tarjetas dotadas de un microchip que almacena datos y un circuito impreso a modo de antena emisora, utilizado para comunicarse a través de las señales de radio frecuencia. Las etiquetas RFID se pueden unir a cualquier artículo y se pueden adaptar a cualquier condición (humedad, polvo, suciedad, entre otros).

Las tarjetas o etiquetas son clasificadas en dos categorías generales, activas y pasivas, dependiendo de su fuente de energía eléctrica. Las etiquetas activas tienen su propia fuente de energía y al transmitir una señal más fuerte los lectores pueden acceder a ellas desde una distancia más lejana (de 20 a 100 metros), así mismo, estas características hacen que este tipo de etiquetas sean más grandes y más costosas. Por otro lado, las etiquetas pasivas son más baratas y más pequeñas puesto que al no contar con una batería integrada recogen la energía del campo electromagnético creado por el lector[56][57].

3.2.1.2. Lectores RFID

Un lector es un dispositivo electrónico que se comunica con las etiquetas a través de la antena y lee la información almacenada en la etiqueta RFID. El lector puede tener diversas formas de diseño ya sea como una forma fija o como un terminal móvil[55].

El lector de RFID crea un campo de frecuencia de radio que detecta las ondas y puede ser capaz de leer datos desde un transpondedor (dispositivo transmisor de señales) y escribir datos hacia este. Cuando una etiqueta RFID pasa a través de un campo de radiofrecuencia generado por un lector compatible la etiqueta refleja de vuelta al lector la información de identificación sobre el objeto al que está unido identificando así ese objeto.

3.2.1.3. Middleware RFID

Es un tipo especial de software que se utiliza para recoger y filtrar datos de los dispositivos de lectura RFID. A través de este software se gestiona en tiempo real la información de lectura que han hecho los lectores, se recopilan los datos procesados, se transforman y se transfieren a otros sistemas de información existentes[20].

3.2.1.4. EPC

El Código Electrónico de Producto (EPC) es un identificador universal basado en Identificadores Universales de Recursos (URIs); este código proporciona una identidad única para cada objeto físico en cualquier parte del mundo y para todos los tiempos [58].

El EPC está diseñado para facilitar los procesos y aplicaciones que requieren manipular los datos de visibilidad de los objetos físicos. El EPC es un esquema de numeración que proporciona una identificación única para objetos físicos y sistemas. Esta identificación debe ser lo suficientemente grande como para enumerar todos los objetos y para dar cabida a todos los métodos de asignación de nombres actuales y futuros.

La numeración en EPC está basada en EPC-64, EPC-96 y EPC-256, tres modos de codificación, que son respectivamente 64, 96 y 256 bits de longitud[59]. No importa qué tipo de modo de codificación se utilice, el EPC incluye:

- Cabecera que identifica la longitud, tipo, estructura, versión y generación del EPC.
- Número de Administrador que identifica la empresa o fabricante del objeto.

- Clase de objeto.
- Número de serie, que es la instancia específica de la clase de objeto que se etiqueta.

Las etiquetas de RFID almacenan un EPC único en un chip y transmiten este código a través de una antena para lectores de RFID. En un entorno de red EPC global, un lector lee el código EPC en los objetos y se transmite al servidor de asignación de nombres de objeto ONS. Al consultar en el servicio EPC-información del servidor utilizando el código EPC se puede obtener la información detallada del objeto[60].

3.2.2. NFC

NFC (Comunicación en Campo Cercano) es una tecnología inalámbrica de comunicación entre dispositivos (especialmente teléfonos móviles y asistentes personales). Esta tecnología fue desarrollada por Philips y Sony en 2002, y combina la tecnología de conectividad inalámbrica RFID y tecnologías de interconexión para ofrecer una comunicación inalámbrica de corto alcance y de alta frecuencia entre dos dispositivos NFC ubicados a menos de 20 cm[57].

Los sistemas NFC constan de dos elementos: a) el iniciador, el cual comienza y controla el intercambio de información y b) el objetivo, que es el dispositivo que responde al requisito del iniciador[54].

En un sistema NFC hay dos modos de funcionamiento: activo y pasivo. En el modo activo los dispositivos generan su propio campo de radio frecuencia para transmitir datos. En el pasivo, sólo uno de estos dispositivos genera el campo de radiofrecuencia mientras que el otro se utiliza para cargar de modulación para transferencias de datos[4].

3.3. Plataformas para Internet de las Cosas

3.3.1. ThingSpeak

ThingSpeak es una plataforma Open Source con una API para almacenar y recuperar datos de los objetos, diseñada para permitir conectar personas con

objetos, por lo que su principal aplicación es el Smart Home[61]. ThingSpeak permite la integración de su plataforma con:

- *Arduino*
- *Raspberry Pi*
- *IoBridge / RealTime.io*
- *Electric Imp*
- *Móviles / Aplicaciones web*
- *Redes Sociales*

3.3.2. Carriots

Carriots es una plataforma en la nube, orientada a proyectos IoT y máquina a máquina (M2M), a través de un servicio PaaS. Carriots es una plataforma propietaria, se caracteriza por una gran compatibilidad de hardware, una API muy completa y documentada[62]. Carriots ofrece mecanismos integrados que podrán ser accesibles mediante su SDK (DropBox, Twitter, Mailing, SMS)[63]. Los ámbitos más importantes de aplicación son:

- Smart City
- Smart Energy
- Smart Oil
- Smart Agriculture
- Smart Buildings
- Smart Retail
- Smart Banking
- Smart Consumer
- Smart Logistic

3.3.3. Social Internet of Things

SIoT es una red IoT enmarcado en lo que se conoce como IoT Social, en inglés *Social IoT*. SIoT fue construida sobre el núcleo de ThingSpeak. SIoT es una aplicación RESTful y ofrece soporte para los formatos *Comma-Separated Values* (CSV), JSON y *eXtensible Markup Language* (XML) usando métodos GET

y POST[13][64]. La principal meta de SIoT es permitir la creación de listas de amigos con objetos para crear sus relaciones.

3.3.4. *Electronic Imp*

Esta plataforma abarca tanto soluciones hardware como software en forma de servicios. La plataforma se basa en su módulo Wifi, el cual tiene una completa integración con cualquier dispositivo hardware, y software. Básicamente este módulo Wi-Fi, actúa como un gateway para conectar un dispositivo a internet independientemente del hardware del dispositivo. Cloud es la pieza central de la plataforma Electronic imp, para conectar toda clase de dispositivos a través de internet. Cloud nos ofrece un servicio SaaS[65]. La plataforma Electronic Imp, no se centra en un único campo de aplicación, por lo que diversifica un poco su oferta ofreciendo diversas soluciones en el mundo del IoT, ante todo en los siguientes ámbitos:

- Smart Logistic
- Smart Home
- Smart Consumer
- Smart Industry

3.3.5. *Spark*

El propósito de la plataforma de Spark Core es dotar de inteligencia a objetos, es aquí donde arduino nos da todo ese poder y los mecanismos necesarios a la hora de programar. Su principal uso está enfocado al desarrollo de temas de Smart Home[66].

3.3.6. *Kaa*

Es una plataforma IoT de código abierto. Esta red IoT permite conectar tu aplicación con otros objetos utilizando como *middleware* 5 el servidor Kaa para crear los *endpoints* 6. Además, tiene una integración con sistemas de gestión de datos y sistemas de análisis. No obstante, como ocurre con otras plataformas,

hay que programar la aplicación que debe funcionar en los diferentes objetos y conectarse con el servidor Kaa[67]

3.3.7. Blaulabs

Es una plataforma SaaS modular dedicada al IoT, centrándose en la monitorización y análisis en tiempo real de los dispositivos y sensores. Es una plataforma que permite capturar cualquier tipo de datos desde cualquier dispositivo haciendo uso de la API abierta. Blaulabs Suite es una línea de negocio que se basa en dar soluciones tipo Smart City, Gestión Energética, Smart Manufacturing, etc[68]

3.3.8. Thinking Things

Es una plataforma del internet de las cosas, que ofrece tanto hardware como software. Thinking Things es la propuesta de Telefónica para el internet de las cosas se basa en unos módulos tipo LEGO, compuestos básicamente de una batería, un sensor y un módulo de conectividad GSM[30][69].

3.3.9. Zatar

Es una plataforma IoT orientada para construir aplicaciones empresariales, basado en cloud computing. Esta solución incluye componentes básicos de IoT como son los sensores, RFID y código de barras, permitiendo gestionar la información a través de la web[70].

3.4. Campos de aplicación de internet de las cosas

Las aplicaciones de IoT actualmente se enfocan en las *Smart Home*, la industria 4.0, llamada *Industrial Internet of Things (IIoT)*, los municipios o poblados agrupados en el concepto *Smart Town*, las ciudades también llamadas *Smart Cities* y el medio ambiente que se sitúa bajo el concepto *Smart Earth*. Cada una de estas divisiones tiene sus propias aplicaciones, aunque todas se basen en IoT. Las Smart home buscan priorizar la automatización, la industria

mejorar el proceso industrial, los pueblos el mantener su identidad cultural, las ciudades su habitabilidad y el medioambiente la comunicación con los edificios y la naturaleza. Con base a esto, la finalidad es crear un mundo inteligente, conducente al *Smart World* [71].

El dominio de las áreas de aplicación para el IoT está limitado sólo por el tiempo y la imaginación.[72] En la revisión de la literatura realizada para esta investigación, y con el propósito de proponer una arquitectura y metamodelo de integración, se efectuó la descripción de cinco áreas de aplicación, principalmente de aplicaciones a la industria (Internet de las Cosas Industrial: Industria 4.0).

3.4.1. Infraestructura inteligente

La integración de objetos inteligentes en la infraestructura física puede proporcionar flexibilidad, confiabilidad y eficiencia en el funcionamiento de la infraestructura. Estos beneficios pueden reducir los costos y los requisitos de mano de obra, así como aumentar la seguridad. Las redes inteligentes utilizan la tecnología IoT para recolectar datos sobre energía[73].

Consumo y poner los datos disponibles en línea. Los datos suelen incorporarse a los informes que muestran los patrones de uso e incluyen recomendaciones sobre cómo reducir el consumo de energía y el costo [74]. Las tecnologías IoT también se están utilizando dentro de hogares y oficinas. Los hogares y edificios están equipados con sensores y actuadores que rastrean el consumo de servicios públicos, monitorean y controlan la infraestructura del edificio, tales como luces y sistemas HVAC, y llevan a cabo las necesidades de seguridad de la vigilancia[75]. En una escala más amplia, las tecnologías de IoT pueden emplearse para hacer las ciudades más eficientes. El objetivo de las ciudades inteligentes es aprovechar el IOT para mejorar la vida de los ciudadanos, mejorando el control del tráfico, controlando la disponibilidad de plazas de aparcamiento, evaluando la calidad del aire e incluso notificando cuando los contenedores de basura están llenos[76].

3.4.2. Asistencia sanitaria

El IOT se propone para mejorar la calidad de la vida humana mediante la automatización de algunas de las tareas básicas que los seres humanos deben realizar. En ese sentido, el seguimiento y la toma de decisiones pueden ser delegados de los seres humanos a la máquina. Una de las principales aplicaciones de la IOT en salud[77] es en escenarios de vida asistida. Los sensores se pueden colocar en el equipo de monitoreo de salud utilizado por los pacientes. La información recolectada por estos sensores se pone a disposición de los médicos, miembros de la familia y otras partes interesadas en Internet para mejorar el tratamiento y la capacidad de respuesta[21]. Además, los dispositivos IoT pueden usarse para monitorear los medicamentos actuales del paciente y evaluar el riesgo de nuevos medicamentos en términos de reacciones alérgicas e interacciones adversas[78].

3.4.3. Cadenas de suministro / logística

RFID y las redes de sensores ya tienen un papel establecido en las cadenas de suministro. Los sensores han sido utilizados durante mucho tiempo en líneas de montaje en instalaciones de fabricación y la RFID se utiliza con frecuencia para rastrear los productos a través de la parte de la cadena de suministro controlada por una empresa específica. Si bien el uso de estas tecnologías en las cadenas de suministro no es nuevo[79], la omnipresencia y la ubicuidad prometidas por el IoT permitirán el uso de estas tecnologías a través de las fronteras organizativas y geográficas[80]. Concretamente, el IoT puede mejorar aún más la logística y la eficiencia de la cadena de suministro[81], proporcionando información detallada y actualizada que la actualmente disponible, mitigando el efecto de chicle, reduciendo la falsificación y la mejora de la trazabilidad de los productos.

3.4.4. Aplicaciones sociales

Dado que es probable que los dispositivos estén conectados a muchos objetos e incluso a las propias personas, es absolutamente esencial examinar los impactos sociales y personales potenciales del IoT. Los dispositivos IoT

permiten una serie de funcionalidades que pueden promover la interacción social y las necesidades personales. Una posible aplicación del IoT en un contexto social es la interacción de los dispositivos IoT con los servicios existentes de redes sociales como Facebook o Twitter (Vázquez y López-de-Ipina, 2008). El uso de dispositivos IoT para proporcionar información sobre las actividades y la ubicación de una persona puede ahorrarle tiempo al usuario. Además, las aplicaciones que recopilan e integran automáticamente esta información pueden informar a las personas cuando están cerca de amigos, eventos sociales u otras actividades que puedan interesarles[82]. Además, los teléfonos móviles compatibles con IoT pueden conectarse directamente a otros teléfonos móviles y transferir información de contacto cuando son compatibles los perfiles predefinidos de citas o amigos[82].

3.4.5. Internet de las cosas Industrial: Industria 4.0

Internet de las Cosas Industrial (IIoT) conocido como Industria 4.0, siendo así la combinación de Internet de las Cosas con la industria. De acuerdo con el reporte del *Industrie 4.0 Working Group* del Ministerio Federal de Educación e Investigación Alemán [83][84], en los últimos años el número de aplicaciones de **IoT** para la industria viene en aumento[85], indicando que se sigue evolucionando. Como tecnología emergente se espera que **IoT** ofrezca soluciones para transformar los sistemas de transporte y manufactura [85], mejorándolos para obtener una mejor eficiencia, ahorro de costos y una reducción del tiempo en los procesos.

El Internet industrial de las cosas (IIoT) está conectando las máquinas entre sí con el mundo físico de los sensores, cada vez más omnipresentes, aumentando la velocidad de los negocios y el desarrollo industrial de forma exponencial. El IIoT va más allá de la comunicación M2M [86], se trata de la conectividad de los sensores, dispositivos y máquinas a través de Internet. Comprende la conexión de redes industriales y de servicios con diferentes infraestructuras de almacenamiento de información a través de la prestación de servicios de software y su control autónomo en la nube. La Internet industrial de las cosas es la nueva

revolución industrial y el aumento en el uso de sensores, análisis avanzados de la información y la toma de decisiones inteligentes está cambiando el mundo a sí mismo.

Los continuos avances tecnológicos han permitido el desarrollo de nuevos métodos de comunicación entre personas y objetos[65], que permiten el intercambio de información dentro de los parámetros de velocidad y seguridad. Internet de las cosas ha aprovechado estos avances tecnológicos y la inclusión de nuevos elementos dentro de los sistemas de información para permitir el acceso remoto y el control de los diferentes sistemas. Bajo la filosofía de comunicación ubicua y comunicación máquina a máquina[10], Internet de las cosas se define como un conjunto de tecnologías diseñadas para permitir la conexión de objetos heterogéneos a través de diferentes redes y métodos de comunicación[87]; Su principal objetivo es posicionar dispositivos inteligentes en diferentes ubicaciones para capturar, almacenar y administrar la información para que sea accesible en cualquier parte del mundo para cualquier persona.

El interés en **IoT** sigue creciendo en diferentes tipos de industria como es en la agricultura, la alimenticia, la farmacéutica y la minera, entre muchas otras. En estos casos, la tecnología que más ha sido utilizada es RFID[20][88], empleada para mejorar la eficiencia de las cadenas de suministro [89] y mejorar la información obtenida durante todo el proceso en tiempo real [56][90].

Existe gran cantidad de aplicaciones en el IIoT, ejemplo de ello un sistema para poder localizar el transporte y monitorizarlo [11], permitiendo predecir su localización futura y el tráfico que tendrá en la carretera.

Sin embargo, la principal aplicación de IIoT se tiene en las cadenas de suministro donde se utiliza la tecnología RFID con mayor énfasis[56][90]. Por eso mismo IoT tiene un rol tan importante en las industrias logísticas [9]. En estas se utiliza IoT para monitorizar el movimiento de objetos desde su origen hasta su destino a través de la cadena de suministro para identificar productos y obtener información acerca de su almacenamiento, sin embargo al incluir el uso de sensores se podría obtener más datos del entorno, como la localización y condiciones del ambiente[91][92].

3.4.5.1 Arquitectura del Internet de las Cosas Industrial IIoT

Con base en el desarrollo de la tecnología IoT para aplicaciones industriales, ha llamado la atención la implementación de sensores y las ventajas de la inteligencia de detección, con las cuales se ha logrado un eficiente monitoreo y control en la reducción de costos y el consumo de energía de las industrias de Producción de bienes y/o servicio[93][94]. De la misma manera con la ayuda de sensores y dispositivos inalámbricos incorporados en máquinas y sistemas industriales, el mantenimiento de estas máquinas y sistemas es controlable y automatizable[91].

Las diferentes arquitecturas propuestas IoT no han convergido a un modelo de referencia [95] o una arquitectura común. En la literatura actual, se puede distinguir entre varios modelos, como se puede ver en la Figura 6 [96]. El modelo básico que tiene tres capas (capa de aplicación, de red y de percepción) fue diseñado para dirigirse a tipos específicos de canales de comunicación y No cubre todas las tecnologías subyacentes que transfieren datos a una plataforma IoT, otros modelos de cuatro capas (capa de sensores, de red, de servicios e interfaces)[59] están diseñados en el contexto de industria 4.0 [97].

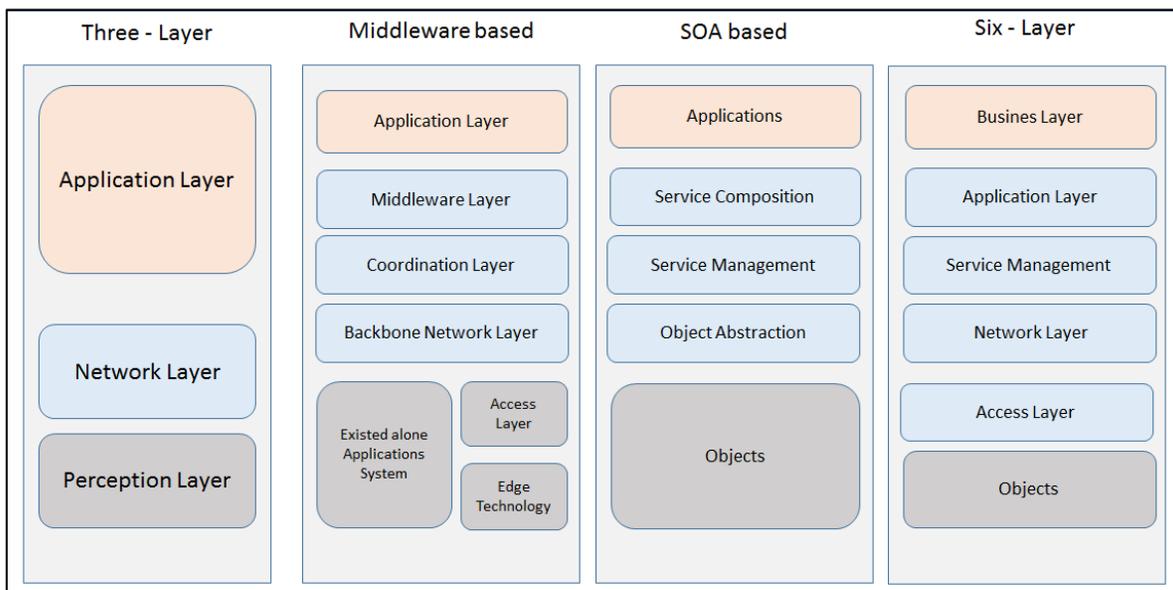


Figura 6. Arquitecturas IoT [96].

CAPÍTULO 4 – REDES SOCIALES

4.1 Definición

Una red social es un conjunto de personas o grupos con una estructura de relaciones o interacciones entre ellos. Desde los años treinta, los sociólogos tienen en cuenta este tipo de estructuras para comprender el funcionamiento de la sociedad[98]. En el contexto sociológico los estudios típicos se dirigen a definir, analizar y medir variables como la centralidad de los individuos (quiénes son más importantes, están mejor conectados con otros o tienen mayor poder), la centralización (distribución de la centralidad), o la cohesividad de grupos (tendencia de sus miembros a mantenerse unidos persiguiendo ciertas metas o, también, resistencia del grupo a romperse)[99].

De igual manera, una red social es una estructura social que se puede representar a través de uno o varios grafos, en la cual los nodos representan individuos u otras redes, denominados actores, y las aristas, relaciones entre ellos. Las relaciones pueden ser de distinto tipo, pero están principalmente destinadas a la realización de operaciones de intercambio.

Las redes sociales se distinguen de las redes físicas, en el sentido de que las conexiones entre estas últimas se dan entre objetos, tales como edificios, equipos, sistemas de información, unidos a través de canales que permiten el flujo de intercambios. Dado lo anterior, es evidente la importancia económica de las redes físicas, puesto que ellas incluyen infraestructura de transporte (vías aéreas, terrestres y marítimas), infraestructura de comunicaciones (telecomunicaciones e Internet) o sistemas de distribución.

Si bien las redes sociales y las físicas son diferentes, están relacionadas de manera cercana. La explotación de una red física con fines económicos involucra la coordinación que proveen las redes sociales. Por ejemplo, una red para exportación de los productos de un emprendimiento a mercados internacionales es una red física, pero las redes de comunicación entre los comerciantes de un país y otro son sociales. Las redes físicas consisten en el flujo de bienes y servicios, mientras que las redes sociales consisten, entre otros aspectos, en el flujo de necesidades, acuerdos de precios y pactos de

colaboración. Así, una red física compromete inversión en capital financiero, representada, por ejemplo, en la infraestructura usada para el transporte o en la compra de una carta de crédito; una red social representa la existencia de capital social definido por relaciones de compromiso y confianza entre los actores. La inversión en capital financiero supone el flujo de bienes tangibles; por su parte, el capital social supone el flujo de bienes intangibles.

Las redes sociales también varían en duración, dependiendo del tipo de relaciones de intercambio o transacciones que realicen sus actores. Existen redes sociales que son solamente transitorias, en las que grupos o uniones temporales se relacionan para aprovechar oportunidades específicas de mercado y luego los actores se dispersan.

4.1.1. Antecedentes de análisis de Redes Sociales.

El origen del análisis de redes nace de la Sociología, enfatizando que las relaciones de interacción y comunicación son la clave para entender la vida social.

Con este enfoque se desarrolló la **sociometría**, la cual buscaba la investigación de las relaciones y como estas servían de limitación u oportunidad para tomar acciones en el ambiente social[100].

4.1.2. Redes Sociales: Características y Propiedades.

Noel Tichy, Michael Thusman y Charles Fombrun[101] clasifican las principales propiedades de las redes de la siguiente forma:

a) Por la Naturaleza de las Relaciones – Donde se involucra la fortaleza de la relación entre dos actores:

- Intensidad – Representa la fuerza de la interrelación entre los actores.
- Claridad de Expectativas – Grado en que se han definido las expectativas acerca del comportamiento del otro actor.
- Reciprocidad – Es cuando se tiene el mismo grado de intensidad a todas las partes de la relación.

- Multiplexity– Grado en el cual varios actores son unidos por varias relaciones.

b) Por Contenido Transaccional – Según la naturaleza del intercambio.

- De Amistad.
- De influencia.
- De Intercambio de Información.
- De Intercambio de Bienes o Servicios.

c) Por las Características Estructurales – Donde se toma en cuenta el patrón de relaciones de los actores.

- Redes externas.
- Redes Internas.
- Clusters (sub-grupos) dentro de la red.
- Individuos como nodos especiales dentro de la red.

4.1.3. Red Social Abierta (RSA).

La red social abierta integra activamente a las personas con el mundo público, abriendo canales con las estructuras comunitarias e institucionales de su sociedad. Estos canales posibilitan, por un lado, la obtención de los recursos fundamentales para su existencia como ciudadano, y por otro, la participación activa, en función de sus valores y modelos de sociedad, en la toma de decisiones que conciernen la calidad de su existencia social.

- **Es un sistema de comunicación.** La RSA no se estructura en torno a un nodo/foco como lo hace la red social focal. La RSA constituye un sistema más amplio y flexible de conversaciones y articulaciones multidimensionales entre organizaciones, instituciones, asociaciones, grupos y actores individuales. En rigor, una RSA no es un sistema que posea una positividad social distinta de las redes focales. Su diferencia está más bien construida por el observador (científico, interventor) que practica una distinción basada en intereses teóricos e interventivos diferentes.

- **Es un sistema de apoyo.** Lo que interesa en la RSA es cómo un conjunto de entidades sociales se articulan e intercambian recursos en relación a un objetivo común determinado. La RSA se articula e intercambia recursos en torno a un tema, en otros términos, en esta red la focalización es temática. La RSA se estructura en torno a la prevención del consumo de drogas, de la salud mental, de la ecología, de los derechos humanos, del Sida, de la Infancia, de la Mujer, etc.
- **Es un sistema de participación social.** Este tipo de red puede constituirse fuera de las instituciones establecidas (ejemplo, asociaciones ecológicas que proponen modelos alternativos a las políticas oficiales), o incorporando esta institucionalidad (ejemplo, sistemas comunales de prevención que articulan organizaciones comunitarias e institucionales, con el municipio como centro estratégico) según sea el caso. Se conforman cuando las personas se organizan para cambiar algo en conjunto y de manera no burocrática. Su fuerza radica en la aceptación global de parte de las personas que la componen de sistemas de recompensa mutua y de solidaridad que refuerzan la cohesión. Sin sistemas de recompensa, sin el placer compartido de participar a una causa común, la red se desagrega y desaparece. Su cohesión interna nace del consenso, del compartir los mismos valores que cimentan y unen a sus componentes. Cada uno de estos componentes puede ser a su vez "nodo" y "vínculo", no hay funciones especializadas, sino roles múltiples. En una RSA la información es compartida y el poder distribuido.

4.1.4. Estructura de la Red Social Abierta.

Las RSA se distinguen de las organizaciones formales porque poseen un grado de estructuración más débil:

- No hay organigrama ni cronogramas estructurados.
- No poseen locales propios.
- No tienen presupuesto propio.
- Su funcionamiento es más bien sociográfico (horizontal).

- No hay constreñimientos institucionales u obligaciones formales para los participantes.
- Su estabilidad y permanencia dependen de la eficacia y eficiencia de su funcionamiento y de los procesos motivacionales de los participantes (nodos).

En su dimensión operativa una RSA está compuesta de **actores** que pertenecen y representan a organizaciones. Los actores son personas significativas en el sistema social al que pertenecen ya sea éste una organización institucional o comunitaria. Una persona se transforma en actor cuando perteneciendo significativamente a una organización representa su voz y se vincula en red con otros actores pertenecientes a otras organizaciones.

En una RSA podemos distinguir dimensiones de tipo estructural, procesos motivacionales y procesos afectivos.

Las principales dimensiones de tipo estructural son las siguientes:

1. Densidad.
2. Frecuencia de contacto.
3. Tipo de contacto.
4. Naturaleza de la relación.
5. Accesibilidad.
6. Reciprocidad.

4.1.5. Mediciones del análisis de las Redes.

Las diferencias en el acceso de las redes sociales se examinan usando mediciones sociométricas de la **atractividad interpersonal**:

- Amistad – Amigo más cercano.
- Influencia percibida – La persona que tiene más que decir acerca de cómo la organización está funcionando.
- Respeto Profesional – Respeto o admiración organizacional.
- Conocimiento Organizacional – Individuo que es fuente de información.

Los resultados de estas preguntas, se revisan según el análisis de redes sociales, a través de la **centralidad** que ayuda a identificar a los actores que

tienen más conexiones o relaciones que los demás, lo cual puede indicar un alto grado de centralidad[102].

La centralidad de una red, es la mínima distancia entre una persona y todas las demás del grupo. Se mide contando el número de relaciones o ligas entre la persona y cada una de las demás, para resultar en un total de centralidad.

Según Freeman[103], tanto Harold Leavitt como Sidney Smith, Bavelas y Barrett, concluyen en sus estudios que la Centralidad está relacionada con la eficiencia de un grupo para resolver problemas, con la satisfacción del personal de los miembros y con la percepción que se tiene del liderazgo.

La Centralidad también aporta ventajas, explica que estas ventajas son las que proporcionan acceso a estructuras de comunicación, interacción e intercambio profesional, lo que es crítico para el buen desempeño del personal.

Las principales mediciones de centralidad que fueron definidas como:

- **Intermediación** (“Betweenness”) medida de centralidad global que determina en qué medida un punto hace de “intermediario” entre otros puntos por estar situado en el camino entre ellos.
- **Grado** (“Degree”) centralidad de los puntos del grafo. Un punto es central si tiene un grado alto, lo que se corresponde con la idea intuitiva de centralidad según la cual un punto es central si está bien conectado con los demás puntos de su entorno.
- **Cercanía** (“Closeness”) es una medida de centralidad global de cada punto respecto a los demás y expresada en términos de la distancia entre los puntos.
- **Prestigio** (“Prestige”) es una medida que refleja la popularidad, es el número de veces en que un actor es elegido por otros, y los lazos directos en el cual el actor es el objeto de la relación.
- **Poder** (“Power”) es una medida que involucra algunas interdependencias entre dos o más actores, la habilidad de un actor para vencer la resistencia y lograr el resultado deseado o simplemente de afectar los resultados y conseguir que las cosas se hagan.

Donde la Intermediación (betweenness), remarca la ventaja que da a una persona el estar estratégicamente en las líneas de comunicación que ligan a otras personas, ya que esta persona puede influir en el grupo ocultando o

distorsionando la información que transmite. Esto representa el control potencial, o la posibilidad de incrementar la dependencia de los demás en un actor.

La cercanía (Closeness) se refleja en la organización como independencia, donde se evita el control de otros, o eficiencia, donde el actor puede llegar a todos los demás en el número más corto de relaciones o ligas directas o indirectas. Y Un actor que tiene alto grado (degree) es capaz de intercambiar su punto de vista con muchos de la organización, esto lo convierte en un actor influyente.

El prestigio o atractividad es medida sociométricamente por el número de selecciones recibidas en las preguntas anteriores de amistad, influencia percibida y respeto profesional.

4.1.6. Ámbitos de articulación.

Una RSA puede producir por lo menos tres tipos de articulaciones entre nodos:

- **Una articulación intracomunitaria.** En este caso la red vincula nodos pertenecientes al ámbito comunitario. Entendemos aquí por ámbito comunitario el espacio de vida de las personas. Los nodos en este caso son organizaciones comunitarias, ya sea territoriales (juntas de vecinos, uniones comunales, organizaciones barriales, comités de adelanto), como funcionales (centros de madres, asociaciones culturales, religiosas, micro comunidades, clubes deportivos, etc.).
- **Una articulación intrainstitucional.** En este caso la red vincula nodos pertenecientes al ámbito institucional, es decir, el espacio de las organizaciones formales, tanto públicas como privadas: ministerios, servicios estatales, municipios, hospitales, consultorios, iglesias, ONG's, corporaciones, fundaciones, etc.
- **Una articulación institucional/comunitaria.** Como su nombre lo dice, en este caso la red vincula organizaciones institucionales con organizaciones comunitarias.

Estos tres tipos de articulación pueden darse en forma separada (redes intersectoriales, redes comunitarias, etc.) o en forma simultánea y combinada.

4.1.7. Historia de las redes sociales (RS) en la sociedad de la información y el conocimiento (SIC).

La primera gran revolución de las redes sociales acontece influida por el impacto del telégrafo, cuyo uso supuso el paso una organización de la información y el poder centralizada y nacional, propia del Estado moderno, a la descentralizada e internacional de los siglos XIX y XX. A partir de los años 90, un impacto similar acontece aunado al fenómeno de la globalización, los cambios tecnológicos y el surgimiento de la Sociedad de la Información y la Comunicación, con una nueva valoración del acceso al conocimiento y a su construcción. El acceso directo a la información facilitado por las tecnologías emergentes, han propiciado nuevas transformaciones en las RS y por ende en la sociedad. En la actualidad, las tecnologías como Internet la impulsan a asumir cada vez más una forma “*distribuida*” en la que cualquiera puede, potencialmente, encontrar, reconocer y comunicar con cualquiera. En consecuencia, las redes sociales han evolucionado hacia estructuras distribuidas.

Si bien, los sitios web dedicados a las comunidades de interés siguen existiendo y prosperando, cada vez más las redes sociales se organizan principalmente en torno a las personas y no entorno a intereses.

En la actualidad, las redes sociales tienden a estructurarse como redes personales (o “egocéntricas”), con los actores en el centro de su propia comunidad. En este proceso de cambio y expansión, en muchos casos las RS en se han convertido en negocios exitosos, lo que ha transformado su naturaleza esencial, incorpora nuevos intereses y modifica su filosofía de funcionamiento libre e inclusiva.

Del 2003 en adelante, numerosas redes sociales son lanzadas al espacio web y algunas se hacen muy populares, como fue el caso de *Tribe* y *Myspace*. Esta actividad provoca el interés de empresas para hacer uso de las RS con visión de negocios. *Google*, lanza en enero de 2004 *Orkut*, a partir de un

experimento que uno de sus empleados realizaba en su tiempo libre. En 2005 se incorpora *Yahoo 360°*, entre otras.

Adicionalmente, se evidenciaba que la identificación de intereses comunes, de una oferta y una demanda, parecía la mejor e interminable estrategia para el lanzamiento de nuevas redes sociales: *Couchsurfing* apareció para conectar viajeros con personas que ofrecían alojamiento; *MyChurch* unía a iglesias cristianas y a sus miembros, entre muchas otras opciones.

Paulatinamente, los sitios web centrados en compartir información y generar contenidos propios por los usuarios, comenzaron convertirse en redes sociales. Ejemplos de ello son *Flickr* (para compartir fotos), *Last.FM* (hábitos de escuchar música) y *YouTube®* (para compartir vídeos). A su vez, en 2004 se inicia *Facebook®* en Harvard University, pasatiempo del estudiante Mark Zuckerberg, con el cual empezó a dar servicio a los otros alumnos. La inmediata popularidad alcanzada por esta red social marcó su acelerado desarrollo. Pronto se expande a otras instituciones como MIT, Boston University y Boston College, así como otras prestigiosas universidades norteamericanas y un año después, contaba con más de un millón de usuarios. Para el 2005, *Facebook®* se amplió para incluir a estudiantes de secundaria, profesionales dentro de redes corporativas, y, con el tiempo, al mundo entero.

La mayoría de las RS siguieron como *Facebook®*, otras se centraron expresamente en un público más restringido, como una estrategia para perfilarse como sitios “selectivos”. De esta manera, diferentes RS han marcado la delantera en sus ámbito de actuación específica, en sus modos y sus medios de funcionamiento, así como en los modelos de negocio desarrollados, las cuales se han constituido en nuevos paradigmas en la Sociedad de la Información y el Conocimiento, SIC.

4.2. Redes sociales e internet de las cosas

Las redes sociales conocidas como Online Social Networks (OSN) o Social Network Sites (SNS) [98], son una pieza importante en la convergencia entre el mundo real y el mundo digital en la Web 2.0 [104].

Otras redes sociales que proporcionan la base para encontrar, conocer y mantener relaciones sociales con otros usuarios que tienen intereses similares [105], las OSNs son consideradas para ponerse al día acerca de la información personal y las actividades actuales de las relaciones sociales [106].

Las redes sociales online tiene aplicaciones académicas, siendo muy importantes por su facilidad de uso, velocidad y alcance [107][108], estableciendo tendencias en temas como el medio ambiente, la política, la tecnología y la industria [108]. Varias propuestas sobre la combinación de OSNs e Internet de las Cosas basadas en una deducción hecha por científicos de Ericsson [109], quienes observaron que las personas se familiarizan mucho mejor con las tecnologías de Internet de las Cosas si existe una analogía entre estas y los hábitos diarios en las redes sociales como Twitter o Facebook. Las propuestas realizadas se basan en aportar a los objetos inteligentes habilidades de socialización con el fin de establecer relaciones entre los propios objetos. Este concepto se denomina Internet de las Cosas Social (SIoT) [51], [109]. Sin embargo, utilizar una OSN en el marco de IoT requiere que se tengan servicios de identificación y autorización, una API que permita acceder y manipular el grafo de la OSN, publicar, leer y añadir lugares del mundo físico, y ofrecer una forma fácil de crear aplicaciones de terceros [104].

Al uso de servicios web para diversos objetivos dentro de Internet de las Cosas se denomina la Web de las Cosas o Web of Things (WoT), permite analizar el uso de contenedores de aplicaciones de terceros que las redes sociales ofrecen como servicio. Actualmente Twitter tiene la capacidad de publicar eventos procedentes de objetos inteligentes, sensores y actuadores, en las redes sociales [110]. Por otro lado, hay investigaciones que permiten el acceso a los objetos en Twitter [111]. Esta propuesta la denominaron Social Access Controller (SAC).

Otros autores [112] presentan como los ciudadanos pueden servir como sensores humanos para proveer información suplementaria, alternativa y complementaria sobre las ciudades, utilizándolos así como otro sensor más dentro de Smart Cities.

4.3. Redes sociales y Cloud Computing

Cloud Computing es el éxito de la computación orientada a servicios y que ha revolucionado la infraestructura de computación tradicional [113][114][115], al utilizar la nube, ha permitido realizar toda la computación en ella, sin necesidad de poseer servidores o mainframes. Esto ha abierto un nuevo modelo económico en la informática, las empresas ofrecen hardware, plataformas y software como servicios de suscripción y que se encuentran alojados en la nube [116][114][117][118], todos estos empaquetados y bajo demanda, y normalmente a través de Internet [119][120]]. Los servicios que se encuentran en la nube permiten ser gestionados, ya sea para aprovisionarlos, modificarlos o liberarlos con un mínimo esfuerzo gracias a las herramientas que proveen los proveedores de este tipo de servicios[121]. Además, los servicios ofrecidos son robustos, fiables y están disponibles a cualquier hora y desde cualquier lugar [114], [117][122]. Por estos motivos los tres componentes básicos de *Cloud Computing* son la virtualización, la multitenencia y los servicios web[119].

Cloud Computing está dividida en tres niveles [117], [123], [124], [125]: IaaS, PaaS y SaaS. Estos niveles, junto a otros más especializados, son conocidos como XaaS [126][127]. Estos se pueden ver en la figura 7.

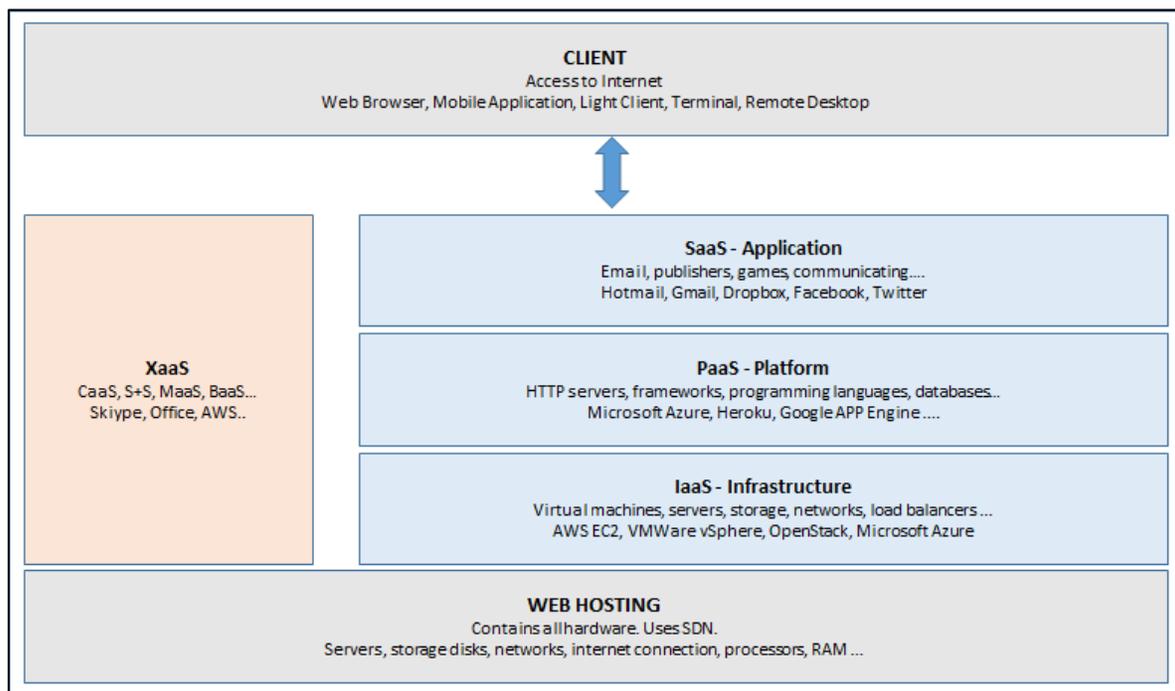


Figura 7. Capas de la nube

Estas tres capas juntas son conocidas como XaaS, no obstante, servicios más específicos que se encuentran dentro de estas capas y poseen su propio nombre. También, como se muestra, las capas superiores pueden usar las capas inferiores, es decir, un SaaS puede estar soportado sobre un PaaS y este a su vez sobre un IaaS, el cual ofrece la abstracción sobre el hardware base que puede utilizar a su vez SDN para controlar todo el hardware de una forma mucho más sencilla.[128]

CAPÍTULO 5 – INGENIERÍA DIRIGIDA POR

5.1. Ingeniería Dirigida por Modelos MDE

Ingeniería Dirigida por Modelos (MDE) es un enfoque para el diseño y desarrollo de software[129], permitiendo una abstracción de diferentes actividades del ciclo de desarrollo de software[130]. Esta abstracción ayuda a que los usuarios finales puedan interactuar con el sistema utilizando los conceptos más comunes, como son la estructura, las entradas y salidas, ya sea de forma textual o gráfica[131] y eliminando toda aquella información no relevante para su propósito. Los enfoques basados en modelos capturan la información relevante del sistema en un modelo específico del dominio de manera que, todos estos conceptos se muestren de forma familiar a los usuarios del sistema [132].

En otras palabras, MDE es la técnica que se basa en utilizar modelos para desarrollar software[133], abstrayendo las partes relevantes del problema, El objetivo de la Ingeniería Dirigida por Modelos es incrementar la productividad y

reducir el tiempo de desarrollo permitiendo que este sea más cercano al dominio del problema y con una abstracción mayor que los lenguajes de programación. La clave de MDE es transformar estos modelos, que están en un nivel de abstracción alto, en modelos específicos de la plataforma por medio de herramientas que puedan transformar los modelos en código fuente[134], ya sea mediante un proceso automático o semiautomático[135].

La Ingeniería Dirigida por Modelos es el último paso en el nivel de abstracción. Los modelos son los artefactos claves de MDE. Un modelo es una descripción de un sistema escrito en un lenguaje bien definido[136]. En este sentido los modelos juegan un papel preponderante ya son los directores de todo el proceso de desarrollo del software[137].

Ingeniería dirigida por modelos (MDE) se caracteriza por: a) poner al frente el nivel de abstracción ocultando los detalles específicos en la plataforma; b) aprovechar el uso de los modelos en todas las fases de desarrollo de software para mejorar la comprensión; c) desarrollar Framework y lenguajes específicos para lograr la comprensión del dominio; y d) obtener provecho de las transformaciones para automatizar trabajo repetitivo y mejorar la calidad del software[138].

5.1.1 Terminología usada en MDE

En la figura 8 se ilustran una serie de términos relacionados con MDE los cuales permiten comprender el funcionamiento de dicha tecnología. A continuación se describirá cada uno de los términos.

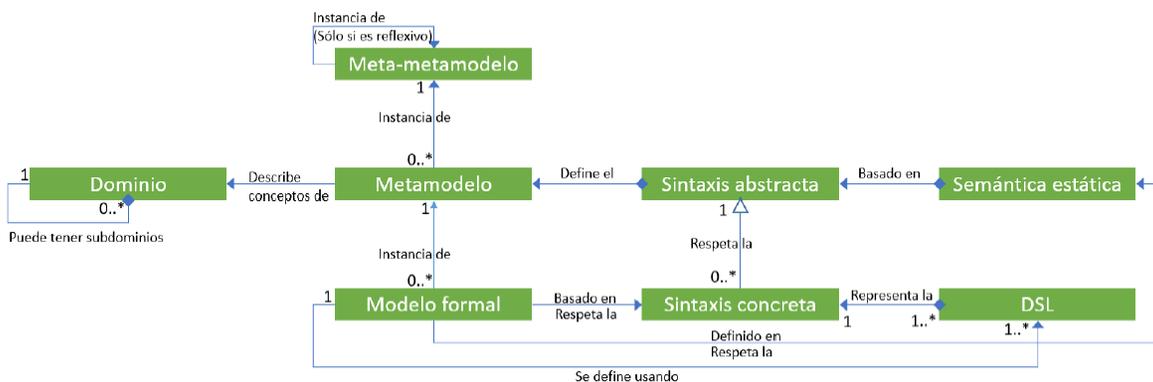


Figura 8. Arquitectura de los conceptos de MDE.

Fuente [139]

Sistema: conjunto de partes y de las relaciones entre estas partes que pueden ser organizados para lograr un propósito. Así, un sistema puede ser un sistema hardware o software, una compañía, procesos de negocio o la combinación de diferentes sistemas. Un sistema está formado por la plataforma y su aplicación [140].

Dominio: el dominio siempre es el punto inicial de MDE y delimita el campo de conocimiento. Así, el dominio es el área de conocimiento sobre la que se trabaja y sobre la que se quiere resolver el problema. Estos se dividen en dominios tecnológicos y profesionales, donde los primeros hacen referencia a la tecnología de desarrollo de software y los segundos a los conceptos que manejará la aplicación. Los dominios a su vez pueden estar compuestos de varios subdominios[141]

Metamodelo: es una especificación del lenguaje de modelado que define las características del modelo y permite verificar el modelo expresado en ese lenguaje determinado de manera formal, es decir, es un modelo del lenguaje de modelado [142].

Metamodelo reflexivo: es cuando el metamodelo de un lenguaje de modelado usa el mismo lenguaje de modelado, es decir, se define el metamodelo utilizando el mismo lenguaje en el que el metamodelo está descrito. Dentro de estos, el metamodelo mínimo reflexivo es aquel que usa el mínimo número de elementos del lenguaje de modelado para los propósitos de ese metamodelo, luego, si se eliminase cualquier elemento sería imposible modelar o expresar cualquier estado esencial [139].

Meta-metamodelo: Es una especificación del metamodelo que define las características del metamodelo y permite verificar el metamodelo expresado en ese lenguaje determinado, es decir, en un modelo del metamodelo. El tener un meta-metamodelo, permite que exista un metamodelo para cada dominio del conocimiento a tratar, mientras se tiene un meta-metamodelo común a todos estos para así poder realizar operaciones sobre ellos, como pueden ser transformaciones automáticas, validaciones y búsquedas[139]

Lenguaje de dominio específico: un DSL está constituido por la estructura, los términos, notaciones, sintaxis, semántica y reglas de integridad

que son usadas para expresar un modelo. Algunos ejemplos de lenguajes de modelado son UML, SQL Schema, Business Process Management and Notation (BPMN), E/R, Ontology Web Language (OWL) y XML Schema [140].

Punto de vista: un punto de vista de un sistema es una técnica de abstracción en la que se seleccionan una serie de conceptos y reglas estructurales de ese sistema con el fin de centrarse en las preocupaciones importantes de ese sistema. Es decir, se crea una abstracción para suprimir detalles irrelevantes y así obtener un modelo simplificado de una parte del sistema [143]. Cada punto de vista puede tener uno o más modelos, también conocidos como vistas.

Vista: una vista es la representación del sistema, desde una perspectiva elegida de un punto de vista. Por ejemplo, si tenemos un sistema para mostrar los datos de los usuarios de un videojuego online, una vista podría ser, la forma en que está la información estructurada, otra vista la que muestra la información que puede ver cada rol existente, otra vista podría ser, la que contiene los protocolos utilizados para transmitir la información y otra vista, para saber cómo se obtiene la información de los usuarios a partir de la información del sistema [139].

Capas de arquitectura del metamodelo[144]: Arquitectura basada en cuatro niveles de abstracción que van a permitir distinguir entre los distintos niveles conceptuales que intervienen en el modelado de un sistema. Esos niveles se les denominan comúnmente con las iniciales M0, M1, M2 y M3.

En la Figura 9 se muestra la arquitectura en cuatro capas y como cada capa inferior es una instancia de la superior, a excepción de la capa M1 que sirve para representar los elementos del mundo real, que se encuentran en la capa M0 y la capa M3, es decir, el meta-metamodelo, que es una instancia de sí mismo en el único caso de que se trate de un metamodelo reflexivo. A la derecha de la imagen se puede ver unos posibles ejemplos correspondientes a cada capa. En la capa M3 se encuentra MOF, que es el meta-metamodelo reflexivo del OMG, en la capa M2 estarían Ecore y UML, que son dos metamodelos basados en MOF, de los cuáles el primero fue creado como el núcleo de Eclipse Modeling Project (EMF) y el segundo fue creado por el OMG

para ser el núcleo de MDA[145]. En la capa M1 se encuentran diferentes modelos realizados utilizando el mismo o diferentes metamodelos y que sirven para representar sus correspondientes elementos de la capa del mundo real, M0.

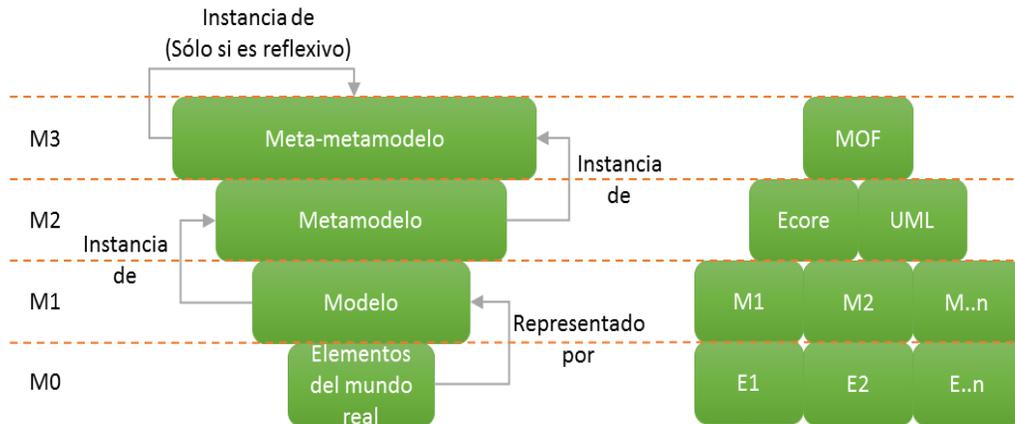


Figura 9. Arquitectura de cuatro capas [67]

- El nivel M0 – Las instancias. El nivel M0 modela el sistema real, y sus elementos son las instancias que componen dicho sistema.
- El nivel M1 – El modelo del sistema. Los elementos del nivel M1 son los modelos de los sistemas concretos. En el nivel M1 es donde se definen los conceptos. Existe una relación muy estrecha entre los niveles M0 y M1: los conceptos del nivel M1 definen las clasificaciones de los elementos del nivel M0, mientras que los elementos del nivel M0 son las instancias de los elementos del nivel M1.
- El nivel M2 – El modelo del modelo (el metamodelo). Los elementos del nivel M2 son los lenguajes de modelado. El nivel M2 define los elementos que intervienen a la hora de definir un modelo del nivel M1. En el caso de un modelo UML de un sistema, los conceptos propios del nivel M2 son “Clase”, “Atributo”, o “Asociación”. Al igual que pasaba entre los niveles M0 y M1, aquí también existe una gran relación entre los conceptos de los niveles M1 y M2: los elementos del nivel superior definen las clases de elementos válidos en un determinado modelo de nivel M1, mientras que los elementos del nivel M1 pueden ser considerados como instancias de los elementos del nivel M2.

- El nivel M3 – El modelo de M2 (el meta-metamodelo). Finalmente, el nivel M3 define los elementos que constituyen los distintos lenguajes de modelado. De esta forma, el concepto de “clase” definido en UML (que pertenece al nivel M2) puede verse como una instancia del correspondiente elemento del nivel M3, en donde se define de forma precisa ese concepto, así como sus características y las relaciones con otros elementos.

Sintaxis abstracta: la sintaxis abstracta de un metamodelo es básicamente un modelo de datos de como almacenar o intercambiar datos en la estructura semántica del modelo a realizar, además de servir para definir las restricciones y estructura de los elementos del modelo[142].

Sintaxis concreta: Puede ser textual o gráfica. Bajo una misma sintaxis gráfica, se pueden tener diferentes sintaxis concretas, lo que nos permite tener diferentes tipos y vistas de esta, permitiendo diferentes tipos de forma para definirla, ya sea gráfica, textual o en forma de árbol, entre otros. Es decir, la sintaxis concreta define cómo se deben representar los modelos y que notaciones deben de utilizar los usuarios para ello [146].

Sintaxis estática: La semántica estática puede estar basada en términos de las cosas observadas en el mundo donde se describe, como puede ser el envío de mensajes, el estado de los objetos o el cambio entre estado, entre otras cosas, o bien, puede estar basada en traducciones de lenguajes de alto nivel construidos a partir de otras construcciones que tienen el significado bien definido[147].

Modelo: describe los elementos que pueden existir en un sistema, así como sus tipos. Por ejemplo, si definimos la clase “Persona” en un modelo, podremos utilizar instancias de dichas clases en nuestro sistema [144].

Infraestructura: conjunto de piezas software o hardware presentes cuando se desarrolla un artefacto software [147].

Plataforma: infraestructura de software que contiene una serie de recursos que permiten implementar o soportar un sistema determinado o implementado una tecnología específica [147]. La plataforma soporta la ejecución de la aplicación y, en conjunto, constituye el sistema. Así, la aplicación provee de la parte funcional del sistema descrito en el modelo.

Arquitectura: el propósito de la arquitectura es definir o mejorar los sistemas de manera que ayude a comprender el ámbito del sistema de interés y los requisitos [140].

Semántica del problema: esto hace referencia a que, cada vez que se incluye un artefacto en el modelo, lo que se está añadiendo es significado, debido a que cada concepto capturado en el modelo tiene su propio significado. Así, cada concepto del lenguaje se mapea directamente a un concepto del dominio que se modela [146].

5.2 Arquitectura Dirigida por Modelos (MDA)

5.2.1 Definición

MDA es quizá la más prometedora y ambiciosa propuesta de la OMG para abordar los retos de los actuales sistemas de software que se encuentran en constante cambio y con alto grado de interconexión. MDA se basa en los conceptos de MDE pero mediante la utilización de estándares de la OMG como son: Meta-Object Facility (MOF), XML Metadata Interchange (XMI), Object Constraint Language (OCL), Unified Modeling Language (UML), Common Warehouse Meta-model (CWM).

Con la combinación de estos lenguajes y el uso de mecanismos tales como la abstracción, refinamiento y vistas, MDA define un conjunto de modelos Computation Independent Model (CIM), Platform Independent Model (PIM), Platform Specific Model (PSM) los cuales definirán el sistema desde diferentes perspectivas y niveles de abstracción [148].

La MDA rescata la importancia de los modelos como estrategia clave para entender y especificar una solución de software y progresivamente obtener la solución final. Aparece como respuesta a dos problemas fundamentales dentro de la industria informática: La diversidad de plataformas y tecnologías y la acelerada evolución tecnológica [149]. La estrategia para alcanzar beneficios fundamentales como productividad, interoperabilidad, portabilidad y facilidad de mantenimiento, se plantean en las ideas del manifiesto MDA [150].

5.2.2. Niveles de abstracción

La especificación MDA define cuatro puntos de vista o niveles de abstracción para analizar sistemas, de modo que la idea principal es empezar desde el nivel de abstracción más alto e ir haciendo transformaciones automática o semi-automáticamente hasta llegar a código ejecutable por una máquina física o virtual. Para cada punto de vista o nivel de abstracción se define un modelo del sistema:

CIM (Computational-Independent Model)

Corresponde con la captura de requisitos o modelo de dominio de la aplicación. El CIM se centra en los requerimientos y representa el nivel más alto del modelo de negocios. El CIM trasciende a los sistemas; cada proceso de negocio interactúa con trabajadores humanos y/o componente de máquina. Un objetivo fundamental del CIM, es que cualquiera que pueda entender el negocio y los procesos del mismo puede comprenderlo, ya que éste evita todo tipo de conocimiento especializado o de sistemas.

PIM (Platform Independent Model)

Cuando se realiza la transformación de CIM a PIM, se baja el nivel de abstracción de modo que se tiene un modelo que está pensado para ejecutarse en un ordenador pero sin tener en cuenta la plataforma. Con el fin de poner en práctica un PIM en una plataforma específica, se utiliza una herramienta para generar el PSM de la PIM. La herramienta comprende la tecnología de destino y sabe cómo traducir las construcciones lógicas del PIM en una forma adecuada para la plataforma elegida. Estas asignaciones pueden ser complejas, un elemento dado de la PIM puede ser asignado a múltiples elementos en el PSM.

Las características clave de PIM son las siguientes [151]–[154]:

1. Abstracción de alto nivel que es independiente de cualquier tecnología de implementación.
2. Describe un sistema de software que soporta algunos negocios y se especifica usando xUML.

3. El sistema es modelado desde el punto de vista que brinde mejor apoyo a la empresa.

4. Describe el comportamiento del sistema, independientemente del entorno informático y las tecnologías de implementación.

5. Los PIM pueden reutilizarse en múltiples plataformas.

PSM (Platform specific Model)

Es un modelo, pero ahora a nivel de diseño y reflejando la asignación a una plataforma de destino. La etapa final es generar a partir de la PSM el código y otros artefactos técnicos necesarios para implementar y ejecutar el sistema.

Las características clave de PSM son los siguientes [152]–[154]:

1. Especifica cómo la funcionalidad en un PIM se realiza en una plataforma informática.

2. Resulta del PIM mediante la adición de atributos específicos de la plataforma a los componentes estándar.

3. Es común tener múltiples PSMs con un solo PIM.

Una vez generado el PSM, el siguiente paso del proceso de desarrollo de software es la generación del código de PSM y desplegando el sistema en el entorno específico. Varios factores clave involucrados al pasar de PSM a código y despliegue son los siguientes [152]–[154]:

1. Generación de un modelo de objetos.

2. Repositorio de modelos que contiene toda la información necesaria para producir un tiempo de ejecución en una plataforma específica.

3. Traducción de abstracciones en los modelos a implementaciones específicas con la ayuda de compiladores de modelos.

4. PSM es un código ejecutable que se ejecuta en la arquitectura de destino.

Un punto central sobre el enfoque de MDA es que la automatización se utiliza para producir el PSM y Código a partir de los objetos de nivel superior. Esta automatización no sólo acelera el proceso de desarrollo, sino también

asegura que el código es una representación fiel de los modelos de más alto nivel, por lo que los modelos son una representación fiel del código.

5.3 Lenguajes de Dominio Específico (DSL)

5.3.1 Definición

La principal idea compartida por todos los paradigmas englobados dentro del Desarrollo de Software Dirigido por Modelos (DSDM) es la conveniencia de que los programadores empleen lenguajes de más alto nivel de abstracción que los lenguajes de programación, esto es, lenguajes que manejen conceptos más cercanos al dominio de la aplicación. Estos lenguajes que proporcionan mayor nivel de abstracción se denominan Lenguajes de Dominio Específico (DSL).

A diferencia de lo que ocurre con los lenguajes de propósito general, gracias al uso de DSLs, se consigue que los conceptos de un lenguaje se mapeen directamente a conceptos del dominio que se modela, sin posibilidad de interpretaciones erróneas. En muchas situaciones del desarrollo de software se evidencia la repetitividad de los problemas en este caso para solucionarlos se puede utilizar un GPL (General Purpose Language o Lenguaje de Propósito General) como Java y C# o se puede recurrir a la utilización de un DSL.

Los expertos en un dominio crean los lenguajes específicos del dominio (DSL) y sus generadores de código, y los desarrolladores los usan para especificar una solución de alto nivel, de una forma más productiva que escribiendo el código en un lenguaje de programación.

5.3.2 Tipos de Lenguajes de Dominio Específico (DSL)

Se pueden distinguir tres clasificaciones para los DSL [141], desde el punto de vista de la construcción del lenguaje, desde el formato de lenguaje y desde el dominio del problema, a continuación se especifican cada uno.

Desde un punto de vista de la construcción del lenguaje:

Internos: Utilizan un determinado lenguaje anfitrión para darle la apariencia de otro lenguaje concreto. Un ejemplo claro son lo que actualmente se conoce como Fluent Interfaces.

Externos: Tiene su propia sintaxis y es necesario un parser para poder procesarlos. Un ejemplo claro de DSL externo es SQL (Structured Query Language)

Desde el punto de vista del formato del lenguaje:

Textuales: La mayoría de los lenguajes informáticos son textuales y están formados por un conjunto ordenado de sentencias. Un ejemplo muy conocido de DSL textual es SQL utilizado para realizar consultas a una base de datos. Una forma de crear DSLs textuales es mediante la creación de una determinada gramática (por ejemplo utilizando EBNF) y posteriormente crear o utilizar un parser para dicha gramática, para en etapas posteriores poder interpretar el DSL o generar código.

Gráficos: En los últimos años están ganando gran aceptación los lenguajes gráficos, podrían citarse como ejemplo UML. La creación de un lenguaje grafico es similar a la de un lenguaje textual, la única diferencia es que en lugar de usar texto para representar los conceptos, se utilizan conectores y figuras simples.

Desde el punto de vista del dominio del problema:

Horizontales: Los DSL horizontales son aquellos en los que el cliente que utilizará el lenguaje no pertenece a ningún dominio específico. Un ejemplo son los editores visuales de entornos de desarrollo que permiten generar interfaces de usuario automáticamente (por ejemplo Windows Forms de visual Studio).

Verticales: A diferencia de los DSL horizontales, el cliente que utilizará el lenguaje pertenece al mismo dominio que el lenguaje en sí. Como en el ejemplo anterior para un lenguaje de definición de encuestas, los usuarios finales serían los expertos en estadística encargados de definir dichas encuestas.

***PARTE III. DESARROLLO DEL
PROTOTIPO***

CAPÍTULO 6 - DESCRIPCIÓN DEL PROTOTIPO

En este documento se propone una arquitectura y un metamodelo de integración de internet de las cosas, las redes sociales y la industria 4.0.

6.1. Propuesta de Arquitectura de IIoT

Con el propósito de integrar los sensores, los actuadores, las redes, la computación en la nube y las tecnologías del internet de las cosas en el contexto de la industria 4.0, se presenta un prototipo de plataforma de IIoT en 5 capas:

- Sensing layer
- Databases layer
- Network layer
- Data response layer
- User layer

La figura 10 muestra la propuesta de arquitectura para la integración de modelos para el desarrollo del internet industrial de las cosas.

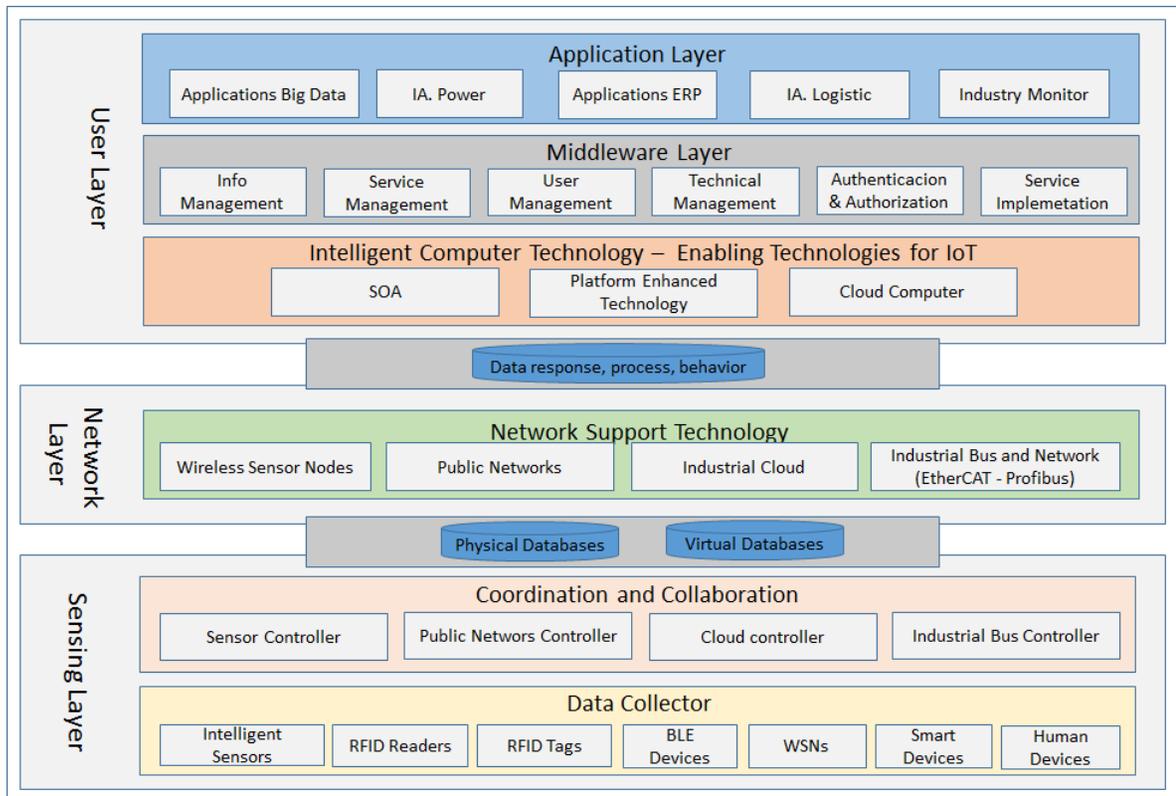


Figura 10. Propuesta Arquitectura de integración

- Sensing layer: Esta capa se encuentra compuesta por los diferentes tipos de dispositivos y determinan directamente la implementación y producción del tipo de dato específico. Desde una perspectiva funcional, es responsable de diferentes actividades como la fabricación, el transporte, la movilidad, la logística y la obtención de datos de sensores u otros dispositivos. La capa de detección se integra con los objetos de hardware disponibles para detectar los estados de las cosas.
- Databases layer: Compuesta por bases de datos físicas y virtuales. Las bases de datos físicas utilizan un conjunto común de SQL y nonSQL (objeto). Permiten una fácil integración con aplicaciones externas, por lo que no deben ser dependientes. En esta arquitectura, hay poca dependencia en características de base de datos propietarias, tales como procedimientos almacenados y disparadores. Los datos IoT provienen de sensores y dispositivos, donde los datos pueden ser recogidos y procesados en tiempo real.

Las bases de datos virtuales permiten exponer esquemas y datos abstractos personalizados. El enfoque de abstracción de datos proporciona vinculación lógica con la base de datos en el nodo de la red, donde los datos están lógicamente y físicamente separados y se puede acceder desde un único esquema virtual. Estas bases de datos abstractas pueden colocarse dentro de contenedores, basadas en las necesidades del dominio del problema. Aquí se almacenan los datos adquiridos sin procesar.

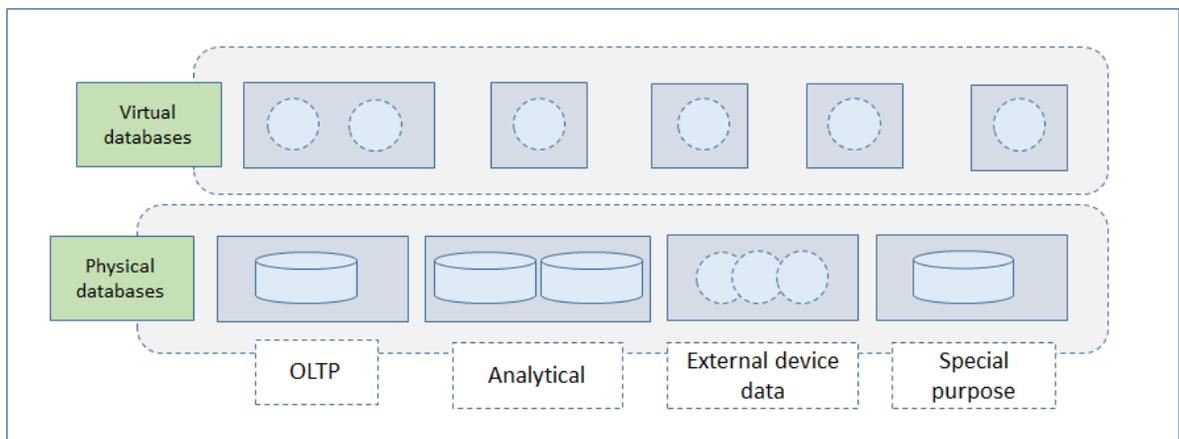


Figura 11. Bases de datos físicas y virtuales
Adaptado de [155]

- Network layer: La capa de red es la infraestructura para soportar conexiones inalámbricas o por cable entre las cosas. La capa de red en IoT, conecta todas las cosas y les permite ser conscientes de su entorno. A través de esta capa se comparten los datos con las todas cosas conectadas. La capa de red agrega datos de Infraestructuras de TI existentes. Es aquí donde se realiza la interacción entre el Sensor layer y el User layer. Se efectúa la gestión de los sensores y actuadores y proporciona información a la siguiente capa. En el contexto de la industria 4.0 se personalizan los servicios proporcionados según los requisitos de la aplicación.
- Data response layer: Esta capa es una colección de Datos que se pueden asignar a Dispositivos, aplicaciones y mantiene la persistencia de otras capas. Se centra en proporcionar respuestas automáticas y aprende a medida que procesa la respuesta. Aquí se mantienen los datos procesados. Los datos se almacenan de modo que la base de datos física se actualiza se

acuerdo a lo definido por el usuario. Todos los datos recibidos de los sensores se procesan en el nodo de la red. Los componentes basados en la nube y los componentes cerca de los dispositivos IoT y sensores están acoplados lógicamente.

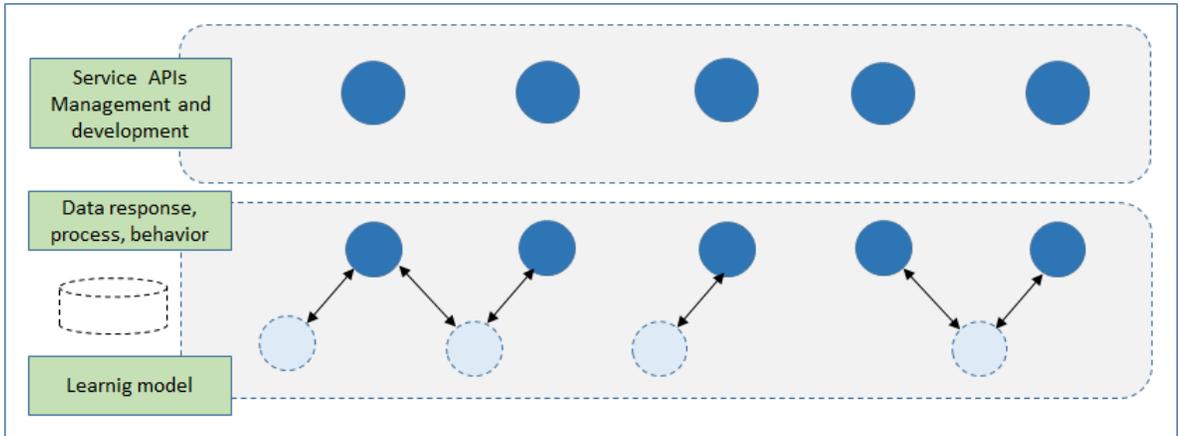


Figura 12. Data response layer
Adaptado de [155]

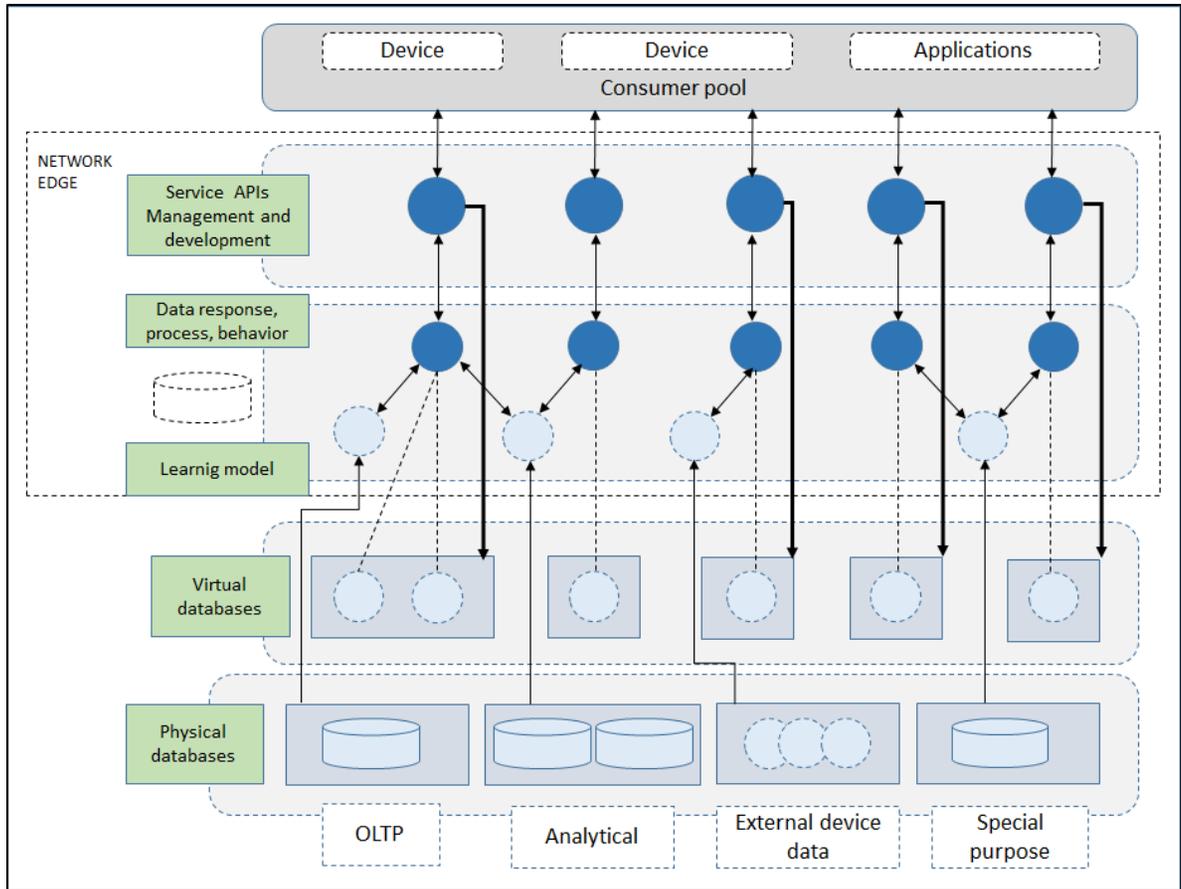


Figura 13. Arquitectura de datos para Internet de las cosas
Adaptado de [155]

- User layer: En esta capa, las APIs son empleadas para el diseño de aplicaciones, en el contexto de industria 4.0 son aplicaciones ERP preferiblemente, las cuales permiten el monitoreo de materias primas, fallas de equipos, control de calidad y programación e producción. Se proporcionan varios servicios, como la recolección de datos, la transmisión y el procesamiento de datos. Esta capa se basa en la tecnología Middleware, la cual es importante en la habilitación de servicios y aplicaciones IoT, permitiendo la reutilización de software y hardware.

6.2. Construcción de Metamodelo de integración

Se presenta la propuesta de creación de un sistema que genera aplicaciones para la industria 4.0 en diversas plataformas y/o sistemas operativos, que

permiten interconectar objetos de Internet de las Cosas a través de las diferentes redes incluidas las redes sociales.

Al definir un dispositivo se deberán definir ciertas propiedades de ese dispositivo y los objetos que contiene (sensores y actuadores). El dispositivo puede contar con reglas que le aporten cierto nivel de inteligencia en forma independiente a los objetos inteligentes.

El sistema se diseñó aplicando MDE, lo que conllevó una serie de pasos. Inicialmente se necesita definir el dominio del problema que se quiere resolver y que acota el campo de conocimientos. En esta propuesta, el dominio es la definición de dispositivos que se puedan conectar a las diferentes redes para publicar datos de sus sensores o para que sus actuadores puedan ser controlados por otros usuarios o dispositivos que hacen uso de reglas que permiten automatizar los procesos.

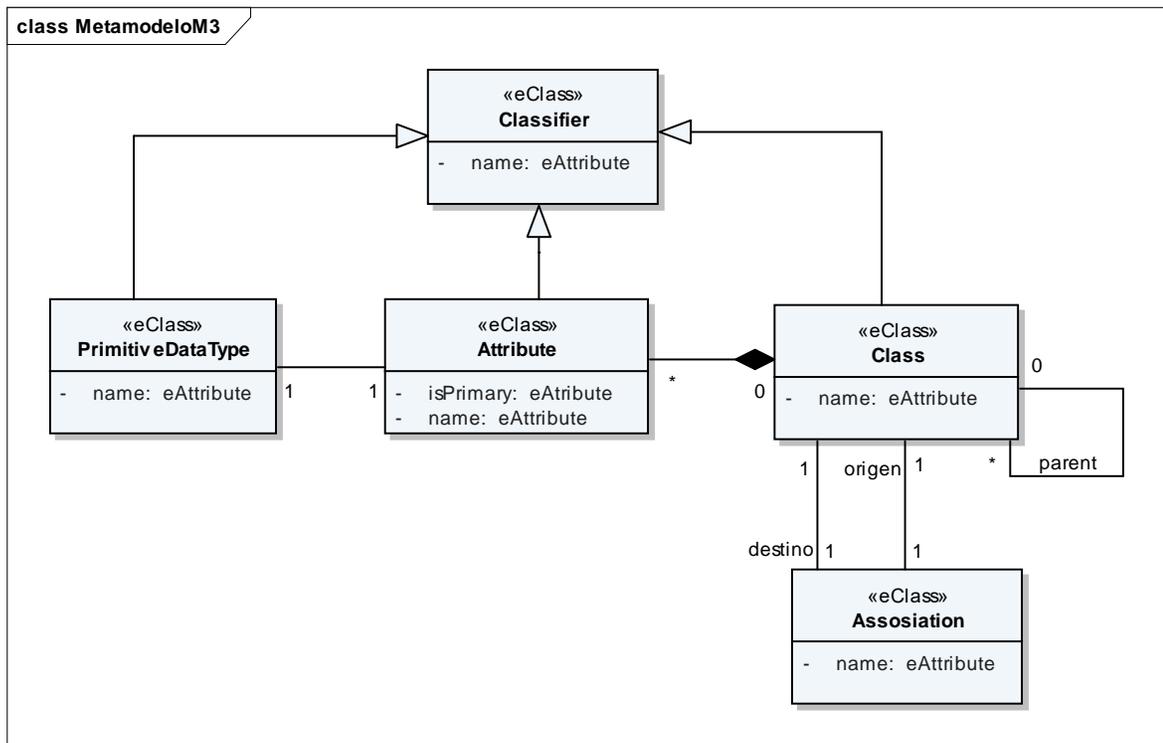
Una vez que se tiene acotado el dominio, se debe definir el metamodelo a partir de un meta-metamodelo, en este caso Ecore (de Eclipse Foundation).

MDE utiliza cuatro tipos de modelos (de mayor a menor nivel de abstracción): CIM (Computation Independent Model), PIM (Platform Independent Model), PSM (Platform Specific Model) y CODE (código fuente de la aplicación). La idea principal, es obtener mediante transformaciones automáticas, modelos más específicos a partir de otros más abstractos; es decir, de un PIM obtener uno o varios PSM (según la tecnología de implementación) y de un PSM, obtener el código fuente en una tecnología específica [1].

Una transformación de modelos es el proceso de convertir un modelo de un sistema en otro modelo del mismo sistema. En esencia, una transformación establece un conjunto de reglas que describen cómo un modelo expresado en un lenguaje origen puede ser transformado en un modelo en un lenguaje destino. Para realizar una transformación entre modelos se debe contar con información de los metamodelos que describen las representaciones de cada elemento del modelo y define las restricciones que debe cumplir un modelo [2].

Diseño de metamodelos para la capa de sensores

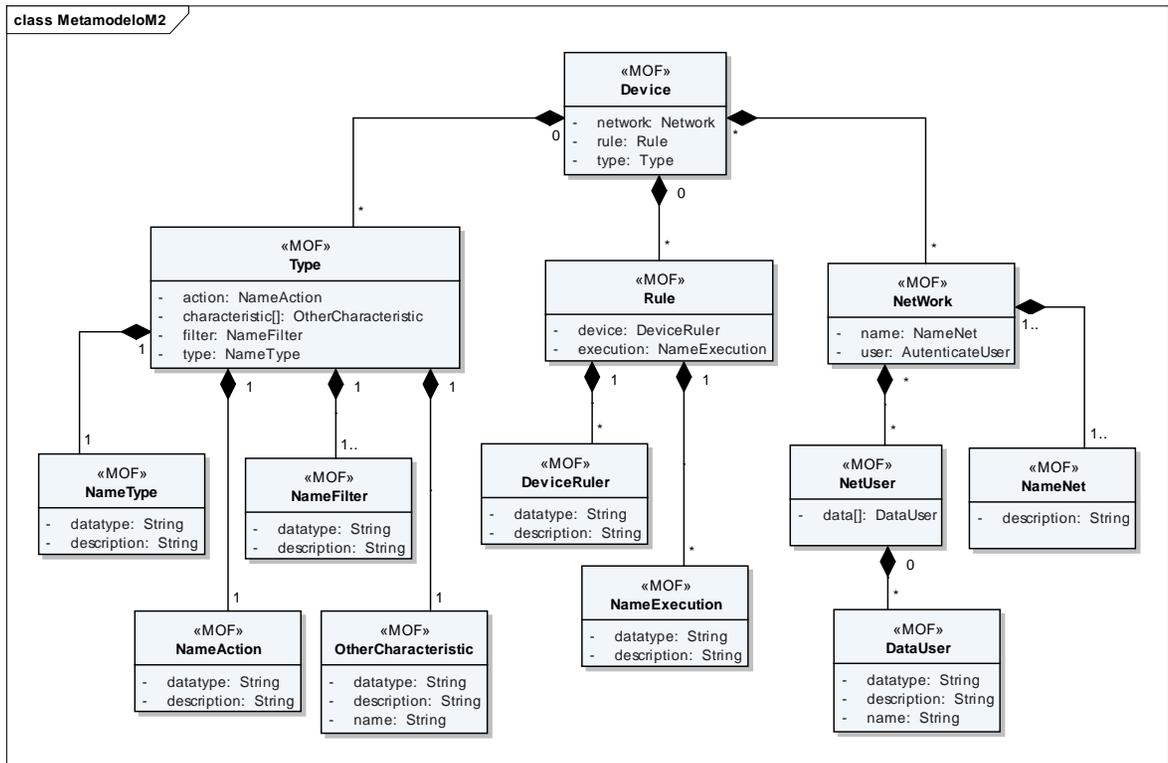
Metamodelo M3 para la capa de sensores



Definición de la transformación para el Metamodelo M3 de la capa de sensores

```
1 transformation classECoreToClassMOF(e:SimpleECore, m:SimpleMOF)
2 {
3     top relation ClassToClass {
4         checkonly domain e ce:Class {
5             name=cn
6         };
7         enforce domain m cm:Class {
8             name=cn
9         }
10    }
11
12    relation AttributeToAttribute
13    {
14        checkonly domain e ce:Class {};
15        enforce domain m cm:Class {};
16        primitive domain prefix:String;
17        where {
18            PrimitiveAttributeToAttribute(ce, cm, prefix);
19        }
20    }
21
22    function EPrimitiveTypeToMOFType(primitiveTpe:String):EString {
23        foreach(eAttribute in Class){
24            eAttributeType = 'MOFAttributeType'
25        }
26    }
27 }
```

Metamodelo M2 después de la transformación



Especificación de la transformación para M2 con QVT

```
1 transformation classMOFTtoClassUML(m:SimpleMOF, u:SimpleUML)
2 {
3     top relation ClassToClass {
4         checkonly domain m cm:Class {
5             name=cn
6         };
7         enforce domain u cu:Class {
8             name=cn
9         }
10    }
11
12    relation AttributeToAttribute
13    {
14        checkonly domain m cm:Class {};
15        enforce domain u cu:Class {};
16        primitive domain prefix:String;
17        where {
18            PrimitiveAttributeToAttribute(cm, cu, prefix);
19        }
20    }
21
22    function PrimitiveTypeToUMLType(primitiveTpe:String):EString {
23        if (primitiveType='INTEGER') then 'EInteger'
24        else if (primitiveType='BOOLEAN') then 'EBoolean'
25            else if (primitiveType='DOUBLE') then 'EDouble'
26                else then 'EFloat'
27            endif
28        endif
29    endif;
30 }
31 }
```

Metamodelo M1 de la capa de sensores después de la transformación

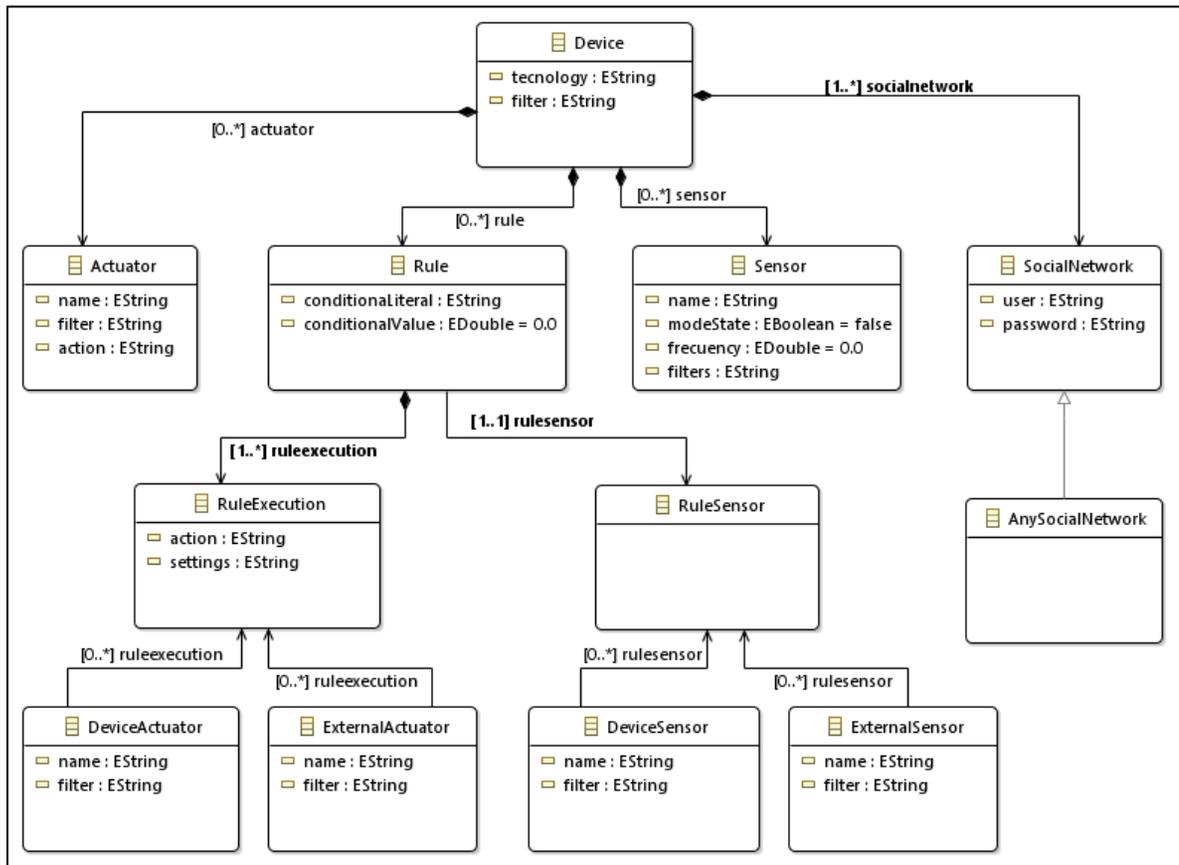


Figura 8. Metamodelo del sistema

- El dispositivo (*Device*): Es el componente principal y su definición es el objetivo del lenguaje. Sus propiedades son la tecnología (*technology*) que representa la plataforma del dispositivo, para el caso aplicado es una Raspberry Pi, plataforma desarrollada en uno de los lenguajes soportados por la Raspberry Pi; y los filtros (*filters*) que representan palabras claves necesarias para identificar los dispositivos en las redes sociales. El dispositivo está conformado por otros componentes: redes sociales, actuadores, sensores y reglas.
- Red Social (*SocialNetwork*): Un dispositivo podrá tener asociadas una o más redes sociales, pero al menos una. Las redes sociales son un componente abstracto que otro componente debe implementar, es decir,

un dispositivo puede tener redes sociales con distintas propiedades, pero todas son redes sociales.

- Actuador (*Actuator*): Un dispositivo podrá estar formado por actuadores además de otros componentes. Mediante este componente se define cada uno de los actuadores del dispositivo. Sus propiedades son el nombre (*name*) del actuador, las acciones (*actions*) que puede efectuar (al menos una es necesaria) y los filtros (*filters*) que se podrán usar para identificar al actuador además del nombre.
- Sensor (*Sensor*): Un dispositivo podrá estar formado por sensores además de otros componentes como actuadores. Mediante este componente se define cada uno de los sensores de los que dispone el dispositivo. Sus propiedades son el nombre (*name*) del sensor, su modo (*modeState*) de funcionamiento que podrá ser automático o manual, la frecuencia (*frequency*) de funcionamiento en caso de ser automático y los filtros (*filters*) que se podrán usar para identificar al sensor además del nombre.
- Regla (*Rule*): Además de componentes físicos como sensores o actuadores, un dispositivo podrá tener definidas reglas que automaticen las ejecuciones de las acciones de sus actuadores o la invocación de acciones de actuadores de otros dispositivos en función de datos de sus sensores o de otros sensores disponibles en la red social. Sus propiedades son el literal de la condición (*conditionLiteral*), por ejemplo «mayor que», y el valor que debe cumplir la condición (*conditionValue*). Además de esas propiedades, un sensor tiene una regla (*RuleSensor*) y una serie de ejecuciones que se realizarán si se cumple la condición (*RuleExecution*).
- Regla del sensor (*RuleSensor*): Un sensor deberá tener una regla del que tomar el valor que validará la condición para automatizar las ejecuciones de acciones. Este componente es un componente abstracto que implementan dos componentes diferentes ya que existen dos posibilidades al momento de seleccionar la regla del sensor: uno o varios sensores del

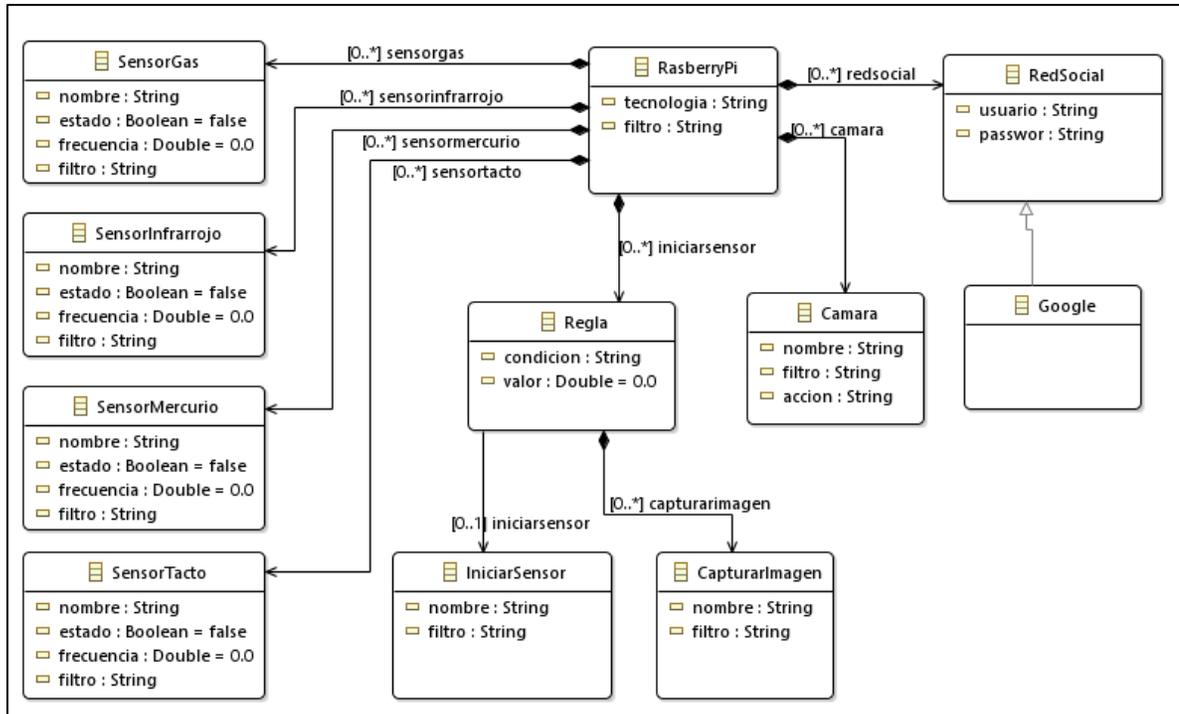
propio dispositivo (*DeviceSensors*), uno o varios sensores de dispositivos externos (*ExternalSensors*).

1. Uno o varios sensores del propio dispositivo (*DeviceSensors*): Las reglas se pueden usar para evaluar la condición, el valor de uno o varios sensores del mismo dispositivo siendo necesario su nombre (*name*) y que contengan los filtros (*filters*) indicados.
 2. Uno o varios sensores de un dispositivo externo (*ExternalSensor*): Las reglas se pueden usar para evaluar la condición, el valor de uno o varios sensores de un dispositivo externo indicando el nombre (*name*) del sensor y los filtros (*filters*) que identifican a los dispositivos y/o sensores.
- Ejecución de una regla (*RuleExecution*): Una regla puede tener varias ejecuciones que realizar si la condición es correcta, pero al menos deberá disponer de una ejecución que hacer. Este componente es un componente abstracto que implementa dos componentes diferentes ya que existen dos posibilidades a la hora de seleccionar el actuador que realizará la ejecución de una acción: uno o varios actuadores del propio dispositivo (*DeviceActuators*), uno o varios actuadores de dispositivos externos (*ExternalActuators*). Tiene una serie de propiedades que comparten todas las posibilidades. Estas son el nombre de la acción (*action*) a realizar y los parámetros (*settings*) que utilizará la acción.
 1. Uno o varios actuadores del propio dispositivo (*DeviceActuators*): Las reglas pueden realizar una acción de uno o varios actuadores del mismo dispositivo siendo necesario su nombre (*name*) y que contengan los filtros (*filters*) indicados.
 2. Uno o varios actuadores de un dispositivo externo (*ExternalActuators*): Las reglas pueden realizar una acción de un actuador de un dispositivo externo indicando el nombre (*name*) de los actuador y los filtros (*filters*) que identifican a los dispositivos y/o actuadores.

Definición de la transformación para el metamodelo M1 de capa de sensores

```
1 transformation UMLmodelToJavamodel(uml:SimpleUML, java:SimpleJava)
2 {
3     top relation ClassUMLToClassJava {
4         cn, prefix: String;
5         checkonly domain u cu:Class {
6             kind='Class', name=cn;
7         enforce domain java cj:Class {name=cn,
8             attribute=at:Attribute {name=cn+'_tid', type='NUMBER'},
9         };
10        where {
11            prefix = '';
12            AttributeToAttribute(cu, cj, prefix);
13        }
14    }
15
16    relation AttributeUMLToAttributeJava
17    {
18        checkonly domain uml cu:Class {};
19        enforce domain java cj:Class {};
20        primitive domain prefix:String;
21        where {
22            PrimitiveAttributeUMLToAttributeJava(cu, cj, prefix);
23        }
24    }
25
26    function PrimitiveTypeUMLToPrimitiveTypeJava(primitiveTpe:ESString):String {
27        if (primitiveType='EInteger') then 'int'
28        else if (primitiveType='EBoolean') then 'boolean'
29            else if (primitiveType='EDouble') then 'double'
30            else 'float'
31        endif;
32    endif;
33    endif;
34 }
```

Modelo M0 después de la transformación



6.3. Prototipo

En el marco de la industria 4.0, el prototipo desarrollado es aplicable para realizar la automatización de procesos manuales dentro de las diferentes actividades productivas en cualquier tipo de industria, mejorando la disponibilidad de la información y su seguridad, facilitando el flujo la información y mejorando los procesos de decisión sobre la aplicación de acciones correctivas y preventivas. Al ser un sistema intermitente (la activación del sistema depende del cambio en la variable de medición) la recopilación de información poco relevante será casi nula y la presentación de la información crítica es presentada en forma organizada y sistemática mejorando de esta forma la minería de datos.

Las ventajas de la aplicación de esta clase de prototipos dentro de procesos productivos o de seguimiento de productos [156] corresponden a la disminución de la incertidumbre y los errores de medición, además tiene como ventaja sobre la medición manual la posibilidad de implementar un proceso de medición interrumpido creando procesos de trazabilidad de datos e información

que no existían anteriormente (sin necesidad de la intervención del recurso humano) dado que la información capturada continuamente es empujada al Big Data o nube informática para su consolidación.

El prototipo desarrollado es una extensión de los sistemas ciber físicos (CPS), los cuales corresponden a una aplicación práctica de la internet de las cosas dentro de la industria 4.0 [157].

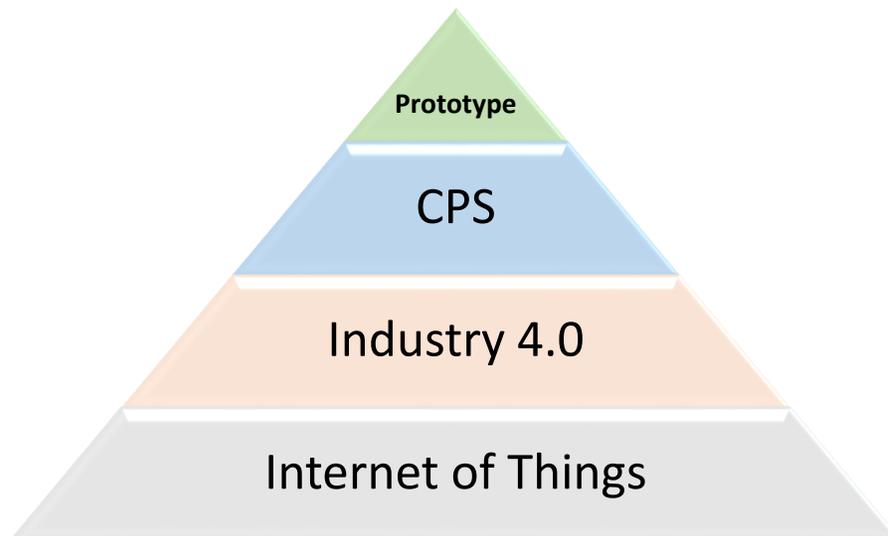


Figura 14. Ubicación del prototipo dentro de la industria 4.0 y la internet de las cosas.

Con base en los conceptos de computación ubicua y comunicación M2M e implementado bajo el marco de la Internet de las cosas, el objetivo de este prototipo es capturar datos y permitir el acceso a estos en tiempo real y a distancia. Se estableció una comunicación entorno-objeto y objeto-objeto para lograr el objetivo propuesto.

El entorno de comunicación-objeto del sistema de vigilancia permitió monitorear diferentes variables, para ello se utilizaron cuatro sensores que permitieron determinar el flujo de productos en una planta de producción, la presencia de gas en el recinto, la alineación y estabilidad del equipo y existencia de contacto entre una persona y el prototipo. La lectura de los cambios en el estado de los sensores realizados se procesó en un micro ordenador Raspberry Pi.

La comunicación objeto-objeto entre el Raspberry Pi y un dispositivo móvil, con el fin de establecer la sincronización de datos directamente al usuario a través de su Smartphone o Tablet, se logró a través de la conexión establecida entre el Raspberry Pi y el servidor de almacenamiento en la nube.

El prototipo tiene las siguientes características:

- Uso de tecnologías de sensores: El prototipo utiliza diferentes sensores para la detección de elementos "extraños" en el entorno de trabajo e inicia la captura y transmisión de datos. El sensor actúa como un enlace entre lo que ocurre en el proceso y el usuario.
- Recopilación de datos en cualquier momento: el prototipo no restringe la captura de imágenes y siempre es consciente de cada movimiento presente en el lugar de instalación del sistema.
- Transferencia de datos: es posible transferir los datos dentro de la red de comunicación compuesta por el usuario (a través de un dispositivo móvil) y el sistema físico (Raspberry Pi) mediante computación en la nube.
- Interfaz de usuario: el prototipo puede comunicarse con el usuario a través de plataformas conocidas o de almacenamiento en servidores.
- Independencia funcional: el prototipo funciona automática e independientemente del usuario, ya que la captura de datos y sus funciones de transferencia se llevan a cabo sólo con la activación de un sensor y una red inalámbrica.
- Comunicación con el dispositivo móvil: el sistema no requiere intervención directa en el hardware o software para operar. La comunicación se realizada bajo los principios de comunicación máquina-máquina en la que el micro-ordenador Raspberry Pi y los sensores representan el prototipo y el dispositivo móvil es el terminal que permite la comunicación con el usuario a través de un servidor y una red de comunicación. Los datos capturados se derivan de una máquina a otra de la misma manera, independiente a que el sensor esté en uso.

- Disposición total de datos: el sistema genera una red de comunicación que permite el acceso al usuario en cualquier lugar y consulta en cualquier momento datos, debido a la conexión del prototipo físico (Raspberry Pi) con el servidor en línea .

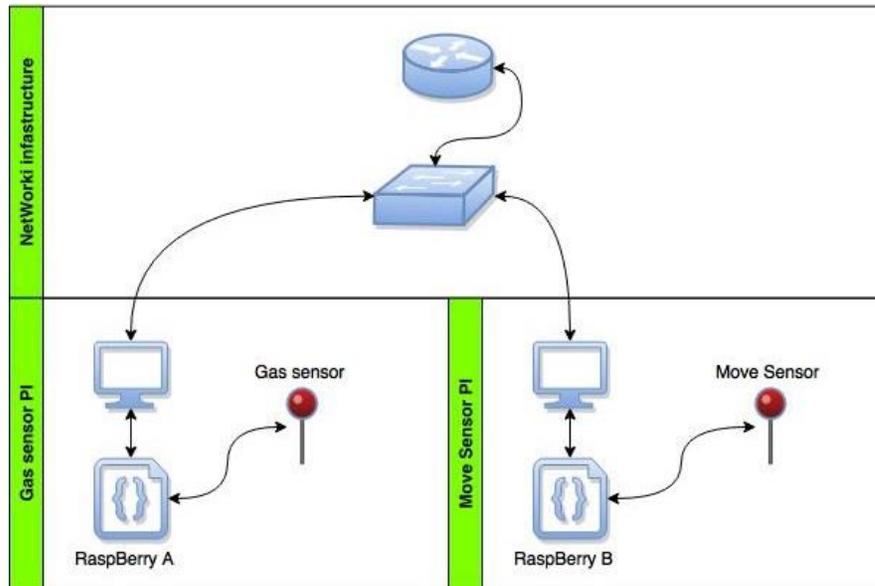


Figura 15. Diagrama de Conexión del prototipo

Si bien se determinó que se emplearía una Raspberry Pi por tener una amplia gama de aplicaciones generalmente orientadas al uso del software y con varios lenguajes de programación, se efectuaron pruebas preliminares de carácter funcional y de conexión con el microcontrolador Arduino que tiene como ventaja el no tener sistema operativo y ser compatible con una gran carga eléctrica, además de ser ideal para proyectos robóticos o de automatización. Las pruebas permitieron evidenciar un mejor desempeño con la integración propuesta y las funciones programadas a la Raspberry Pi.

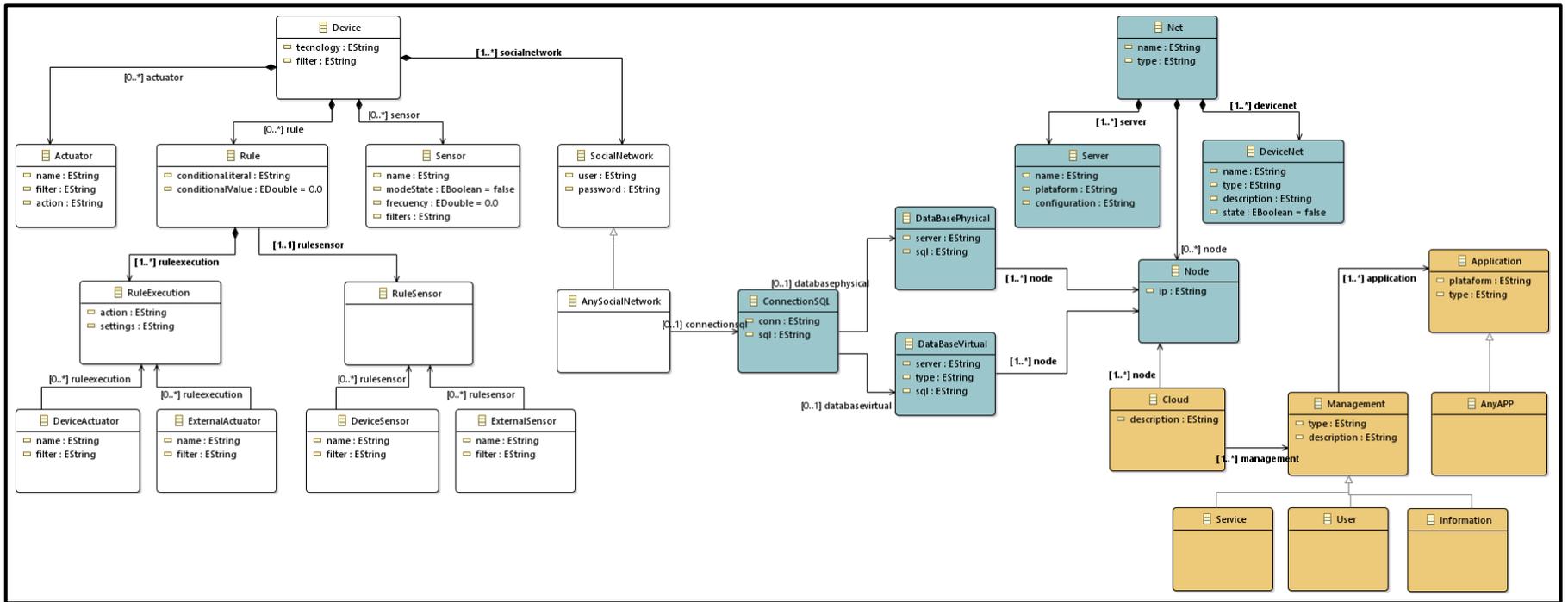


Figura 16. Metamodelo de integración de internet de las cosas, las redes sociales y la industria 4.0

CAPÍTULO 7 - VALIDACIÓN Y PRUEBAS

Con el propósito de validar el cumplimiento del objetivo de la solución propuesta se efectuaron pruebas al sistema de monitoreo con dos sensores, realizando la conexión individual de cada sensor a la Raspberry Pi en cada prueba. Las pruebas se llevaron a cabo en una línea de producción de cerámicas en un periodo de 24 horas de trabajo.

Para validar la funcionalidad del sistema se evaluaron en paralelo los dispositivos, en primera instancia los dispositivos con los que contaba la línea de producción y que generaron reporte de falla, deteniendo el proceso. En segunda instancia el prototipo al funcionar en paralelo registraba de falla y generaba el reporte sin detener el proceso.

7.1. Paradas por fallas en el flujo de producción

Los datos permitieron verificar las condiciones reales de las paradas en la línea de producción, algunas paradas registradas no fueron fallas reales, el sistema las registro al efectuarse un cambio en la velocidad del proceso. Los datos generados por el prototipo también registraron paradas de rutina y verificación calidad lo que conllevó a un ajuste en los parámetros de decisión.

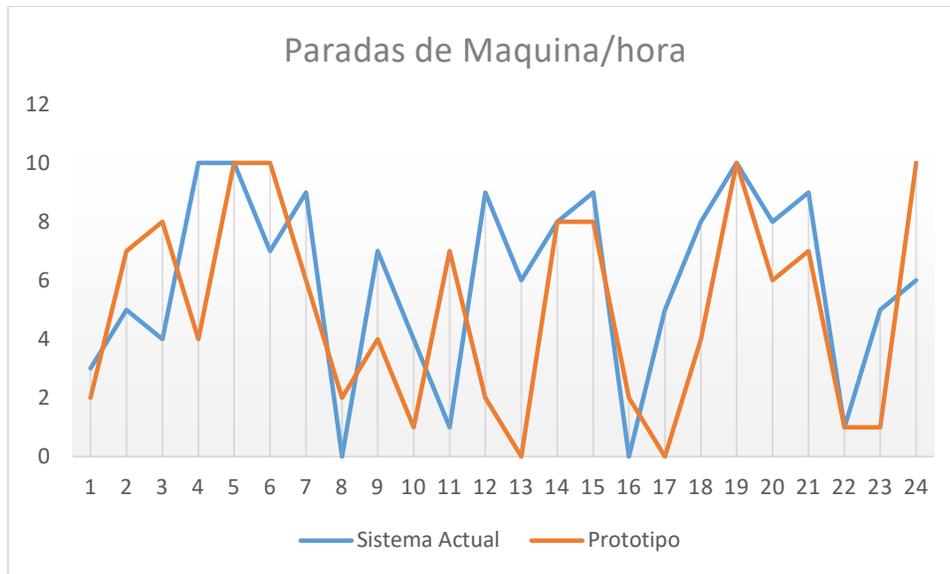


Figura 17. Paradas por fallas en el flujo de producción

7.2. Paradas por fallas alineación y estabilidad del equipo

El equipo actual al no contar con este tipo de información permite que se generen fallas en el proceso y afectación al producto, igualmente afecta al equipo mismo, dado que posteriormente se genera una falla mayor. La información suministrada por el prototipo permitió verificar y ajustar el equipo sin generar detección del proceso.

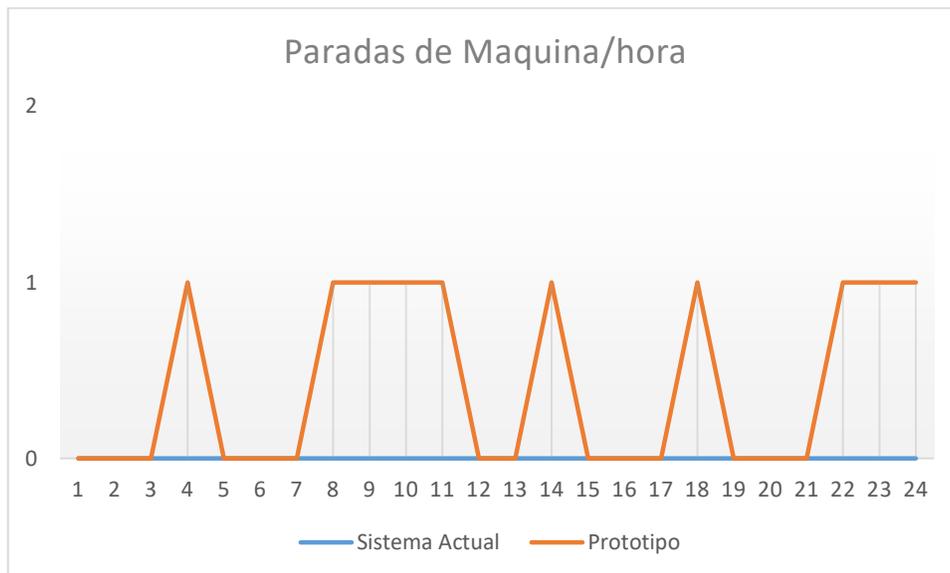


Figura 18. Paradas por fallas alineación y estabilidad del equipo

Se llevó a cabo la prueba del sistema de monitoreo con el sensor de detección de gas y dado que en el medio en el que se realizó no hay presencia de gases LP, metano o humo, las señales enviadas por el sensor a la Raspberry Pi no activaron el sistema de captura ni sincronización.

CAPÍTULO 8 - RESULTADOS

Se efectuaron pruebas al sistema de monitoreo con dos sensores, realizando la conexión individual de cada sensor a la Raspberry Pi en cada prueba. Las pruebas se llevaron a cabo en una línea de producción de cerámicas en un periodo de 24 horas de trabajo.

Para validar la funcionalidad del sistema se evaluaron en paralelo los dispositivos, en primera instancia los dispositivos con los que contaba la línea de producción y que generaron reporte de falla, deteniendo el proceso. En segunda instancia el prototipo al funcionar en paralelo registraba de falla y generaba el reporte sin detener el proceso.

8.1. Paradas por fallas en el flujo de producción

Los datos permitieron verificar las condiciones reales de las paradas en la línea de producción, algunas paradas registradas no fueron fallas reales, el sistema las registro al efectuarse un cambio en la velocidad del proceso. Los datos generados por el prototipo también registraron paradas de rutina y verificación calidad lo que conllevó a un ajuste en los parámetros de decisión.

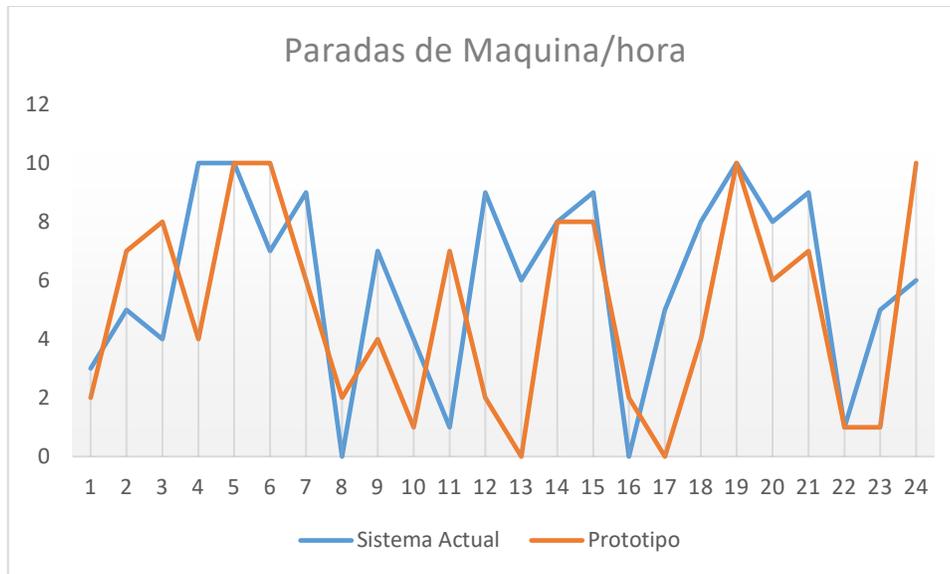


Figura 19. Paradas por fallas en el flujo de producción

8.2. Paradas por fallas alineación y estabilidad del equipo

El equipo actual al no contar con este tipo de información permite que se generen fallas en el proceso y afectación al producto, igualmente afecta al equipo mismo, dado que posteriormente se genera una falla mayor. La información suministrada por el prototipo permitió verificar y ajustar el equipo sin generar detección del proceso.

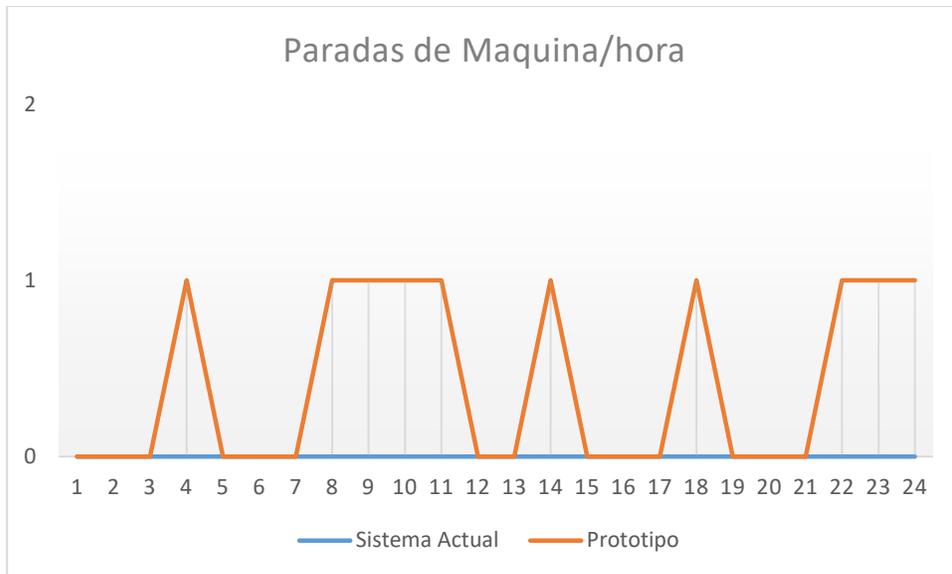


Figura 20. Paradas por fallas alineación y estabilidad del equipo

Se llevó a cabo la prueba del sistema de monitoreo con el sensor de detección de gas y dado que en el medio en el que se realizó no hay presencia de gases LP, metano o humo, las señales enviadas por el sensor a la Raspberry Pi no activaron el sistema de captura ni sincronización.

PARTE IV. CONCLUSIONES

CAPÍTULO 9 - CONCLUSIONES Y TRABAJO FUTURO

La arquitectura y el metamodelo de integración propuestos pretenden contribuir al desarrollo de aplicaciones en el marco de la internet de las cosas. La tecnología en sí misma no es el principal obstáculo para Aplicación de IIoT; Son las interfaces de interoperabilidad de los sistemas pertenecientes a diferentes proveedores que dificultan la adopción. Para lograr interoperabilidad se requiere una estandarización de interfaces de interacción entre los diferentes componentes del sistema.

IIoT es la combinación de tecnologías informáticas, de comunicación y microelectrónica, por lo que existen muchas interfaces entre hardware, software y componentes de red. La arquitectura propuesta en el contexto de la industria 4.0 establece redes de comunicación entre objetos y personas que permiten el flujo de información bidireccional con una intervención mínima en la red. Las redes de comunicación se centran en la transmisión directa y global de datos.

El cambio en la forma de recolectar y almacenar los datos, así como el uso de sensores requiere un cambio en la tecnología. IIOT puede registrar varios parámetros del proceso de producción de forma automática, precisa y oportuna. La producción industrial tradicional realiza la comunicación entre máquinas a través de la tecnología M2M, pero el IIOT puede lograr conexiones entre personas, máquinas y objetos físicos. Sin embargo, en el entorno de IIOT, la función y el rendimiento de los dispositivos de comunicación son diferentes. Algunas aplicaciones requieren un alto rendimiento en tiempo real, mientras que otras no, mientras que algunas tareas de aplicación se realizan periódicamente y otras son activadas por eventos. Estas características aumentan la dificultad de la aplicación práctica del IIOT.

Los trabajos futuros deben enmarcarse en:

- Seguridad de los datos y fiabilidad del sistema
- Estandarización de la Tecnología e interoperabilidad de los sistemas
- Implementación

Al inicio del documento se plantearon un grupo de objetivos, los cuales están estructurados con el fin de validar la hipótesis planteada, los mismos fueron contrastados evidenciando que han sido cumplidos durante el desarrollo de esta tesis doctoral.

9.1. Verificación y contraste de la Hipótesis

La tesis doctoral planteó como hipótesis de partida, el siguiente enunciado **¿Es posible interconectar objetos heterogéneos a través de un metamodelo que integre los beneficios del internet de las cosas, la nube, los sensores, las redes sociales y la gestión de la información de Big Data?**

Como consecuencia de los resultados alcanzados en la investigación se puede inferir que la misma ha sido validada como positiva.

9.2. Verificación, contraste y evaluación de Objetivos

- ***Elaborar el estado de Arte de Internet de las cosas, la nube, los sensores y las redes sociales.***

Se efectuó un ejercicio de revisión de publicaciones relacionadas con el internet de las cosas, las redes sociales, el internet industrial de las cosas, la nube, los sensores y actuadores.

- ***Especificar las herramientas tecnológicas aplicables y los requerimientos para la propuesta de un metamodelo de integración.***

A partir de los resultados del en el primer objetivo se logró determinar el tipo de microcontrolador, el kit de sensores y de actuadores, así como el software sobre el cual se efectuaría el desarrollo.

- ***Identificar los Smart Objects y comprobar la interoperabilidad con las plataformas web y los servicios Internet of Things disponibles para el diseño y desarrollo del metamodelo***

Autores: Paula Catalina Jaraba Navas, Yesid Camilo Guacaneme, José Ignacio Rodríguez Molano

Revista Lecture Notes in Computer Science. Volumen 9714, 2016, pp 449 - 456

ISSN 0302-9743

DOI 10.1007/978-3-319-40973-3

Sitio Web de Publicación: http://link.springer.com/chapter/10.1007/978-3-319-40973-3_45

Artículo: Big Data Meaning in the Architecture of IoT for Smart Cities

Autores: Christian David Gómez, July Katherine Díaz Barriga, José Ignacio Rodríguez Molano

Revista Lecture Notes in Computer Science. Volumen 9714, 2016, pp 457 - 465

ISSN 0302-9743

DOI 10.1007/978-3-319-40973-3

Sitio Web de Publicación: http://link.springer.com/chapter/10.1007/978-3-319-40973-3_46

Artículo: Industrial Internet of Things: An Architecture Prototype for Monitoring in Confined Spaces Using a Raspberry Pi

Autores: José Ignacio Rodríguez Molano, Víctor Hugo Medina, Javier Felipe Moncada Sánchez

Revista Lecture Notes in Computer Science. Volumen 9714, 2016, pp 521-528

ISSN 0302-9743

DOI 10.1007/978-3-319-40973-3

Sitio Web de Publicación http://link.springer.com/chapter/10.1007/978-3-319-40973-3_53

Artículo: Proposal of a Standard Architecture of IoT for Smart Cities

Autores: July Katherine Díaz Barriga, Christian David Gómez, José Ignacio Rodríguez Molano

Revista: Communications in Computer and Information Science. Volumen 620, 2016, pp 77 – 89

ISSN 1865-0929

DOI 10.1007/978-3-319-42147-6

Sitio Web de Publicación: http://link.springer.com/chapter/10.1007/978-3-319-42147-6_7

Artículo: A Proposed Architecture for the Development of a Cluster for Graduates of Higher Education Institutions

Autores: Alejandra Peña Mosquera, Katherin Flórez Vargas, José Ignacio Rodríguez Molano

Revista: Communications in Computer and Information Science. Volumen 657, 2016, pp 392-402

ISSN 1865-0929

DOI 10.1007/978-3-319-50880-1

Sitio Web de Publicación: http://link.springer.com/chapter/10.1007/978-3-319-50880-1_34

Artículo: Introducción al Internet de las Cosas

Autores: José Ignacio Rodríguez Molano, Carlos Enrique Montenregro, Juan Manuel Cueva Lovelle

Revista: Redes de Ingeniería. Volumen 6, 2015, pp 392-402

ISSN 2248-762X

Sitio Web de Publicación: <http://revistas.udistrital.edu.co/ojs/index.php/REDES/issue/view/708/showToc>

9.3.4 Congresos

Se publicó y presentó el paper titulado: “Big Data Meaning in the Architecture of IoT for Smart Cities” en Data Mining and Big Data, First International Conference, DMBD 2016, Bali, Indonesia, June 25-30, 2016. Proceedings. Lecture Notes in Computer Science 9714, Springer 2016, ISBN 978-3-319-40972-6

Se publicó y presentó el paper titulado: "Big Data Tools: Hadoop, MongoDB and Weka" en Data Mining and Big Data, First International Conference, DMBD 2016, Bali, Indonesia, June 25-30, 2016. Proceedings. Lecture Notes in Computer Science 9714, Springer 2016, ISBN 978-3-319-40972-6

Se publicó y presentó el paper titulado: "Industrial Internet of Things: An Architecture Prototype for Monitoring in Confined Spaces Using a Raspberry Pi" en Data Mining and Big Data, First International Conference, DMBD 2016, Bali, Indonesia, June 25-30, 2016. Proceedings. Lecture Notes in Computer Science 9714, Springer 2016, ISBN 978-3-319-40972-6

Se publicó y presentó el paper titulado: "Proposal of a Standard Architecture of IoT for Smart Cities." en Learning Technology for Education in Cloud - The Changing Face of Education - 5th International Workshop, LTEC 2016, Hagen, Germany, July 25-28, 2016, Proceedings. Communications in Computer and Information Science 620, Springer 2016, ISBN 978-3-319-42146-9

Se publicó y presentó el paper titulado "Internet of Things: A Prototype Architecture Using a Raspberry Pi." en Knowledge Management in Organizations - 10th International Conference, KMO 2015, Maribor, Slovenia, August 24-28, 2015, Proceedings. Lecture Notes in Business Information Processing 224, Springer 2015, ISBN 978-3-319-21008-7

9.4 Líneas de Investigación Futuras

El desarrollo de la tesis motiva a continuar el trabajo en los siguientes campos de investigación como líneas futuras:

- Seguridad de los datos y fiabilidad del sistema

Los protocolos de comunicación no cuentan con niveles óptimos de encriptación que permitan guardar la confidencialidad de los datos, la heterogeneidad de los sistemas los hace vulnerables y la calidad de conexión reduce la fiabilidad de los mismos. Es necesario establecer protocolos de seguridad y protección de acceso a la información

- Estandarización de la Tecnología e interoperabilidad de los sistemas

El amplio desarrollo de aplicaciones deberá establecer estándares y protocolos de comunicación que permitan la integración de diferentes objetos a través de diferentes plataformas independiente del hardware o software implementado.

- Implementación práctica

Establecer elementos de integración con tecnologías existentes en los procesos automatizados, incorporando la compatibilidad con los PLCs, y los diferentes actuadores. La implementación de aplicaciones en el ámbito industrial debe efectuarse en un menor tiempo.

REFERENCIAS

- [1] V. García-Díaz, “MDCI: Model-driven continuous integration,” UNIVERSIDAD DE OVIEDO, 2012.
- [2] L. Tan, “Future internet: The Internet of Things,” *2010 3rd Int. Conf. Adv. Comput. Theory Eng.*, pp. V5-376-V5-380, Aug. 2010.
- [3] The US National Intelligence Council, “Disruptive Civil Technologies :Six Technologies with Potential Impacts on US Interests out to 2025,” 2008.
- [4] CERP-IoT, *Internet of Things: Strategic Research Roadmap*. European Commission - Information Society and Media DG, 2009.
- [5] B. Alessandro, Hitachi Europe, and G. Horn, “Internet of Things in 2020: A ROADMAP FOR THE FUTURE,” 2008.
- [6] W. CHONGGANG, D. MAHMOUD, D. MISCHA, H. R. QINGYANG, M. XUFEI, and W. HONGGANG, “Guest Editorial Special Issue on Internet of Things (IoT): Architecture , Protocols and Services,” *IEEE Sens. J.*, vol. 13, no. 10, pp. 3505–3510, 2013.
- [7] A. R. Biswas and R. Giaffreda, “IoT and Cloud Convergence : Opportunities and Challenges,” *2014 IEEE World Forum Internet Things*, pp. 375–376, 2014.
- [8] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A vision, architectural elements, and future directions,” *Futur. Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.
- [9] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.
- [10] T. K. L. Hui and R. S. Sherratt, “Towards disappearing user

interfaces for ubiquitous computing: human enhancement from sixth sense to super senses,” *J. Ambient Intell. Humaniz. Comput.*, pp. 1–17, 2016.

- [11] L. Hao, X. Lei, Z. Yan, and Y. ChunLi, “The application and implementation research of smart city in China,” *2012 Int. Conf. Syst. Sci. Eng.*, no. 70172014, pp. 288–292, 2012.
- [12] González García Cristian, “MIDGAR: Plataforma para la generación dinámica de aplicaciones distribuidas basadas en la integración de redes de sensores y dispositivos electrónicos IoT,” UNIVERSIDAD DE OVIEDO, 2013.
- [13] L. Atzori, A. Iera, G. Morabito, and M. Nitti, “The Social Internet of Things (SIoT) – When social networks meet the Internet of Things: Concept, architecture and network characterization,” *Comput. Networks*, vol. 56, no. 16, pp. 3594–3608, Nov. 2012.
- [14] C. González García, B. C. Pelayo G-Bustelo, J. Pascual Espada, and G. Cueva-Fernandez, “Midgar: Generation of heterogeneous objects interconnecting applications. A Domain Specific Language proposal for Internet of Things scenarios,” *Comput. Networks*, vol. 64, pp. 143–158, May 2014.
- [15] R. Roman, J. Zhou, and J. Lopez, “On the features and challenges of security and privacy in distributed internet of things,” *Comput. Networks*, vol. 57, no. 10, pp. 2266–2279, Jul. 2013.
- [16] J. Pascual Espada, O. Sanjuán Martínez, B. C. Pelayo G-Bustelo, and J. M. Cueva Lovelle, “Virtual Objects on the Internet of Things,” *Int. J. Interact. Multimed. Artif. Intell.*, vol. 1, no. 4, p. 23, 2011.
- [17] B. Xu, L. Da Xu, H. Cai, C. Xie, J. Hu, and F. Bu, “Ubiquitous Data Accessing Method in IoT-based Information System for Emergency Medical Services,” *IEEE Trans. Ind. Informatics*, vol. 3203, no. c, pp. 1–1, 2014.

- [18] K. Gama, L. Touseau, and D. Donsez, "Combining heterogeneous service technologies for building an Internet of Things middleware," *Comput. Commun.*, vol. 35, no. 4, pp. 405–417, Feb. 2012.
- [19] W. Steiner and S. Poledna, "Fog computing as enabler for the Industrial Internet of Things," *e i Elektrotechnik und Informationstechnik*, pp. 310–314, 2016.
- [20] K. Ashton, "That 'Internet of Things' Thing - RFID Journal," *RFID Journal*, 2009. .
- [21] H. G. H. Gu and D. W. D. Wang, "A Content-aware Fridge based on RFID in smart home for home-healthcare," *2009 11th Int. Conf. Adv. Commun. Technol.*, vol. 2, pp. 987–990, 2009.
- [22] E. Mykletun, J. Girao, and D. Westhoff, "Public key based cryptoschemes for data concealment in wireless sensor networks," in *IEEE International Conference on Communications*, 2006, vol. 5, pp. 2288–2295.
- [23] C. Francois, S. Krco, and B. Pokrie, "Designing IoT Architecture (s) A European Perspective," *2014 IEEE World Forum Internet Things*, pp. 79–84, 2014.
- [24] A. Serbanati, C. M. Medaglia, and U. B. Ceipidor, "Building Blocks of the Internet of Things : State of the Art and Beyond," in *deploying-rfid-challenges-solutions-and-open-issues*, no. Challenges, Solutions, and Open Issues, D. C. Turcu, Ed. InTech, 2011.
- [25] CERP-IoT, *Vision and Challenges for Realising the Internet of Things*, no. March. European Commission - Information Society and Media DG, 2010.
- [26] Pascual Espada Jordán, "DISEÑO DE OBJETOS VIRTUALES COLABORATIVOS ORIENTADOS A SERVICIOS EN EL MARCO DE INTERNET DE LAS COSAS," Universidad de oviedo, 2012.

- [27] A. U. N. S. N, C. Sarkar, R. V. Prasad, and A. Rahim, "A Unified Semantic Knowledge Base for IoT," pp. 575–580, 2014.
- [28] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Futur. Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.
- [29] International Telecommunication Union (ITU), "The Internet of Things," 2005.
- [30] M.-W. Ryu, "Survey on Internet of Things: Toward Case Study," *Smart Comput. Rev.*, vol. 2, no. 3, pp. 195–202, Jun. 2012.
- [31] M. Nitti, L. Atzori, and I. P. Cvijikj, "Network Navigability in the Social Internet of Things," pp. 405–410, 2014.
- [32] F. Mattern and C. Floerkemeier, "From the Internet of Computers to the Internet of Things," *Informatik-Spektrum*, vol. 33, no. 2, pp. 107–121, 2010.
- [33] Prajakta Pande and P. Anand, "Internet of Things – A Future of Internet : A Survey," *Int. J. Adv. Res. Comput. Sci. Manag. Stud.*, vol. 2, no. 2, pp. 354–361, 2014.
- [34] The Government Office for Science, "The IOT: making the most of the Second Digital Revolution," *WordLink*, pp. 1–40, 2014.
- [35] K. a Hribernik, Z. Ghrairi, C. Hans, and K. Thoben, "Co-creating the Internet of Things - First Experiences in the Participatory Design of Intelligent Products with Arduino," *Concurr. Enterprising (ICE)*, 2011 *17th Int. Conf.*, no. Ice, pp. 1–9, 2011.
- [36] V. Georgitzikis, O. Akribopoulos, and I. Chatzigiannakis, "Controlling physical objects via the internet using the arduino platform over 802.15.4 networks," *IEEE Lat. Am. Trans.*, vol. 10, no. 3, pp. 1686–1689, 2012.

- [37] V. Vujović and M. Maksimović, “Raspberry Pi as a Sensor Web node for home automation,” *Comput. Electr. Eng.*, vol. 44, pp. 153–171, 2014.
- [38] V. Vujović and M. Maksimović, “Raspberry Pi as a Wireless Sensor node: Performances and constraints,” in *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2014 - Proceedings*, 2014, pp. 1013–1018.
- [39] S. Luo, H. Xia, Y. Gao, J. S. Jin, and R. Athauda, “Smart fridges with multimedia capability for better nutrition and health,” in *Proceedings - 2008 International Symposium on Ubiquitous Multimedia Computing, UMC 2008*, 2008, pp. 39–44.
- [40] T. Yamanoue, K. Oda, and K. Shimosono, “A M2M system using arduino, android and wiki software,” in *Proceedings of the 2012 IIAI International Conference on Advanced Applied Informatics, IIAIAI 2012*, 2012, pp. 123–128.
- [41] A. Laya, L. Alonso, and J. Alonso-Zarate, “Is the random access channel of LTE and LTE-A suitable for M2M communications? A survey of alternatives,” *IEEE Commun. Surv. Tutorials*, vol. 16, no. 1, pp. 4–16, 2014.
- [42] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–105, 2002.
- [43] L. Wang, L. Da Xu, Z. Bi, and Y. Xu, “Data cleaning for RFID and WSN integration,” *IEEE Trans. Ind. Informatics*, vol. 10, no. 1, pp. 408–418, 2014.
- [44] T. Zahariadis, F. Alvarez, and J. P. Moore Olmstead, “An architectural approach towards Future Media Internet,” *Multimed. Tools Appl.*, vol. 70, no. 1, pp. 297–309, 2014.

- [45] C. Perera, C. H. Liu, and S. Jayawardena, "The Emerging Internet of Things Marketplace from an Industrial Perspective: A Survey," *IEEE Trans. Emerg. Top. Comput.*, vol. 3, no. 4, pp. 585–598, 2015.
- [46] K. Di Chang and J. L. Chen, "A survey of trust management in WSNs, internet of things and future internet," *KSII Trans. Internet Inf. Syst.*, vol. 6, no. 1, pp. 5–23, 2012.
- [47] International Telecommunication Union, "Overview of the Internet of things," *Ser. Y Glob. Inf. infrastructure, internet Protoc. Asp. next-generation networks - Fram. Funct. Archit. Model.*, p. 22, 2012.
- [48] M. Hasan, E. Hossain, and D. Niyato, "Random access for machine-to-machine communication in LTE-advanced networks: Issues and approaches," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 86–93, 2013.
- [49] E. Borgia, "The internet of things vision: Key features, applications and open issues," *Computer Communications*, vol. 54, pp. 1–31, 2014.
- [50] G. M. Lee and J. Y. Kim, "Ubiquitous networking application: Energy saving using smart objects in a home," in *International Conference on ICT Convergence*, 2012, pp. 299–300.
- [51] T. Yashiro, S. Kobayashi, N. Koshizuka, and K. Sakamura, "An Internet of Things (IoT) architecture for embedded appliances," *2013 IEEE Reg. 10 Humanit. Technol. Conf.*, pp. 314–319, Aug. 2013.
- [52] X. Jia, Q. Feng, T. Fan, and Q. Lei, "RFID technology and its applications in Internet of Things (IoT)," *2012 2nd Int. Conf. Consum. Electron. Commun. Networks*, pp. 1282–1285, 2012.
- [53] L. A. D. Giusto, A. Iera, G. Morabito, *The Internet of Things*. 2010.
- [54] R. Weinstein, "RFID : A Technical Overview and Its Application to TECHNOLOGY," *IEEE J. Mag.*, vol. 7, no. 3, 2005.
- [55] P. Fuchsíková, V. Kebo, and S. Pavel, "RFID technology for special food moulds," *IEEE Conf. Publitations*, pp. 183–188, 2012.

- [56] C. Sun, "Application of RFID Technology for Logistics on Internet of Things," *AASRI Procedia*, vol. 1, pp. 106–111, 2012.
- [57] G. Chavira, S. W. Nava, R. Hervás, J. Bravo, and C. Sanchez, "Combining RFID and NFC Technologies in an AmI Conference Scenario," *IEEE Conf. Publications*, pp. 165–172, Sep. 2007.
- [58] B. Karakostas, "A DNS Architecture for the Internet of Things: A Case Study in Transport Logistics," *Procedia Comput. Sci.*, vol. 19, no. Ant, pp. 594–601, Jan. 2013.
- [59] S. Li, L. Da Xu, and S. Zhao, "The internet of things: a survey," *Inf. Syst. Front.*, vol. 17, no. 2, pp. 243–259, 2015.
- [60] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, "Future internet: The internet of things architecture, possible applications and key challenges," in *Proceedings - 10th International Conference on Frontiers of Information Technology, FIT 2012*, 2012, pp. 257–260.
- [61] C. Gonz and P. Espada, "Using Model-Driven Architecture Principles to Generate Applications Based on Interconnecting Smart Objects and Sensors," in *Advances and Applications in Model-Driven Engineering*, Vicente Diaz; Juan Lovelle; B. García-Bustelo; Oscar Martinez, Ed. IGI Global., 2013, pp. 73–87.
- [62] I. García Méndez, "Hacia el 'todo conectado': analizamos los nichos de negocio que se abren con el internet de las cosas," *Emprendedores: las claves de la economía y el éxito profesional*, no. 191. pp. 92–96, 2013.
- [63] C. Un and P. D. E. Iot, "Introducción al internet de las cosas," 2013.
- [64] L. Atzori, S. Member, A. Iera, and G. Morabito, "SIoT: Giving a Social Structure to the Internet of Things," vol. 15, no. 11, pp. 1193–1195, 2011.
- [65] V. Baños-Gonzalez, M. Afaqui, E. Lopez-Aguilera, and E. Garcia-

- Villegas, "IEEE 802.11ah: A Technology to Face the IoT Challenge," *Sensors*, vol. 16, no. 11, p. 1960, 2016.
- [66] K. Wang and M. M. H. Khan, "Performance prediction for apache spark platform," in *Proceedings - 2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security and 2015 IEEE 12th International Conference on Embedded Software and Systems, H*, 2015, pp. 166–173.
- [67] C. Gonz, "Tesis Doctoral – Mención Internacional," 2016.
- [68] R. Hariharan and D. C. Misra, "Enterprise Suite for Panchayats," in *9th International Conference on E-Government (ICEG 2012)*, 2012, pp. 29–39.
- [69] H. S. Ning and H. Liu, "Cyber-physical-social-thinking space based science and technology framework for the Internet of Things," *Sci. China Inf. Sci.*, vol. 58, no. 3, pp. 1–19, 2015.
- [70] P. R. Newswire, "Zebra Technologies and Trimble Collaborate on Internet of Things Solution," *IL-ZEBRATECHNOLOGIES*. 2014.
- [71] L. Sanchez *et al.*, "SmartSantander: IoT experimentation over a smart city testbed," *Comput. Networks*, vol. 61, pp. 217–238, Mar. 2014.
- [72] M. Nitti, V. Pilloni, G. Colistra, and L. Atzori, "The Virtual Object as a Major Element of the Internet of Things: A Survey," *IEEE Commun. Surv. Tutorials*, vol. 18, no. 2, 2016.
- [73] D. Gil, A. Ferrández, H. Mora-Mora, and J. Peral, "Internet of things: A review of surveys based on context aware intelligent services," *Sensors (Switzerland)*, vol. 16, no. 7, pp. 1–23, 2016.
- [74] J. Liu, X. Li, X. Chen, Y. Zhen, and L. Zeng, "Applications of Internet of Things on Smart Grid in China," *13th Int. Conf. Adv. Commun. Technol.*, pp. 13–17, 2011.

- [75] M. Darianian and M. P. Michael, "Smart home mobile RFID-based internet-of-things systems and services," in *Proceedings - 2008 International Conference on Advanced Computer Theory and Engineering, ICACTE 2008*, 2008, pp. 116–120.
- [76] H. Schaffers, N. Komninos, M. Pallot, B. Trousse, M. Nilsson, and A. Oliveira, "Smart cities and the future internet: Towards cooperation frameworks for open innovation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6656, pp. 431–446, 2011.
- [77] M. Liu, J. Ma, L. Lin, M. Ge, Q. Wang, and C. Liu, "Intelligent assembly system for mechanical products and key technology based on internet of things," *J. Intell. Manuf.*, 2014.
- [78] Y. Sun, H. Song, A. J. Jara, and R. Bie, "Internet of Things and Big Data Analytics for Smart and Connected Communities," *IEEE Access*, vol. 4, 2016.
- [79] M. Benalla, B. Achchab, and H. Hrimech, "A Distributed Intelligent System for Emergency Convoy," *Int. J. Interact. Multimed. Artif. Intel.*, vol. 4, no. 1, pp. 42–45, 2016.
- [80] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An Information Framework for Creating a Smart City Through Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 2, 2014.
- [81] J. Lee, B. Bagheri, and H. Kao, "A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems," *Manuf. Lett.*, vol. 3, pp. 18–23, 2015.
- [82] B. Guo, D. Zhang, and Z. Wang, "Living with internet of things: The emergence of embedded intelligence," in *Proceedings - 2011 IEEE International Conferences on Internet of Things and Cyber, Physical and Social Computing, iThings/CPSCoM 2011*, 2011, pp. 297–304.

- [83] H. (Deutsche P. A. Henning, Kagermann(National Academy of Science and Engineering). Wolfgang, Wahlster (German Research Center for Artificial Intelligence). Johannes, “Recommendations for implementing the strategic initiative INDUSTRIE 4.0,” *Final Rep. Ind. 4.0 WG*, no. April, p. 82, 2013.
- [84] C. Lesjak, N. Druml, R. Matischek, T. Rupprechter, and G. Holweg, “Security in industrial IoT – quo vadis?,” *e i Elektrotechnik und Informationstechnik*, pp. 324–329, 2016.
- [85] L. Da Xu, W. He, and S. Li, “Internet of things in industries: A survey,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, 2014.
- [86] R. Sanchez-Iborra and M. D. Cano, “State of the art in LP-WAN solutions for industrial IoT services,” *Sensors (Switzerland)*, vol. 16, no. 5, 2016.
- [87] I. Lee and K. Lee, “The Internet of Things (IoT): Applications, investments, and challenges for enterprises,” *Bus. Horiz.*, vol. 58, no. 4, pp. 431–440, 2015.
- [88] L. Da Xu, W. He, and S. Li, “Internet of things in industries: A survey,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.
- [89] E. W. T. Ngai, K. K. L. Moon, F. J. Riggins, and C. Y. Yi, “RFID research: An academic literature review (1995-2005) and future research directions,” *Int. J. Prod. Econ.*, vol. 112, no. 2, pp. 510–520, 2008.
- [90] X. Jia, Q. Feng, T. Fan, and Q. Lei, “RFID technology and its applications in Internet of Things (IoT),” *2012 2nd Int. Conf. Consum. Electron. Commun. Networks*, pp. 1282–1285, 2012.
- [91] Y. Chen, G. M. Lee, L. Shu, and N. Crespi, “Industrial internet of

- things-based collaborative sensing intelligence: Framework and research challenges,” *Sensors (Switzerland)*, vol. 16, no. 2, pp. 1–19, 2016.
- [92] N. Roy, A. Misra, and D. Cook, “Ambient and smartphone sensor assisted ADL recognition in multi-inhabitant smart environments,” *J. Ambient Intell. Humaniz. Comput.*, vol. 7, no. 1, pp. 1–19, 2016.
- [93] B. Rahmani, “Industrial Internet of Things: Design and Stabilization of Nonlinear Automation Systems,” *J. Intell. Robot. Syst.*, 2016.
- [94] D. Georgakopoulos and P. P. Jayaraman, “Internet of things: from internet scale sensing to smart services,” *Computing*, vol. 98, no. 10, pp. 1–18, 2016.
- [95] S. Krco, B. Pokric, and F. Carrez, “Designing IoT architecture(s): A European perspective,” in *2014 IEEE World Forum on Internet of Things, WF-IoT 2014*, 2014, pp. 79–84.
- [96] P. Fraga-lamas, T. M. Fernández-caramés, M. Suárez-albela, and L. Castedo, “A Review on Internet of Things for Defense and Public Safety,” *Sensors*, pp. 1–46, 2016.
- [97] J. Wan *et al.*, “Software-Defined Industrial Internet of Things in the Context of Industry 4.0,” *IEEE Sens. J.*, no. c, pp. 1–1, 2016.
- [98] D. M. Boyd and N. B. Ellison, “Social network sites: Definition, history, and scholarship,” *J. Comput. Commun.*, vol. 13, no. 1, pp. 210–230, 2007.
- [99] N. Mena Díaz, “Redes sociales y Gestión de la Información: un enfoque desde la teoría de grafos,” *Ciencias la Inf.*, vol. 43, no. 1, pp. 29–37, 2012.
- [100] S. P. S. Borgatti, A. Mehra, D. D. J. Brass, and G. Labianca, “Network analysis in the social sciences,” *Science (80-.)*, vol. 323, no. April, pp. 892–5, 2009.

- [101] P. R. Rogers, "The Academy of Management Review," *Acad. Manag. J.*, vol. 20, no. 2, pp. 276–278, 2014.
- [102] D. Sledgianowski and S. Kulviwat, "Using social network sites: The effects of playfulness, critical mass and trust in a hedonic context," *J. Comput. Inf. Syst.*, vol. 49, no. 4, pp. 74–83, 2009.
- [103] L. C. Freeman, "La centralidad de las redes sociales. Clarificación conceptual," *Política y Soc.*, vol. 33, pp. 131–148, 2000.
- [104] M. Blackstock, R. Lea, and A. Friday, "Uniting online social networks with places and things," *Proc. Second Int. Work. Web Things*, p. 5:1--5:6, 2011.
- [105] A. Teutle, "Twitter: Network properties analysis," *Electron. Commun. Comput. (CONIELECOMP), 2010 20th Int. Conf.*, pp. 180–186, 2010.
- [106] M. R. Morris, J. Teevan, and K. Panovich, "What Do People Ask Their Social Networks, and Why? A Survey Study of Status Message Q&A Behavior," in *CHI '10 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2010, pp. 1739–1748.
- [107] D. M. Boyd and N. B. Ellison, "Social Network Sites: Definition, History, and Scholarship," *Journal of Computer-Mediated Communication*, 13(1), 210–230. doi:10.1111/j.1083-6101.2007.00393.xp," *J. Comput. Commun.*, vol. 13, no. 1, pp. 210–230, 2008.
- [108] S. Asur and B. a. Huberman, "Predicting the Future with Social Media," *Web Intell. Intell. Agent Technol. (WI-IAT), 2010 IEEE/WIC/ACM Int. Conf.*, vol. 1, 2010.
- [109] L. Atzori, A. Iera, and G. Morabito, "From 'smart objects' to 'social objects': The next evolutionary step of the internet of things," *IEEE Commun. Mag.*, vol. 52, no. 1, pp. 97–105, 2014.

- [110] M. Kranz, L. Roalter, and F. Michahelles, “Things that twitter: social networks and the internet of things,” *What can Internet Things do* *Citiz. Work. Eighth Int. Conf. Pervasive Comput. (Pervasive 2010)*., no. May 2010, pp. 1–10, 2010.
- [111] D. Guinard, M. Fischer, and V. Trifa, “Sharing using social networks in a composable Web of Things,” *Pervasive Comput. Commun. Work. (PERCOM Work. 2010 8th IEEE Int. Conf.*, pp. 702–707, 2010.
- [112] D. Doran, S. Gokhale, and A. Dagnino, “Human sensing for smart cities,” *Proc. 2013 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Min. - ASONAM '13*, pp. 1323–1330, 2013.
- [113] I. Foster, Y. Zhao, I. Raicu, and S. Lu, “Cloud Computing and Grid Computing 360-degree compared,” in *Grid Computing Environments Workshop, GCE 2008*, 2008.
- [114] S. P. Mirashe and N. V. Kalyankar, “Cloud Computing,” *Commun. ACM*, vol. 51, no. 7, p. 9, 2010.
- [115] W. Voorsluys, J. Broberg, and R. Buyya, “Introduction to Cloud Computing,” *Cloud Computing: Principles and Paradigms*, pp. 1–41, 2011.
- [116] R. Buyya, S. Pandey, and C. Vecchiola, “Cloudbus toolkit for market-oriented cloud computing,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2009, vol. 5931 LNCS, pp. 24–44.
- [117] S. Pandey and S. Nepal, “Cloud computing and scientific applications - Big data, scalable analytics, and beyond,” *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1774–1776, 2013.
- [118] M. D. Assunção, R. N. Calheiros, S. Bianchi, M. A. S. Netto, and R. Buyya, “Big data computing and clouds: Trends and future directions,” *J. Parallel Distrib. Comput.*, vol. 79, pp. 3–15, 2014.

- [119] S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, and A. Ghalsasi, "Cloud computing - The business perspective," *Decis. Support Syst.*, vol. 51, no. 1, pp. 176–189, 2011.
- [120] W. I. C. A. Computing, "What is Cloud Computing," *Commun. ACM*, vol. 101, no. 2, pp. 1–37, 2008.
- [121] P. Mell and T. Grance, "The NIST Definition of Cloud Computing (Draft) Recommendations of the National Institute of Standards and Technology," *Nist Spec. Publ.*, vol. 145, no. 6, p. 7, 2011.
- [122] D. Agrawal, S. Das, and A. El Abbadi, "Big data and cloud computing," *Proc. 14th Int. Conf. Extending Database Technol. - EDBT/ICDT '11*, vol. 443, no. 7108, p. 530, 2011.
- [123] A. Botta, W. De Donato, V. Persico, and A. Pescapé, "Integration of Cloud computing and Internet of Things: A survey," *Futur. Gener. Comput. Syst.*, vol. 56, 2016.
- [124] A. Iosup, R. Prodan, and D. Epema, "IaaS Cloud Benchmarking : Approaches , Challenges , and Experience," *Proc. ACM/IEEE Conf. High Perform. Netw. Comput.*, pp. 1–8, 2013.
- [125] P. Arora, "Cloud Computing Security Issues in Infrastructure as a Service," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 2, no. 1, pp. 707–711, 2012.
- [126] R. Prodan and S. Ostermann, "A survey and taxonomy of infrastructure as a service and web hosting cloud providers," in *Proceedings - IEEE/ACM International Workshop on Grid Computing*, 2009, pp. 17–25.
- [127] I. Weidner, "SaaS, PaaS , IaaS und XaaS: Was die Cloud Buzzwords bedeuten," *Comput. online*, pp. 1–5, 2015.
- [128] A. Juan-Verdejo and B. Surajbali, "XaaS Multi-Cloud marketplace architecture enacting the industry 4.0 concepts," in *IFIP Advances in*

Information and Communication Technology, 2016, vol. 470, pp. 11–123.

- [129] B. Selic, “Model-driven development of real-time software using OMG standards,” in *ISORC 2003: SIXTH IEEE INTERNATIONAL SYMPOSIUM ON OBJECT-ORIENTED REAL-TIME DISTRIBUTED COMPUTING, PROCEEDINGS*, 2003, pp. 4–6.
- [130] B. Hailpern and P. Tarr, “Model-driven development: The good, the bad, and the ugly,” *IBM Syst. J.*, vol. 45, no. 3, pp. 451–461, 2006.
- [131] T. A. Bapty and J. Sztipanovits, “Model-based engineering of large-scale real-time systems,” *Proc. Int. Work. Eng. Comput. Syst.*, pp. 467–474, 1997.
- [132] D. Thomas and B. M. Barry, “Model driven development: the case for domain oriented programming,” *Opsla*, pp. 2–7, 2003.
- [133] E. Seidewitz, “What models mean,” *IEEE Softw.*, vol. 20, no. 5, pp. 26–32, 2003.
- [134] B. Al-Batran, B. Schätz, and B. Hummel, “Model Driven Engineering Languages and Systems,” *MoDELS*, vol. 7590, no. OCTOBER, pp. 258–272, 2012.
- [135] B. Selic, “Models, Software Models and UML,” in *UML for Real*, 2003, pp. 1–16.
- [136] J. Trejo and A. Robles, “Conceptos fundamentales de Ingeniería dirigida por Modelos y Modelos de Dominio Específico,” *Rev. Investig. Sist. e Informática*, vol. 7, no. 2, pp. 9–19, 2010.
- [137] E. R. Núñez-Valdez, V. García-Díaz, J. M. C. Lovelle, Y. S. Achaerandio, and R. González-Crespo, “A model-driven approach to generate and deploy videogames on multiple platforms,” *J. Ambient Intell. Humaniz. Comput.*, pp. 1–13, 2016.
- [138] J. Ledgard, T. Trejo, A. David, and E. Robles, “Conceptos

fundamentales de Ingeniería dirigida por Modelos y Modelos de Dominio Específico,” vol. 7, no. 2, pp. 9–19, 2010.

- [139] C. Gonzalez, “MIDGAR: interoperabilidad de objetos en el marco de Internet de las Cosas mediante el uso de Ingeniería Dirigida por Modelos,” Universidad de Oviedo, 2017.
- [140] T. Stahl, M. Völter, J. Bettin, A. Haase, and S. Helsen, *Model-driven software development: technology, engineering, management*. John Wiley & Sons, 2013.
- [141] V. Diaz, “MDCI : Model-Driven Continuous Integration,” Universidad de Oviedo, 2011.
- [142] B. Selic, “Models, Software Models and UML,” in *UML for Real: Design of Embedded Real-Time Systems*, L. Lavagno, G. Martin, and B. Selic, Eds. Boston, MA: Springer US, 2003, pp. 1–16.
- [143] S. Kent, “Model driven engineering,” in *International Conference on Integrated Formal Methods*, 2002, pp. 286–298.
- [144] L. Fuentes and A. Vallecillo, “Una Introducción a los Perfiles UML,” *Novatica*, vol. 168, pp. 6–11, 2004.
- [145] O. M. G. A. Specification and C. Bars, “OMG Unified Modeling Language (OMG UML),” *Language (Baltim).*, no. November, pp. 1–212, 2007.
- [146] J. Sutherland, “Business Objects in Corporate Information Systems,” *ACM Comput. Surv.*, vol. 27, no. 2, pp. 274–276, 1995.
- [147] G. Poels and G. Dedene, “Complexity metrics for formally specified business requirements,” in *Proceedings of the Annual Oregon Workshop on Software Metrics (AOWSM’97)*, 1997, pp. 1–11.
- [148] S. Meliá, “WebSA: un método de desarrollo dirigido por modelos de arquitectura para aplicaciones web,” Universidad de Alicante, 2006.

- [149] J. Quintero and R. Anaya, “MDA y el papel de los modelos en el proceso de desarrollo de software,” *Rev. EIA*, vol. 8, pp. 131–146, 2007.
- [150] G. Booch, A. Brown, S. Iyengar, J. Rumbaugh, and B. Selic, “An MDA manifesto,” *IBM Ration. Softw.*, pp. 1–9, 2004.
- [151] Y. Liu and Y. Ma, “An Approach for MDA Model Transformation Based on JEE Platform,” in *4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08.*, 2008.
- [152] S. Sendall, R. Hauser, J. Koehler, J. Küster, and M. Wahler, “Understanding model transformation by classification and formalization,” in *Proc. Of Workshop on Software Transformation Systems*, 2004.
- [153] T. Mens and P. Van Gorp, “A Taxonomy of Model Transformation,” *Electron. Notes Theor. Comput. Sci.*, vol. 152, pp. 125–142, 2006.
- [154] K. Czarnecki and S. Helsen, “Classification of Model Transformation Approaches,” in *OOPSLA'03 Workshop on Generative Techniques in the Context of Model-Driven Architecture*, 2003, pp. 1–17.
- [155] D. Linthicum, “Responsive Data Architecture for the Internet of Things,” *Computer (Long. Beach. Calif.)*, vol. 49, no. 10, pp. 72–75, Oct. 2016.
- [156] J. Lee, B. Bagheri, and H. Kao, “A Cyber-Physical Systems architecture for Industry 4 . 0-based manufacturing systems,” *Manuf. Lett.*, vol. 3, pp. 18–23, 2015.
- [157] J. Lee, H. D. Ardakani, S. Yang, and B. Bagheri, “Industrial Big Data Analytics and Cyber-physical Systems for Future Maintenance & Service Innovation,” in *Procedia CIRP*, 2015, vol. 38, pp. 3–7.