Yacchirema-Vargas, DC.; Palau Salvador, CE. (2016). Smart IoT Gateway For Heterogeneous Devices Interoperability. IEEE Latin America Transactions. 14(8):3900-3906. doi:10.1109/TLA.2016.7786378.

Smart IoT Gateway For Heterogeneous Devices Interoperability

D. C. Yacchirema and C.Palau, *Senior Member, IEEE*

***Abstract*— The Internet of things (IoT) will interconnect a huge amount of devices, leading to a new way of interaction in the physical and virtual world, inspired by the idea of ubiquity, where all the objects around us, such as: sensors, automobiles, refrigerators, thermostats, industrial robots, tablets, smartphones, etc. could be connected anytime and anywhere. However, one of the main challenges that faces IoT is the high degree of heterogeneity in terms of communication capabilities of the devices, protocols, technologies or hardware. We focus on the implementation of a new Smart IoT Gateway designed to allow interconnection and interoperability between heterogeneous devices in the IoT. The proposed gateway offers significant advantages: (i) it enables connectivity of different protocols and traditional communication technologies (Ethernet) and wireless (ZigBee, Bluetooth, WiFi); (ii) it uses a flexible protocol that translates all the data obtained from the different sensors into a uniform format, performing the analysis of the data obtained from the environment-based-rules related to the different types of sensors; (iii) it uses a lightweight and optimal protocol on the use of devices with limited resources for delivering information environment; and (iv) it provides local data storage for later use and analysis. Our proof of concept demonstrates the performance and capacity of the proposed Smart IoT Gateway is related with Active and Healthy Aging (AHA).**

***Keywords*— Internet of Things, Gateway, Interoperability, Interconnection, MQTT.[1]**

I. INTRODUCCIÓN

UN PRONÓSTICO reciente realizado por la Corporación Internacional de Datos (IDC) proyecta al Internet de las Cosas (IoT) y al ecosistema asociado a ser un mercado de $7,1 billones en el 2020, [1] que incluirá 50.000 millones de dispositivos conectados [2]. Estas previsiones dejan ver un nuevo modo de interacción en el mundo físico, inspirado en la idea de ubicuidad, donde todos los objetos que nos rodean (sensores, automóviles, refrigeradoras, termostatos, robots industriales, tables, smartphones, etc.) se puedan conectar a Internet en cualquier momento y en cualquier lugar. Los recientes avances en las tecnologías de comunicación, así como el costo cada vez menor de dispositivos, potencia de procesamiento y conectividad de datos, habilitan esta posibilidad.

La visión de IoT es que todos estos dispositivos se comuniquen e interactúen entre sí, [3] siendo capaces de recoger información, procesarla y compartirla. Sin embargo, uno de los principales obstáculos al que se enfrenta IoT es el alto grado de heterogeneidad de las diferentes capacidades de comunicación (protocolos, tecnologías y hardware) de los dispositivos. En particular, el requisito de interoperabilidad es uno de los grandes retos que debe abordarse para la integración y el desarrollo de nuevas plataformas IoT [4] [5]. Sin interoperabilidad el 40 % de los beneficios potenciales de IoT no se podrán obtener. [6]

La interoperabilidad se considera como la capacidad de dos o más sistemas o componentes para intercambiar datos y utilizar la información [4] [7]. En el contexto de IoT la interoperabilidad puede establecerse a diferentes niveles. Interoperabilidad a nivel de dispositivo [8]: consiste en integrar de manera transparente dispositivos que soportan hardware y software heterogéneos. Interoperabilidad a nivel de protocolos [9]: consiste en habilitar la comunicación directa entre diferentes tecnologías de red. Interoperabilidad a nivel de datos [7]: está asociada con el formato de los datos de los dispositivos. Los mensajes transmitidos por los protocolos necesitan una sintaxis y codificación definida.

Solucionar el problema de interoperabilidad permitirá eliminar los denominados ecosistemas cerrados o silos verticales de la información, [10] obteniendo el verdadero valor de IoT que reside en los datos que se crean y transmiten cuando los dispositivos interactúan entre sí [2]. Grupos de estandarización como la ITU [11] y estudios en [12] proponen la utilización de Gateways para habilitar la interoperabilidad de IoT, incorporando en este dispositivo toda la funcionalidad requerida como, por ejemplo, el procesamiento y almacenamiento de la información de los datos de los dispositivos. La investigación presente hasta la fecha se ha centrado en el diseño y desarrollo de Gateways IoT para solucionar la interoperabilidad de protocolos y tecnologías de comunicación específicos, sin considerar las capacidades limitadas de los dispositivos IoT.

En este trabajo se propone la implementación de un Smart IoT Gateway que permite la interoperabilidad de dispositivos heterogéneos a nivel de dispositivo, protocolos y datos, mediante la incorporación de las funcionalidades de conversión de protocolos de comunicación, transformación, procesamiento y almacenamiento de datos. A nivel de dispositivo; abstrae las características específicas de hardware y software de los dispositivos heterogéneos para permitir la interconexión e interoperabilidad entre sí. A nivel de protocolo; permite la conectividad de los diferentes protocolos y tecnologías de comunicación tradicionales (Ethernet) e inalámbrica (ZigBee, Bluetooth, WiFi) a través de la función de conversión de protocolos. A nivel de datos; utiliza un protocolo flexible que traduce todos los datos obtenidos de los diferentes dispositivos sensores en un formato uniforme, mediante las funciones de transformación y procesamiento de datos. Para el intercambio de datos entre los dispositivos se utiliza un protocolo abierto, ligero y óptimo en el uso de

[D]. C. Yacchirema, Universitat Politècnica de Valencia (UPV), Valencia, Spain, diayac1@posgrado.upv.es

C. Palau, Universitat Politècnica de Valencia (UPV), Valencia, Spain, cpalau@dcom.upv.es

dispositivos con pocos recursos. El Smart IoT Gateway también permite el almacenamiento local de los datos y la visualización de los mismos en tiempo real.

Este artículo está estructurado de la siguiente manera: en la Sección II, se presenta el estado del arte de trabajos previos relacionados y su respectiva contribución en el área de la interoperabilidad. En la Sección III, se presenta la arquitectura del Smart IoT Gateway propuesto. La aplicación del Smart IoT Gateway a un caso de estudio se describe en la Sección IV. La evaluación y resultados se proporcionan en la Sección V. Finalmente en las Sección VI se presenta las conclusiones y líneas futuras de trabajo.

II. Estado Del Arte

Los dispositivos de IoT pueden interconectarse a través de múltiples protocolos como: Bluetooth, WiFi, Ethernet, etc. pero eso no significa que sean capaces de entenderse. El principal objetivo de este trabajo es implementar un Gateway que permita resolver el problema de interoperabilidad a diferentes niveles: dispositivo, protocolos y datos.

Existen diferentes trabajos relacionados con el diseño e implementación de Gateways para IoT. Un primer conjunto de investigaciones proponen la implementación de un Gateway para solucionar la interoperabilidad entre protocolos y tecnologías de comunicación específicos. Los autores en [13] proponen un "IoT Gateway" basado en los protocolos ZigBee y GPRS con el objetivo de facilitar la transmisión de datos de las redes de sensores inalámbricas (WSNs Wireless Sensor Networks) y las redes de comunicación móviles. El Gateway propuesto permite la conversión de protocolos y la administración de los dispositivos. El trabajo en [14], presenta la implementación de un "Smart IoT Gateway" para WSNs, el cual es responsable de la conversión de protocolos y fusión de datos de los diferentes sensores. El Gateway soporta tres tipos de comunicación de datos: Ethernet, 3G y bus RS485. Una investigación similar se propone en [15] a través de la implementación de un middleware, que habilita en el Gateway la ejecución de código de aplicaciones para ofrecer las funcionalidades de: conversión de protocolos, cache de solicitudes, descubrimiento y cacheo inteligente. Los autores plantean mapear la semántica entre los protocolos: CoAP y TCP/HTTP mediante el uso de la pila de protocolo EZnet.

Otras contribuciones proponen Gateways para dominios de aplicación específicos. Por ejemplo, los autores en [8] describen un Gateway denominado "Smart e-Salud" que actúa como una capa intermedia entre la WSN e Internet, ofreciendo servicios de alto nivel como: procesamiento de datos central, almacenamiento local (temporal) y minería de datos para el despliegue de aplicaciones de monitoreo de la Salud. La propuesta apunta a establecer interoperabilidad entre las diferentes redes heterogéneas (WiFi, Bluetooth y 6LoWPAN) y los protocolos subyacentes mediante la utilización de WebSockets. Un Gateway para el hogar basado en OSGI se propone en [16], el cual permite la interconexión e interoperabilidad de diferentes protocolos de red doméstica: x10, Insteon, ZigBee y UPnp El Gateway habilita el descubrimiento automático de dispositivos.

Otro tipo de Gateways se basan en el soporte de Cloud Computing. Estas propuestas se enfocan en utilizar las ventajas de potencia de cálculo y almacenamiento escalable de este paradigma. Los autores en [17] presentan un Gateway capaz de enviar los datos de sus respectivas redes locales a las aplicaciones IoT alojadas en el Cloud. El Gateway provee servicios de análisis y formato de datos. Finalmente en [18], se propone un Gateway basado en Fog Computing [5] para permitir la comunicación entre los dispositivos IoT y Cloud Computing. Esta propuesta tiene por objetivo disminuir la carga que se produce en el Cloud debido al procesamiento de los datos de los dispositivos IoT. El Gateway ofrece los servicios de pre-procesamiento de los datos, mientras que el almacenamiento de los mismos se realiza en servidores Fog Computing.

De la misma manera para la construcción de Gateways IoT existen propuestas de código abierto como: *Kura* [19] y *Mihini* [20], las cuales son impulsadas por las iniciativas M2M Eclipse. *Kura* [19] es un framework basado en Java y OSGI. Este framework ofrece un conjunto de APIs para simplificar la administración de las configuraciones de red (Ethernet, WiFi y módems celulares), la comunicación con plataformas de integración IoT y la administración remota del Gateway. Uno de sus principales objetivos es la integración de diferentes protocolos y estándares en un entorno heterogéneo. Sin embargo, utilizan bundles en Java, que limitan las capacidades del Gateway a un solo lenguaje de programación [21]. *Mihini* [20] es un framework basado en el lenguaje de programación Lua, el cual proporciona una capa de abstracción del hardware y protocolos subyacentes de los Gateways y permite la transmisión inteligente de datos entre dispositivos y servidores IoT. Este framework puede ser implementado en dispositivos con recursos limitados. Sin embargo, incrementa el consumo de energía de este tipo de dispositivos.

La literatura disponible hasta la fecha se ha centrado, en su mayoría, en la implementación de gateways para la interconexión de WSNs con Internet o entornos de Cloud Computing, algunos de estos esfuerzos se enfocan en dominios de aplicación específicos, por ejemplo, el hogar inteligente (Smart Home). Otros, simplemente limitan su nivel de inteligencia a la conversión de protocolos específicos. Las propuestas descritas, sin embargo, no abordan la interoperabilidad e interconexión de dispositivos heterogéneos IoT a nivel de dispositivo, protocolos y datos, tampoco realizan evaluaciones de las prestaciones del mismo.

III. Arquitectura Smart IoT Gateway

El requisito principal del Smart IoT Gateway es permitir la interoperabilidad a nivel de dispositivo, protocolos de comunicación y datos con el objetivo de establecer la conectividad sin fisuras entre dispositivos heterogéneos. La Fig. 1, detalla la arquitectura funcional del Smart IoT Gateway. El Smart IoT Gateway cumple cuatro funciones principales: (i) Transformación de Datos, (ii) Procesamiento de Datos, (iii) Conversión de Protocolos y (iv) almacenamiento de los diferentes Datos de los dispositivos. Estas funciones permiten llevar a cabo la secuencia de acciones que se realizan a nivel de las capas de la arquitectura

básica de una solución IoT [22] [23]. Capa de Percepción: recopilar y transmitir los datos por parte de los dispositivos conectados; Capa de Transporte: transportar los datos de los dispositivos a través de una red que puede ser fija, inalámbrica, móvil, etc.; Capa Middleware: almacenar los datos de los dispositivos generados en el entorno, extraer información de valor a partir del procesado de los datos y tomar decisiones automáticas basadas en los resultados (generar notificaciones que se envían a los dispositivos, personas o sistemas para que lleven a cabo determinadas acciones de control); y Capa de Aplicación: compartir datos con otras plataformas y entornos.

La arquitectura propuesta soporta la conexión de dispositivos heterogéneos a través diferentes protocolos y tecnologías de comunicaciones tradicionales (Ethernet) y comunicaciones inalámbricas (ZigBee, WiFi, Bluetooth). La arquitectura incorpora un gestor de eventos a través de un servidor MQTT Broker, con la finalidad de: (i) enviar notificaciones a los dispositivos actuadores para que ejecuten las acciones fijadas y (ii) enviar alertas a los usuarios finales en función de reglas de medición especificadas. De la misma forma, se incorpora un Servidor Web que permite el acceso a los datos de los dispositivos desde Internet o desde entornos Cloud, los cuales previamente se encuentran almacenados en un servidor de BBDD incorporado en el Gateway.

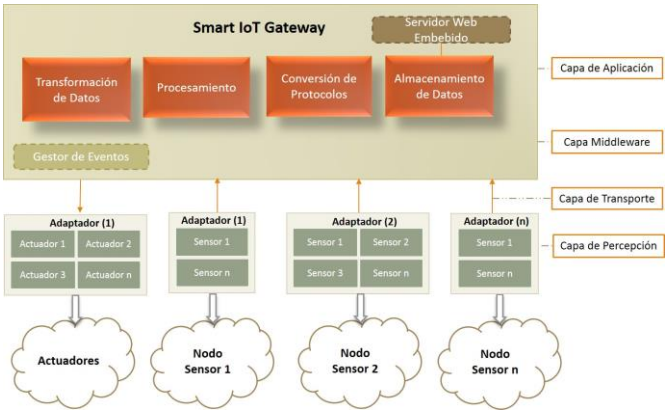

Figura 1. Arquitectura de alto Nivel del Smart IoT Gateway.

A. *Transformación de Datos*

Cada dispositivo sensor captura datos sobre ellos mismos y sobre su entorno. Los conjuntos de datos generados por estos dispositivos están en varios formatos. Una de las primeras funciones que realiza el Smart IoT Gateway es transformar y normalizar estos datos a un formato estándar que permita el intercambio y representación de los diferentes dispositivos heterogéneos. Para describir las propiedades de los diferentes dispositivos el Smart IoT Gateway utiliza el formato de datos JSON, que ha sido seleccionado por las ventajas que ofrece [24]; frente a otros lenguajes más extendidos para la representación e intercambio de datos como XML y, lo más importante, que puede ser empleado en dispositivos con recursos limitados y entornos donde el flujo de datos es alto, como a futuro se prevé que será el Internet de las Cosas [22]. Al normalizar los datos, se puede crear modelos simples para representar cualquier dispositivo IoT. La Fig.2, muestra un ejemplo de la transformación de los datos provenientes de dispositivos heterogéneos al formato JSON.

```
Temperature:\0x0921.9 C
Humidity:\0x0945.9 rH
Pressure:\0x091023.6 mbar
Magneto:\0x09-432/-4029/1513 (X/Y/Z) mgauss
Acceleration:\0x09-267/-189/16664 (X/Y/Z) mg
Gyroscope:\0x0923/4/-25 (X/Y/Z) mdps
```

```
{
  "Sensor": [
    {
      "tipo": "Temperature",
      "valor": "21.9",
      "unidad": "C",
      "nombre": "STM32",
      "fechaHora": "12/02/16:11:47:13",
      "protocolo": "Wifi"
    }
  ]
}{
  "Sensor": [
    {
      "tipo": "Humidity",
      "valor": "45.9",
      "unidad": "rH",
      "nombre": "STM32",
      "fechaHora": "12/02/16:11:47:14",
      "protocolo": "Wifi"
    }
  ]
}{
  "Sensor": [
    {
      "tipo": "Pressure",
      "valor": "1023.6",
      "unidad": "mbar",
      "nombre": "STM32",
      "fechaHora": "12/02/16:11:47:14",
      "protocolo": "Wifi"
    }
  ]
}
```

Figura 2. Transformación de los datos provenientes del sensor de ambiente y de movimiento STM32 al formato de datos JSON.

B. *Procesamiento de Datos*

El procesamiento de la información en el Smart IoT Gateway permite reducir el tiempo de latencia o el jitter al encontrarse cercano a los dispositivos finales (sensores y actuadores). Tomando ventaja con respecto a las propuestas presentadas en [17] y [18]. Los datos generados por los dispositivos sensores producen un efecto sobre los dispositivos actuadores tras ser procesados por los diferentes servicios y aplicaciones. El Smart IoT Gateway procesa estos datos en base al análisis de reglas; condiciones previamente definidas, que determinan las acciones que deben ejecutar los dispositivos actuadores. Si los datos obtenidos de los sensores cumplen las condiciones fijadas por la aplicación, se genera un evento. El Smart IoT Gateway detecta el evento y envía señales a los actuadores y notificaciones a los usuarios externos. La Fig.3, describe la funcionalidad del procesamiento de Datos.

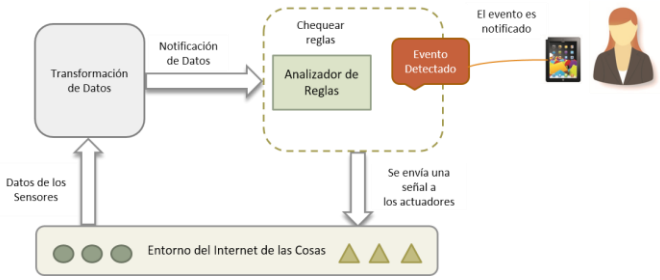

Figura 3. Detección de eventos en base a reglas previamente definidas.

C. *Conversión de Protocolos*

Para la interconexión e interoperabilidad entre dispositivos heterogéneos es necesario la conversión de protocolos. Varias organizaciones como: AllSeen Alliance [25], OpenIoT [26] e IPSO Alliance [27], están trabajando sobre la estandarización

de protocolos de comunicación para proveer la interoperabilidad entre dispositivos de diferentes fabricantes.

El Smart Iot Gateway propuesto actúa como un puente de conexión y traductor entre diferentes protocolos de comunicación, debido a que soporta un amplio rango de interfaces de red: ZigBee, WiFi, Bluetooth y Ethernet. De esta manera el Smart IoT Gateway se encuentra siempre escuchando nuevas peticiones de conexión, a través de las interfaces y puertos establecidos para la comunicación con los diferentes dispositivos. Cuando el Smart IoT Gateway recibe un paquete de datos por una interfaz física, descomprime la carga útil del paquete origen y mapea a un formato de paquete, de acuerdo al protocolo de comunicación destino. Esta funcionalidad se lleva a cabo una vez que los datos se han normalizado y procesado. Posteriormente se transmite por la red a los dispositivos actuadores a través del protocolo de transporte de mensajes MQTT, óptimo en el uso de dispositivos con pocos recursos [5]; puede ser implementado en dispositivos con menos de 64 kb de RAM.[28]

MQTT implementa una arquitectura de tipo publicación/suscripción a través de un nodo central denominado "broker". La comunicación se basa en la definición de temas, que son mensajes enviados por los publicadores al broker y recibidos por los subscriptores. El Smart IoT Gateway está diseñado para asumir el rol de bróker; para ello crea temas específicos en base a los diferentes tipos de sensores (presencia, temperatura, humedad, etc.). Estos temas son publicados por el Smart IoT Gateway a los dispositivos actuadores y usuarios externos, quienes previamente se encuentran subscritos a uno o varios temas de interés. En la Fig.4, se presenta el diagrama asociado al envío de señales y notificaciones a los actuadores y usuarios externos respectivamente.

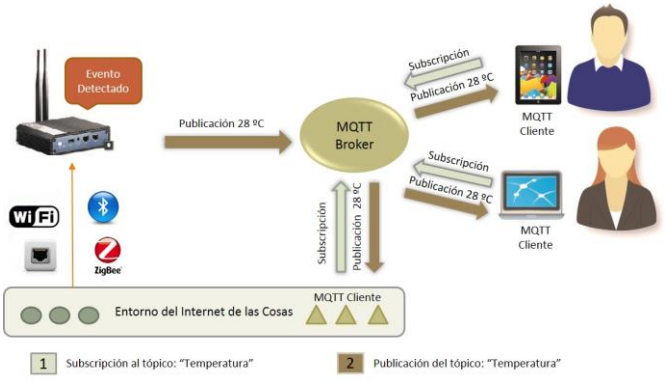

Figura 4. Procedimiento de Subscripción /Publicación de dispositivos actuadores y usuarios externos.

D. Almacenamiento de los Datos

El Smart IoT Gateway permite el almacenamiento local de los datos de los dispositivos en una base de datos. En el caso de los dispositivos sensores se almacena los valores obtenidos del entorno y en el caso de los dispositivos actuadores el valor de su estado. Los datos almacenados se pueden obtener y visualizar a través de una aplicación web.

IV. CASO DE ESTUDIO

Existen diferentes dominios de aplicación en el entorno de IoT, por ejemplo, sociedad, industria y medio ambiente. [29] En este trabajo describimos un caso de estudio del Smart IoT Gateway aplicado al dominio de la sociedad, específicamente al envejecimiento activo y saludable de las personas mayores.

En países como Estados Unidos, más de 10,000 personas mayores se jubilan cada día. Hasta un 90 por ciento de ellos desean permanecer en su propia casa, manteniendo su independencia sin sobrecargar a sus familias o al personal de cuidado en el hogar. Para adquirir la capacidad de afrontar y disfrutar de un envejecimiento activo y saludable, las personas mayores requieren que las tecnologías de información y comunicación den respuesta a sus necesidades: *Seguridad y Salud*; estar y sentirse seguro dentro del hogar, *Autonomía*; realizar funciones sin la ayuda de otras personas, *Participación Social*; evitar el sentimiento de soledad o abandono, *Movilidad*; facilitar la movilidad dentro del hogar. IoT deber ser considerado como la piedra angular para responder a estas necesidades. De esta manera el Smart IoT servirá como base para ofrecer una arquitectura completa que promueva la autonomía, independencia y seguridad de las personas mayores dentro del hogar.

Las principales funcionalidades que ofrece la arquitectura son: (i) Monitorizar la movilidad de las personas mayores dentro del hogar, a fin de emitir señales que habilitan la ejecución de acciones; por ejemplo, encender la luz ante la detección de una señal de presencia. (ii) Monitorizar las condiciones ambientales del hogar, (iii) Notificar y detectar situaciones de riesgo para la salud; por ejemplo, cuando se produce una caída, en la que la persona pueda necesitar apoyo médico, (iv) Almacenar y presentar la información a usuarios externos al hogar; por ejemplo, a familiares y personal de cuidado.

Para lograr realizar con éxito estás funciones y demostrar la interoperabilidad a nivel de dispositivo, protocolos y datos, se propone una arquitectura que consta de tres componentes principales: Dispositivos Sensores y Actuadores, Smart IoT Gateway y Aplicación Cliente. (Fig.5)

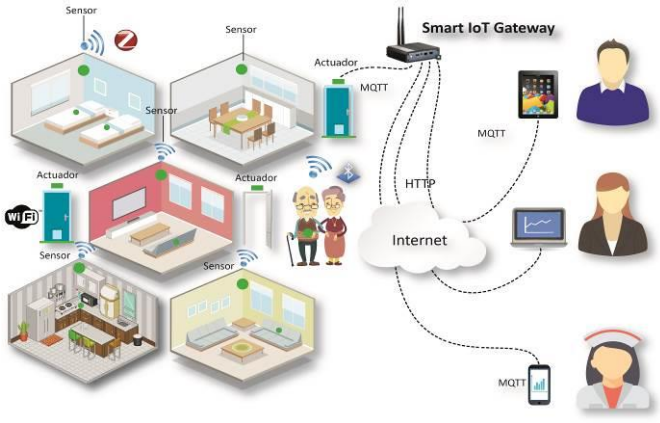

Figura 5. Arquitectura del caso de estudio aplicado al envejecimiento activo y saludable de las personas mayores.

1) Dispositivos Sensores y Actuadores

En un escenario de envejecimiento activo y saludable varios dispositivos sensores son distribuidos estratégicamente por todo el hogar e incorporados en el cuerpo de las personas mayores. Los sensores detectan los datos de la movilidad; traslación de sillón/cama, acceso al baño, riesgo de caídas, etc. y actividades de las personas mayores dentro del hogar; levantarse de la cama, tomar los medicamentos, preparar comidas, usar el teléfono, etc.

Tres tipos de sensores son utilizados para este caso de estudio: sensores portátiles, sensores de presencia y sensores ambientales. La utilización de sensores portátiles; por ejemplo, para monitorizar las caídas que pueden sufrir las personas mayores usualmente se basan en el uso de uno o más acelerómetros. Los sensores de presencia son utilizados para la ubicación de las personas mayores dentro del hogar; detectan todos los movimientos en el área que cubren y envían una señal cada vez que se produce un movimiento [30]. Su ubicación es importante dentro del hogar; por ejemplo, en la mesa de cocina, la cama, el sofá etc. De la misma forma se utilizan sensores de temperatura, humedad y gas para detectar las condiciones ambientales del hogar.

Los datos obtenidos de los sensores son enviados al Smart IoT Gateway a través del módulo de comunicación de cada nodo sensor. Por otra parte, los actuadores reciben órdenes para realizar ciertas acciones, como por ejemplo abrir las puertas, encender o apagar la luz, etc. Para demostrar la interoperabilidad a nivel de dispositivo, nosotros utilizamos diferentes actuadores y nodos sensores: Arduino, NodeMCU V2 ESP8266 y Sensor de Ambiente y de Movimiento STM32.

2) Smart IoT Gateway

El Smart Iot Gateway es el componente principal de la arquitectura propuesta. Protocolos de bajo nivel como WiFi, Bluetooth, ZigBee y tradicionales como Ethernet son utilizados para la conexión entre el Smart IoT Gateway y los nodos sensores. Para demostrar la interoperabilidad a nivel de protocolo, cada nodo sensor trabaja con una tecnología de red diferente. El módulo de traducción de protocolos soporta la comunicación entre las redes IP como WiFi y Ethernet y las redes no IP como Bluetooth y ZigBee. La principal ventaja del dispositivo es su capacidad para ampliarse a otras redes abiertas o propietarias.

Los datos provenientes de los nodos sensores son normalizados y mapeados al formato JSON por el Smart IoT Gateway a través de la función de transformación de datos. El Smart IoT Gateway envía señales a los dispositivos actuadores o alertas a los familiares o personal de cuidado utilizando el protocolo MQTT mediante la función de procesamiento de datos y conversión de protocolos. Para la construcción del prototipo de Smart IoT Gateway se adoptó un esquema de diseño de alto rendimiento. Utilizamos una Raspberry Pi 2 modelo B que cuenta con un procesador ARM Quad-Core a 900 MHz, 1 GB de memoria RAM, 32GB de memoria de almacenamiento SD, 4 puertos USB, 1 puerto HDMI , 1 puerto RJ-45 y un consumo energético de 700 mA, (3.5 W). Para demostrar la interoperabilidad a nivel de protocolos de comunicación varios módulos se han integrado como: Ethernet 10/100, 802.11 b/g/n, Xbee serie2 y Bluetooth v4.0. La memoria SD externa permite el almacenamiento local de los datos de los dispositivos en una base de datos MySQL, para su posterior visualización por parte de los usuarios.

3) Aplicación Web

Los datos de movilidad y de las condiciones ambientales del hogar pueden ser visualizados en tiempo real por los familiares o cuidadores desde cualquier dispositivo móvil (tablets, smartphones, etc.) o fijo (computador de escritorio) a través de una aplicación web construida en el lenguaje de programación PHP, que se encuentra alojada en servidor web incorporado en el Smart IoT Gateway. (Fig.6)

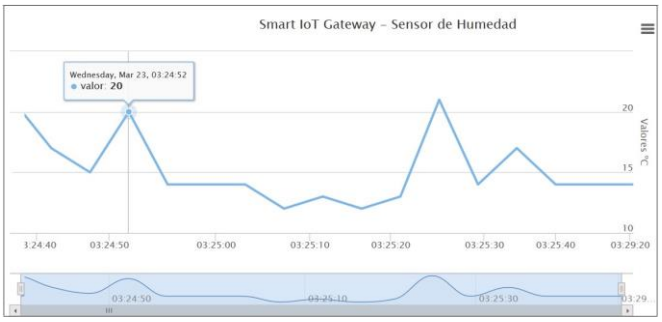

Figura 6. Datos obtenidos del Sensor de Temperatura NodeMCU V2 ESP8226.

V. Evaluación y Resultados

A. Entorno de Pruebas

Con el fin de evaluar el rendimiento y la capacidad del Smart IoT Gateway en situaciones de carga, se ha realizado un banco de pruebas utilizando los dispositivos especificados en la tabla I.

TABLA I.
CASO DE ESTUDIO

Dispositivo –Protocolo	Función	Tipo de Sensor
Arduino – ZigBee	NS	Gas
Arduino – Bluetooth	NS	Presencia
Arduino - Ethernet	NS	Sonido
NodeMCU V2 ESP8226 -WiFi	NS	Humedad y temperatura
STM32 – WiFi	NS	Presión atmosférica, giroscopio y acelerómetro
Arduino-WiFi	A	-
Raspberry Pi 2B	GW	-

NS = Nodo Sensor, A = Actuador, GW = Smart IoT Gateway

B. Escenarios de Prueba

Para caracterizar el rendimiento de un sistema, la latencia es una de las medidas más utilizadas. Sin embargo, en un escenario IoT debido a la sincronía entre los relojes internos de los diferentes dispositivos, no se puede obtener un valor de latencia exacta. Por lo tanto, en lugar de ello, hemos calculado la variación del retardo de los mensajes recibidos (Jitter), la mediana y la desviación estandar. De esta manera, las medidas no se ven afectadas por una posible asíncrona entre los dispositivos sensores y actuadores. Se ha tomado varias

muestras sucesivas de 1000 paquetes de diferentes tamaños cada una: 64, 128, 256, 512 y 1024 bytes. Los paquetes fueron enviados desde los nodos sensores al Smart IoT Gateway y del Smart IoT Gateway a los actuadores.

La evaluación de la capacidad se ha realizado en términos del consumo de memoria y de CPU. Para asegurarse de que las condiciones de red no afecta el rendimiento drásticamente del Smart IoT Gateway, realizamos una evaluación exhaustiva de tres semanas, promediando las mediciones obtenidas al enviar desde los diferentes nodos sensores 5000 paquetes al Smart IoT Gateway para que sean ejecutados en base a las funcionalidades: Transformación de datos, conversión de protocolos, procesamiento de datos y almacenamiento de datos.

C. Resultados

1) Rendimiento

Como se muestra en la Fig. 7 (a), el Smart IoT Gateway exhibe una escalabilidad lineal del jitter en relación a la longitud de paquetes enviados. Aun cuando el número de paquete enviados es de 1000 y el tamaño de la trama es de 1024 bytes, los valores del jitter son aceptables en cada una de las muestras enviadas. El valor mínimo del retardo entre las muestras es de 0,964ms y el valor máximo es de 1039,6ms siendo una duración relativamente corta no mayor a 2 segundos. Por otra parte, la mediana (Fig. 7 (b)) y la desviación estándar (Fig. 7 (c)) presentan la misma tendencia creciente conforme la longitud de los paquetes aumenta.

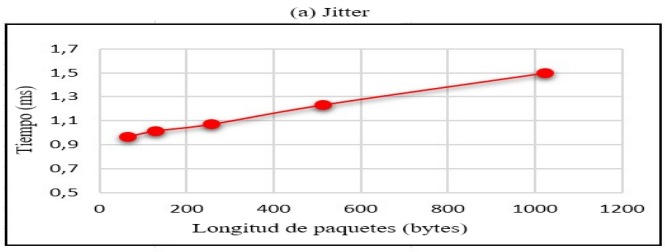

Figura 7. Rendimiento (a) Jitter.

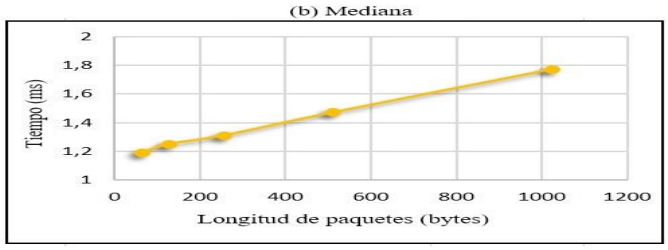

Figura 7. Rendimiento (b) Mediana.

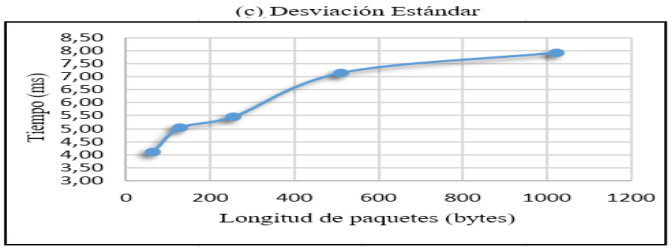

Figura 7. Rendimiento (c) Desviación Estándar.

2) Capacidad

En los resultados de la Fig. 8 (a), es posible observar que la cantidad de memoria utilizada es extremadamente baja y no supera el 50% de utilización del total de la memoria (1GB). Por otra parte, se demuestra que todo el procesamiento que realiza el Smart IoT Gateway para cumplir con sus funciones no tiene un impacto significativo en el consumo de CPU (Fig 8 (b)), dado que los valores permanecen bastantes constantes a pesar del aumento de paquetes enviados. Más específicamente, la utilización de CPU es del 53,2 % al enviar 1000 paquetes, incrementándose su utilización en 4,3% al enviar 5000 paquetes. Por lo tanto, el Smart Iot Gateway tiene la capacidad suficiente, para procesar más datos, considerando que el dispositivo sobre el que se ha implementado es de recursos limitados.

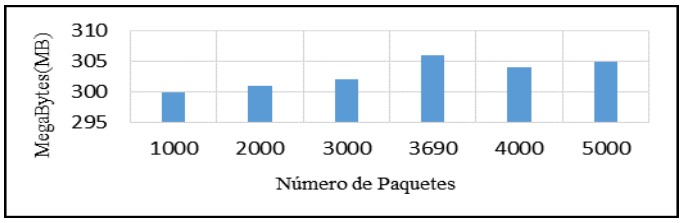

Figura 8. Capacidad (a) Consumo de Memoria.

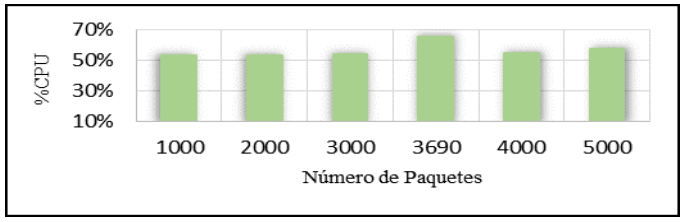

Figura 8. Capacidad (b) Consumo de CPU.

VI. CONCLUSIONES

La existencia de diferentes dispositivos heterogéneos es uno de los principales obstáculos para la adopción y el despliegue de IoT. En este artículo se ha implementado un nuevo Smart IoT Gateway que permite la interoperabilidad a nivel de dispositivos, protocolos y datos, aprovechando la utilización de protocolos flexibles y óptimos en el uso de recursos. El Smart IoT Gateway propuesto actúa como un elemento central permitiendo la interconexión de dispositivos que trabajan con diferentes protocolos y tecnologías de comunicación: ZigBee, WiFi, Bluetooth y Ethernet mediante la implementación de las funcionalidades de conversión de protocolos, transformación, procesamiento y almacenamiento de datos. El Smart IoT Gateway ha sido aplicado al dominio de la sociedad, específicamente al envejecimiento activo y saludable de las personas mayores, no obstante, puede aplicarse a varios dominios debido a que permite llevar a cabo la secuencia de acciones que se realizan a nivel de las capas de la arquitectura básica de un solución IoT. Las prestaciones del Smart IoT Gateway se han evaluado, obteniendo resultados aceptables a nivel de rendimiento y capacidad. Como trabajo futuro, proponemos extender las funcionalidades del Smart IoT Gateway para que los usuarios externos puedan controlar remotamente a los dispositivos heterogéneos como respuesta a la notificación de un evento.

AGRADECIMIENTOS

Esta investigación fue apoyada por el Gobierno del Ecuador a través de la Secretaría de Educación Superior, Ciencia, Tecnología e Innovación (SENESCYT) y ha recibido la financiación del Programa de Investigación e Innovación Horizonte 2020 de la Unión Europea en el marco de la "Interoperabilidad de Plataformas Heterogéneas de IoT" (INTER-IoT), proyecto bajo acuerdo de subvención n° 687283.

REFERENCIAS

[1] D. Lund and M. Morales, "Worldwide and Regional Internet of Things (IoT) 2014-2020 Forecast : A Virtuous Circle of Proven Value and Demand," 2014.

[2] D. Evans, "The Internet of Things - How the Next Evolution of the Internet is Changing Everything," 2011.

[3] 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, 2013.

[4] ETSI, "Interoperability Best Practices," 2013.

[5] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols and Applications," *IEEE Commun. Surv. Tutorials*, no. 99, pp. 1–1, 2015.

[6] Mckinsey Global Institute, "the Internet of Things : Mapping the Value Beyond the Hype," 2015.

[7] IERC, "IoT Semantic Interoperability:Research Challenges, Best Practices, Recommendations and Next Steps," 2011.

[8] A. Rahmani, N. K. Thanigaivelan, T. N. Gia, J. Granados, B. Negash, P. Liljeberg, and H. Tenhunen, "Smart e-Health Gateway : Bringing Intelligence to Internet-of-Things Based Ubiquitous Healthcare Systems," in *Consumer Communications and Networking Conference (CCNC), 12th Annual IEEE*, 2015, pp. 826–834.

[9] M. Unis, A. Nettsträter, F. Iml, J. Stefa, C. S. D. Suni, A. Salinas, and U. Sapienza, "Internet of Things-Architecture IoT-A Final architectural reference model for the IoT v3.0," 2013.

[10] S. Tilkov, "Semantic Gateway as a Service architecture for IoT Interoperability," in *IEEE International Conference on Mobile Services*, 2015, vol. 32, no. 2, pp. 116–116.

[11] Telecommunication Standarization Sector of ITU, "Common requirements and capabilities of a gateway for Internet of things applications," 2014.

[12] A. Castellani, S. Loreto, N. Bui, and M. Zorzi, "Quickly interoperable Internet of Things using simple transparent gateways," in *Interconnecting Smart Objects with the Internet*, 2011, pp. 1–2.

[13] Q. Zhu, R. Wang, Q. Chen, Y. Liu, and W. Qin, "IOT Gateway: BridgingWireless Sensor Networks into Internet of Things," in *EEE/IFIP International Conference on Embedded and Ubiquitous Computing*, 2010, pp. 347–352.

[14] S. Guoqiang, C. Yanming, Z. Chao, and Z. Yanxu, "Design and implementation of a smart IoT gateway," in *Green Computing and Communications (GreenCom), IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing*, 2013, pp. 720–723.

[15] D. Bimschas, H. Hellbrück, and R. Mietz, "Middleware for smart gateways connecting sensornets to the internet," in *Proceedings of the 5th International Workshop on Middleware Tools, Services and Run-Time Support for Sensor Networks. ACM.*, 2010, pp. 8–14.

[16] J. E. Kim, G. Boulos, J. Yackovich, T. Barth, C. Beckel, and D. Mosse, "Seamless Integration of Heterogeneous Devices and Access Control in Smart Homes," in *Eighth International Conference on Intelligent Environments*, 2012, pp. 206–213.

[17] F. Anon, V. Navarathinarasah, M. Hoang, and C.-H. Lung, "Building a Framework for Internet of Things and Cloud Computing," *2014 IEEE Int. Conf. Internet Things(iThings), IEEE Green Comput. Commun. IEEE Cyber, Phys. Soc. Comput.*, no. iThings, pp. 132–139, 2014.

[18] M. Aazam and E.-N. Huh, "Fog Computing and Smart Gateway Based Communication for Cloud of Things," in *International Conference on Future Internet of Things and Cloud*, 2014, pp. 464–470.

[19] "Kura," 2015. [Online]. Available: http://www.eclipse.org/kura/. [Accessed: 01-Dec-2015].

[20] "Mihini." [Online]. Available: https://wiki.eclipse.org/Mihini. [Accessed: 01-Dec-2015].

[21] D. Wilusz and J. Rykowski, "Comparison of architectures for service management in IoT and sensor networks by means of OSGi and REST services," in *Computer Science and Information Systems (FedCSIS)*, 2014, vol. 2, pp. 1207–1214.

[22] N. Tan, L., & Wang, "Future Internet: The Internet of Things," in *Advanced Computer Theory and Engineering (ICACTE), 3rd International Conference*, 2010, pp. 376–380.

[23] 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*, 2012, pp. 257–260.

[24] "Introducing JSON." [Online]. Available: http://www.json.org/. [Accessed: 01-Jan-2016].

[25] "Allseen Alliance." [Online]. Available: https://allseenalliance.org/. [Accessed: 12-Jan-2016].

[26] "OpenIoT." [Online]. Available: http://www.openiot.eu/. [Accessed: 12-Jan-2016].

[27] "IPSO Alliance." [Online]. Available: http://www.ipso-alliance.org/. [Accessed: 12-Jan-2016].

[28] V. Gazis, M. Görtz, M. Huber, A. Leonardi, K. Mathioudakis, A. Wiesmaier, F. Zeiger, and E. Vasilomanolakis, "A Survey of Technologies for the Internet of Things," in *International Wireless Communications and Mobile Computing Conference (IWCMC), Machine - to - Machine Communications (M2M) & Internet of Things (IoT) Workshop*, 2015, pp. 1090–1095.

[29] S. Agrawal and M. L. Das, "Internet of Things — A paradigm shift of future Internet applications," in *Nirma University International Conference on Engineering*, 2011, pp. 1–7.

[30] A. Fleury, M. Vacher, and N. Noury, "SVM-based multimodal classification of activities of daily living in health smart homes: Sensors, algorithms, and first experimental results," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 2, pp. 274–283, 2010.

Diana C. Yacchirema received the M.Sc. degree in Communications Technologies, Systems and Networks from the Universitat Politécnica de Valencia in 2011 and the M.Sc. degree in Management of Communications and Information Technology from the Escuela Politécnica Nacional, Quito-Ecuador, in 2009. Currently, she is a Ph.D Student in the Escuela Técnica Superior de Ingenieros de Telecomunicación at the Universitat Politécnica de Valencia, Spain. Her research activities and interests cover a wide range of subjects related to Internet of Things, sensor networks and network security.

Carlos E. Palau received his M.Sc. and Ph.D (Dr.Ing.) degrees, both in telecommunication engineering, from the Universitat Politécnica de Valencia in 1993 and 1997, respectively. He is Full Professor in the Escuela Técnica Superior de Ingenieros de Telecomunicación at the Universitat Politécnica de Valencia. He has more than 18 years of experience in the ICT research area in the area of Networking. He has collaborated extensively in the R&D of multimedia streaming, security, networking and wireless communications for government agencies, defence and European Commission. He has been the main UPVLC researcher in the FASYS project, which has funded this work, He is author and co-author of more than 120 research papers and member of the TPC of several IEEE, ACM and IFIP conferences. He is Senior Member of IEEE.