

Un Framework de Programación Dinámica para IoT

Alejandro Pérez-Vereda¹, Daniel Flores-Martín², Carlos Canal¹ and Juan M. Murillo²

¹ University of Malaga, ² University of Extremadura, Spain
apvereda@uma.es, dfloresm@unex.es, canal@lcc.uma.es,
juanmamu@unex.es

Resumen. El crecimiento del *Internet of Things* está permitiendo la conexión a la red de muchos dispositivos. La tecnología debería permitir a estos dispositivos adaptarse automáticamente a las necesidades de sus usuarios. Con este propósito, desarrollamos en trabajos anteriores la arquitectura de referencia *People as a Service*, para crear perfiles virtuales de los usuarios almacenados en sus *smartphones*. Sin embargo, para la obtención de un perfil completo necesitamos información de contexto, que solo pueden proporcionarnos estos dispositivos del entorno. Nuestro objetivo es desarrollar un *framework* en el que usuarios y dispositivos conectados se integren de manera transparente y dinámica, permitiendo una actualización programática de los perfiles y el comportamiento de los dispositivos. De esta forma, damos un primer paso hacia un Mundo Programable.

Palabras clave: *Internet of Things*, *People as a Service*, Mundo Programable, Beacons, Perfiles virtuales de usuarios, *Framework* de programación.

1 Introducción y antecedentes

En un futuro próximo en el que interactuemos con docenas de dispositivos diariamente [6], será impensable tener que configurarlos manualmente en función de nuestras necesidades cada vez que queramos usarlos. Para prevenir este problema, necesitamos soluciones que conviertan en “inteligentes” a estos dispositivos, permitiéndoles adaptar su comportamiento a nuestras preferencias de forma automática y dinámica.

Existen varios trabajos de investigación que ya tratan de obtener y procesar la información contextual de los usuarios creando perfiles virtuales de los mismos [2, 5]. Incluso los autores de este artículo hemos trabajado en una arquitectura de computación móvil de este tipo, llamada *People as a Service* (PeaaS) [4], que aprovecha la gran presencia de los *smartphones* en nuestra vida diaria y los sensores de los que disponen para la inferencia y mantenimiento de un perfil virtual del usuario.

Sin embargo, para obtener un perfil lo suficientemente completo, en muchas situaciones necesitaremos información acerca del entorno del usuario que solo la comunicación con estos objetos inteligentes podría darnos. Además, la interacción entre estos perfiles y los dispositivos inteligentes no puede reducirse a un mero intercambio de datos. Tal y como ya se prevé en [7], tiene que producirse un salto del IoT centrado en los datos de hoy día hacia una red de objetos programables dinámica y automáticamente. Esto supone una evolución hacia un software flexible que permita la ejecución

de código durante el funcionamiento tanto de los objetos inteligentes como de los *smartphones* de los usuarios. De esta forma ambos podrán aprender y evolucionar con el tiempo de forma automática y transparente.

De acuerdo con esto, nuestro objetivo es el de usar el perfil virtual de los usuarios para adaptar dinámicamente el comportamiento de los dispositivos que usan en su día a día. La propuesta que presentamos en este artículo consiste en un nuevo *framework* de programación dinámica que permita abrir un amplio abanico de nuevas posibilidades de interacción entre las personas y los dispositivos conectados de su entorno.

2 Arquitectura de la propuesta

Nuestra meta entonces consiste en el desarrollo de un *framework* de programación que, en primer lugar, permita la aportación y/o actualización de forma dinámica de la información contenida en el perfil del usuario y, en segundo lugar, aporte herramientas para adaptar el comportamiento del dispositivo conectado disparando ciertas acciones relacionadas con la nueva información disponible. Así, tanto el perfil como los objetos inteligentes se actualizarán programáticamente y en tiempo de ejecución.

En primera instancia, la propuesta se basará en el uso de *beacons* para representar los objetos inteligentes, que interactuarán con los *smartphones* mediante *Bluetooth Low Energy* (BLE). Así, los *beacons* transmitirán una URL a un script, que el móvil descargará y ejecutará cuando entre en su rango de emisión. Esta ejecución permitirá la actualización del perfil virtual además del comportamiento del dispositivo inteligente.

Para el manejo de los perfiles virtuales, nos basaremos en nuestro trabajo anterior sobre la arquitectura PeaaS ya mencionada. Esta otorga al *smartphone* capacidad para inferir, almacenar y por supuesto compartir el perfil virtual de su dueño, convirtiéndose en su interfaz automática con los dispositivos del entorno negociando sus interacciones de forma transparente para el usuario [1].

Los perfiles virtuales estarán escritos en JSON, ya que se trata de un lenguaje abierto y fácilmente extensible. Un detalle importante de esta arquitectura a tener en cuenta es que el perfil solo se almacena en el teléfono del usuario, por lo que necesitamos una base de datos orientada a documentos que almacene los datos en el propio teléfono.

En cuanto al script, en principio deberá estar escrito en algún lenguaje de propósito general y que disponga de intérprete para una aplicación móvil. Una primera opción podría ser Python aunque aún hay que estudiar su viabilidad.

Para que la idea quede más clara proponemos un posible escenario en el cual podría desplegarse esta tecnología:

Supongamos una situación cotidiana como ir a la farmacia. Esta solo necesitaría tener un *beacon* en su interior emitiendo la dirección a su script. De esta forma, cada vez que un cliente entre en la farmacia, su *smartphone* detectará la señal del *beacon* y descargará el script. La aplicación móvil del *smartphone* no es específica para la farmacia, es una aplicación muy general cuya única función es ejecutar los scripts que encuentra. Un posible script para la farmacia podría ser una tarjeta de visita que insertará en el perfil del cliente toda la información acerca de la farmacia y lanzará notificaciones con productos en oferta para el cliente. Adicionalmente, el script puede conectarse a la base

de datos de la farmacia e insertar los datos del cliente si sus restricciones de privacidad lo permiten obteniéndolos del propio perfil. Ahora la farmacia crearía las fichas de sus clientes de forma autónoma y el cliente recibiría ofertas que puedan interesarle, además de tener toda la información de su farmacia de confianza a mano.

3 Conclusiones y trabajos futuros

En trabajos previos [4], hemos abogado por el uso de los *smartphones* para inferir perfiles virtuales de las personas. Estos perfiles son el elemento clave para el avance hacia un mundo en el que la tecnología se adapte a nosotros automáticamente [1]. El propósito es construir una interfaz transparente entre las personas y los dispositivos de su entorno IoT en la era de un Mundo Programable [7]. De esta forma, nuestro objetivo a medio plazo es el desarrollo de un *framework* que permita la adaptación dinámica de forma programática de los perfiles y de los dispositivos conectados de nuestro entorno.

En líneas de trabajo futuras, los propios *smartphones* podrán actuar también como emisores además de receptores de la señal BLE de forma que puedan intercambiar información y llegar a objetivos comunes.

En un *framework* como el que proponemos, la autenticación, privacidad y seguridad toman una importancia vital por lo que deberá ser considerada como un objetivo futuro fundamental. Como resultado, es probable que, en lugar de usar un lenguaje de propósito general para los scripts, se desarrolle un lenguaje propio de alto nivel con primitivas para la interacción con el perfil virtual y otras funcionalidades como notificaciones.

Agradecimientos. Este trabajo ha sido parcialmente financiado por el Gobierno de España a través de los proyectos TIN2015-67083-R y TIN2015-69957-R (MINECO/FEDER, UE), por el proyecto 4IE 0045-4IE-4-P financiado por el programa Interreg V-A España-Portugal (POCTEP) 2014-2020, y por el Gobierno Regional de Extremadura (proyecto GR15098).

Referencias

1. Berrocal, J., Garcia-Alonso, J., Canal, C., Murillo, J. M.: Situational-context: a unified view of everything involved at a particular situation. ICWE 2016, LNCS 9671, Springer, 476-483 (2016)
2. Gronli, T.M., Ghinea, G., Younas, M.: Context-aware and automatic configuration of mobile devices in cloud-enabled ubiquitous computing. Personal and Ubiquitous Computing, 18(4), 883–894 (2014)
3. Guillen, J., Miranda, J., Berrocal, J., Garcia-Alonso, J., Murillo, J. M., Canal, C.: People as a Service: A Mobile-centric Model for Providing Collective Sociological Profiles. Software, IEEE, 31(2), 48-59 (2014)
4. Hong, J.Y., Suh, E.H., Kim, S.J.: Context-aware systems: a literature review and classification. Expert Systems with Applications, 36(4), 8509–8522 (2009)
5. Perera, C., Liu, C.H., Jayawardena, S., Chen, M.: Context-aware computing in the internet of things: a survey on internet of things from industrial market perspective. CoRR (2015)
6. Taivalsaari, A., Mikkonen, T.: A roadmap to the programmable world: Software challenges in the IoT era. Software, IEEE, 34(1), 72-80 (2017)