

# Interconectando sistemas de Domótica

Ing. Jorge Dignani – UNPSJB Sede Puerto Madryn y Cenpat Conicet

Ing. Santiago Drangosch – Universidad Católica Argentina

## Resumen

Desde los años 70, han aparecido diferentes ideas y tecnologías para el control inteligente de dispositivos eléctricos/electrónicos de una casa ó edificio. Este proyecto busca construir un puente entre los diferentes standards, de forma que se pueda utilizar una mezcla de las tecnologías existentes mejorando el desempeño global.

## 1 - Introducción

Domótica es la aplicación de las tecnologías de computación y robótica con el fin de controlar y automatizar determinados aspectos de edificios y casas.

Es el uso de una red inteligente de dispositivos electrónicos que, mediante programas, monitorean y controlan los dispositivos eléctricos de una casa o edificio. [1]

Existe una gran variedad de standards, cada uno con sus particularidades en cuanto a medios físicos de comunicación, funcionalidad, protocolos de comunicación, dispositivos existentes, etc.

Algunos ejemplos son:

- X10 – Uno de los standards más antiguos. Comunicación muy simple (ON – OFF – atenuadores de luces) a través de la línea de alimentación.[2]
- ZigBee – Protocolo inalámbrico optimizado para una baja tasa de transferencia y muy poco consumo de energía. Permite armar redes con varias topologías y generar perfiles de control de dispositivos [3].
- LonWorks – Desarrollo pensando en un ambiente distribuido. No requiere de un control centralizado, sino que los dispositivos se comunican en lo que sería una “red inteligente”. Basado en chips y transceptores, que simplifican la comunicación entre varios medios de transmisión diferentes [4].
- SpinOff – Sistema de control inalámbrico pensado para un sistema hogareño, basado en comunicación serial. No es un standard, sino un prototipo por los autores , partiendo de varios proyectos en los que se estaba trabajando.

Cada standard en el área de domótica presenta cualidades y ventajas únicas, y también desventajas muy propias y puntuales.

Asimismo, al elegir uno, se cierran automáticamente las puertas al resto, ya que la compatibilidad entre los mismos es casi inexistente.

Este proyecto busca intercomunicar protocolos de domótica en forma general, logrando sistemas compuestos por varios standards, donde se mitiguen las debilidades de cada uno y se potencien y aprovechen las fortalezas. También se busca poder acoplar nueva tecnología a lugares con sistemas más viejos y que se logre una interoperabilidad de forma transparente.

## **2 - Arquitectura**

El resultado de esta investigación es una arquitectura e implementación de la misma, a fin de poder interconectar sistemas de domótica de una forma sencilla, abierta y expansible.

Dicha arquitectura fue denominada Suricata Architecture y permite:

- *Aprovechar Instalaciones y sistemas existentes.*
- *Intercomunicar tecnologías en forma genérica.*
- *Visualizar los dispositivos independientemente de su tecnología.*
- *Expandir el control hacia servicios web y red celular.*
- *Proveer al usuario un sistema sencillo de control a través de elementos visuales (en desarrollo).*
- *Independizar el control y la comunicación del acceso físico a los sistemas.*

## **3 - Intercomunicación de Protocolos**

La fortaleza de una cadena, está determinada por el más débil de sus eslabones. De la misma forma, la complejidad de la comunicación entre sistemas estará dada por el más básico o restrictivo.

Al mismo tiempo, cada tecnología de domótica debe tener disponible, de alguna forma, una representación del sistema integrado como un todo. Debe poder visualizar o acceder a los dispositivos de otras tecnologías como si se tratara, simplemente, de uno más de los suyos.

Para este fin, se hace uso de dispositivos virtuales “fantasma”, que proveen a un dispositivo real de la tecnología A, una dirección en la tecnología B. En el momento requerido, automáticamente se rutea el mensaje y se realiza la traducción correspondiente, para que le llegue al dispositivo real.

Se utiliza una base de datos para mantener control de los dispositivos existentes y registrados, sus atributos, valores y tipos de datos.

## **4 - Modelo de Capas**

La arquitectura fue dividida en capas lógicas, envolviendo cada una cierto tipo de funcionalidad.

Consta de tres capas verticales necesarias al diseño, cortadas transversalmente por una cuarta capa de conexión a base de datos.

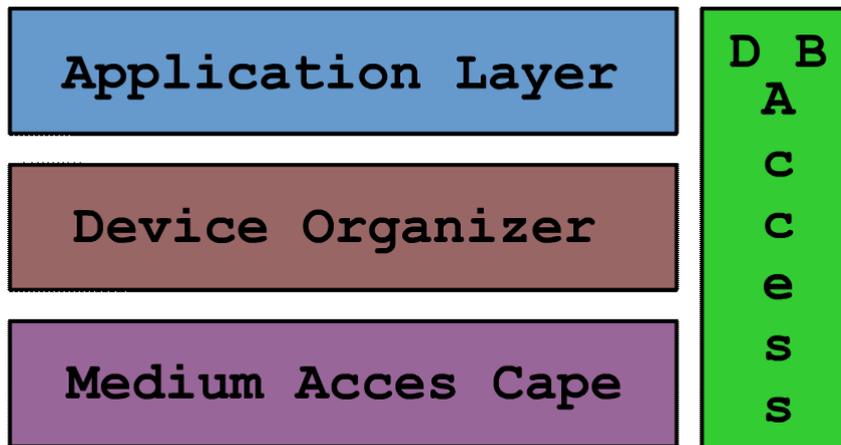


Figura 1: Capas de la Suricata Architecture

#### 4.1 - *Application Layer*:

Se encarga de interactuar con el usuario, permitiéndole visualizar estados globales del sistema y cambiarlos a gusto.

En este nivel se abstraen los dispositivos existentes de su tipo. Es decir todos son vistos como iguales.

#### 4.2 - *Device Organizer*:

Dispone de una estructura donde tiene almacenada una abstracción de cada dispositivo existente en el mundo físico.

Se conecta con las interfaces de acceso de las diversas tecnologías.

Conoce a los dispositivos como son realmente, a qué interface está conectado cada uno y cómo intercambiar mensajes entre ellos. Les provee además, direcciones fantasma en las tecnologías a las que no pertenecen.

#### 4.3 - *MACs - Medium Access Capes (en lugar de Medium Access Control)*:

Capa que combina obligatoriamente software y hardware. Trabaja sobre las capas de Aplicación de las tecnologías de domótica a las que se comunica. Traduce mensajes.

Esta es la única capa personalizable, que debe ser ajustada a cada tecnología en particular. Cuando se desee conectar nuevas tecnologías, simplemente se construye una interfaz con el medio físico y se programa la correspondencia funcional. La única restricción es respetar las interfaces de software definidas.

#### 4.4 - *DataBase Access*:

Los dispositivos y sus direcciones y estados son mantenidos en una base de datos relacional.

## 5 - Ruteo de Mensajes

La imagen que se incluyen a continuación, muestra en qué forma se mueven los mensajes dentro y a través de las capas del sistema.

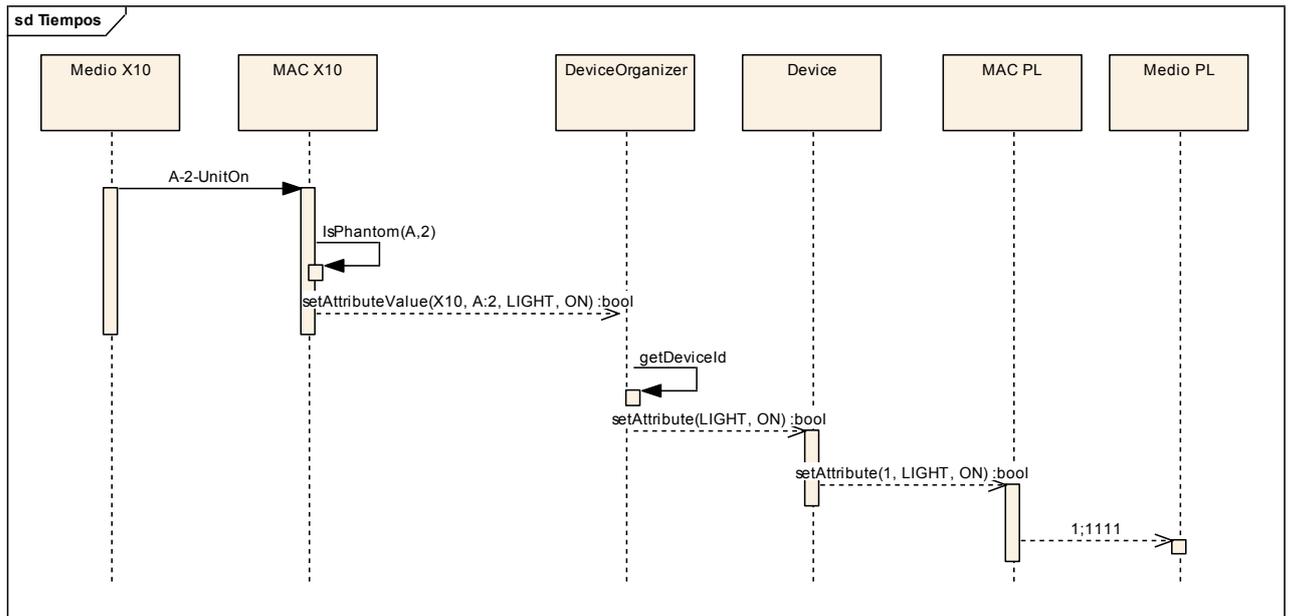


Figura 2: Diagrama de comunicación entre dos capas MAC

## 5 - Implementación

La arquitectura es independiente de plataforma y de lenguaje de programación utilizado. Es un diseño que podría ser implementado sobre distintas tecnologías.

El prototipo se implementó en una PC, utilizando C#.NET y una base de datos Microsoft SQL Server 2005. Los principales motivos de elección fueron:

- Experiencia en el uso.
- Velocidad de desarrollo.
- Componentes de ambiente distribuido: "Remoting".

Las MACs que se construyeron para este prototipo fueron con interfaces a:

- Simulador de standard X-10.
- Sistema de comunicaciones inalámbricas SpinOff, diseñado por este mismo equipo.

## 6 - Estado actual del proyecto:

Se han realizado pruebas piloto en la UCA y UNPSJB en una red que controla 15 dispositivos conectados a las capas MAC prototipo antes descritas.

El rendimiento es muy aceptable en lo que respecta a dispositivos con funciones de encendido y apagado solamente. Se quiere investigar si expandiendo el sistema a dispositivos que intercambian otro tipo de información (enteros, flotantes, etc.), la performance continúa intacta.

Para ello se está trabajando en el diseño y construcción de las MAC para el protocolo ZigBee, utilizando los módulos Picdem Z de Microchip.

Asimismo, se quiere avanzar con el diseño de una interfase que sea amigable al usuario y permita de una forma simple y visual, agregar, interconectar y controlar dispositivos.

## **7 - Referencias**

[1] Wikipedia. Home Automation [On-Line]. Citado diciembre de 2007. Contenido: Building Automation. Infrastructure. Protocols and Industry Standards. Disponible en Internet en: [http://en.wikipedia.org/wiki/Building\\_automation](http://en.wikipedia.org/wiki/Building_automation)

[2] X-10 Home Solutions. X10 Prowerline Carrier Technology [On-Line]. Citado octubre de 2006. Disponible en Internet en: <http://www.x10.com/support/technology1.htm>

[3] Microchip. AN965: Microchip Stack for the ZigBee Protocol, 2006 [On-Line]. Disponible en Internet en <http://www.microchip.com/>

[4] EBV Elektronik. An introduction to control networks based on LonWorks Technology, 2001 [On-Line]. Disponible en: <http://www.ebv.com>