

# Resumen

Hoy en día los avances en la miniaturización de sistemas electrónicos han impulsado el desarrollo de dispositivos o artefactos que incorporan capacidades computacionales y de comunicación. Estos dispositivos pueden proveer de una serie de servicios en diferentes entornos gracias a su tecnología empotrada, como por ejemplo: reconocimiento de personas, localización de usuarios en un entorno, ajuste automático de la temperatura e iluminación de un entorno, etc. Este potencial de procesamiento y comunicación, está permitiendo crear nuevas aplicaciones distribuidas, donde el rol principal no lo tiene el computador personal, sino los diferentes dispositivos empotrados en el entorno: sensores, interfaces, actuadores, teléfonos móviles, etc., Esto ha generado novedosas áreas de aplicación como: Internet de Cosas (Internet of Things), Computación Móvil, Redes de Sensores, Sistemas Ubícuos, Inteligencia Ambiental, etc.

Estos avances han conducido al desarrollo de un nuevo paradigma, *computación orientada a la interacción*, es decir, la computación ocurre a través de los actos de comunicación entre las entidades. Por lo tanto, es lógico pensar que este paradigma requiere, desde un punto de vista de diseño, el desarrollo de aplicaciones en diferentes plataformas de software y de hardware, debido a lo heterogeneo de los sistemas de computación, lenguajes, sistemas operativos, y objetos (dispositivos físicos: sensores, actuadores, interfaces, etc.) dispersos en el entorno. Dicha heterogeneidad presente en los sistemas ubícuos, representa todo un reto a la hora de diseñarlos.

La ingeniería de software basada en sistemas multi-agente, en particular, los sistemas multi-agente abiertos (como las Organizaciones Virtuales), tiene la capacidad de abordar los retos al diseñar sistemas ubícuos. A ello hay que unir, que varias metodologías de desarrollo de software han adoptado el enfoque “dirigido por modelos” (*model-driven*) para realizar el análisis y el diseño del software. Dicho enfoque puede ser adoptado en los sistemas multi-agente, para mejorar el proceso de desarrollo y la calidad del software basado en agentes. Así mismo, el desarrollo dirigido por modelos proporciona un soporte apropiado para abordar este tipo de sistemas, ya que nos permite el uso de modelos como principal elemento abstracto para el diseño del sistema, por medio de la interconexión de un

## XVIII

conjunto de componentes visuales.

En este trabajo proponemos el desarrollo de sistemas ubícuos utilizando una organización virtual, creando una *Organización Virtual Ubícua*, la cual es diseñada usando el enfoque de desarrollo dirigido por modelos. De forma más detallada, este trabajo presenta tres propuestas. La primera, presenta un conjunto de meta-modelos para diseñar una *Organización Virtual Ubícua*, llamado  $\pi$ VOM, que utiliza conceptos generales que se abstraen de las metodologías y de las plataformas de agentes, lo permite diseñar aplicaciones utilizando abstracciones generales de alto nivel, evitando los detalles de implementación de bajo nivel. Así mismo, se presentan dos modelos de transformaciones, que permiten obtener el modelo de implantación de la organización (con los agentes, entidades y dispositivos), por medio de transformaciones semi-automáticas dadas por la metodología *model-driven*, reduciendo la brecha entre las fases de diseño y de implementación para este tipo de sistema.

La segunda propuesta presenta una arquitectura de implantación que define una estructura de capas funcionales basada en servicios, que soporta la interacción de las entidades de la organización virtual. La arquitectura de implantación permite la interoperabilidad de diferentes entidades, plataformas de software y hardware, proporcionando a los miembros de la organización virtual la capacidad de administrar y controlar los dispositivos del entorno (del sistema ubícuo).

La tercera propuesta presenta una plataforma de ejecución de agentes empujados llamada ANDROMEDA, que permite ejecutar agentes empujados sobre sistema operativo *Android* que cumplen con el modelo de agente de  $\pi$ VOM. Los agentes en ANDROMEDA pueden acceder a los dispositivos del entorno, tal y como sucede en los sistemas ubícuos.

Las propuestas presentadas fueron evaluadas empíricamente con dos ejemplos, que permiten mostrar sus bondades.