

Capa de abstracción de dispositivos para el diseño de software para entornos inteligentes

A. Fernandez-Montes¹, J. A. Ortega¹, M. Muñoz Organero², V. Cordoba Magaña²

¹Departamento de Lenguajes y Sistemas Informáticos, Universidad de Sevilla, Spain
{afdez,jortega}@us.es

²Departamento de Ingeniería Telemática, University Carlos III, Leganés, Spain
{munozm,vcordoba}@it.uc3m.es

Abstract

La popularización de la computación ubicua y la inteligencia ambiental se ha visto potenciada por el incremento de los dispositivos inteligentes que nos rodean. Las viviendas inteligentes o los ambientes inteligentes incluyen dispositivos de automatización y control para ahorro energético, realización de tareas, asistencia de personas y proporcionar un mayor confort de manera que se satisfagan las preferencias de los usuarios.

Este artículo se centra en la capa software que trata de proporcionar una abstracción superior sobre los dispositivos, de manera que sea transparente para el desarrollador el uso de unas u otras tecnologías implicadas con cada dispositivo.

1 Introducción

Un entorno inteligente se puede definir como *aquel que es capaz de adquirir y aplicar conocimiento sobre el entorno y sus habitantes para mejorar su experiencia con ese entorno* (1). Las tecnologías de entornos inteligentes son una parte importante de la computación ubicua. Mark Weiser (2) subrayó los principios para describir la Computación Ubicua de entre los cuales debemos enfatizar que el objetivo principal de un computador es ayudar a alguien a hacer algo. Hoy en día, debido a la popularización de los dispositivos computacionales y sus aplicaciones, la computación ubicua está llamada a ser la revolución para el desarrollo de entornos inteligentes.

Sin embargo los artefactos software relacionados con la computación ubicua, en conjunto con un amplio número de dispositivos computacionales (y el software necesario para completar su misión) son demasiado heterogéneos y difíciles de comparar o clasificar. Cada pieza de software evoluciona aislada o sólo en relación con el hardware para el que fue desarrollada. El problema que se trata en este artículo es la orquestación de una capa de abstracción sobre los dispositivos que permita realizar un desarrollo de software cómodo sobre estos dispositivos.

Esta capa de abstracción es análoga al modelo Open System Interconnection (OSI), que es una recomendación para caracterizar y estandarizar las funciones de un sistema de comunicación en términos de capas de abstracción.

Los beneficios de esta propuesta es el mejor entendimiento de los problemas que deben afrontarse cuando se desarrolla soluciones para entornos inteligentes. Esta capa de abstracción reduciría los costes de los ciclos de vida clásicos del software: diseño, desarrollo, despliegue y mantenimiento y favorecerá la interoperabilidad entre distintas soluciones.

2 Trabajo relacionado

La inteligencia ambiental es una línea de investigación muy popular, por lo que distintas iniciativas de investigación han aparecido alrededor de ésta. Una de las aplicaciones más comunes en inteligencia ambiental son los entornos inteligentes. Muchos investigadores alrededor del mundo desarrollan proyectos relacionados con estos entornos.

En esta sección los proyectos más famosos serán revisados, prestando especial atención a la arquitectura que siguen para construir una solución para entornos inteligentes.

El objetivo del proyecto MavHome (3) es crear una vivienda que actúe como un agente inteligente. Su arquitectura organiza el entorno como un agente que puede ser dividido en agentes inteligentes que interactúan entre ellos. Las tecnologías de cada agente son separadas en cuatro capas que cooperan entre sí: la capa de decisión, capa de información, capa de comunicación y capa física.

El proyecto ATRACO (4) trata de dar soporte a actividades del día a día. Usa una plataforma OSGi conectada con una pasarela residencial, que gestiona los dispositivos, sensores y actuadores a través del protocolo UPnP. Estos elementos trabajan conjuntamente para ofrecer un servicio dependiente del contexto para el entorno. El proyecto DomoSEC (5) ofrece una solución para la automatización que cubre las necesidades domóticas para el interior. DomoSEC se compone de una computadora embebida que centraliza la inteligencia de la vivienda, y se comunica con los dispositivos, todo basado también en la plataforma OSGi.

CASAS (6) es un sistema adaptativo para viviendas inteligentes que usa técnicas de aprendizaje automático para descubrir patrones en las actividades de las habitantes y genera reglas automáticas. La información capturada por los sensores se analiza para buscar por patrones de actividad que sean interesantes para ser automatizados. Los patrones se modelan continuamente en una estructura multinivel para construir el contexto. Finalmente se realiza una selección de actividades a automatizar.

Energy Aware Smart Home (7) se centra en la integración de dispositivos embebidos y heterogéneos y en enchufes con control de gasto realizada por un middleware llamado Hydra. El propósito de este middleware es recoger los datos de consumo energético para monitorizar y analizar el consumo de energía para usar un programa de consumo eficiente y controlar los dispositivos de la vivienda.

El proyecto Aware Home (8) intenta crear un entorno capaz de gestionar información de sí mismo, sus habitantes y sus actividades. Está compuesto de un conjunto de subsistemas que son responsables de las tecnologías desplegadas en él: interacción hombre-máquina, aprendizaje automático, percepción computacional, computación vestible, ingeniería del software y sensores.

El proyecto Place Lab (9) se centra en capturar el comportamiento de los habitantes a través de un conjunto de componentes que contienen un microcontrolador, altavoz, cámaras y un conjunto de sensores. Estos sensores graban un registro audiovisual de la actividad. Este conjunto de datos es usado para desarrollar nuevos algoritmos de detección del contexto y aplicaciones dependientes del contexto.

Otros sistemas como Sentient Computing System (19), pueden cambiar su comportamiento basado en un modelo de entornos que ellos construyen utilizando datos de los sensores. Trata de evitar las dificultades compartiendo y configurando los dispositivos para la explotación del modelo.

A partir de la información de estos proyectos podemos concluir que las mejoras de la tecnología han llenado los espacios inteligentes con sensores para percibir, técnicas para razonar y actuadores para modificar, pero todas estas propuestas de espacios inteligentes sufren de un mismo problema: sus componentes o dispositivos son muy dependientes de la arquitectura por lo que integrar nuevos componentes o sustituir los existentes cuando quedan obsoletos es un trabajo complejo. La consecuencia es que los entornos inteligentes se encuentran en un estado más académico que de explotación y comercialización. Esta es la principal razón que nos empujó a avanzar en la creación de una capa de abstracción de dispositivos que pueda ser usada en los distintos modelos de entornos inteligentes.

3 Modelo de abstracción de dispositivos

Este modelo se basa en la idea de un modelo de entorno inteligente proporcionando una capa de abstracción que describe un conjunto de estándares y convenciones para el control, configuración y acceso a los datos generados desde todos los dispositivos relacionados. La integración de los sensores y actuadores se lleva a cabo siguiendo el modelo de abstracción de dispositivos en el software de control.

Este modelo jerárquico se muestra en las figuras 1 y 2, y unifica los criterios para facilitar el acceso a los dispositivos, sus funcionalidades y los datos generados por ellos. De esta manera, los métodos usados no dependen de los protocolos subyacentes y permiten una manera más sencilla de intercambiar dispositivos. El uso de esta capa de abstracción es un paso más allá en la estandarización de entornos inteligentes.

Básicamente, esta capa de abstracción clasifica los dispositivos en sensores y actuadores. Los sensores se clasifican

a su vez en medidores y detectores y los actuadores dependiendo de su funcionalidad como pulsos, switches, dimmers y movimiento. Cada categoría de dispositivo proporciona un conjunto específico de métodos para el control de las acciones de los dispositivos.

Además, hay otro artefacto software entre la capa de abstracción y los dispositivos: la API que traduce los métodos de alto nivel hacia las peticiones dependientes del protocolo. Estas APIs son proporcionadas habitualmente por los fabricantes de estos dispositivos.

4 Conclusions

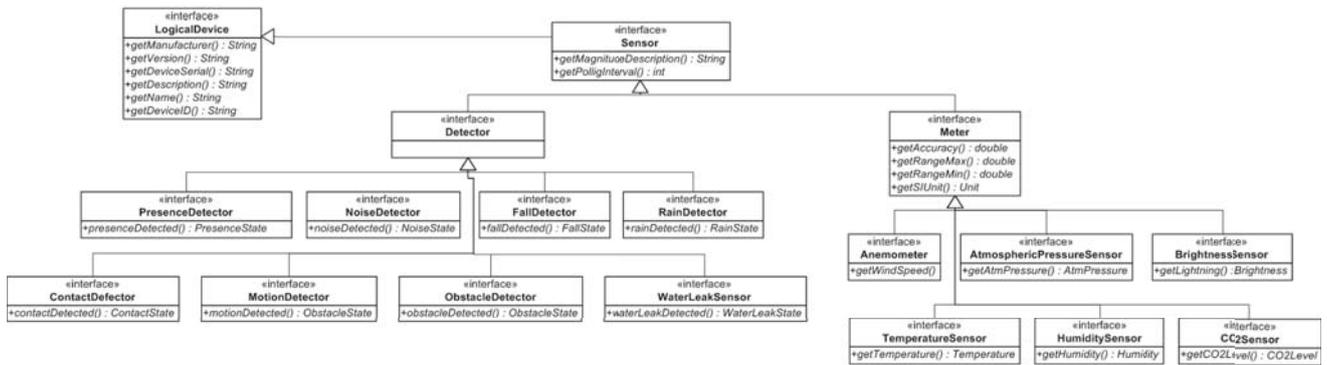
En este trabajo se ha presentado la capa de abstracción sobre los dispositivos indentificando los artefactos software necesarios para este objetivo.

El trabajo futuro en esta materia debería centrar su esfuerzo en tres áreas. La primera sería describir los procesos de razonamiento y actuación. La segunda es el desarrollo de un framework que facilitase la implementación de entornos inteligentes proporcionando los servicios más comunes y dando soporte a diversos módulos como un sistema común de información, comunicación y control. La tercera será el desarrollo de técnicas innovadoras para cada uno de estos módulos que implementen estos procesos.

Otras áreas importantes deberían ser tenidas en cuenta por este framework como las interfaces de usuario y la seguridad. Ambas áreas son de vital importancia para una experiencia completa de entornos inteligentes. Es un reto muy interesante no sólo considerar la capacidad de los entornos inteligentes para adaptarse a las preferencias del usuario sino usarlo para cambiar los comportamientos de éstos que podría permitir aplicar políticas de sostenibilidad de manera que pueda entrenar a los habitantes para ser más cuidadosos con el medio ambiente.

References

- [1] G. Youngblood, E. Heierman, L. Holder, D. Cook. *Automation intelligence for the smart environment*, Proceedings of the International Joint Conference on Artificial Intelligence, 2005.
- [2] M. Weiser. *The computer for the 21st century*, SIGMOBILE Mob. Comput. Commun. Rev., vol. 3, no. 3, pp. 3-11, 1999.
- [3] D. Cook, M. Youngblood, S. Das. *A multi-agent approach to controlling a smart environment*, Lecture notes in computer science, vol. 4008, p. 165, 2006.
- [4] A. Meliones, D. Economou, I. Grammatikakis, A. Kameas, C. Goumopoulos. *A context aware connected home platform for pervasive applications*, Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, pp. 120-125, 2008.
- [5] Zamora-Izquierdo, M. A., Santa, J., Gómez-Skarmeta, A. F. *Integral and Networked home automation solution towards indoor ambient intelligence*, IEEE Pervasive Computing, 2010.



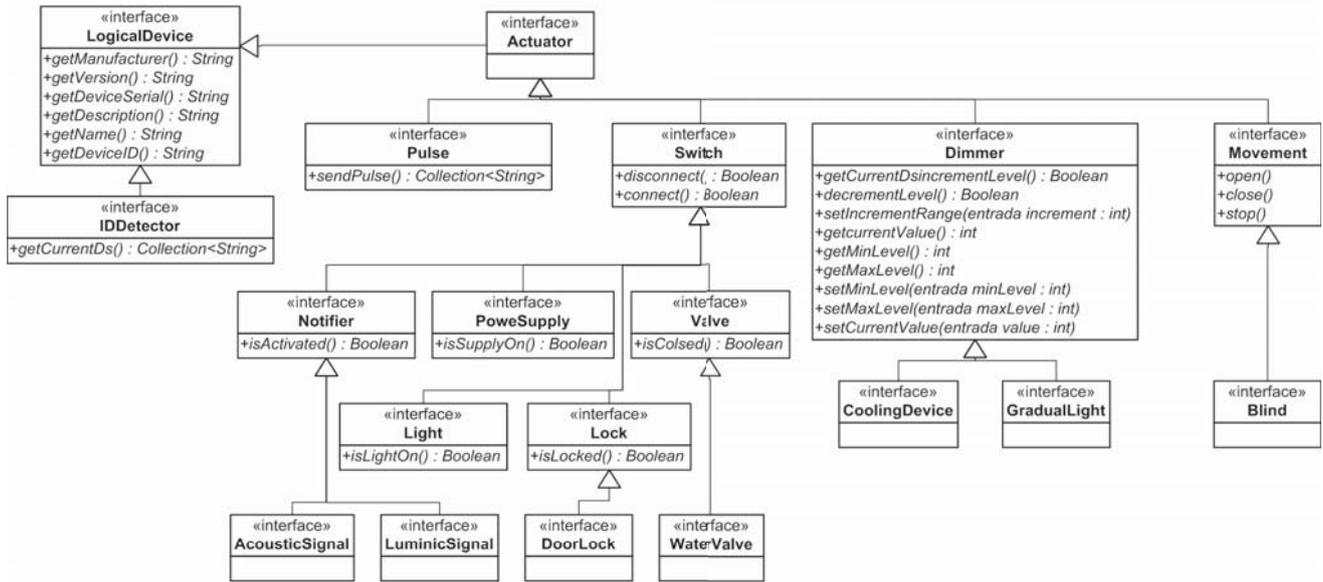


Figure 2: Device Abstraction Class Model

[6] P. Rashidi, D.J. Cook. *Keeping the resident in the loop: adapting the smart home to the user*, Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, vol.39, no.5, pp.949-959, 2009.

[7] M. Jahn, M. Jentsch, C.R. Prause, F. Pramudianto, A. Al-Akkad, R. Reiners. *The energy aware smart home*, 5th International Conference on Future Information Technology, pp. 1-8, 2010.

[8] C.D. Kidd. *The Aware Home: A living laboratory for ubiquitous computing research*, In Proc. The Second International Workshop on Cooperative Buildings - CoBuild, vol. 99, pp. 93-95, 1999.

[9] S.S. Intille, K. Larson, J.S. Beaudin, M. Tapia, P. Kaushik, J. Nawyn, T.J. Mcleish. *The PLACELAB, A live in laboratory for pervasive computing research (video)*, 2005.

[10] T.Weis, M.Knoll, A.Ulbrich, G.Muhl, A.Brandle. *Rapid prototyping for pervasive applications*, Pervasive Computing, IEEE, vol.6, no.2, pp. 76-84, 2007.

[11] D.Bannach, P.Lukowicz, O.Amft. *Rapid prototyping of activity recognition applications*, Pervasive Computing, IEEE, vol.7, no.2, pp. 22-31, 2008.

[12] C.da Costa, A.Yamin, C.Geyer. *Toward a general software infrastructure for ubiquitous computing*, Pervasive Computing, IEEE, vol.7, no.1, pp. 64-73, 2008.

[13] Román, M., Hess, C., Cerqueira, R., Campbell, R. H. *A middleware infrastructure for active spaces*, Pervasive Computing, IEEE, 1(4), 74-83. IEEE, 2002.

[14] Bardram, J. E. *Design, implementation, and evaluation of the Java Context Awareness Framework (JCAF)*, Context, (April), 1-14. Technical Report CfPC, 2005.

[15] Messer, A., Kunjithapatham, A., Sheshagiri, M., Song, H., Kumar, P., Nguyen, P., Yi, K. H. *InterPlay: A middleware for seamless device integration and task orchestration in a networked home*, Fourth Annual IEEE International Conference on Pervasive Computing and Communications PERCOM06, 2(1), 296-307, 2006.

[16] Brumitt, B., Meyers, B., Krumm, J., Kern, A., Shafer, S. *EasyLiving: technologies for intelligent environments*, Handheld and Ubiquitous Computing, 1927/2000, 12-29. Springer, 2000.

[17] Kameas, A., Mavrommati, I. *Computing in tangible: using artifacts as components of ambient intelligence environments*, Intelligence, 121-142, 2005.

[18] Ballagas, R., Szybalski, A., Fox, A. *Patch panel: enabling control-flow interoperability in UbiComp environments*, Event (London).

[19] Adlesee, M., Curwen, R., Hodges, S., Newman, J., Steggle, P., Ward, A., Hopper, A. *Implementing a sentient computing system*, Computer, 2001.

[20] D.Cook, S.Das. *How smart are our environments? An updated look at the state of the art*, Pervasive and Mobile Computing, vol.3, no.2, pp. 53-73, 2007.

[21] S.Das, D.Cook. *Designing smart environments: A paradigm based on learning and prediction*, Mobile, Wireless, and Sensor Networks, 2005.

- [22] Li, Jun, Bu, Yingyi, Chen, Shaxun, Tao, Xianping, Lu, Jian. *FollowMe: on research of pluggable infrastructure for context-awareness*, 20th International Conference on Advanced Information Networking and Applications - Volume 1 (AINA'06), pp. 199–204, 2006.
- [23] http://thewiki4opentech.org/index.php/OSAmI-ES_Device_Abstraction_HowTo.
- [24] I. Witten, E.Frank. *Data Mining: practical machine learning tools and techniques*, Morgan Kaufmann, 2005.