

Tesis Doctoral

Diseño, Especificación, Validación y Aplicación de una Arquitectura modular de gestión de Redes Inalámbricas de Sensores



Autor: Salvatore Flavio Pileggi
Director: Dr. Carlos E. Palau Salvador

Marzo 2011

“ a mi chica Beatriz ”



Diseño, Especificación, Validación y Aplicación de una Arquitectura modular de gestión de Redes Inalámbricas de Sensores



Autor: **Salvatore Flavio Pileggi**

Director: **Dr. Carlos E. Palau Salvador**

ÍNDICE

Introducción. Metodología y Objetivos 15

1 Estado del arte de Arquitectura de Redes Inalámbricas de Sensores 23

1.1	Introducción	23
1.2	Tecnología básica	26
1.2.1	Hardware	27
1.2.1.1	Arquitectura tipo MOTE	27
1.2.1.2	Otras soluciones	28
1.2.2	Tecnología Inalámbrica	29
1.2.2.1	Tecnología basada en 802.15.4: ZigBee y Xbee	30
1.2.2.2	Soluciones basadas en 802.15.1: Bluetooth	31
1.2.2.3	Wireless HART	32
1.2.2.4	WiFi	32
1.3	Plataformas Software: Sistemas Operativos y Lenguajes de Programación	33
1.3.1	Sistemas Operativos	33
1.3.1.1	TinyOS	34
1.3.1.2	Otras soluciones	35
1.3.2	Lenguajes de Programación	36
1.4	Arquitectura de protocolos	37
1.4.1	Funcionalidades de comunicación básicas	37
1.4.2	Configuración básica o control de topología	38
1.4.3	Enrutamiento	39
1.4.3.1	Soluciones Energy-aware y Position-aware	40
1.4.3.2	Enrutamiento en redes móviles	41
1.4.4	Monitorización y control	42
1.4.5	Calidad de servicio y Seguridad	43
1.4.6	Configuración avanzada y Optimización	44
1.4.6.1	Clustering	45
1.4.7	Aplicación	47
1.5	Tecnología para Middleware	47
1.5.1	Arquitecturas orientadas a servicios: Web-Services	49

1.5.1.1	Entornos empotrados	51
1.5.2	Organizaciones Virtuales y Grid-Computing	53
1.5.2.1	Open Grid Service Architecture (OGSA)	54
1.5.2.2	Frameworks para desarrollo en tecnología Grid	55
1.5.3	Sistemas Multi-Agente	57
1.5.3.1	Negociación en Sistemas Multi-Agente	58
1.5.3.2	Plataformas Multi-Agentes	58
1.6	Soporte semántico en sistemas complejos	59
1.6.1	<i>Estándares para definición de esquemas semánticos</i>	61
1.6.1.1	RDF y RDF Schema	61
1.6.1.2	OWL	61
1.6.1.3	SWRL	62
1.6.2	<i>Servicios semánticos</i>	62
1.6.3	<i>Requisitos y limitaciones para tecnologías semánticas</i>	63
1.7	Modelo, Simulación y Evaluación	63
1.8	Entornos lógicos	64
1.8.1	<i>Recurso físico</i>	65
1.8.2	<i>Recurso lógico o Virtual</i>	66
1.8.3	<i>Sensor Web</i>	66
1.8.3.1	<i>Actuales límites del Sensor Web</i>	67
1.8.4	<i>Semantic Sensor Web</i>	68
1.8.4.1	<i>Interoperabilidad Semántica</i>	69
1.9	Limitaciones de las Redes Inalámbricas de Sensores	70
1.10	Lineas de investigación activas y futuro de las Redes Inalámbrica de Sensores	71
2	Descripción de la Arquitectura	75
2.1	Introducción	75
2.2	Visión general	77
2.3	Recursos Físicos	80
2.3.1	<i>Modelo de referencia de los recursos físicos</i>	80
2.3.1.1	<i>Modelo de dispositivos</i>	81
2.3.1.2	<i>Modelo de red</i>	81
2.3.2	<i>Arquitectura de protocolos de Redes Inalámbricas de Sensores</i>	82
2.3.2.1	<i>Evolución de la arquitectura</i>	82
2.3.2.2	<i>Arquitectura modular multi-dominio</i>	84
2.4	Recursos Lógicos o Virtuales	88
2.4.1	<i>Sistemas autónomos y Sistemas abiertos</i>	90
2.4.2	<i>Enfoque distribuido y Grid Computing</i>	92
2.4.2.1	<i>Evaluación preliminar de GT4</i>	93
2.4.2.2	<i>Ejemplo de infraestructura avanzada: monitorización de productos en aeropuertos</i>	96
2.5	Aplicación de la arquitectura en el Sensor Web	98
2.6	Soporte semántico y aplicación en el Semantic Sensor Web	100
2.6.1	<i>Interoperabilidad Semántica: Entornos de conocimiento y Actores Semánticos</i>	101

2.6.2	<i>Ingeniería del conocimiento Semántico: Ontología de Dominio y de Procesado de Datos</i>	103
2.6.2.1	<i>Ontología de dominio</i>	104
2.6.2.2	<i>Ambiente semántico orientado a eventos para procesado de datos</i>	109

3 Modelo, Simulación y Evaluación de Redes de Sensores Inalámbricas 115

3.1	Introducción	115
3.2	Modelo de Redes Inalambricas de Sensores	117
3.2.1	<i>Configuración lógica</i>	117
3.2.2	<i>Rutas</i>	120
3.2.2.1	<i>Rutas convergentes</i>	121
3.2.2.2	<i>Rutas divergentes y nodo a nodo</i>	124
3.2.3	<i>Cluster</i>	124
3.2.4	<i>Comunicación no ideal</i>	126
3.2.5	<i>Movilidad</i>	127
3.2.6	<i>Modelo de consumo energético</i>	130
3.3	Evaluación de Redes Inalámbricas de Sensores	130
3.3.1	<i>Entornos estáticos</i>	131
3.3.2	<i>Entornos móviles</i>	133
3.3.3	<i>Análisis de consumo energético</i>	135

4 Implementación y Aplicaciones 141

4.1	Introducción	141
4.2	Implementación del Framework: modulo CORE	142
4.2.1	<i>Funcionalidades extendidas de acceso al medio</i>	143
4.2.2	<i>Protocolo de configuración</i>	146
4.2.3	<i>Protocolo de enrutamiento</i>	148
4.2.4	<i>Reinforcement</i>	150
4.2.5	<i>Mecanismo de monitorización y control</i>	151
4.3	Aplicaciones	153
4.3.1	<i>Plataforma para adquisición de datos</i>	154
4.3.1.1	<i>Estructura de la plataforma para adquisición de datos</i>	155
4.3.1.2	<i>Mecanismos avanzados</i>	158
4.3.1.3	<i>Red de sensores</i>	158
4.3.2	<i>Multi-Mode WSN</i>	160
4.3.2.1	<i>AC y ABC en Redes inalámbricas de Sensores</i>	160
4.3.2.2	<i>Multi-Mode Wireless Sensor Node</i>	163
4.3.2.3	<i>La plataforma hardware</i>	164
4.3.2.4	<i>La interfaz de programación</i>	165
4.3.2.5	<i>La plataforma software</i>	168
4.3.2.6	<i>Evaluación preliminar</i>	170
4.3.3	<i>Convergencia entre WSN y tecnología RFID: aplicaciones industriales</i>	170
4.3.3.1	<i>Tecnología RFID y su convergencia con redes inalámbricas de sensores</i>	171
4.3.3.2	<i>Arquitectura para identificación y localización de objetos en entornos industriales</i>	173
4.3.3.3	<i>Pruebas de detección de RSSI</i>	178

4.3.4	<i>Redes inalámbricas de sensores móviles: entornos vehiculares</i>	181
4.3.4.1	<i>Redes de sensores inalámbricos en entornos vehiculares: potencialidades y modelos de negocios</i>	182
4.3.4.2	<i>Arquitectura de redes de sensores para entornos vehiculares</i>	183
4.3.4.3	<i>Particularización del framework</i>	184
4.3.4.4	<i>Limitaciones actuales</i>	185
4.3.5	<i>Integración de sensores en sistemas de mando y control</i>	187
4.3.5.1	<i>Sistemas de mando y control: situation awareness</i>	187
4.3.5.2	<i>Integración de redes inalámbricas de sensores</i>	188
4.3.5.3	<i>El nivel de presentación</i>	189

Conclusiones y Trabajo Futuro.....195

Anexo A: Contribuciones.....199

Anexo B: Implementación de un Actor Semántico 201

Anexo C: Ambientes de Simulación.....203

Referencias.....209

INDICE DE FIGURAS

Figura 1.1.a: Resumen de las aplicaciones de redes de sensores inalámbricos. Figura original de [Cha1].

Figura 1.1.b: Visión simplificada de los aspectos que requieren mejoras significativas en aplicaciones reales (*OnWorld report*). Figura original de [Cha1].

Figura 1.2: Ejemplos relevantes de arquitecturas de redes de sensores inalámbricos.

Figura 1.3: Unas de las más relevantes plataformas comerciales.

Figura 1.4: Comparativa de la robustez de diferentes tecnologías inalámbricas. Figura original de [Fer1].

Figura 1.5: Ejemplo de modelo de referencia para la pila de protocolos de una red de sensores inalámbricos.

Figura 1.6: Ejemplo de protocolo de enrutamiento distribuido (figura original de [Con1]) y centralizado.

Figura 1.7: Ejemplo de configuración en clusters.

Figura 1.8: Arquitectura protocolar de una *Open Grid Service Architecture* (OGSA). Figura original de [Fos2].

Figura 1.9: Esquema básico de la arquitectura de GT4. Figura original de [Fos3].

Figura 1.10: Entornos lógicos para arquitecturas de redes inalámbricas de sensores.

Figura 2.1: Esquema representante los posibles entornos lógicos para una arquitectura de sensores.

Figura 2.1.1: Izquierda: Visión general de la Arquitectura (Figura original de [Con1]). Derecha: Visión de la Arquitectura basada en clusters (Figura original de [Con4]).

Figura 2.2: Visión esquemática de la arquitectura.

Figura 2.2.1: Entorno lógico: *Physical Resource*.

Figura 2.3: Modelo con reserva de recursos. Figura original de [Pos2].

Figura 2.4: Arquitectura modular multi-dominio (Figura original de [Con9]).

Figura 2.4.1: Entorno lógico: *Virtual Resource*.

Figura 2.5: Visión de un sistema autónomo con acceso a múltiples recursos físicos.

Figura 2.6: Visión de un sistema abierto donde diferentes recursos físicos interactúan entre ellos o con aplicaciones web-centric.

Figura 2.6.1: Tiempos de respuesta de servicios Grid con estado para las diferentes operaciones básicas (red local).

Figura 2.6.2: Tiempos de respuesta de servicios Grid con estado para las diferentes operaciones básicas (red externa).

Figura 2.6.3: Visión global de la arquitectura para monitorización de productos en aeropuertos (Figura original de [Con3]).

Figura 2.6.4: Gestión del estado de los diferentes servicios de la infraestructura Grid (Figura original de [Con3]).

Figura 2.6.5: Entorno lógico: *Sensor Web*.

Figura 2.7: Visión de un Sensor Web y posible rol de las tecnologías semánticas (Figura original de [Cha1]).

Figura 2.7.1: Entorno lógico: *Semantic Sensor Web*.

Figura 2.8: Interacción semántica entre redes de sensores (Figura original de [Con11]).

Figura 2.8.1: Enlace lógico entre conceptos semánticos (Figura original de [Con11]).

Figura 2.9: Estructura lógica del actor semántico (Figura original de [Con11]).

Figura 2.10: Estructura lógica de la ontología de dominio propuesta (Figura original de [Con11]).

Figura 2.11: Visión la ontología de dominio propuesta empleando Protege (Figura original de [Con10]).

Figura 2.12: Relaciones lógicas fundamentales de la ontología de dominio (Figura original de [Con10]).

Figura 2.13: Ejemplos de implementación de conceptos semánticos en OWL (Figura original de [Con10]).

Figura 2.14: Validación de la Ontología utilizando *OwlSight* (Figura original de [Con10]).

Figura 2.14.1: Capas funcionales que componen un sistema de procesamiento de datos en tiempo real. Figura original de [Con12].

Figura 2.15: Estructura lógica de la Ontología de Proceso. Figura original de [Con12].

Figura 2.16: Visión en Protege de la jerarquía de conceptos de la Ontología de Proceso. Figura original de [Con12].

Figura 3.1: Configuración lógica (teórica) de un cluster (Figura original de [Con4]).

Figura 3.2: Posible configuración en entornos reales (derecha) a frente de la configuración ideal representada a la derecha (Figura original de [Con4]).

Figura 3.3: Ejemplo de ruta directa.

Figura 3.4: Ejemplos de rutas paralelas.

Figura 3.5: Ejemplo de ruta indirecta limitada.

Figura 3.6: Estrategia de enrutamiento nodo a nodo.

Figura 3.7: Configuración a paridad de condiciones lógicas.

Figura 3.7.1: Disposición simétrica de estación base en un área.

Figura 3.8: Estadísticas de conexión relacionadas con el despliegue aleatorio de nodos estáticos con número variable de estaciones base dispuestas simétricamente. El número de nodos es variable así como el rango de comunicación y el número de estaciones base. Cada valor reportado es el promedio de 100 experimentos independientes.

Figura 3.9: Estadísticas de configuración en clusters relacionadas con el escenario estático.

Figura 3.10: Estadísticas de conectividad relacionadas con el escenario móvil. Se considera una topología con 5 estaciones base. Cada valor reportado es el promedio de 100 experimentos independientes.

Figura 3.11: Configuración en clusters relacionadas con el entorno móvil. Se consideran rangos de comunicación de 20 metros (a) y de 30 metros (b).

Figura 3.12: Estadísticas de conexión en función de la velocidad de los nodos. Se consideran rangos de comunicación de 20 metros (a) y de 25 metros (b).

Figura 3.13: Topología y conectividad entre nodos para el análisis de consumo energético. Figura original de [Con1].

Figura 3.14: Valor medio del número de saltos de las rutas de comunicación. Figura original de [Con1].

Figura 3.15: Transmisiones por nodo (izquierda) y valor medio de transmisiones por nivel. Figura original de [Con1].

Figura 4.1: Modulo *CORE* en el modelo de referencia.

Figura 4.2: Funcionalidades extendidas de acceso al medio en el modelo de referencia.

Figura 4.2.1: Típico escenario de colisión de mensajes en WSN. Figura original de [Con5].

Figura 4.2.2: Colisiones en función de k utilizando el modelo con slot aleatorio y la topología de referencia representada en Figura 4.2.4. Figura original de [Con5].

Figura 4.2.3: Modelo con slot temporal predefinido.

Figura 4.2.4: Topología de referencia para la evaluación de los mecanismos extendidos de acceso al medio.

Figura 4.3: Protocolo de configuración en el modelo de referencia.

Figura 4.3.1: Diagrama de actividad del protocolo de configuración. Figura original de [Con4].

Figura 4.4: Protocolo de enrutamiento en el modelo de referencia.

Figura 4.4.1: Protocolos de enrutamiento.

Figura 4.5: Mecanismos de *Reinforcement*.

Figura 4.5.1: Mecanismo de monitorización y control en el modelo de referencia.

Figura 4.6: Sistema autónomo de adquisición de datos (izquierda) y sistema empotrado de adquisición de datos (derecha).

Figura 4.7: Sistema distribuido de adquisición de datos.

Figura 4.8: Interfaz para monitorización y control local.

Figura 4.9: Dispositivos *MicaZ* (izquierda) e *IRIS* (derecha).

Figura 4.10: Dispositivos para conexión con estaciones base: *MIB520* (izquierda), *MIB600* (centro) y *MIB510* (derecha).

Figura 4.11: Tarjetas de adquisición de datos: *MTS101* (izquierda), *MTS420* (centro) y *MDA300* (derecha).

Figura 4.12: Dos ejemplos de despliegue aleatorio con diferente grado de conectividad respecto a la estación base (Figura original de [Con9]).

Figura 4.13: Fallo de un enlace (Escenario a); despliegue aleatorio ineficiente (Escenario b); intento de reconfiguración de un modo multi-modo inteligente (Escenario c). Figura original de [Con9].

Figura 4.14: Arquitectura de *MM-Node*.

Figura 4.15: Componentes hardware de *MM-Node* (a la izquierda). *Adapter* (en alto a la derecha) y *R-device* (en bajo a la derecha).

Figura 4.16: Modelo teórico de gestión de recursos a eventos de *MM-Node* (Figura original de [Con8]).

Figura 4.17: Emulador para prueba y validación de la API desarrollada.

Figura 4.18: Diagrama de actividad simplificado de ACP. Figura original de [Con9].

- Figura 4.19:** Ejemplo de etiquetas pasivas (Figura original de Con[8]).
- Figura 4.20:** Visión esquemática de la área industrial de AERNOVA (Figura original de Con[8]).
- Figura 4.21:** Componentes de la arquitectura (Figura original de Con[8]).
- Figura 4.22:** Estructura lógica de la arquitectura (Figura original de Con[8]).
- Figura 4.23:** Composición de celdas básicas. Figura original de [Con3].
- Figura 4.24:** Valores medios de RSSI detectados en entornos indoor y outdoor. Figura original de [Con8].
- Figura 4.24.1:** Posiciones de referencia para la detección de RSSI en áreas regulares.
- Figura 4.24.2:** RSSI en función de la distancia media.
- Figura 4.24.3:** Valores medios de la relación entre RSSI y distancia.
- Figura 4.25:** Redes inalámbricas de sensores en entornos vehiculares.
- Figura 4.26:** Arquitectura para redes inalámbricas de sensores en entornos vehiculares. Figura original de [Con5].
- Figura 4.27:** Particularización del framework (Figura original de [Con5]).
- Figura 4.28:** Entornos de simulación (Figura original de [Con5]).
- Figura 4.29:** Posible arquitectura de un sistema de mando y control (Figura original de [Isr1]).
- Figura 4.30:** Sistema de adquisición de datos y puesto de mando local (estación base).
- Figura 4.31:** Nodo sensor geo-referenciado (izquierda) y nodo sensor no geo-referenciado (derecha).
- Figura 4.32:** Ejemplos de representación de datos geo-referenciados.
- Figura 4.33:** Ejemplos de representación de datos no geo-referenciados asociados a una unidad de mando y control local.
-
- Figura C.1:** Resumen de los resultados obtenidos.
-
- Figura AB.1:** Métodos principales de la interfaz proporcionada.
- Figura AB.2:** Interfaz usuario.
-
- Figura AC.1:** Módulos funcionales que componen los ambientes de simulación.
- Figura AC.2:** Interfaces de usuario principales para el ambiente con actores estáticos (arriba) y móvil (abajo).
- Figura AC.3:** *ToolBar*.
- Figura AC.4:** Ejemplo de despliegue aleatorio con diferentes números de estaciones base dispuestas simétricamente..
- Figura AC.5:** Diferentes puntos de vista de la red.
- Figura AC.6:** Panel de estadísticas.
- Figura AC.7:** Panel de estadísticas.

INDICE DE TABLAS

Tabla 2.1: Conceptos inferidos de la ontología de dominio (Tabla original de [Con10]).

Tabla 4.1: RCF 1662 (Tabla original de [Con8]).

Tabla 4.2: API para el acceso a las funcionalidades del componente activo gemelo en MM-Node (Tabla original de [Con9]).

Introducción. Metodología y Objetivos

Durante los últimos años las redes de sensores inalámbricas han sido objeto, como consecuencia de un creciente interés comercial, de una intensa actividad de investigación que ha determinado relevantes avances tanto en la tecnología base como en los aspectos de ingeniería a todos los niveles.

Las redes de sensores inalámbricas [Aky1] se basan en el concepto de nodo sensor autónomo de bajo coste que proporciona recursos limitados en términos de cálculo y capacidad de almacenamiento de información, baja potencia de transmisión y sensorica avanzada. Se caracterizan por el tamaño extremadamente reducido y una ingeniería orientada a la eficiencia energética (normalmente las plataformas se gestionan según un paradigma orientado a eventos que asume la permanencia del dispositivo en un estado de bajo consumo durante gran parte de su ciclo de vida) y al networking. La transmisión y la recepción suelen considerarse operaciones críticas en términos de recursos energéticos y, por lo tanto, aconsejan la utilización de tecnologías inalámbricas de comunicación de baja potencia expresamente diseñadas para soportar eficientemente el tráfico a ráfagas de pequeñas cantidades de datos en entornos de comunicación multi-hop.

El campo de estudio comúnmente denominado “redes de sensores inalámbricas” es un sector científico multidisciplinar en el que convergen aspectos (hardware, software y de ingeniería de redes) profundamente diferentes y caracterizados por sus retos, posibilidades y limitaciones [Tse1][Til1].

A pesar de la disponibilidad de soluciones altamente avanzadas, caracterizadas por la eficiencia y la flexibilidad, la difusión comercial masiva se ha planteado más veces como hipótesis plausible y además parece tardar en concretarse de forma definitiva. Las

principales causas están relacionadas, directamente o indirectamente, con dos factores: coste elevado y falta de suficiente fiabilidad/robustez. El principio de “bajo coste”, ha demostrado ser relativo y en la mayor parte de los casos, el coste real de las arquitecturas es con diferencia superior a lo esperado debido a la necesidad de disponer de plataformas (hardware, software, mecanismos, protocolos, etc.) a medida. Por otra parte, la mayoría de los experimentos relacionados con redes de sensores inalámbricas se limitan a experiencias de laboratorio o a experimentación supervisada y, especialmente enfocada al estudio del crecimiento de la escala de la red. En muchos casos carecen de fiabilidad y robustez implicando una continua (y costosa) intervención humana, que en un entorno de operación real podría resultar difícil, y, en algunos casos, imposible.

La obtención de resultados inferiores a los esperados está influyendo notablemente sobre la actividad investigadora, que sigue siendo intensa pero no (casi) exclusivamente centrada en la producción de soluciones “de laboratorio” como hasta fechas recientes. Los nuevos trabajos proponen extraer el máximo provecho de los continuos avances en la tecnología básica para incrementar la aplicabilidad de las arquitecturas en el mundo real mediante desarrollos significativos en temas de robustez, fiabilidad, tolerancia de fallos, seguridad y soluciones generalizadas caracterizadas por costes más razonables.

Otra consecuencia del desarrollo de arquitecturas “ad-hoc” que caracteriza actualmente las redes de sensores inalámbricas es la de garantizar una gran cantidad de óptimos locales siendo la causa principal de una preocupante ausencia de estándares tanto en términos de protocolos de comunicación como en términos de organización y representación de información. También nuevos modelos de negocio y de explotación dentro de las organizaciones virtuales de última generación son actualmente temas de atención en el seno de la comunidad científica internacional.

Este trabajo se sitúa en el marco de las últimas líneas de investigación orientadas a conciliar soluciones avanzadas, caracterizadas por una ingeniería innovadora, con su aplicación efectiva en el mundo real siguiendo principios de escalabilidad y manteniendo siempre el principio de bajo coste. Considerando la aplicación de la arquitectura, en términos no solamente tecnológicos si no también comerciales, como elemento primario y determinante, la solución propuesta en la presente tesis se caracteriza por su independencia respecto a un dominio aplicativo predefinido al proporcionar un marco flexible que puede ser particularizado en función del entorno, de la aplicación y relacionarlo con requisitos funcionales y no funcionales. La arquitectura propuesta, al contrario de la mayoría de las soluciones existentes, sería operativa tanto para entornos estáticos (o semi-estáticos) como para entornos caracterizados por la presencia de nodos y actores móviles.

Desde un punto de vista metodológico, se considera de gran importancia proporcionar un modelo teórico de referencia para diferentes aspectos de la arquitectura propuesta: modelo de conectividad-cobertura, *clustering* dinámico y comportamiento de los actores móviles. El objetivo principal de la modelización es proporcionar una sólida base teórica, independiente de implementaciones concretas, para el desarrollo de ambientes de simulación avanzados orientados a soportar tanto la fase de diseño previo como la fase de evaluación preliminar con especial atención a la escalabilidad.

Paralelamente, la implementación de la plataforma y la experimentación de la misma en entornos de trabajo reales proporcionan un feedback vital para el refinado de los modelos y de los ambientes de simulación (caracterizando y validando entornos, plataformas y aplicaciones mediante datos reales). Al mismo tiempo permite comparar las prestaciones reales con las esperadas evaluar el impacto de fenómenos impredecibles (interferencias temporáneas o permanentes, obstáculos, etc.).

Partiendo del modelo orientado a eventos que caracteriza la mayoría de los dispositivos inalámbricos de última generación, la hipótesis de partida es que los nodos sensores implementen mecanismos y protocolos complejos en el cual cada uno de los actores mejore la economía de la red como elemento activo e inteligente capaz de modificar, según un patrón adaptativo, su comportamiento en función de los eventos detectados respecto al entorno o en la red. Aspectos determinantes como la eficiencia, la fiabilidad y la tolerancia a fallos pueden resultar afectados. La aplicación de las redes de sensores inalámbricos está actualmente en auge tanto en entornos civiles como militares. Siendo un hecho y una opinión casi unánime el considerar dichas redes como uno de los entornos con mayor potencial y perspectiva comercial, en gran parte poco explotada, debido a las limitaciones técnicas de los sistemas actualmente en el mercado. Considerando el estado del arte global de las arquitecturas de redes sensores inalámbricos, los cinco ejes del desarrollo de la presente tesis serían:

- **Método “horizontal” en contraposición al tradicional método de estudio “vertical” que caracteriza la mayoría de los proyectos sobre redes de sensores inalámbricos.** El campo de estudio comúnmente definido como “redes de sensores inalámbricos” es un sector multidisciplinar por definición. Los proyectos de investigación suelen centrarse, de forma vertical, sobre un aspecto específico o, bien, solucionar de forma eficiente un problema concreto. En la tesis desarrollada, el área de estudio de mayor interés es la ingeniería de protocolos y mecanismos avanzados de comunicación y gestión. Considerando especialmente: tecnologías básicas de comunicaciones, elementos software básicos, tecnologías para middleware y algoritmos de clustering. El objetivo fundamental es proporcionar, dada una tecnología básica de referencia, una solución integral y generalizada para redes de sensores inalámbricos que no solucione de forma cerrada un problema concreto si no que pueda ser particularizada o integrada para garantizar unas altas prestaciones, fiabilidad y robustez en cualquier dominio de aplicación o, más concretamente, en gran parte de los dominios de aplicación de redes de sensores inalámbricos.

Bajo este punto de vista se consideran resultados relevantes todos aquellos modelos, mecanismos, protocolos y marcos de referencia que no hacen asunciones sobre aspectos relacionados con aplicaciones específicas y que pueden ser particularizados y/o integrados en función del entorno de aplicación para garantizar eficiencia y calidad de servicio. La mayor dificultad en este sentido es representada por las profundas diferencias que pueden existir entre ellos (por ejemplo entornos estáticos y móviles, aplicaciones reactivas o para adquisición periódica de datos, etc.).

- **Ingeniería orientada a la reducción de los costes a través de soluciones flexibles multi-dominio.** El ciclo de vida para el desarrollo de aplicaciones basadas en redes de sensores se compone de una fase previa de diseño; una fase de implementación y una fase de experimentación preliminar. El éxito de esta última fase determina la experimentación del prototipo en contextos reales; los datos proporcionados constituyen una información vital para modificar el prototipo en función de los resultados conseguidos respecto a los esperados. Este ciclo puede repetirse un número importante de veces y suele ser costoso en términos de tiempo y recursos.

Ninguna de las fases precedentes es prescindible pero el proceso puede ser optimizado si la aplicación a desarrollar se diseña y posteriormente se desarrolla sobre la base de un marco de soporte; dicho marco proporciona, al mismo tiempo, un ambiente de simulación completo y extensible y la posibilidad de desarrollar la aplicación deseada como particularización de la plataforma básica proporcionada, en

función de las exigencias o requisitos concretos de acuerdo a varias filosofías de enrutamiento, de control y gestión.

- **Elaboración de aspectos teóricos con sucesiva evaluación experimental de los modelos propuestos en varios entornos de aplicación.** La experimentación previa del marco de trabajo en el ámbito de esta tesis proporcionará datos suficientes para caracterizar entornos específicos, fenómenos ambientales indeseados y patrones de comportamiento complejos para los nodos respecto a su posición (movilidad) o a su funcionamiento (fallos, etc.). La capacidad de análisis resulta por lo tanto incrementada respecto a la capacidad proporcionada por las solas medidas experimentales y puede efectuarse de acuerdo a niveles de abstracción creciente. La fase de experimentación pretende proporcionar una realimentación para el refinamiento de los modelos analíticos propuestos y, al mismo tiempo, experimentar y evaluar la plataforma en contextos reales. Asimismo, la resolución de problemas complejos concretos que requieren extensiones específicas del modelo básico se considerarán resultados relevantes en el contexto de la tesis como ejemplos de ingeniería de protocolos sobre un entorno básico abstracto.
- **Soporte para entornos estáticos, nomádicos y con nodos sensores/actores móviles.** La plataforma propuesta se adapta, indiferentemente, a entorno de funcionamientos que asumen nodos estáticos (los nodos no cambian su posición durante el tiempo de vida de la aplicación), a entornos nomádicos (los actores pueden cambiar su posición puntualmente durante el ciclo de vida de la aplicación o accidentalmente como consecuencia de agentes externos) y a entornos móviles (los nodos sensores o una parte de ellos pueden moverse, así como las estaciones base, e incluso se contempla una estructura mallada y/o adhoc). Al contrario, la mayoría de las soluciones corrientes suelen centrarse en entornos concretos. La plataforma a desarrollar en la tesis configura y gestiona las rutas de comunicación en función del entorno de ejecución: rutas con estado y refresco periódico en el caso de entornos estáticos; rutas con estado y refresco dinámico en caso de entornos semi-estáticos o estáticos con requisitos de fiabilidad en ambientes hostiles y rutas “*soft-state*” para entornos móviles.
- **Diseño de una arquitectura flexible y dinámica apta a soportar diferentes modelos de interacción e interoperabilidad en entornos aplicativos de diferente escala y complejidad.** Esta arquitectura pretende ser efectiva y funcional como sistema autónomo y, al mismo tiempo, poder trabajar de forma eficaz en entornos de escala creciente, de acuerdo a diferentes modelos de interacción e interoperabilidad (básica, funcional y semántica). Para poder alcanzar este objetivo se requiere un entorno funcional heterogéneo que incluye tecnologías orientadas a servicios, tecnologías Grid/Cloud y tecnologías Semánticas.

El trabajo propuesto se estructura en tres partes estrechamente relacionadas y que conforman la tesis doctoral desarrollada:

- La primera parte describe el diseño de la arquitectura propuesta. Desde un punto de vista estructural, dicha arquitectura consta de tres componentes fundamentales: redes de sensores, infraestructuras middleware y sistema de información. Desde un punto de vista conceptual, se estructura de acuerdo a cuatro conceptos progresivos: recursos físicos, recursos virtuales, Sensor web y, finalmente, Semantic sensor web.
- La segunda parte, con enfoque fundamentalmente teórico, se centra en la elaboración de un modelo analítico, tanto para entornos estáticos como nomádicos y móviles, de

la arquitectura definida. El conjunto de los modelos propuestos se caracteriza por su independencia respecto a aspectos específicos de dominios de aplicación concretos; este aspecto se considera absolutamente fundamental para proporcionar un enfoque general a los varios análisis y resultados relacionados. Dicho modelo es el fundamento teórico para el entorno de simulación propuesto. También se proporciona una introducción a los posibles modelos de análisis y evaluación de arquitecturas reales complejas.

- La tercera parte del trabajo de tesis, con carácter más aplicativo y experimental se centra en el desarrollo del framework de referencia propuesto y en su particularización e integración para soporte de aplicaciones concretas. Los dominios aplicativos objeto de interés en el ámbito del trabajo de tesis se caracterizan por requisitos funcionales complejos que determinan soluciones innovadoras y conceptualmente avanzadas. La arquitectura se ha implementado como se presentará en el capítulo correspondiente mediante la utilización de los nodos MicaZ/IRIS de Crossbow.

La tesis se compone de cuatro capítulos fundamentales: el primero describe las características fundamentales de las redes inalámbricas de sensores y el estado del arte de los principales elementos de interés (tecnología básica, protocolos y algoritmos, etc.); el segundo capítulo propone una descripción general de la arquitectura propuesta con el objetivo fundamental de describir las características de los diferentes actores componentes la arquitectura en términos de lógicos/funcionales e interacciones potenciales entre los mismos; el tercer capítulo propone el modelo analítico fundamental de la red, tanto por entornos estáticos como nomádicos y móviles, junto con el simulador desarrollado y los resultados obtenidos; finalmente, el cuarto capítulo describe los detalles de implementación más relevantes de los componentes fundamentales y la aplicación de la arquitectura propuesta, oportunamente integrada y/o particularizada, en entornos concretos complejos.

CAPÍTULO 1

1 Estado del arte de Arquitectura de Redes Inalámbricas de Sensores

Este Capítulo proporciona un estado del arte exhaustivo de las Redes Inalámbricas de Sensores. Este campo de estudio es, realmente, un sector científico multidisciplinar en el cual convergen aspectos (hardware, software y de ingeniería de redes) profundamente diferentes entre ellos, cada uno caracterizados por sus retos, posibilidades y limitaciones. Por lo tanto, el Capítulo resume las más relevantes soluciones para cada uno de los aspectos fundamentales que caracterizan a los sistemas basados en redes de sensores con especial énfasis en los aspectos objeto de este trabajo de tesis.

1.1 Introducción

Las redes de sensores inalámbricas [Aky1][Ily1] difieren con respecto a sensores autónomos en cuanto implícitamente asumen diferentes puntos de observación y la necesidad de organizarse en red para poder ejecutar tareas funcionales básicas (comunicación hacia el centro de control por ejemplo) o tareas cooperativas de nivel aplicación.

Dichas redes suelen basarse en el concepto de nodo sensor autónomo de bajo coste que dispone de recursos limitados en términos de cálculo y capacidad de almacenamiento de información, baja potencia de transmisión y recursos sensores variados. Se caracterizan por un tamaño extremadamente reducido y una ingeniería orientada a la eficiencia energética (normalmente las plataformas se gestionan según un paradigma orientado a eventos que supone que el dispositivo está en un estado de bajo consumo durante gran parte de su ciclo de vida) y al entorno de red. La transmisión (y la recepción) suelen considerarse operaciones críticas en términos de recursos energéticos y, por lo tanto, aconsejan la utilización de estándares de comunicación de baja potencia expresamente diseñados para soportar eficientemente el tráfico a ráfagas de pequeñas cantidad de datos en entornos de comunicación multi-salto. Aunque no sea habitual, se pueden encontrar diferentes

plataformas que difieren de forma importante respecto al modelo de bajo coste por una o más características (comunicación de largo alcance por ejemplo).

Las redes inalámbricas de sensores de bajo coste contienen implícitamente una (o más) estación base también denominadas sumideros (sink). Una estación base suele ser un dispositivo caracterizado por recursos más potentes respecto a los de un dispositivo sensor y suele actuar, respecto a la red circundante, como el punto de destino de toda la información generada y retransmitida por los nodos sensores. Una estación base que también proporciona funciones de pasarela para la información hacia centros de control o de mando remoto asume el rol de Gateway, cuando este dispositivo tiene interconexión a dos redes de comunicación diferentes. También existen otros actores con funciones específicas que pueden ser parte activa de la red: se consideran actores de interés todos aquellos dispositivos que pueden actuar o modificar su comportamiento a consecuencia de observaciones, medidas o comportamiento de componentes sensores. El estudio específico y profundizado de actores con funciones diferentes a la de estación base o pasarelas se considera fuera de los objetivos fundamentales de este trabajo de tesis.

El campo de aplicación de las redes de sensores inalámbricos es muy variado y extendido (Figura 1.1.a): tanto en el sector civil como el militar, para aplicaciones orientadas a: la adquisición de datos; monitorización y control de procesos; aplicaciones medicas y cuidado de salud; aplicaciones para seguridad; vigilancia y seguimiento; aplicaciones para agricultura y ganadería; automoción; control de tráfico; monitorización ambiental; de estructuras y de fenómenos naturales; entre otras muchas. En la Figura 1.1.b se propone una visión simplificada, proporcionada por OnWorld [Onw1], de las principales causas actualmente consideradas que limitan el empleo masivo de redes de sensores inalámbricos en el mundo real.



Figura 1.1.a: Resumen de las aplicaciones de redes de sensores inalámbricos. Figura original de [Cha1].

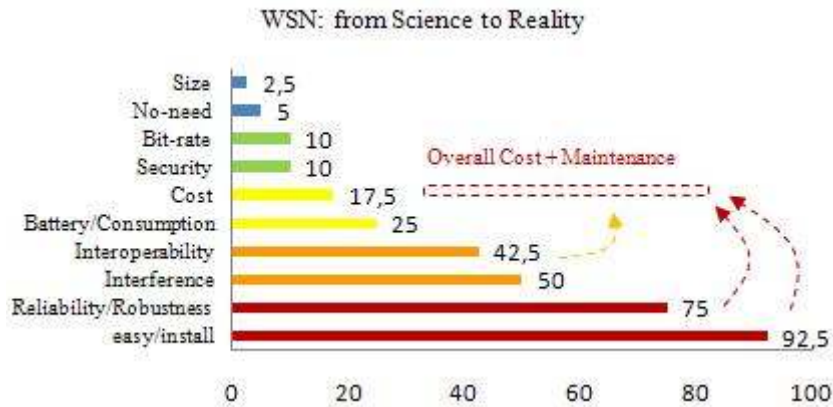


Figura 1.1.b: Visión simplificada de los aspectos que requieren mejoras significativas en aplicaciones reales (*OnWorld report*). Figura original de [Cha1].

Como se puede apreciar, hay un grado de satisfacción importante respecto a aspectos relacionados con el tamaño de los dispositivos, satisfacción parcial respecto al coste y a la autonomía de los dispositivos que se alimentan con baterías y absoluta disconformidad respecto al nivel de interoperabilidad; al coste de mantenimiento y a la fiabilidad de las soluciones proporcionadas. Este análisis se puede considerar como una referencia aceptable aunque, probablemente, no considerar la interdependencia entre los factores resulta en una simplificación del análisis respecto a la situación real.

Las arquitecturas más recientes asumen que las redes inalámbricas de sensores funcionan como servicio autónomo o, bien, en el contexto de organizaciones virtuales con una clara diferenciación entre recurso físico y lógico, así como servicios empotrados o de soporte. A estos aspectos los diseñadores, en el desarrollo de infraestructuras middleware eficientes y flexibles que puedan soportar modelos de interacción y gestión profundamente diferentes entre ellos.

Las prestaciones de las redes inalámbricas de sensores se degradan de forma creciente al aumentar la escala de la red. En parte, por sus limitaciones intrínsecas (*energy-hole*, recursos limitados, centralidad de la estación base) y en parte por la dificultad de garantizar requisitos no funcionales en entornos con muchos nodos. Las soluciones escalables propuestas suelen caracterizarse por arquitecturas de protocolos para configuración/gestión, control y enrutamiento bastante más extendidas y avanzadas respecto a las correspondientes destinadas a entornos de media/pequeña escala.

Independientemente de la escala de la red, la arquitectura básica de protocolos suele proporcionar las funciones mínimas (sensorización y comunicación) de la red de forma eficiente y fiable, junto con requisitos funcionales avanzados (auto-configuración, monitorización, etc.) así como la garantía de requisitos no funcionales (robustez, fiabilidad, seguridad, etc.). Incrementos significativos de la escala de la red implican restricciones topológicas (presencia simultánea de varias estaciones base) y mecanismos de configuración avanzada (por ejemplo *clustering*) para limitar la degradación de las prestaciones con relación al tamaño de la red.



Figura 1.2. Ejemplos relevantes de arquitecturas de redes de sensores inalámbricos actualmente empleadas en el mundo real.

Muchas de las arquitecturas experimentales diseñadas y desarrolladas en el ámbito de proyectos de investigación han demostrado una escasa aplicabilidad en el mundo real. En muchísimos casos los puntos de partida son poco realistas, las pruebas supervisadas y elementos fundamentales (económicos, por ejemplo) no se suelen tener en debida consideración. También la escala de trabajo de la red es poco realista en la mayoría de los casos. En la Figura 1.2 se representan 4 relevantes aplicaciones que concretamente trabajan en el mundo real. Como se puede apreciar, independientemente de la aplicación considerada, el tamaño máximo de la red no supera los 60 nodos sensores en contraposición a experimentos que asumen redes formadas por cientos y miles de nodos, que se han evaluado por simulación o de forma analítica.

Un aspecto relevante en el estudio y análisis de redes de sensores inalámbricos es tener en consideración que se trata, realmente, de un sector científico multidisciplinar en el cual convergen aspectos (hardware, software y de ingeniería de redes) profundamente diferentes entre ellos, cada uno caracterizados por sus retos, compensaciones y limitaciones. Los principales temas de interés desde un punto de vista científico y comercial se detallaran brevemente en las siguientes secciones centradas en tecnología básica (procesadores, memorias, tecnología inalámbrica), plataformas software (sistemas operativos, lenguajes de programación), pila de protocolos (funcionalidades básicas y avanzadas), middleware, soporte semántico, entornos lógicos y, finalmente, aspectos más teóricos (modelo, simulación y evaluación).

1.2 Tecnología básica

Tecnología básica es el término empleado para referir la ingeniería de los nodos inalámbricos en todos sus componentes: recursos de cálculo (hardware en general con particular enfoque a procesadores de bajo consumo y tamaño y memorias), tecnología inalámbrica de baja potencia (orientada a soportar eficientemente el tráfico, periódico o a

ráfagas, de pequeñas cantidades de datos), componentes sensores y soluciones para la alimentación de los nodos (baterías).

Los avances en los componentes sensóricos se consideran una de las claves para la utilización masiva de redes de sensores inalámbricas. Actualmente, existe una amplia gama de componentes de medida avanzados (temperatura, humedad, acústico, magnetómetro, acelerómetro, presencia, detectores químicos/biológicos, biomédicos, etc.). Más recientemente, también componentes avanzados como módulos GPS y cámaras están disponibles en el mercado en un contexto de relativo bajo coste [Cro1].

La mayoría de las redes inalámbricas emplean alimentación con batería; las baterías se consideran un elemento crítico tanto para los nodos sensores como para las estaciones base. Avances relevantes sobre baterías tradicionales que proporcionan prestaciones siempre superiores y se caracterizan por aproximaciones innovadoras (como baterías recargables a través de pequeños paneles solares por ejemplo) son novedades recientes, con el objetivo principal de incrementar el tiempo de vida de las redes y, consecuentemente, la competitividad de los productos en diferentes mercados.

1.2.1 Hardware

Un número creciente de plataformas hardware de bajo coste se encuentra disponible en el mercado. Todos los nodos comerciales se caracterizan por procesadores (FPGA, micro-controladores, micro-procesadores) de bajo consumo orientados a la máxima eficiencia energética. El concepto fundamental es la gestión del ciclo de vida del sensor que suele asumir una gestión orientada a eventos y que fuerza al dispositivo a trabajar durante la mayor parte del tiempo en un estado de consumo energético mínimo.

Las clases de memoria más comúnmente empleadas en estos sensores inalámbricos son las memorias integradas (volátiles) y flash (no volátiles); el tamaño de ambas se incluye en un rango de pocos Kbytes a unos Mbytes; las memorias no se suelen considerar componentes críticos en términos de coste de los nodos utilizados en este tipo de redes.

Otro elemento de seguro interés a efectos prácticos es representado por las fuentes de alimentación portátiles: se están proponiendo soluciones extremadamente avanzadas (alimentación mecánica o solar por ejemplo) como alternativa a las baterías tradicionales (siempre más avanzadas).

A continuación se realiza un breve análisis de las principales plataformas hardware. En la Figura 1.3 se representan las plataformas comerciales más relevantes proporcionadas por diferentes fabricantes.

1.2.1.1 Arquitectura tipo MOTE

Desde 1998, Crossbow [Cro1] es el mayor fabricante de nodos sensores y el mayor proveedor dentro de los principales grupos de investigación a nivel mundial, así como para aplicaciones reales desplegadas. Todas las plataformas propuestas por Crossbow se caracterizan por una ingeniería básica de tipo “MOTE”, diseñada y desarrollada por la Universidad de Berkeley, así como la plataforma software de gestión TinyOS [Tin1]. Crossbow también comercializa un conjunto completo de módulos sensores y dispositivos avanzados (GPS, Cámara) compatibles con los módulos inalámbricos proporcionados según

una política orientada a favorecer el diseño y desarrollo de prototipos sin tener que recurrir al costoso hardware a medida que suele caracterizar productos finales.

Actualmente la arquitectura de tipo MOTE incluye tres clases de dispositivos: nodos inalámbricos de bajo coste, nodos para control y nodos para aplicaciones avanzadas con requerimiento de recursos. El primer conjunto de soluciones (Mica2, MicaZ, Iris) se caracteriza por componentes de base en línea con el concepto típico de nodo sensor de bajo coste (memorias del orden de los kbytes, etc.). Los nodos para control (TelosB, por ejemplo) también se pueden considerar nodos de bajo coste; su peculiaridad es disponer de interfaces especiales (USB normalmente) para favorecer la comunicación directa con otros dispositivos electrónicos. Finalmente, nodos avanzados (IMOTE2) se caracterizan por tamaños de memorias del orden de los Mbytes y múltiples plataformas de gestión (TinyOS o .Net Micro Framework [Net1]); también monta un procesador con características superiores respecto a lo de los demás dispositivos de la familia; evidentemente esta clase de nodo está diseñada para soportar aplicaciones avanzadas (multimedia por ejemplo) que requieren alta capacidad de procesamiento local.

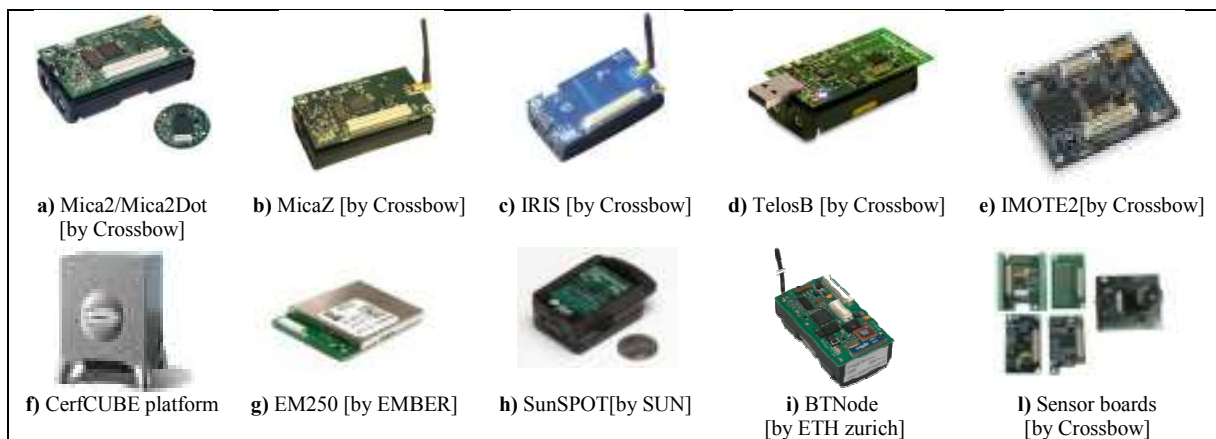


Figura 1.3. Plataformas comerciales más relevantes

1.2.1.2 Otras soluciones

Otras empresas han proporcionado plataformas similares a los MOTEs o, bien, con características diferentes, ya que no existe una arquitectura estándar y todos los sistemas son propietarios. Las principales soluciones son:

- *SENTILLA* (antes conocida como MoteIV) [Sen1] debe su popularidad principalmente a la comercialización de los nodos TMote, también diseñados de acuerdo a la arquitectura tipo MOTE.
- *SHOCKFISH* [Sho1] ha conseguido, durante los últimos años, una posición relevante en el mercado de los nodos sensores gracias a su política comercial que la sitúa como puente ideal entre mundo científico/académico y mundo real. Su mayor campo de acción es el entorno industrial por el cual ha diseñado y desarrollado la plataforma TinyNode, actualmente considerada una de las pocas alternativas validas a los nodos de tipo MOTE.
- *BTNode* [Btn1] fabrica nodos sensores desarrollados por ETH Zurich; actualmente, dichos componentes son a la base de varios experimentos y proyectos de notable interés científico y comercial.

- *Ember* [Emb1] es una de las mayores promotoras de ZigBee Alliance [Zig1]; la tecnología desarrollada, basada en comunicación ZigBee, es ideal para soluciones escalables de bajo consumo que requieren topologías en mallas.
- *Sun* [Sun1] propone una interesante solución (Sun SPOT [Sun2]) basada en el estándar físico IEEE 802.15.4 [Bou1][Zhe1] operativa sobre máquina virtual JAVA Squawk.
- *Nano-RK* [Nan1] ha desarrollado recientemente una solución de bajo coste y consumo (FireFly) con el objetivo primario de proporcionar pleno y eficiente soporte para servicios y aplicaciones de tiempo real.

1.2.2 Tecnología Inalámbrica

La tecnología inalámbrica es uno de los temas fundamentales para el efectivo desarrollo de nodos sensores avanzados aptos al funcionamiento en el mundo real. El entorno de comunicación para nodos sensores inalámbricos de bajo coste se puede caracterizar como un medio orientado a soportar la transmisión de pequeñas cantidad de datos, transmitidos de manera periódica o impulsiva [Iee2], utilizando transmisores de muy baja potencia en el contexto de redes data-centric [Pyo1]. Estos requisitos orientan la tecnología inalámbrica de referencia hacia tecnologías inalámbricas para redes de área personal (PAN).

En el reciente pasado, era bastante común el uso de transmisores con frecuencia entre 400 y 900Mhz (una frecuencia muy común, utilizada por ejemplo por la plataforma Mica2, era 868Mhz). Actualmente la tendencia más marcada propone frecuencias más altas (2.4Ghz). Una excepción significativa es representada por dispositivos que trabajan bajo el agua [Yan1] que suelen comunicar a través de tecnologías con frecuencias notablemente inferiores.

Hay dos soluciones concretas, en términos de tecnología inalámbrica, actualmente en el mercado para aplicaciones en redes PAN: Bluetooth [Blu1] y ZigBee [Zig1]. El primero se caracteriza por un mayor ancho de banda y consumo energético, el segundo por una mayor cobertura y menor tasa de transmisión. Evidentemente las soluciones tipo ZigBee se adapta con más naturalidad que Bluetooth a las redes de sensores inalámbricas, posiblemente más indicado para redes personales de pequeña escala a soporte de aplicaciones user-centric.

Una de las extensiones de las redes de sensores es para soporte de aplicaciones multimedia (Visual Sensor Networks) [Aky2] (nodos equipados con cámara, micrófono y/o otros sensores) que, posiblemente, podrían necesitar mayores capacidades, en términos de ancho de banda superiores a las proporcionadas por ZigBee. Esta clase de redes, aunque emergente, tiene gran importancia científica y de gran potencial comercial, no es de interés específico en este trabajo de tesis.

Tecnologías inalámbricas orientadas a proporcionar comunicación optimizada en entornos concretos (industriales por ejemplo) son objeto de estudio e investigación con el objetivo primario de definir nuevos estándares y tecnologías orientadas a dominios específicos; de momento no parecen proporcionar avances significativos en relación al coste. Algunos ejemplos son Wireless HART [Wir1] e ISA100 [Isa1].

Independientemente de la tecnología considerada, hay que tener en cuenta que un número importante de plataformas tan solo cumplen con una parte de los estándares de referencia, normalmente la capa física.

A continuación se proponen las principales características de las tecnologías ZigBee y Bluetooth. Soluciones con características tipo WiFi son de difícil aplicación a menos que en entornos muy específicos (aplicación de redes de sensores a redes vehiculares, en las que se están afianzando soluciones en sistemas caracterizados por escalas significativas); no son, por lo tanto, de interés específico en esta tesis. Como soluciones alternativas se propone Wireless HART. También se proporciona un breve análisis de la tecnología WiFi; esta tecnología no se adapta especialmente bien a las redes inalámbricas de sensores con nodos de bajo coste a causa de su consumo; de todos modos, hay que tener en cuenta que es una de las tecnologías más utilizada para comunicación inalámbrica entre elementos middleware y que existen modelos de nodos sensores que difieren bastante del modelo de bajo coste.

1.2.2.1 Tecnología basada en 802.15.4: ZigBee y Xbee

ZigBee [Bar1][Zig1] se basa en el estándar físico IEEE 802.15.4 [Iee1][Pan1] y propone la integración del estándar que se amplía con capas de niveles superiores (acceso al medio especialmente) de la pila de protocolos de comunicación.

IEEE 802.15.4 trabaja con frecuencia a 2.4GHz, velocidad de 250Kbps y modulación DSSS. El estándar trabaja, por lo tanto, sobre las bandas ISM de uso no regulado. Se definen hasta 16 canales en el rango de 2,4 GHz, cada uno de ellos con un ancho de banda de 5 MHz.

Zigbee está principalmente diseñado para el soporte de transmisión a baja potencia de pequeñas cantidades de datos siendo, por lo tanto, un estándar de gran perspectiva y potencialidad [Zhe1][Bou1] en diferentes campos de aplicación como WPAN, redes demóticas y redes de sensores inalámbricos.

Es probablemente el estándar que mejor se adapta a las Personal Area Networks (PANs) y, por tanto, a las redes inalámbricas de sensores en cuanto ZigBee privilegia el bajo consumo respecto al ancho de banda, aunque proporcionando rangos de operación interesantes (hasta 75 metros). Otro punto fuerte de ZigBee, como se puede apreciar en Figura 1.4, es la robustez bastante superior a la garantizada por otros estándares.

ZigBee propone dos posibles versiones de protocolo de acceso al medio [Bar1] para ampliar ulteriormente su campo potencial de aplicación: RFD (Reduced Function Device) and FFD (Full Function Device). El primero proporciona funcionalidades mínimas, el segundo pleno soporte para acceso al medio como en otros estándares. La mayoría de los dispositivos sensores inalámbricos suelen hacer referencia a la solución básica (RFD) lo que proporciona al desarrollador la posibilidad de diseñar soluciones a medida (normalmente implementadas vía software como complemento de los protocolos de enrutamiento). De esta forma se consigue optimizar la comunicación en entornos específicos aunque limitando el nivel de abstracción en el desarrollo de los sistemas de comunicación.

En su versión integral (que también incluye el nivel de red), ZigBee permite conectar un elevado número de nodos eventualmente distribuidos en subredes. No obstante, el nivel de red no es compatible entre las diferentes versiones de ZigBee, al no poder combinarse routers de versiones distintas. Un router ZigBee (ZR, Zigbee Router) es un tipo de dispositivo que básicamente interconecta nodos separados en la topología de red. Estos nodos pueden ser o bien nodos finales (ZED, ZigBee End Device) o bien coordinadores (ZC, ZigBee Coordinator).

Xbee es una variante de ZigBee a nivel físico y nivel de enlace de datos. A nivel físico, la diferencia fundamental es la variación progresiva del control de potencia para alcanzar

mayores rangos de cobertura. A nivel de enlace de datos, la complejidad se incrementa al ofrecer una serie de nuevos campos hacia el nivel de aplicación.

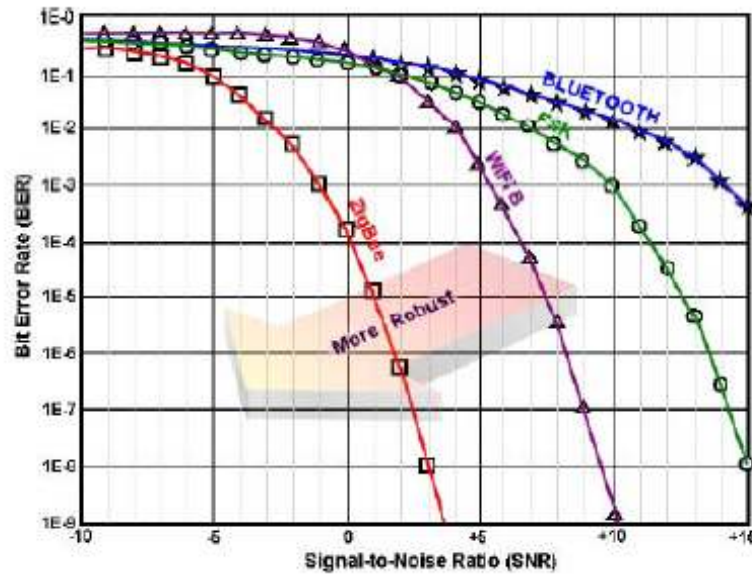


Figura 1.4. Comparativa de la robustez de diferentes tecnologías inalámbricas [Fer1].

1.2.2.2 Soluciones basadas en 802.15.1: Bluetooth

Bluetooth [Blu1] es una especificación industrial para Redes Inalámbricas de Área Personal normalmente empleado para posibilitar la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia segura y en banda ISM (2,4 GHz).

Compatible con el estándar físico IEEE 802.15.1 [Iee3], proporciona, respecto a ZigBee, un mayor ancho de banda y un radio de comunicación inferior (en su versión de bajo consumo), normalmente a coste de un mayor gasto energético. La cobertura alcanzada por un dispositivo BlueTooth depende de la potencia de transmisión, criterio que clasifica a estos. Otra clasificación puede efectuarse respecto a la velocidad de transmisión (BT 1.0, con una velocidad de hasta 1 Mbps, y BT 2.0 con una velocidad de 3Mbps; los futuros dispositivos, UWB Bluetooth, con velocidades entre 50 y 480 Mbps).

El estándar Bluetooth, análogamente a WiFi, emplea la técnica FHSS (Frequency Hopping Spread Spectrum), que divide la banda de frecuencia de 2.402 - 2.480 GHz en 79 canales (saltos) de 1 MHz y transmite la señal utilizando una secuencia de canales conocida tanto para la estación emisora como para la receptora. Al cambiar de canal con una frecuencia de 1600 veces por segundo, el estándar Bluetooth puede evitar la interferencia con otras señales de radio.

El estándar Bluetooth define un cierto número de perfiles de aplicación (denominados perfiles Bluetooth) para definir qué tipos de servicios ofrece un dispositivo Bluetooth. Por lo tanto, cada dispositivo puede admitir múltiples perfiles. Algunos de ellos son: Perfil de distribución de audio avanzado (A2DP), Perfil de control remoto de audio y vídeo (AVRCP), Perfil básico de imagen (BIP), Perfil básico de impresión (BPP), Perfil de telefonía inalámbrica (CTP) y Perfil de fax (FAX).

En principio las características citadas no parecen ideales para soportar eficientemente aplicaciones sobre redes de sensores en el contexto tecnológico actual; aun sin considerar el factor consumo, Bluetooth garantiza una conectividad notablemente inferior respecto a ZigBee a causa de su reducido radio de comunicación; por contra, el ancho de banda proporcionado se quedaría prácticamente desaprovechado visto el típico entorno de trabajo del tipo de redes que lo utilizan.

Excepciones de interés relevante pueden ser aplicaciones caracterizadas por transferencia masiva de datos, poco usuales en redes de sensores a menos de aplicaciones multimediales; en este último contexto, actualmente poco difuso aunque emergente, Bluetooth puede ser un válido antagonista de ZigBee o, incluso, proporcionar mejores prestaciones especialmente en un contexto de comunicación multimodo.

Desde un punto de vista institucional, el Bluetooth SIG (Special Interest Group) es una asociación privada sin ánimo de lucro formada por más de 9000 compañías de telecomunicaciones, informática, etc. Los miembros del SIG (Intel, IBM, Nokia, etc.) dirigen el desarrollo de la tecnología inalámbrica Bluetooth, además de implementar y comercializar la tecnología en sus productos. El Bluetooth SIG por sí mismo no fabrica ni vende dispositivos.

1.2.2.3 Wireless HART

Otra alternativa es el estándar WirelessHART [Wir1] que se basa en el estándar HART de amplia difusión, tal como lo evidencian los 20 millones de dispositivos de campo compatibles con HART.

HART se concibió originalmente como una ampliación del bucle de corriente 4 a 20 mA común, con el fin de proporcionar dispositivos de campo con mayor funcionalidad. WirelessHART es la simbiosis entre la ampliamente difundida y probada tecnología HART y la nueva tecnología de radio (por lo menos, nueva en la tecnología de proceso).

Además de la conocida aplicación HART para la parametrización de dispositivos, HART ya se ha utilizado ampliamente para: (i) Supervisión de valores de instrumentos y medioambientales; (ii) Gestión y optimización de activos; (iii) Mantenimiento preventivo; (iv) Supervisión del rendimiento y (v) Gestión de energía. WirelessHART utiliza la banda ISM como medio de transferencia como varias tecnologías de radio, incluido WLAN, Bluetooth y ZigBee. Con el fin de evitar colisiones en la banda de frecuencia de 2,4 GHz, WirelessHART lleva a cabo una búsqueda especial de los canales no utilizados en esta banda de frecuencia y comprueba las interferencias mutuas de las tecnologías de radio.

Se considera, actualmente, una de las referencias para comunicación inalámbrica en entornos industriales [Gun1].

1.2.2.4 WiFi

Wifi [Wif1] es un sistema de envío de datos inalámbrico estandarizado en IEEE 802.11 [Iee4]. Básicamente se trata de portar la flexibilidad de las redes Ethernet a un entorno radio, con las consiguientes adaptaciones de un medio inalámbrico.

Existen varios estándares de 802.11. El primero de ellos, 802.11b permite velocidades de hasta 11 Mbps en la banda libre de 2.4 GHz. Nótese que, en un entorno radio donde la recepción de la señal es altamente variable, se emplean diferentes modulaciones para

garantizar la robustez de la comunicación, a costa de sacrificar velocidad. Es por ello por lo que 802.11b permite descender hasta 2 Mpps si es necesario. El siguiente estándar, 802.11g permite velocidades de hasta 54 Mbps, opera en la banda libre de 2.4 GHz y es actualmente el más utilizado en la actualidad, al permitir sesiones multimedia debido a su capacidad de transmisión y ser retro-compatible con 802.11b.

El estándar 802.11a opera hasta 54 Mbps en la banda de 5 GHz, por lo que se pierde retro-compatibilidad en este aspecto, si bien esta banda está más libre de interferencias al no coexistir con otras tecnologías como BT, Zigbee y microondas.

El último estándar recientemente aprobado es 802.11n, que permite velocidades de hasta 300 Mbps empleando modulaciones más robustas y sistemas MIMO.

La seguridad en redes Wifi es aportada a través de diferentes alternativas o protocolos de cifrado, como son WEP, WPA y WPA2, éste último el más seguro de todos. Otras mejoras relativas a la seguridad son el filtrado por MAC y el uso de túneles IP en VPNs.

Desde el punto de vista de la arquitectura, la conexión mediante Wifi requiere básicamente un punto de acceso (AP) que permite y coordina el acceso inalámbrico y el acceso a una red cableada (ya sea una LAN o directamente Internet). Los clientes Wifi simplemente deben asociarse (y registrarse, si la autenticación lo requiere) a un punto de acceso Wifi para empezar a transmitir información. También es posible la comunicación entre dos clientes Wifi sin punto de acceso, mediante una configuración ad-hoc.

1.3 Plataformas Software: Sistemas Operativos y Lenguajes de Programación

Los nodos sensores necesitan ser gestionados por sistemas operativos, empotrados o de propósito general, dependiendo de su tamaño, potencia y capacidad. Asimismo se requieren ambientes bien definidos para el desarrollo de firmwares concretos. Respecto a entornos de cálculo tradicionales o de tipo genérico, los nodos sensores inalámbricos se caracterizan por ser elementos de recursos extremadamente limitados que requieren un software de gestión más sencillo y básicamente orientado a la eficiencia. En otras palabras, hay dos elementos básicos que caracterizan el entorno de las redes de sensores respecto a la ingeniería de sistemas operativos: por una parte las restricciones hardware que hacen la aplicación de sistemas operativos tradicionales ardua o imposible (por ejemplo no se pueden implementar técnicas de virtualización de memoria), por la otra también los objetivos finales en términos de funcionalidades proporcionadas pueden ser bastante diferentes (por ejemplo, no se desea un entorno de software interactivo como en el caso de los sistemas operativos genéricos).

1.3.1 Sistemas Operativos

Los sistemas operativos para redes inalámbricas de sensores se clasifican en tres categorías principales, cada una refleja una filosofía de diseño concreta:

- *Sistemas operativos derivados de sistemas operativos genéricos.* La idea básica de esta clase de solución es proporcionar, sobre la plataforma de gestión de versiones extremadamente reducidas de sistemas operativos genéricos, un conjunto software para gestión avanzada y específica de nodos con recursos limitados. Las principales soluciones están desarrolladas sobre distribuciones Linux. Recientemente se ha proporcionado una versión ultra compacta de .NET (.NET Micro Framework [Net1])

que, en las primeras evaluaciones parece carecer de eficiencia respecto a las correspondientes soluciones basadas en Linux.

- *Sistemas operativos para sistemas empotrados.* Desde un punto de vista conceptual un nodo sensor inalámbrico no es diferente respecto a otros dispositivos empotrados de capacidad similar. Por lo tanto la utilización de sistemas operativos diseñados para dispositivos empotrados (eCos y uC/OS por ejemplo) es una opción más que realista.
- *Sistemas operativos ad-hoc.* Sistemas operativos diseñados y desarrollados a medida para garantizar las máximas prestaciones. El más relevante de esta categoría es TinyOS [Tin1].

TinyOS es la solución más empleada para trabajar con sensores de pequeña capacidad. Es el sistema operativo seleccionado para la realización de la tesis, en la siguiente sección se proporciona una descripción detallada así como una visión panorámica de sus características.

1.3.1.1 TinyOS

TinyOS [Tin1] es un conjunto avanzado de programas de código abierto desarrollado por la Universidad de Berkeley. *TinyOS* proporciona una plataforma completa tanto para la gestión como para el desarrollo a través de una amplia y extensible API que cuenta con el apoyo de gran parte de la comunidad científica internacional.

TinyOS proporciona interfaces, módulos y configuraciones específicas, que permiten a los programadores construir programas como una serie de módulos que hacen tareas específicas. Los módulos de *TinyOS* proporcionan interfaces para los tipos estándar de entradas y salidas de hardware y sensores.

El lenguaje de programación es un metalenguaje orientado a módulos derivado de C (Network Embedded Systems C, *NesC* [Nes1]). *NesC* se puede considerar un dialecto del lenguaje de programación C optimizado para las limitaciones de memoria de las redes de sensores.

El diseño de TinyOS está basado en responder a las características y necesidades de las redes de sensores, tales como reducido tamaño de memoria, bajo consumo de energía, operaciones de concurrencia intensiva, diversidades en diseños y usos, y finalmente operaciones robustas para facilitar el desarrollo confiable de aplicaciones. Además se encuentra optimizado en términos de uso de memoria y eficiencia de energía.

El diseño del kernel de *TinyOS* contempla un esquema bastante más sencillo que lo propuesto por sistemas operativos tradicionales y se articula en dos niveles de planificación:

- *Eventos:* Es una ejecución de una porción de código como consecuencia de la detección de un evento interno (caducidad de un temporizador) o externo (recepción de un mensaje). Suelen interrumpir las tareas que se están ejecutando al momento de la detección del evento.
- *Tarea:* Es una ejecución de código en contexto temporal non critico; esto significa básicamente que las tareas, aunque ejecutándose en su totalidad, pueden ser interrumpidas por eventos que, por lo tanto, se consideran con prioridad más alta y, normalmente, implican procesamientos mínimos.

Visto el paradigma de gestión de *TinyOS* que es claramente orientado a eventos (conurrencia intensiva) y teniendo en cuenta que a efectos prácticos, el gestor de eventos es el elemento crítico de todo el sistema, así como las políticas de acceso concurrente a datos y la ejecución síncrona de código.

La versión 2.x de *TinyOS* [Tin2] presenta varias mejoras, optimizaciones y mecanismos avanzados, como, por ejemplo, un sistema de gestión de Threads introducido en la versión 2.1 [Tin3].

Existen varias herramientas que permiten el estudio y desarrollo de aplicaciones para las redes de sensores, que van desde aplicaciones para la obtención y manejo de datos, hasta sistemas completos de simulación.

1.3.1.2 Otras soluciones

Existen otras soluciones alternativas a TinyOS. Las más relevantes se detallan a continuación así como soluciones compactas de sistemas operativos de propósito general:

- *.NET Micro Framework* es la solución .NET para dispositivos empotrados de pequeñas dimensiones y capacidades [Mic1]. Además de estar totalmente integrado en Visual Studio, esta solución dispone de un módulo avanzado y extensible para emulación de diferentes capacidades hardware. Esta solución está en su estado inicial en el ámbito de las redes inalámbricas de sensores (una versión de la plataforma IMote2 está disponible con plataforma de gestión .NET) proponiendo un paradigma de programación (en C#).. Ya dispone de una interfaz avanzada de gestión de la batería por lo cual muchas empresas están apostando de manera firme sobre esta plataforma que, en perspectiva, podría convertirse en una plataforma de primer plano en el ámbito de las redes de sensores. De momento no es fácil comparar las prestaciones de esta plataforma de forma completa porque, actualmente, su uso implica un hardware relativamente avanzado (como lo de IMote2). Su punto fuerte parece ser la convergencia con otras plataformas basadas en .NET, su punto débil las prestaciones de momento menos efectivas respecto a soluciones más específicas como TinyOS.
- *eCos* (embedded Configurable operating system [Fer1]) es un sistema operativo de código abierto para sistemas empotrados con operaciones en tiempo real; se programa en C y dispone de un mecanismo flexible para adaptar los programas generados a diferentes características hardware y modos de funcionamiento a través de una capa de abstracción del hardware.
- *uC/OS* (MicroC/OS-II [Fer1]) es un sistema operativo multitarea en tiempo real basado en prioridad preventiva diseñado para gestión de sistemas empotrados. Como sugiere su nombre, se trata de la evolución de un sistema anterior (MicroC/OS) que incluye unas cuantas mejoras que adaptan el prototipo inicial a requerimientos más propiamente comerciales.
- *Contiki* [Fer1] es un sistema extremadamente compacto que se caracteriza por soportar aplicaciones y mecanismos avanzados (gestión de aplicación multitareas, pila de protocolos TCP/IP) utilizando varios kbytes de código y tan solo unos cuantos kbytes de RAM. Consiste en un núcleo completamente orientado a eventos con carga y descarga dinámica de programa en proto-hilos. También soporta multi-hilo por cada proceso y comunicación entre procesos mediante intercambio, gestionado a eventos, de mensajes.

- *MANTIS* (Multimodal system for Networks of In-situ wireless Sensors [Fer1]) se distingue por intentar optimizar las características típicas de los nodos sensores inalámbricos a través de una utilización avanzada de los recursos (buffers por ejemplo).
- *BTnut* [Fer1] fue diseñado para funcionar sobre arquitecturas tipo MOTE; es, por lo tanto, un sistema parecido a TinyOS aunque, al contrario de TinyOS, se programa en lenguaje C.
- *SOS* [Fer1] intenta compensar unas cuantas limitaciones de sistemas básicos de gestión como TinyOS. Su mayor contribución es la capacidad de carga y descarga dinámica de aplicaciones (interpretadas como módulos software) que pueden, por lo tanto, alternarse en tiempo de ejecución sin interrumpir la base del sistema operativo. Sistemas basados en SOS pueden proporcionar esquemas de reconfiguración dinámica extremadamente mejoradas especialmente en sistemas tolerantes a fallos o sensibles a eventos complejos. Además se dispone de un modelo de programación a prioridad.
- *Nano-RK* [Fer1] es un sistema completamente preventivo basado en reserva en tiempo real. Soporta multitarea preventiva con prioridad a través de un kernel con recursos empotrados.

1.3.2 Lenguajes de Programación

El lenguaje de programación más común para nodos sensores inalámbricos es C y el conjunto de sus derivados como el ya mencionado NesC [Nes1]. Más recientemente se han propuesto soluciones conceptualmente diferentes basadas en tecnologías JAVA o, incluso, en paradigmas de programación lógica [Tav1][Chi1] (normalmente lenguajes PROLOG-like como DATALOG) que, a pesar de su indudable interés científico, tienen difícil aplicación en contextos comerciales.

En las secciones anteriores se ha indicado que los lenguajes de programación tipo C son los más utilizados tan como en dispositivos empotrados. Hay, básicamente, dos tendencias relacionadas con los lenguajes tipo C:

- Por una parte se tiende a limitar el paradigma de programación general y a orientarlo a eventos. Es el caso del ya mencionado NesC que propone un paradigma completamente orientado a módulos con soporte para eventos asíncronos, comandos y tareas.
- Otra posibilidad es la de mantener el paradigma general de programación proporcionado por C integrando el mismo con un conjunto de librerías especializadas orientadas a soportar el programador en la gestión de operaciones de tiempo real.

Aspectos comerciales o, bien, el intento de convergencia entre sistemas con recursos limitados y sistemas generales son las causas principales de existencias de alternativas a lenguajes de tipo C en el contexto de las redes inalámbricas de sensores. Las soluciones más conocidas están basadas en: JAVA y son empleadas principalmente en sensores comercializados por SUN y C# en el contexto de la plataforma .NET.

La mayoría de los casos de uso relacionados con redes de sensores inalámbricos se modelan sobre grafos, eventualmente orientados y pesados. Un gran número de programadores

considera el paradigma de programación declarativo como especialmente apto a la resolución de este tipo de problemas así como para la programación de nodos inteligentes [Chi1][Tav1]. En el contexto de otros sistemas, el paradigma declarativo podría proporcionar un significativo nivel de abstracción para el desarrollo de módulos compactos que podrían incrementar tanto la comprensión como el tiempo de implementación del código.

Esta línea de investigación ha estado muy activa en los últimos tiempos cuando soluciones basadas en versiones simplificadas de PROLOG (por ejemplo DATALOG) se utilizaron para proporcionar plataformas de desarrollo declarativas. Estos experimentos se limitaron a traducir código declarativo a código NesC. A pesar del énfasis inicial esta línea de investigación no parece especialmente activa en la actualidad, aunque la mayoría de los especialistas reconozcan la convergencia entre inteligencia artificial y redes de sensores como un tema de gran interés científico. No se tiene constancia que se hayan propuesto lenguajes declarativos a medida.

1.4 Arquitectura de protocolos

La arquitectura de protocolos que constituye el software, ejecutado por un nodo sensor inalámbrico, puede variar en función de los requisitos funcionales y no funcionales de la aplicación considerada; puede incluir tan solo funcionalidades extremadamente básicas, extender/integrar estas últimas o proporcionar un conjunto complejo de mecanismos avanzados.

Aunque no exista, de momento, un modelo de referencia universalmente reconocido para la pila de protocolos de una red inalámbrica de sensores, en el presente capítulo se hace referencia al modelo representado en Figura 1.5

A continuación se propone una clasificación esquemática de los principales protocolos que suelen componer el comportamiento de un nodo sensor inalámbrico.

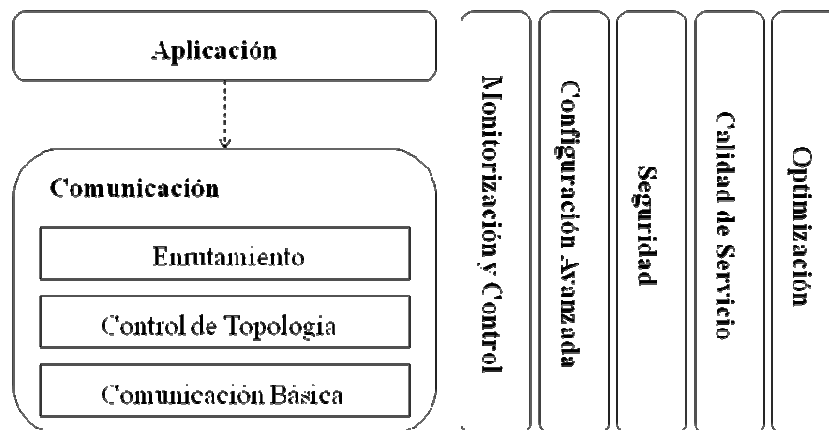


Figura 1.5. Ejemplo de modelo de referencia para la pila de protocolos de una red de sensores inalámbricos.

1.4.1 Funcionalidades de comunicación básicas

La mayoría de los nodos inalámbricos se comunica de acuerdo con la versión de funciones reducidas de ZigBee [Bar1] que, de hecho, no proporciona ninguna funcionalidad de acceso

al medio. Esta forma de funcionar aparentemente ineficiente e incómoda para el programador proporciona, realmente, un grado de libertad añadido al programador que puede, en función del dominio aplicativo considerado (y la relativa densidad de nodos), extender las funcionalidades de base para maximizar las prestaciones.

Considerando una tecnología inalámbrica de referencia extremadamente limitada a nivel de acceso al medio, cualquier red de una cierta densidad requiere una extensión de dichas funcionalidades básicas de comunicación para garantizar prestaciones aceptables. En un contexto de comunicación mono-salto, las funcionalidades de comunicación básica suelen constituir el único protocolo de comunicación.

La mayoría de las soluciones más eficientes se basan en TDMA [Cio1]; los protocolos de tipo TDMA suelen ser bastante más sencillos de los correspondientes para redes ad-hoc. La solución más lógica parece ser TDMA con ack que proporciona un protocolo de acceso al medio conforme a los adoptados en redes tradicionales; considerando las características peculiares de comunicación de los nodos sensores inalámbricos, en presencia de alta densidad, se prefiere replicar la información a través de múltiples rutas más que comunicación bidireccional (mensaje y confirmación).

Soluciones tipo TDMA suelen ser poco funcionales en el contexto de aplicaciones extremadamente impulsivas que requieren una comunicación extremadamente reactiva; un ejemplo clásico son las redes que detectan actividad sísmica. Para esta clase de aplicaciones suelen adoptarse soluciones de enrutamiento orientadas a privilegiar flujos impulsivos de datos que se caracterizan por intentar garantizar un cierto porcentaje de información correctamente transmitido (y recibido); por lo tanto resulta difícil considerar soluciones de acceso al medio estándar.

1.4.2 Configuración básica o control de topología

Es un conjunto de protocolos orientados a proporcionar a la red la configuración mínima para garantizar sus funciones básicas [Soh1][San1], normalmente de enrutamiento. Por esta razón, en muchos casos esta clase de protocolo suele considerarse parte integrante del protocolo de enrutamiento, aunque soluciones altamente flexibles multi-dominio y/o caracterizadas por múltiples modos de funcionamiento aconsejan una ingeniería altamente modular de acuerdo con los fundamentos de la ingeniería del software moderna.

Normalmente el protocolo de configuración básica proporciona a cada nodo sus parámetros de configuración (normalmente resultado de operaciones de descubrimiento) junto con su visión del entorno (normalmente tabla de vecinos y características relacionadas). Eventualmente puede también proporcionar el conocimiento de ciertos parámetros (lógicos o físicos) relacionados con una parte restringida de nodos (normalmente de los vecinos) o de todos los nodos (configuración centralizada).

Ejemplos típicos de información adicional, relacionada con los nodos de entorno, pueden ser la posición física (soporte de soluciones de enrutamiento location-aware [Rao1][Shu1]) o la energía (soporte de protocolos de enrutamiento energy-aware [Lin1][Vid1]).

Otras funciones típicas se consideran la estimación de la estabilidad de las rutas (a soporte de los protocolos de enrutamiento que la requieren) y la disseminación dinámica de parámetros [Cha1][Hwa1] (por ejemplo identificadores dinámicos de nodos o grupos).

1.4.3 Enrutamiento

El enrutamiento es la funcionalidad más relevante de una red de sensores inalámbricos; las prestaciones de la red completa dependen, en gran parte, de la eficacia del mecanismo de enrutamiento que suele tener una estricta dependencia de la configuración básica y avanzada (clustering por ejemplo) de la misma. Los protocolos de enrutamiento para redes de sensores inalámbricos [Akk1][Kar1][Kar2] han sido (y siguen siendo) objeto de gran interés para la comunidad científica.

Desde el punto de vista del enrutamiento, una red de sensores inalámbricos se caracteriza por su topología (números de bases, números de nodos, densidad de nodos), por el patrón de comportamiento de los nodos (estático, semi-estático, móvil) y por las características específicas de las aplicaciones (pasivas, on-demand, reactivas, etc). La mayoría de las redes tan solo necesitan de comunicación convergente (desde los nodos sensores hacia la estación base); de todos modos muchos algoritmos también proporcionan soporte para comunicación divergente (desde la base a todos o unos cuantos nodos) y/o nodo a nodo.

La clasificación más elemental para protocolos de comunicación suele diferenciar soluciones centralizadas y distribuidas. Un contexto distribuido proporciona a los nodos una visión muy limitada de la red: un nodo suele disponer de información relacionada con los nodos a su alrededor y, eventualmente, de la información relacionada con los mismos (parámetros lógicos y/o físicos); a la hora de tener que encaminar un paquete (Figura 1.6, izquierda), el nodo en cuestión elige la ruta en función de la información que dispone; su entorno de conocimiento al salto i se limita, por lo tanto, al paso $i+1$.

Al contrario un algoritmo centralizado proporciona a cada nodo una visión completa de la red. Con esta información global un nodo está capacitado para elegir la ruta óptima. No siempre un algoritmo distribuido configura rutas óptimas. En Figura 1.6 a la derecha, se propone, a título de ejemplo, una sencilla topología en el cual cada salto se caracteriza por un cierto factor de coste que tiene que minimizarse; en presencia de dos posibles rutas formadas por el mismo número de saltos, un nodo operante con algoritmo distribuido elige la ruta marcada por línea continua disponiendo tan solo de información de coste directamente relacionada con sus vecinos; contrariamente, una solución centralizada optaría por la ruta de coste mínimo (marcada con línea discontinua).

Evidentemente los algoritmos centralizados proporcionan prestaciones mejores aunque tan solo en un contexto teórico. De hecho, a causa de la necesidad de diseminar por la red cantidad relevantes de información sujeta a continuos cambios, las soluciones centralizadas son poco difusas porque costosas en términos de recursos y difícilmente escalables.

Redes complejas caracterizadas por pequeña/mediana escala pueden trabajar en un contexto distribuido extendido (semi-centralizado) en el cual el entorno de conocimiento de un nodo se extiende a un conjunto de nodos superior a lo de sus vecinos aunque no a toda la red. Dicha extensión del entorno de conocimiento implica un mayor gasto de recursos en las tareas de configuración/reconfiguración. Este tipo de solución, por ejemplo extendida a dos saltos, puede configurar rutas óptimas en topologías como la propuesta en Figura 1.6 aunque requiriendo mayores recursos para las tareas de configuración/reconfiguración.

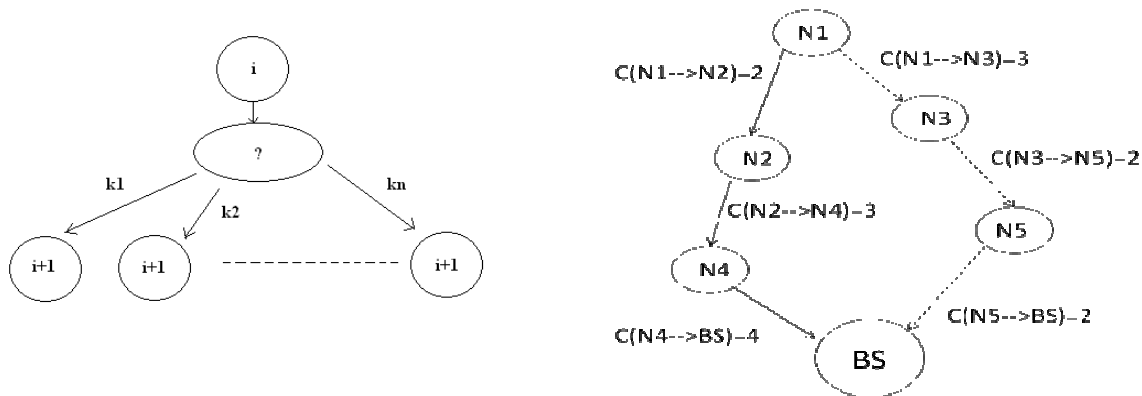


Figura 1.6. Ejemplo de protocolo de enrutamiento distribuido (figura original de [Con1]) y centralizado.

Durante los últimos años se han propuesto una gran variedad de soluciones avanzadas caracterizadas por eficiencia, robustez y fiabilidad para entornos estáticos y nomádicos [Akk1]; dichas soluciones se caracterizan por principios básicos diferentes y suelen optimizar la comunicación configurando rutas con número mínimo de saltos y/o balanceando el tráfico entre las rutas de forma de maximizar el tiempo de vida de la red.

Las soluciones más sencillas se basan simplemente en el balanceo de tráfico o, bien, sobre la configuración aleatoria de las rutas. Todas y cada una de las soluciones disponibles pueden, en ciertos entornos aplicativos, necesitar de mayor robustez y fiabilidad que suele proporcionarse a través de información redundante o, bien, a través de la utilización de protocolos de enlace de datos más robustos a coste, obviamente, de un mayor gasto de recursos.

A continuación se describen las dos clases de soluciones más relevantes: enrutamiento *energy-aware* y enrutamiento *position-aware* (o *location-aware*); finalmente se describen brevemente las peculiaridades del entorno móvil.

1.4.3.1 Soluciones *Energy-aware* y *Position-aware*

Soluciones basadas en *energy-awareness* [Has1][Vid1][Lin1] asumen que cada nodo conozca el nivel de carga de la batería de sus vecinos (*energy-awareness* local) o, incluso, de toda o gran parte de la red (*energy-awareness* centralizado); aprovechando esta información las rutas de comunicación pueden configurarse de forma tal que se intente la comunicación entre nodos con mayores cargas de baterías maximizando el tiempo de vida global de la red. Obviamente, en el caso de soluciones distribuidas, quedan las limitaciones descritas en la sección anterior.

Evidentemente esta clase de solución garantiza altas prestaciones en aplicaciones caracterizadas por tráfico de información altamente irregular que puede generar consumos energéticos bastante diferentes entre los nodos; es prácticamente inútil en el caso contrario especialmente considerando que el estimador de energía no suele ser extremadamente preciso en dispositivo de bajo coste y, por lo tanto, se considera fiable solo por diferencias notables de energías.

Contrariamente el coste de diseminación de la información suele ser poco relevante en un contexto distribuido aunque, evidentemente, tiene que ser periódico siendo el nivel de carga de la batería un valor variable en el tiempo en función de la actividad de la red.

Otra clase de solución de altas prestaciones se basa en *location* o *position-aware* [Sat1][Shu1]; en este caso el factor guía para determinar las rutas de comunicación es la

posición física de los nodos y es concretamente aplicable gracias a la disponibilidad de dispositivos para localización (GPS) de bajo coste. Evidentemente el enrutamiento geográfico podría, por lo menos en teoría, ser funcional tanto en contextos estáticos como nomádicos/móviles. En general las soluciones de enrutamiento geográfico garantizan altas prestaciones para la comunicación divergente o nodo a nodo.

En un contexto estático esta clase de solución proporciona buenas prestaciones especialmente considerando que, si la posición no varía, no se requiere diseminación continua de información. La limitación mas grande, en este sentido, es representada por el mismo dispositivo GPS que implica un hardware bastante más avanzado a lo usual; además su utilizo se limita a contextos outdoor. Otras soluciones [Rao1] position-aware se basan en la estima de la posición de los nodos en función de las potencias detectada en la recepción de los mensajes; aparecen poco interesantes en contextos generales, posiblemente funcionales en unos entornos aplicativos específicos.

La aplicación de soluciones *position-aware* en entornos móviles y outdoor con posicionamiento GPS esta, de momento, limitada a causa de las propias característica de los dispositivos GPS que están diseñados para geo-referenciar la información más que a soporte de mecanismos básicos.

1.4.3.2 Enrutamiento en redes móviles

Mención particular merece el entorno móvil que, también gracias a la disponibilidad de dispositivos de posicionamiento integrados en los nodos inalámbricos y, por lo tanto, de generar información geo-referenciada, es objeto de particular atención en el contexto científico-comercial.

Una red de sensores se considera móvil si la posición de los nodos que la componen puede cambiar en el tiempo de vida de la aplicación. Realmente la movilidad de los nodos puede generar diferentes escenarios: sinks móviles y nodos sensores estáticos, sinks estáticos y nodos sensores móviles, sinks y nodos sensores móviles o, finalmente, soluciones híbridas caracterizadas por coexistencia de nodos móviles y fijos.

Evidentemente la tecnología de referencia por excelencia es MANET (Mobile Ad-hoc NETwork [Jey1][Dha1]), con la cual hay muchos puntos en común y unas diferencias sustanciales.

Aunque ambos entornos requieren soluciones diferentes respecto a redes con topologías predefinidas y presentan muchos elementos de convergencia, no sería del todo correcto considerar las redes móviles de sensores inalámbricos simplemente como caso particular de MANET. En primer lugar las redes de sensores, móviles o estáticas, suelen ser sistemas data-centric [Hon1] al contrario de las redes ad-hoc que proponen un enfoque id-centric o, eventualmente, user-centric con fuerte interacción entre usuario y nodo. Además, la presencia de las estaciones base implícitamente centraliza las redes de sensores, al contrario de las MANETs que, propone un entorno distribuido potencialmente peer-to-peer. Finalmente el hardware de los nodos en MANET se asume potencialmente limitado solamente en términos de alimentación y suele diseñarse para soportar aplicaciones avanzadas de alta capacidad; al contrario, el contexto aplicativo de los nodos sensores es bastante más limitado (normalmente a la interacción con el ambiente circundante) así como su hardware.

Las soluciones diseñadas para redes de sensores móviles suelen ser simplificaciones de las equivalentes diseñadas para MANET o bien, especializaciones de las soluciones diseñadas

para entornos de redes de sensores estáticos con diferente gestión del estado relativo a las rutas de comunicación. Estas últimas soluciones se basan, en función del entorno de aplicación, sobre configuración soft-state o stateless (on-demand) de las rutas de comunicación: en el primer caso las rutas se consideran validas tan solo por un cierto periodo de tiempo (al contrario de las soluciones con estado que asumen rutas estables entre dos tareas de configuración) requiriendo, por lo tanto, continuos refrescos; soluciones on-demand (normalmente iniciadas por la estación base) suelen basarse en la configuración de las rutas en tiempo real asumiendo las mismas validas tan solamente para la comunicación corriente.

El entorno nomádico asume movimientos limitados de los nodos debidos a fenómenos naturales o accidentales y/o cambios de posicionamiento puntuales; redes nomádicas de sensores suelen trabajar con protocolos diseñados para redes estáticas; así como para redes estáticas que trabajan en entornos de comunicaciones hostiles (interferencias temporales, obstáculos, etc.), los mecanismos de monitorización y gestión suelen garantizar la fiabilidad necesaria.

En el contexto de las redes de sensores móviles hay un siempre creciente interés en las aplicaciones “group-based”, especialmente en entornos militares y en aplicaciones vehiculares. En este ultimo campo de aplicación, las redes de sensores inalámbricos proporcionan una interesante alternativa de bajo coste para el soporte de aplicaciones innovadoras para monitorización de grandes áreas aprovechando la movilidad de los vehículos.

1.4.4 Monitorización y control

Cualquier aplicación comercial requiere una cierta fiabilidad, mínima en unos casos, máxima en otros; esto implica la necesidad de conocer el estado de la red y, en consecuencia, su nivel de operatividad.

Arquitecturas caracterizadas por tráfico de datos periódico y continuo se monitorizan implícitamente. Otras (aplicaciones reactivas a eventos, por ejemplo) requieren mecanismos específicos para monitorización. Es difícil identificar patrones generales para los mecanismos de monitorización; normalmente suelen comprobar la operatividad de la red a través de la diseminación de paquetes de test con espera de notificación por parte de los nodos interesados; visto el coste implícito de la diseminación de dichos paquetes (test y notificaciones), unas soluciones prefieren reconfigurar la red periódicamente sin monitorizar explícitamente la misma.

De hecho, otro de los parámetros comerciales más comunes es la robustez que impone que la red se reconfigure, periódicamente o de forma reactiva, para asegurar funcionalidad en presencia de interferencias, obstáculos y otros factores que pueden limitar, de forma temporánea o permanente, la operatividad de la red.

Considerando la intervención humana como extremadamente costosa (imposible en unos casos), la auto-gestión de la red [Sh1] es uno de los requisitos principales de muchas arquitecturas reales. Todos los aspectos relacionados con monitorización y control tienen importancia en presencia de despliegue aleatorio de los nodos (muy común), donde diferentes despliegues pueden presentar prestaciones significativamente diferentes entre ellas y, tal vez, lejanas de las esperadas.

El entorno móvil representa, una vez más, una excepción en cuanto los mecanismos de monitorización y control son implícitamente parte del propio mecanismo de enrutamiento de base.

1.4.5 Calidad de servicio y Seguridad

En campo de redes de sensores el concepto de calidad de servicio [Daz1][Wan1] puede ser bastante ambiguo. A parte de los ya mencionados requisitos no funcionales (fiabilidad, robustez, eficiencia energética, todos directamente gestionados por los mecanismos de comunicación, monitorización y gestión), los requisitos de las aplicaciones reales suelen referirse a calidad de servicio especialmente en términos de tolerancia de fallos.

En otras palabras, aunque existan ciertos parámetros de calidad de servicio unánimemente reconocidos, requisitos reales de calidad de servicio no suelen definirse de acuerdo con métricas estándares como en otros tipos de red. Una técnica bastante utilizada es medir el “beneficio” introducido por un cierto mecanismo de calidad de servicio en función de los recursos utilizados.

La capacidad de garantizar una cierta calidad de servicio (como definida en el entorno considerado) puede ser una de las claves para el éxito comercial de las redes de sensores inalámbricos.

Como en otros entornos, la tolerancia a fallos en redes de sensores [Oul1][Xue1][Kou1] se caracteriza por dos fases (detección del fallo y reacción al mismo [Rui1]). Unos fallos simplemente son indetectables, ciertas arquitecturas auto detectan los fallos, otras necesitan mecanismos específicos de monitorización. Asimismo, en unos casos las arquitecturas son implícitamente tolerantes a los fallos en otros necesitan mecanismos reactivos. En el caso específicos de las redes de sensores, el concepto de fallo hardware/software de un nodo puede integrarse con el “fallo” debido al acabarse de la batería. Tanto los mecanismos de detección como de reacción a fallos suelen ser característica de sistemas concretos; es, por lo tanto, difícil identificar patrones. El caso más crítico es, seguramente, representado por las tareas cooperativas que, a frente de fallos no detectados, podrían producir resultados erróneos o indeseados.

Los aspectos relacionados con la seguridad de los sistemas informáticos han representado siempre un punto fijo de interés para la comunidad científica y, además, uno de los elementos clave para la difusión comercial de determinados productos.

Históricamente se han propuesto diferentes modelos teóricos de ataque (Denial of Service, Sybil Attack, Blackhole/Sinkhole Attack, Hello Flood Attack, Wormhole attack, etc.); una manera moderna y sugestiva de relacionarse al problema de la seguridad es plantearlo de acuerdo con la teoría de juegos en el cual un jugador propone un ataque y el otro se defiende.

Inevitablemente, el problema de la seguridad se repropone sobre redes inalámbricas de sensores [Shi1][Per1] tanto a nivel de acceso a los recursos como a nivel de protección de los datos. Problemas de mayor relevancia se consideran el propio entorno (inalámbrico), la posibilidad de capturar uno o más nodos e intentar de comprometer el entero sistema y el entorno a recursos limitados que impone el uso de mecanismos de seguridad más sencillo y, por lo tanto, más vulnerables. De hecho la gran mayoría de los mecanismos de seguridad para redes de sensores suelen ser simplificaciones de las correspondientes para redes ad-hoc.

El objetivo básico de un sistema de seguridad sobre redes inalámbricas de sensores [Pat1] es de proporcionar un buen nivel de defensa (acceso seguro y encriptación de datos) y, al

mismo tiempo, en el caso de ataque físico de un sensor evitar comprometer todo el sistema; el mecanismo diseñado tiene que ser evaluado en función de los recursos que requiere y, por tanto, proporcionar un compromiso entre nivel de seguridad proporcionado y recursos utilizados.

Técnicas típicas de seguridad usan llaves para habilitar un enlace [Cam1]; dependiendo del nivel de seguridad deseado y de los recursos disponibles se pueden diseñar mecanismos basados en la compartición de claves globales, en claves de grupo o, incluso, de pareja [Saz1]. La generación de las claves puede ser un punto relevante y puede basarse en varias asunciones (generación casual [Cha1][Hwa1] o de acuerdo con determinados modelos matemáticos). En general, la organización jerárquica de la red puede aconsejar una política de distribuciones de las claves basadas sobre principios de localidad de la información.

1.4.6 Configuración avanzada y Optimización

Los protocolos de configuración avanzada suelen proporcionar un nivel lógico adicional que complementa y se integra con el nivel lógico básico. Pueden necesitarse para garantizar altas prestaciones (o parámetros no funcionales) al crecer de la escala a través de arquitecturas multi-base que requieren *clustering* dinámico o para soporte de soluciones avanzadas (por ejemplo arquitecturas orientadas a roles) o para solucionar problemas específicos.

El más relevante es, seguramente, el *clustering* dinámico [Abb1] que será objeto explícito de la sección siguiente; entre los otros protocolos de configuración dinámica de relevancia se citan:

- *Protocolos para asignación de roles*: las arquitecturas orientadas a roles [Ort1] se basan sobre la definición de unos cuantos roles de base que tienen que cumplir con unos cuantos requisitos lógicos (ciertas propiedades de configuración básica), topológicos (por ejemplo un cierto número de vecinos) y/o físicos (cierto hardware en el caso de redes heterogéneas). Para que estas arquitecturas puedan trabajar correctamente se necesita evaluar las condiciones de configuración básica y de la red y asignar (eventualmente de forma optimizada) los roles en función de las condiciones detectadas [Rom2]. Evidentemente se trata, en la mayoría de los casos, de algoritmos centralizados aunque existan variantes distribuidas. Un ejemplo típico de rol, muy utilizado para optimizar aplicaciones de adquisición periódica de datos, es el data-collector; su función es de actuar como “hole” para los nodos circunstantes que envían datos; el data-collector se preocupa de comunicar con la estación base optimizando el throughput a nivel de aplicación.
- *Distribución aleatoria de parámetros* [Chg1][You1]: ciertas aplicaciones requieren la diseminación de ciertos parámetros de forma aleatoria, tal vez con unos vínculos o condiciones. Un ejemplo típico es la diseminación dinámica de los identificadores que no tiene que repetirse. Otros parámetros podrían relacionarse con aspectos de seguridad o de prioridad en la comunicación o de prioridad en la asignación de recursos.
- *Diseminación y configuración de tareas*: no siempre los nodos sensores actúan de forma aislada en la red; ciertas aplicaciones requieren tareas cooperativas entre sensores; dichas tareas necesitan configuración [Hog1], dinámica normalmente, especialmente en presencia de hardware homogéneo. De la misma forma, ciertas aplicaciones, en función de ciertos eventos externos (control por parte del usuario) o internos (condiciones detectadas), necesitan la configuración del comportamiento de

los nodos sensores; estos últimos pueden actuar todos de acuerdo a un mismo patrón o actuar de forma diferente; en ambos casos necesitan de configuración dinámica. La diseminación de tareas suele ser un mecanismo centralizado aunque soluciones distribuidas puedan ser funcionales a ciertos entornos aplicativos.

- *Otros protocolos:* otros protocolos orientados a proporcionar funcionalidades añadidas. Dichos mecanismos también pueden tener el objetivo de configurar parámetros complejos así como de garantizar alguna propiedad o a solucionar algún problema específico (de cálculo por ejemplo). Un ejemplo de protocolos que añaden funcionalidades en función de la aplicación a soportar son los protocolos de nivel de transporte (encargados de la transferencia libre de errores de los datos entre el emisor y el receptor, aunque no estén directamente conectados, así como de mantener el flujo de la red). Por sus propias características (transmisión de pequeñas cantidades de datos en modalidad convergente), las redes inalámbricas de sensores suelen asumir funcionalidades de transporte extremadamente reducidas [Gan1][Man1] o ausentes. Sin embargo, arquitecturas a soporte de aplicaciones caracterizadas por transferencia de flujos de datos importantes (redes de sensores multimediales por ejemplo) pueden presentar capas de transporte extendidas [Alm1][Bou2].

Independientemente de la presencia o menos de protocolos de configuración avanzadas, arquitecturas de protocolos complejas que incluyen varios mecanismos operantes simultáneamente suelen proponer una capa de optimización; el objetivo de esta capa es optimizar la comunicación (minimizando el número de mensajes intercambiados) multiplexando la información de los varios protocolos y mecanismos en mensajes optimizados. Esta última capa lógica no es de confundirse con la optimización a nivel de aplicación aunque, en muchos casos, puede coincidir especialmente en el contexto de aplicaciones que trabajan con ciclos periódicos.

1.4.6.1 Clustering

Una arquitectura orientada a clusters supone la división o partición del sistema principal en unos cuantos subsistemas que interoperan entre sí. Un cluster clásico se compone de una cabeza o header, normalmente con funciones de gestión y representación del entero cluster cara al sistema global, y por el contexto de los actores componentes el cluster que suelen organizarse de forma jerárquica. En el modelo más común la interacción entre clusters (o con el sistema de gestión centralizado) es gestionada por las cabeceras de los clusters.

En un sistema general la configuración mediante clusters puede aplicarse por varios motivos con objetivo la optimización de factores relacionados con la arquitectura, las infraestructuras o la información. El análisis de dichos sistemas no es de interés específico de este trabajo de tesis.

En el caso de las redes inalámbricas de sensores se presentan, básicamente, dos posibles aplicaciones de técnicas de clusterización:

- *La cabecera de los cluster es representada por una Estación Base* (Figura 1.7): es el caso más relevante en cuanto, en presencia de nodos sensores con recursos limitados, permite una mayor escalabilidad del sistema dividiendo un sistema formado por n nodos sensores en m subsistemas (clusters) compuestos medianamente por n/m nodos. La gestión de cada cluster es autónoma respecto a su cabecera por lo cual las rutas de comunicación resultan reducidas de un factor proporcional a m así como el

energy-hole. Por el contrario, tanto las prestaciones (tiempo de vida de la red) como los típicos parámetros no funcionales (fiabilidad y robustez) resultan beneficiarse de dicha organización lógica. La organización en clusters suele introducir una complejidad añadida a las infraestructuras de control así como, en el caso de configuración dinámica, la necesidad de configuración avanzada. Los aspectos positivos de la organización en clusters se consideran bastante más relevantes que los aspectos negativos así que esta técnica está progresivamente ganando importancia en el seno de las aplicaciones reales siendo, de hecho, la única técnica válida para garantizar ciertas prestaciones, en un contexto de fiabilidad y robustez, para arquitecturas operantes sobre larga escala [Abb1].

- *La cabecera de los clusters está representada por un nodo sensor:* es un caso realmente menos relevante del anterior; su aplicación, tal vez solamente teórica, se enfoca básicamente dentro de las arquitecturas orientadas a roles o, más en general, en un contexto de cálculo distribuido sobre redes de sensores. Su aplicación real resulta, por lo tanto, muy limitada.

Con referencia a clusters caracterizados por cabeceras representadas por Estaciones Base, se distinguen, básicamente, dos posibilidades en términos de configuración:

- *Configuración Estática:* cada nodo sensor se asocia a un cierto cluster a priori; este tipo de configuración puede aplicarse a contextos con topologías más o menos predefinidas o despliegues aleatorios controlados. Evidentemente no implica ningún mecanismo de configuración avanzado.

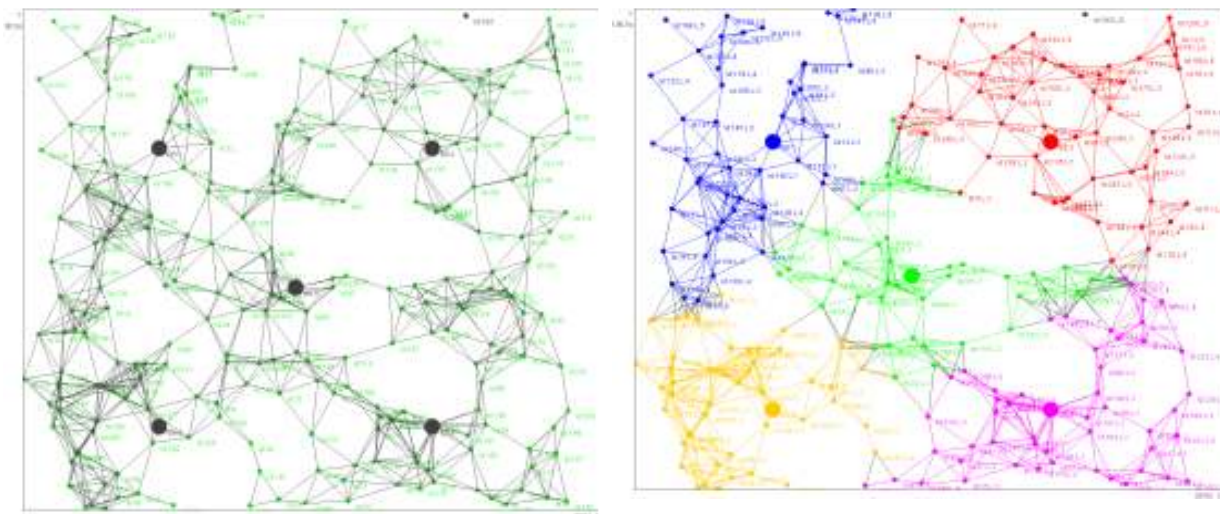


Figura 1.7. Ejemplo de configuración en clusters.

- *Configuración Dinámica:* al contrario del caso anterior, cada sensor se asocia dinámicamente a un cluster más que a otro; a lo largo del tiempo de vida de la aplicación el mismo sensor puede, por lo tanto, asociarse a cluster diferentes en función de las condiciones de entorno, de vínculos y restricciones. Potencialmente un nodo sensor puede asociarse a más que un cluster contemporáneamente. Evidentemente esta clase de solución (que requiere mecanismos de configuración avanzada) proporciona soluciones mucho más complejas y dinámicas que mejor se adaptan a despliegues completamente

aleatorios y a cambios significativos de topologías y condiciones de entorno. Un mecanismo dinámico de configuración avanzada en clusters se caracteriza por la política según el cual asocia los nodos a un cluster mas que a otro; se distinguen, normalmente, dos tendencias de base: una orientada a las prestaciones y otras orientada al balanceo de las dimensiones. La primera, con diferencia la más difusa, configura el cluster intentando minimizar las rutas en términos de saltos; evidentemente aspectos relacionados con la comunicación y con la optimización de los recursos suelen ser beneficiados aunque los diferentes clusters que componen el sistema puedan resultar bastante desequilibrados respecto al tamaño. Soluciones orientadas al balanceo, al contrario, intentan configurar los clusters de forma que el tamaño medio sea lo más parecido posible; esta clase de solución puede causar una cierta ineficiencia en términos de comunicación y, por lo tanto, se suele asociar a entornos de aplicación muy específicos. De todos modos, las dos soluciones descritas suelen converger al disminuir de la escala del sistema a paridad de densidad de nodos sensores y números de bases.

1.4.7 Aplicación

Exactamente como en las arquitecturas de protocolos tradicionales, los protocolos de nivel de aplicación suelen implementar las funcionalidades de la aplicación considerada apoyándose en las funcionalidades proporcionadas por las capas inferiores.

En el contexto de las redes inalámbricas de sensores, el nivel de aplicación suele implementar el software para la adquisición y la interpretación de la información proporcionada por los transductores o, bien, la implementación del algoritmo distribuido que define el comportamiento del nodo en la red.

Así como visto para capas inferiores, también el nivel de aplicación puede incluir una capa virtual de optimización; el objetivo es el mismo de lo relativo a protocolos más básicos: se pretende, básicamente, optimizar las tareas de comunicación consideradas notablemente más dispendiosas respecto a la elaboración local de datos. Las técnicas más comunes actúan, por lo tanto, para optimizar el throughput [Che1]: técnicas basadas en data-fusion [Yua1][Du1] intentan privilegiar, donde posible, la elaboración local de los datos generados (por ejemplo enviando tan solo la media sobre unos valores más que cada singulo valor); técnicas basadas en data-aggregation [Das1][Kal1] intentan, visto el pequeño tamaño de los datos generados, de agrupar mas tramas informativas (collación de datos) en cada mensaje de aplicación para minimizar el numero de mensajes transmitidos.

1.5 Tecnología para Middleware

Un número importante de arquitecturas reales suelen basarse en topologías formadas por más de una estación base, algunas veces organizadas de forma jerárquicas. La funcionalidad básica de este tipo de infraestructura es recoger la información, eventualmente integrarla y/o procesarla, y finalmente transmitirla hacia un centro de control remoto.

Un número creciente de redes de sensores inalámbricos se integra en sistemas complejos como sistemas empotrados. Más recientemente, también el modelo de explotación de las redes de sensores se ha complicado progresivamente y los servicios proporcionados por dichas redes se manejan dentro de Comunidades Virtuales [Fos2] siempre más complejas y

articuladas que requieren políticas de acceso y manejo de la información y de los servicios extremadamente complejas (normalmente *context-aware* o *content-aware*).

Todos estos aspectos imponen (o aconsejan) la utilización de un modelo middleware extremadamente flexible que pueda soportar eficientemente tanto sistemas propietarios como sistemas embebidos o interoperables, eventualmente integrando políticas de acceso y gestión compleja. El middleware es un software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas. Funciona como una capa de abstracción de software distribuida, que se sitúa entre las capas de aplicaciones y las capas inferiores (sistema operativo y red). El middleware abstrae de la complejidad y heterogeneidad de las redes de comunicaciones subyacentes, así como de los sistemas operativos y lenguajes de programación, proporcionando una API para la fácil programación y manejo de aplicaciones distribuidas.

En el contexto tecnológico actual, el modelo tecnológico de referencia al respecto es, sin duda, el modelo Web-service especificado por W3C [W3c1]. El concepto clave de un web-service es la clara división entre interfaz (definida según un modelo de datos interoperable) del servicio e implementación del mismo con el objetivo de enfatizar la interoperabilidad y la modularidad de los sistemas distribuidos de acuerdo con los más relevantes principios de la ingeniería del software.

Más recientemente, modelos avanzados de web-services se han diseñado con el intento de incrementar las prestaciones en el contexto del cálculo paralelo y distribuido o bien, la flexibilidad en entornos virtuales extremadamente complejos y articulados. Es el caso de Grid Computing [Fos1][Fos2][Glo1] que se caracteriza por un modelo de servicio con gestión compleja de su estado: un web-service tradicional es sin estado por definición. El entorno Grid proporciona, en su última versión, un modelo mucho más complejo en el cual un servicio puede ser sin estado (y por lo tanto compatible con el estándar W3C) o, bien, con estado manejable de forma estática (un servicio se instancia y se destruye explícitamente) o según políticas de gestión dinámicas (instancias temporizadas o relacionadas con eventos). Recientemente se han propuesto varias soluciones middleware para redes inalámbricas de sensores que se basan en tecnología Grid [Ben1][Gay1].

Una de las últimas tendencias en redes de sensores es su progresiva evolución conceptual hacia el modelo lógico-computacional conocido como “Internet-of-things” [Zou1][Gui1][Spi1]. El concepto básico es la consideración de cada nodo sensor como un objeto lógico siempre conectado y la información asociada (o generada) como la base para construir modelos de conocimientos más avanzados y abiertos de los proporcionados por servicios propietarios. Evidentemente, vista la imposibilidad (o el elevado coste) de conectar cada sensor a la Red como unidad independiente, este modelo implica un middleware de virtualización extremadamente eficiente y flexible que asocie una instancia de servicio a cada nodo. La solución Grid, en este sentido, aparece especialmente eficaz. Temas de investigación relacionados se consideran el modelo de datos para garantizar la interoperabilidad de la información en sistemas heterogéneos y la semántica asociadas para garantizar soporte para interacciones complejas.

Una posible alternativa en el desarrollo de componentes middleware es el modelo de computación multi-agente. Respecto al modelo a servicios, los Sistemas Multi-Agente son meno modulares, medianamente menos interoperables a causa de su modalidad de funcionamiento que, normalmente, requiere un acuerdo entre las partes para garantizar entornos seguros.

Las principales opciones son:

- Arquitecturas orientadas a servicios: *Web Services*
 - Entornos empotrados
- Organizaciones Virtuales y Grid-Computing
 - Open Grid Service Architecture (OGSA)
 - Frameworks para desarrollo en tecnología Grid
- Sistemas Multi-Agente
 - Negociación en Sistemas Multi-Agente
 - Plataformas Multi-Agentes

1.5.1 Arquitecturas orientadas a servicios: Web-Services

Una arquitectura orientada a servicio (SOA, Service Oriented Architecture) es una arquitectura de aplicación distribuida que proporciona interfaces para un conjunto de servicios que ejecutan procesos de negocio. Estos interfaces son típicamente interfaces de red basados en HTTP (SOAP [W3c3], REST [Xin1]) y son definidos empleando un lenguaje descriptivo como WSDL [W3c2] o WADL [Wad1]. En general, una SOA se basa en una serie de principios fundamentales como débil acoplamiento, existencia de un contrato, abstracción del servicio, reusabilidad, orquestación, autonomía y descubrimiento.

Una SOA basada en web services expone sus servicios mediante interfaces basados en HTTP, básicamente interfaces SOAP y REST. SOAP (Simple Object Access Protocol) es un protocolo que permite el intercambio de información estructurada en XML (eXtensible Markup Language) y típicamente utiliza HTTP como protocolo de nivel de aplicación, Los interfaces SOAP son descritos mediante el uso de WSDL (Web Services Description Language), que define, mediante XML, los servicios como una agrupación de puntos finales de red, o puertos. La definición abstracta de puertos y mensajes está separada de su uso concreto, permitiendo la reutilización. La mayoría de lenguajes de programación proporcionan herramientas que generan automáticamente plantillas de código SOAP y WSDL. REST (Representational State Transfer) es una forma alternativa de definir e identificar servicios web (modelo RESTful) o, de una manera menos estricta, cualquier interfaz sencilla capaz de transmitir datos sobre HTTP sin una capa adicional de mensajería como SOAP (modelo REST-like). Los interfaces REST se describen usando WADL (Web Application Description Language) que, de manera análoga a WSDL, es independiente del lenguaje y de la plataforma.

En la actualidad existen tecnologías para la implementación sencilla de soluciones basadas en SOA, fomentando el dinamismo, la modularidad y la simplicidad. En entornos Java, OSGi (Open Services Gateway Initiative) [Ogs1] está siendo adoptado cada vez más como plataforma de servicio, ofreciendo un modelo de programación de servicio sencillo, capaz de soportar despliegue dinámico. Esta plataforma proporciona un sistema de módulos dinámico para Java, un modelo de componentes y adaptabilidad en tiempo real. Según el consorcio oficial, OSGi Alliance, esta tecnología está siendo empleada por numerosas empresas, como Fortune 100, en todo tipo de mercados embebidos, desktop, corporativo y móvil. Además de OSGi, existen otros estándares para desarrollar plataformas SOA como son JBI (Java Business Intelligence) [Jbi1] y SCA (Service Components Architecture) [Sca1]. En entornos Microsoft, la meta-arquitectura .NET dispone de una framework para el

desarrollo de servicios web denominado WCF (Windows Communication Foundation) [Wcf1].

La modularidad de SOA hace de este paradigma la referencia también en el campo de los sistemas empotrados, de gran interés para redes inalámbricas de sensores; las arquitecturas empotradas pueden ser extremadamente sencillas o bien de complejidad creciente en función de la complejidad de la interacción entre los módulos que componen el sistema; normalmente, la interacción entre módulos implica la presencia de un middleware de soporte que suele proporcionar, por una parte, un entorno de ejecución global y, por la otra, un sistema de gestión.

La aplicación de arquitecturas orientadas a servicios y, más concretamente, su implementación a través de Servicios Web, conlleva numerosas ventajas de las que pueden beneficiarse los sistemas desarrollados. Una de las mayores ventajas de la tecnología de los Servicios Web es la utilización de protocolos estándar como HTTP. Esta medida permite el paso de mensajes entre servicios sin necesidad de preocuparse por los sistemas de seguridad y, en particular, los cortafuegos de las distintas organizaciones. En general, el uso de protocolos estándar facilita la interoperabilidad entre plataformas de distintos fabricantes.

Por otro lado, los Servicios Web aportan interoperabilidad entre aplicaciones software con independencia de su implementación, del lenguaje de programación utilizado y de la plataforma en que estén instaladas. Esto ha provocado que esta tecnología se utilice, cada vez con más frecuencia, para permitir el uso de sistemas heredados por parte de las nuevas aplicaciones que se desarrollan en el ámbito de una organización. Por su parte, la independencia con respecto a la implementación aporta flexibilidad al sistema, de forma que se puede modificar la implementación del servicio sin necesidad de actualizar su descripción. Además, los Servicios Web permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.

Los Servicios Web y el paradigma SOA poseen algunos problemas, como por ejemplo su rendimiento más bajo si comparado con otros modelos de computación distribuida como RMI, DCOM o CORBA. Este problema viene ocasionado por el uso de XML como lenguaje para la codificación de mensajes. En particular, el problema radica en la necesidad de parsear y componer los ficheros XML, lo cual requiere mayor capacidad computacional y hace que las aplicaciones se ejecuten más lentamente. Como resultado del esfuerzo por resolver este limitante, la W3C creó en 2003 el grupo de trabajo para la caracterización binaria de XML (“XML Binary Characterization Working Group”, <http://www.w3.org/XML/Binary/>) que en 2005 se transformó en el grupo de trabajo para el intercambio eficiente de XML (“Efficient XML Interchange Working Group”, <http://www.w3.org/XML/EXI/>). El objetivo final es el desarrollo de una especificación de un formato de codificación que permita el intercambio eficiente de documentos XML.

Otro de los inconvenientes de los Servicios Web es que el grado de madurez de algunas de sus especificaciones no puede compararse con el grado de desarrollo de estándares abiertos en computación distribuida como CORBA. Hasta ahora, los investigadores en Servicios Web se han preocupado más de cómo realizar tareas de alto nivel como el descubrimiento y la invocación de servicios, dejando sin cubrir aspectos más básicos como la seguridad, la gestión de transacciones, etc. Hoy en día existen numerosos grupos de trabajo para el desarrollo de este tipo de especificaciones (WS-Security, WS-Trust, WS-Policy, WS-Privacy, WS-ReliableMessaging, WS-Routing, WS-Addressing, etc.). Esto introduce un plus de riesgo cuando se trata del desarrollo de sistemas comerciales que deben ser aplicados en entornos reales.

Disponiendo de todos los componentes que constituyen la tecnología de Servicios Web, cualquiera podría hacer uso de los servicios ofrecidos por todos los Servicios Web disponibles en Internet. Sin embargo, y a medida que la Web crece en tamaño y diversidad, cada vez es más necesaria la automatización de aspectos relacionados con los Servicios Web como el descubrimiento, la selección, la composición y la invocación. De hecho, una de las principales ventajas de esta tecnología es la posibilidad de componer de forma dinámica servicios utilizando componentes software independientes y reutilizables. El problema, entonces, es que la tecnología actual, en torno a UDDI, WSDL y SOAP no proporciona los medios para conseguir este dinamismo.

1.5.1.1 Entornos empotrados

Una de las ventajas de las Arquitecturas Orientadas a Servicios es su extrema modularidad que hace de este paradigma la referencia en el campo de los sistemas empotrados; las arquitecturas empotradas se caracterizan por componer el sistema principal con diferentes componentes preexistentes o diseñados para proporcionar servicios específicos y, por lo tanto, potencialmente independientes o funcionalmente autónomas. Las arquitecturas empotradas pueden ser extremadamente sencillas (interacción nula entre los módulos que actúan de forma independiente) o bien de complejidad creciente en función de la complejidad de la interacción entre los módulos que componen el sistema; normalmente, la interacción entre módulos implica la presencia de un middleware de soporte que suele proporcionar, por una parte, un entorno de ejecución global y, por la otra, un sistema de gestión.

En este contexto, evidentemente, aspectos típicos de la ingeniería del software que favorecen la modularidad de los sistemas, la reusabilidad del código, la escalabilidad y el acceso estandarizado y pervasivo de los componentes se consideran de importancia capital para limitar los costes e incrementar la eficacia de los sistemas embebidos. En referencia a los sistemas empotrados, normalmente el término “eficiencia” se suele referir a diferentes aspectos del tiempo de vida del sistema: en la fase de diseño, la modularidad es evidentemente el aspecto determinante en cuanto a que incrementa la abstracción del diseño a nivel de módulo; en la fase de implementación el aspecto determinante es la accesibilidad del componente que simplifica la implementación de interacciones complejas a nivel de middleware; finalmente, a nivel de ejecución pueden intervenir diferentes factores aunque se consideran determinantes la escalabilidad y, eventualmente, la portabilidad.

Aunque CORBA [Cor1], en su versión empotrada (CORBA/E [Cor2]) se considera un paradigma de desarrollo extremadamente eficiente y completo, el modelo orientado a servicios esta progresivamente afirmándose como alternativa más concreta y sencilla. Tanto el modelo CORBA como el modelo SOA pueden garantizar eficiencia a nivel de diseño, implementación y ejecución [Thr1][Sch1][Bar2].

Realmente, no hay una correspondencia teórica completa entre el concepto de modulo (definido en general como componente autónomo o auto-controlado de un sistema que se caracteriza por su diseño que facilita su integración dentro del sistema principal) y servicio (definido como función, en principio sin estado, auto-contenida y accesible mediante una interfaz bien definida). El concepto de servicio aparece más específico que el de módulo pero, al mismo tiempo, parece cumplir con todas las especificaciones relevantes de módulo. Utilizando el modelo orientado a servicios, resulta bastante natural dividir el sistema en subsistemas, cada uno compuesto por servicios.

Todas las tecnologías actuales relativas a la computación orientada a servicios ofrecen implícitamente pleno soporte para arquitecturas empotradas en cuanto a que están caracterizadas por interfaces definidas de acuerdo con especificaciones estándares (por ejemplo WSDL) y pervasividad respecto a la implementación (normalmente con soporte a más de un lenguaje de programación). Además, durante los últimos años, los modelos avanzados de servicios proporcionan alternativas relevantes en cuanto a la infraestructura de gestión del estado del servicio, que puede, al margen de sus especificaciones originarias, ser con estado (state-full service) eventualmente con gestión dinámica del mismo. También los modelos de interacción están en continua evolución; por ejemplo el clásico modelo de registro e indexación de servicios basados en criterios funcionales se puede integrar con modelos semánticos que habilitan mecanismos de descubrimiento y gestión extremadamente avanzados. Estos avances pueden resultar importantes en los sistemas empotrados que se presentan siempre más dinámicos en todos sus aspectos en cuanto parte de organizaciones virtuales: el modelo computacional orientado a servicios proporciona un soporte relevante en todo aquellos entornos que se caracterizan por una gestión avanzada de la información (por ejemplo servicios context-aware/content-aware), por la gestión avanzada de los recursos (por ejemplo servicios multiple-resource) o por binding dinámico (la implementación del servicio se decide en tiempo de ejecución en función de alguna política dinámica) de los servicios.

La mayor parte de las nuevas tecnologías orientadas a servicios proponen versiones compactas y portátiles de sus frameworks de referencia para favorecer ciertos aspectos prácticos de los sistemas empotrados como la eficiencia en ejecución o el explícito soporte de sistemas con recursos limitados.

Soluciones SOA para entornos con recursos limitados suelen ser simplificaciones de las correspondientes para entornos generalizados.

DPWS (Devices Profile for Web Services [Dpw1]) define un conjunto mínimo de especificaciones para implementar un entorno web services completo (mensajes, descubrimiento, descripción y eventos) en dispositivos con recursos limitados. En otras palabras, DPWS es un entorno web service compacto.

El enfoque propuesto por DPWS es parecido, por objetivos y funcionalidades, a Universal Plug and Play (UPnP); DPWS se diferencia por su estructura orientada a servicios que lo hace más flexible y adaptable a diferentes escenarios; además se considera potencialmente extensible respecto a muchas de las funcionalidades propuestas en función de los requisitos de aplicaciones concretas y recursos disponibles.

En DPWS dos tipos de servicios pueden ejecutarse en un dispositivo: hosting services y hosted services. Los Hosting services están directamente asociados al dispositivo en el cual se ejecutan y, tiene, por lo tanto, un rol fundamental en procesos típicos y básicos como el descubrimiento. Los Hosted services implementan servicios funcionales de carácter aplicativo.

Típicas funcionalidades básicas son implementadas por el Discovery service (conjunto funcional que permite a un dispositivo de descubrir otros dispositivos o servicios y de ser descubierto), por el Metadata exchange service (acceso dinámico a servicios y metadatos relacionados) y por el Publish/subscribe eventing service (gestión de la subscripción de servicios o dispositivos externos a la notificación asíncrona de mensajes generados a consecuencia de eventos internos).

DPWS utiliza la gran mayoría de los estándares Web Service: WSDL 1.1, XML Schema, SOAP 1.2, WS-Addressing, WS-MetadataExchange, WS-Transfer, WS-Policy, WS-Security, WS-Discovery y WS-Eventing.

Actualmente las plataformas Windows Vista y Windows Embedded CE6R2 integran DPWS en una pila llamada WSDAPI que es parte de la tecnología Windows Rally.

1.5.2 Organizaciones Virtuales y Grid-Computing

La sociedad moderna presenta una continua y siempre creciente dispersión, a nivel geográfico y organizativo, de recursos, humanos en particular, y una notable jerarquización y heterogeneidad a nivel individual e institucional (Organizaciones Virtuales [Fos2]). En este contexto, ya no es posible pensar en términos acceso estáticos a recursos y esto se traduce, en campo tecnológico, en la búsqueda de soluciones innovadoras y altamente avanzadas, aptas a soportar aplicaciones realmente complejas en términos de prestaciones y calidad de servicios, compartición (centralmente coordinada) de recursos sobre larga escala y un alto nivel de seguridad. Además reglas de negocio complejas, profundamente diferentes entre ellas, pueden condicionar el diseño y la implementación de las arquitecturas siempre más orientadas a reflejar políticas de gestión context-aware. La abstracción de los recursos y la interoperabilidad de las infraestructuras son elementos clave desde un punto de vista de las tecnologías concretas [Fos2].

En el contexto de las organizaciones virtuales, se pueden definir “computacional Grid” y “Grid Problem” de la siguiente forma:

- *Computacional Grid*: infraestructura hardware y software que proporciona acceso fiable, persistente, dinámico y de bajo coste, a recursos de cálculo de alto nivel [Fos1].
- *Grid Problem*: compartición coordinada, fiable, segura entre conjuntos dinámicos de individuos e instituciones, orientada al logro de uno o más objetivos comunes [Fos1].

Antes de Grid Computing, ninguno de los paradigmas computacionales distribuidos parecía soportar, de manera completa y eficiente, el Grid Problem en el contexto social y tecnológico de una organización virtual compleja. Grid Computing pretende representar la ambiciosa respuesta de la comunidad científica a las exigencias de un Organización Virtual. Es evidente que la definición de Grid Computing es realmente (y voluntariamente) genérica y por muchos aspectos ambigua; aunque el concepto de Grid Computing no se identifica unívocamente en una tecnología concreta, la mayoría de las soluciones concretas tienden a extender los paradigmas de cálculo distribuidos tradicionales sin estado (especialmente web-service) con particulares infraestructuras que introducen el estado y, normalmente, una gestión flexible del mismo. Se considera oportuno remarcar unos cuantos puntos que son comúnmente causa de malas interpretaciones:

- Grid Computing no es, de manera absoluta, una alternativa a Internet, sino una arquitectura de protocolos, orientada a servicios, diseñada sobre Internet misma; en este sentido, la interpretación más correcta es considerar las aplicaciones diseñadas sobre Grid Computing como una nueva generación de aplicación en red.
- No hay porqué pensar en Grid Computing en términos de restricciones de acceso.
- Desde un punto de vista más amplio y teórico, Grid Computing se puede ver como una máquina virtual distribuida.
- Aunque se intente insertar Grid Computing en el contexto tecnológico actual, una

revisión, más o menos profunda, de los métodos y de los modelos de diseño y proyecto de sistemas es necesaria para aprovechar las características proporcionadas para este paradigma de cálculo.

Se propone, a continuación, una descripción teórica de la arquitectura de protocolos de Grid Computing (OGSA) y la descripción de los principales paquetes para desarrollo en dicha tecnología. Como un marco de referencia que ha sido empleado en la siguiente tesis.

1.5.2.1 Open Grid Service Architecture (OGSA)

La descripción más efectiva y directa de una arquitectura Grid es el análisis de su estructura protocolar teórica (Figura 1.8). La arquitectura protocolar descrita es bastante coherente con la estructura de base de las implementaciones reales a menos de dos precisiones:

- Las arquitecturas reales integran los protocolos de nivel red y transporte de Internet.
- Cada nivel descrito no es realmente un único protocolo o mecanismo, sino un conjunto más o menos complejo de protocolos y mecanismos.

Una OGSA se compone básicamente de cinco niveles de protocolos; cada una de esta capa software es brevemente descrita a continuación:

- *Fabric*: Una de las ideas clave de Grid Computing es el concepto de recurso virtual; el nivel *Fabric* es exactamente el nivel de virtualización de los recursos, en el cual, se proporcionan las interfaz de acceso a los recursos y un conjunto de servicios orientados a facilitar operaciones típicas sobre recursos físicos como localización, interrogación, monitorización y gestión. El conjunto de posibles clases de recursos reales es prácticamente infinito; independientemente del tipo de recursos (cálculo, almacenamiento, recurso de red), el nivel *Fabric* proporciona una descripción computacional del recurso (o del conjunto de recursos) y de los servicios que el recurso (o el conjunto de recursos) proporciona.
- *Connectivity*: El nivel *Connectivity* es directamente diseñado sobre la información virtual proporcionada por el nivel “*Fabric*”. Proporciona, básicamente, dos clases de servicios: servicios relacionados con la comunicación y servicios relacionados con la seguridad. Los primeros garantizan el cambio de datos entre recursos y todos los servicios relacionados: naming, routing y transporte. Los servicios orientados a la seguridad están básicamente enfocados a la posibilidad de garantizar políticas de acceso y autenticación flexibles y fiables (autenticación sencilla, posibilidad de delegar todos o una parte de los privilegios poseídos, integración con niveles de seguridad locales, políticas de acceso inteligente basadas en principios de identidad local, conjunta y transitiva).

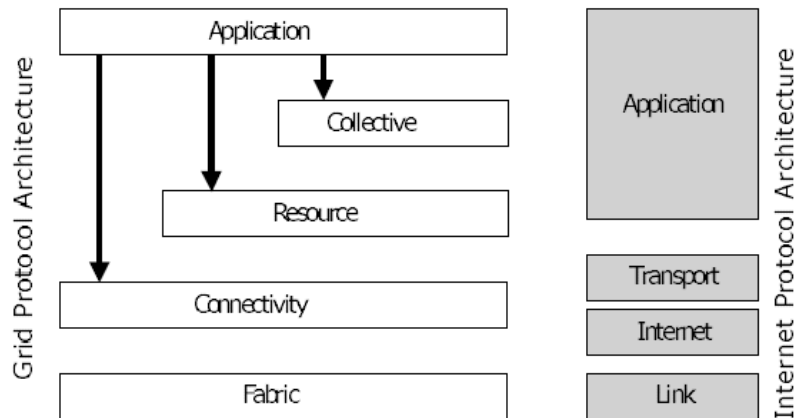


Figura 1.8. Arquitectura protocolar de una *Open Grid Service Architecture* (OGSA). Figura original de [Fos2].

- *Resource*: Proporciona un conjunto de servicios orientado al soporte para la compartición, centralmente coordinada, de recursos singulares. Más concretamente, implementa mecanismos para la gestión de la información relativa a estado y estructura del recuso y mecanismos relativos a la compartición del recurso.
- *Collective*: Es el complemento natural y lógico del nivel “Resource”; proporciona todas las funcionalidades necesarias para compartir recursos múltiples. Un recurso múltiple es intrínsecamente más complejo que su correspondiente en versión de recurso singular en tema de gestión local y global.
- *Application*: El nivel de aplicación es un conjunto de protocolos de alto nivel que permiten utilizar de manera, más o menos directa, los servicios proporcionados por una OGSA. Es un nivel bastante “abierto” en términos de extensibilidad. Incluye, por una parte, implementaciones de protocolos existentes (como FTP) en Grid (GridFTP), por la otra un conjunto de protocolos innovadores orientados a altas prestaciones.

1.5.2.2 Frameworks para desarrollo en tecnología Grid

Como se puede deducir observando la arquitectura protocolar de OGSA propuesta en el apartado anterior, aunque bien definida y orientada a solucionar una clase concreta (aunque amplia) de problemas, OGSA es realmente más próxima al concepto de modelo de referencia que de arquitectura real.

Este aspecto es bastante relevante a la hora de diseñar y desarrollar concretamente sistemas Grid; paquetes concretos para el desarrollo podrían presentar características relativas a protocolos y mecanismos bastante diferentes; por una parte este aspecto garantiza más opciones y, probablemente, más flexibilidad; al mismo tiempo afecta negativamente aspectos clave como es la interoperabilidad y la compatibilidad que en un contexto de organización virtual podría ser uno de los elementos clave a nivel comercial.

Desde el momento de su definición, Grid Computing ha sido objeto de gran atención por parte de unas de las principales empresas del campo tecnológico; tanto IBM, como HP, CISCO y otros colosos de la informática mundial, han desarrollado (o están desarrollando) proyectos de gran entidad y portada en tecnologías Grid.

A nivel de soluciones concretas, el mundo de la investigación hace, generalmente, referencia al paquete *Globus Toolkit* (GT), proporcionado por Globus Alliance [Glo1], y recientemente llegado a la versión 4 [Fos3].

El framework Globus Toolkit extiende o integra el concepto tradicional de servicio según un modelo basado en la coexistencia entre Web Service y recurso para proporcionar WS-Resources. La arquitectura propuesta por Globus mantiene una cierta coherencia con el concepto común de web-service (división y absoluta independencia entre interfaz y implementación del servicio para garantizar interoperabilidad, transparencia y, eventualmente, binding múltiple, etc) proponiendo, pero, un modelo de recurso mucho más extendido y avanzado respecto a la gestión del estado de la instancia de servicio. Un Web Service supone, por su propia definición, ausencia de estado (que es generalmente implícito o local); esta tendencia se considera una buena estrategia para programación en contextos distribuidos.

La actual versión de Globus Toolkit, GT4, propone múltiples posibilidades en términos de modelo de gestión del estado de una instancia de servicio: puede simplemente seguir el modelo de servicio sin estado (servicio transiente) propuesto por WSs tradicionales, o, bien, un modelo con estado, eventualmente gestionado de forma dinámica.

El caso de instancia con estado incluye servicios permanentes (su ciclo de vida tiene que ser explícitamente gestionado tanto en su creación como en su destrucción) y servicios semipermanentes o a estado dinámico (el servicio se crea por un cierto tiempo de vida; se auto-destruye al final de este tiempo en ausencia de renegociación del tiempo de vida). GT4 utiliza un estándar para la definición de interfaz (WSRF) que garantiza compatibilidad con Web Service estándar. Su esquema simplificado se representa en Figura 1.9.

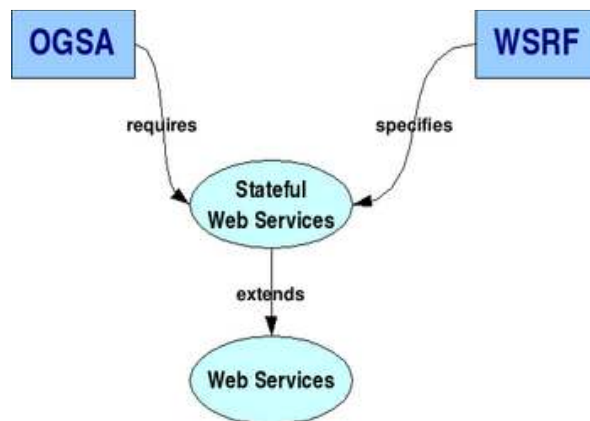


Figura 1.9. Esquema básico de la arquitectura de GT4. Figura original de [Fos3].

GT4 gestiona los aspectos de seguridad a través del GSI (Grid Security Infrastructure) que extiende y completa el Transport Layer Security Protocol (TLS) con el uso de certificados X.509 y con GAA (Generic Authorization Access) para la gestión de la integración de los niveles locales de seguridad.

La información relacionada con los recursos está gestionada por el GRIP (Grid Resource Information Protocol) basado sobre LAPD; el protocolo GRRP (Grid Resource Registration Protocol) se usa para registrar los recursos; GRAM (Grid Resource Access and Management) trabaja sobre HTTP y proporciona un conjunto de servicios orientados a la instanciación, monitorización y gestión de los recursos.

GT4 incluye otros protocolos, de nivel de aplicación, que se basan en los mencionados o que son la versión en tecnología Grid de protocolos existentes (GridFTP por ejemplo).

Una de las versiones más exitosas de GT es la versión 3 que implementa de forma bastante rigurosa el concepto de WS-Resource. GT3 siempre asume un modelo de servicio con estado, eventualmente gestionado dinámicamente; se puede considerar, por lo tanto, bastante menos flexible respecto a su evolución. El estándar para la definición de interfaz es OGSF 1.0 (Open Grid Service Infrastructure).

Evidentemente GT propone una solución Grid, completa y efectiva, que extiende el concepto habitual de WS sin revolucionarlo. Más recientemente, *GridGain* [Gri1] ha propuesto, en el contexto de las tecnologías Grid, una válida y original alternativa a GT. GridGain propone un modelo de programación Grid estructuralmente diferente basado completamente en programación JAVA (en muchos casos más amigable) manteniendo las características de fiabilidad, flexibilidad e interoperabilidad de GT. Recientemente GridGain se está claramente orientando y especializando en contextos de Cloud Computing.

1.5.3 Sistemas Multi-Agente

Un Sistema Multi-Agente (SMA) se define como una red poco acoplada de agentes software que interactúan para resolver problemas que van más allá de las capacidades o conocimiento individual de cada uno de los componentes que resuelven problemas [Olf1].

Cuando se forma un SMA, surge la necesidad de disponer de unos mecanismos de coordinación y de un lenguaje para permitir la comunicación entre ellos.

Los agentes pueden ser cooperativos o competitivos. En el caso de los agentes cooperativos, los mecanismos de cooperación más comunes son las estructuras organizacionales, la planificación multi-agente, redes de contratos y cooperación funcionalmente exacta. En el caso de agentes competitivos es necesario un mecanismo de negociación, de los que los más comunes son la formación de coaliciones, los mecanismos de mercados, la teoría del regateo, la votación, las subastas y la asignación de tareas entre dos agentes.

Para que la comunicación entre agentes pueda llevarse a cabo se han definido varios lenguajes, entre los que destacan:

- KQML: Knowledge Query and Manipulation Language [Fin1].
- KIF: Knowledge Interchange Format.
- FIPA-ACL: FIPA Agent Communication Language. Es estándar desde el 2005 [Fip2].

El uso de SMA implica una serie de beneficios sobre la aplicación de otros paradigmas de programación, por ejemplo:

- Fiabilidad: Los SMA son más robustos y tolerantes a fallos.
- Modularidad y escalabilidad: los agentes pueden ser añadidos o borrados de un entorno concreto sin la necesidad de interrumpir o detener el sistema.
- Adaptabilidad: los agentes tienen la habilidad de reconfigurarse de forma que se adapten a fallos y cambios.
- Concurrencia: los agentes son capaces de razonar y realizar tareas en paralelo.
- Dinamismo: los agentes pueden colaborar dinámicamente para compartir sus recursos y resolver problemas.

1.5.3.1 *Negociación en Sistemas Multi-Agente*

Existen tres técnicas o aproximaciones principales a la negociación automatizada:

- Técnicas basadas en teoría de juegos [Wen1]: intentan determinar la estrategia óptima analizando la interacción como si fuera un juego entre participantes idénticos, y buscando su equilibrio. Generalmente asumen que tienen infinitos recursos computacionales y que el espacio de los resultados es completamente conocido, lo cual suele ser falso y supone una limitación computacional importante.
- Técnicas basadas en heurística [Mad1]: ofrecen aproximaciones a las decisiones que se pueden llegar a tomar a partir de estudios basados en teoría de juegos. Sin embargo aplicando estas técnicas se obtienen resultados sub-óptimos (ya que no examinan el espacio completo de resultados posibles) y es muy difícil predecir como el sistema se comportará.
- Técnicas basadas en argumentación [Lop1]: Intentan superar las limitaciones de los anteriores modelos permitiendo a los agentes intercambiar información adicional o argumentar acerca de sus creencias y otras actitudes mentales durante el proceso de negociación, con el objetivo de aumentar la probabilidad de alcanzar un acuerdo y maximizar la calidad de éste intercambiando argumentos que influyan en el estado del resto de los agentes.

1.5.3.2 *Plataformas Multi-Agentes*

Las herramientas que permiten la interconexión y ejecución de de los agentes de un SMA se denominan plataformas de agentes o multi-agentes.

Hay una gran cantidad de plataformas que siguen el modelo de referencia estándar definido por FIPA (The Foundation for Intelligent Physical Agents) [Fip1], que engloba a grupos industriales y de investigación con el objetivo de estandarizar los modelos y tecnologías de agentes. De todas formas pueden hallarse diferencias entre las diferentes implementaciones debido a diferentes interpretaciones del estándar, por lo que suele ser imposible la correcta comunicación entre agentes implementados en diversas plataformas.

Las plataformas más significativas son las siguientes:

- JADE (Java Agent DEvelopment Framework). Es la implementación más extendida del estándar FIPA. Está basada en una arquitectura peer-to-peer, ha sido desarrollada completamente en Java y es de código abierto. JADE [Jad1] ha sido diseñada de acuerdo con los más comunes principios básicos en entornos distribuidos: interoperabilidad, uniformidad y portabilidad, facilidad de uso, filosofía de “pago por uso” (lo que no se utiliza no es necesario conocer y no supone coste computacional).
- FIPA-OS (FIPA Open Source) [Fip3]. Es un marco de trabajo de agentes desarrollado con el fin de construir plataformas multi-agente heterogéneas, agentes y servicios que se ajusten a los estándares FIPA.
- ZEUS [Zeu1]. Aglutina un conjunto de herramientas con entorno visual que facilitan el desarrollo de sistemas multi-agente, así como proporcionar una infraestructura en la que puedan ser ejecutados.

1.6 Soporte semántico en sistemas complejos

Las tecnologías semánticas pueden considerarse como una extensión de las tecnologías convencionales de información [Jan1]. Lo que significa que todas las tecnologías semánticas son tecnologías de información pero que no todas las tecnologías de información pueden ser consideradas tecnologías semánticas. Ejemplos son XML, bases de datos y otras tecnologías de información en las que la semántica de la información no está representada de forma explícita. Para comprender las principales propiedades de las tecnologías semánticas es necesario observar el significado del término semántico, consistente en que explícitamente además del dato está representado su significado y la lógica del programa. Representación que puede ser comprendida e interpretada tanto por humanos como por máquinas, haciendo posible que diferentes programas o equipos puedan trabajar con la semántica sin apenas intervención humana.

Las tecnologías semánticas proporcionan la extracción del significado de los datos, así como las reglas que definen el enlace entre las aplicaciones, los datos y el significado de los datos. Este hecho indica que las aplicaciones desarrolladas sobre tecnologías semánticas son mucho más flexibles respecto a mecanismos de comunicación, nuevas formas de codificar los datos, cooperación con aplicaciones no conocidas a priori. Lo que por otra parte permite que los desarrolladores empleen menos tiempo en la programación de aplicaciones, y puedan emplear más tiempo en la definición de modelos para los diferentes ámbitos de funcionamiento.

Todos los problemas que las tecnologías semánticas pueden resolver, pueden ser resueltos también por mecanismos convencionales relacionados con tecnologías de la información. Un ejemplo es la gestión de información semántica, ya que una aplicación tradicional podría gestionarla, pero los programadores deberían hacerlo de una forma explícita en la fase de desarrollo. El principal problema es el coste, en desarrollo y en la adaptación ante un cambio del significado de los datos. Por el contrario una aplicación desarrollada sobre tecnologías semánticas tendría la semántica representada fuera del código de la aplicación, haciendo que sea mucho más dinámica y flexible ante cambios de requerimientos, ya que un cambio en la semántica no requeriría una recompilación del código fuente. La principal razón es que las tecnologías semánticas ofrecen una separación entre el aparato de razonamiento y la lógica del programa. En otras palabras, ya que la información “contiene” su significado, los actores que interactúan a través de modelos semánticos son más ligeros porque se preocupan, básicamente, de interpretar un modelo de datos estándar más que implementar políticas o reglas de razonamiento a medida.

La definición explícita y formal de los modelos semánticos, hace que la unión y el mapeo de los mismos sean más sencillos y permite crear el efecto de la existencia de un modelo de datos común, aunque este realmente no exista. Mediante la definición de interrogaciones sobre este modelo, se pueden definir diferentes vistas sobre la información disponible. En pocas palabras la información es más fácil de combinar y reutilizar y por tanto es mucho más adaptable.

Las tecnologías semánticas permiten que la información especificada según varios esquemas coexista en un mismo espacio de almacenamiento. Haciendo que esta familia de tecnologías sea mucho más abierta y permita extensiones de modelos de datos, muy útiles en aplicaciones multi-dominio.

Más concretamente, los beneficios de las tecnologías semánticas se pueden resumir en los siguientes puntos:

- Ayuda en el proceso de adquisición del conocimiento. Con el uso de principios

ontológicos, la información toma sentido, pasa a formar parte de un microcosmos en el que ésta es clasificada y relacionada. Evidentemente la ingeniería de los modelos de conocimiento es a la base de cualquier sistema: los beneficios introducidos se pueden considerar, en una cierta medida, proporcionales a la expresividad y flexibilidad de las ontologías y modelos de conocimiento.

- Descubrimiento, búsqueda, consulta y recuperación automatizada de la información. Estructuras de datos semánticos pueden mejorar significativamente la relación entre datos disponibles e información/conocimiento en cuanto el mismo esquema incluye los datos y el “código” para interpretarlos.
- Modelo de interoperabilidad (interoperabilidad semántica [Kha1]) notablemente más avanzado respecto a otros modelos (interoperabilidad básica o funcional). Este aspecto es absolutamente fundamental tanto para la interacción entre sistemas como para el acceso y la interpretación de la información por parte de aplicaciones de largo alcance (web-centric por ejemplo).
- Integración y fusión de información de fuentes semánticamente heterogéneas, la principal idea relacionada con la integración de la información semántica, es la de resolver el desacople entre diferentes formatos a nivel semántico. Estableciendo una separación entre las ontologías y los sistemas, se pueden definir mapeos entre las diferentes ontologías empleadas, debido a que las mismas son modelos formales basados en la lógica y es posible emplear herramientas automáticas que realicen dichos mapeos. Esto puede ser empleado para integrar información procedente de diferentes fuentes de forma automática, el resultado sería un mecanismo de resolver y gestionar interacciones semánticas.
- Interrogaciones federadas, consiste en enviar una interrogación a un motor centralizado, el cuál realiza una traducción y envía sub-interrogaciones en las diferentes bases de datos. Mediante la combinación de las respuestas a estas interrogaciones, se puede obtener nueva información no disponible previamente. Este es un mecanismo muy beneficioso en el entorno de un sistema de sistemas. El valor añadido que las tecnologías semánticas dan a esta técnica radica en la forma de almacenar los datos, así como a las ontologías. Las interrogaciones federadas de tipo semántico utilizan la integración de información semántica para interrogar las bases de datos en diferentes organizaciones mientras se preserva la autonomía de las bases de datos.
- Capas de inteligencia extendidas. Como afirmado en precedencia, un esquema ontológico incluye la información y su semántica, expresada como conjunto de esquema jerárquico y reglas o relaciones lógicas entre los varios conceptos, campos y propiedades. Estas características del modelo de datos habilitan la deducción automatizada de información así como diferentes modelos de computación (ontology-driven, ontology-aware).

Las tecnologías semánticas se pueden interpretar como una revolución o, bien, como un concepto progresivo. Vista la intensa actividad científica que está interesando diferentes (aunque relacionados) sectores de la tecnología de la información es difícil concretar las posibles desventajas introducidas por modelos y tecnologías semánticas. Seguramente, a pesar de los muchos beneficios, los emergentes modelos semánticos, si aplicados de forma masiva, podrían resultar (por lo menos considerando las soluciones actuales)

significativamente más rígidos respecto a los correspondientes sintácticos introduciendo filtros indeseados y otros motivos de decaimiento de las prestaciones.

Como más veces mencionado, los modelos semánticos, fundamentalmente, se basan sobre modelos avanzados de descripción y especificación del conocimiento (ontologías). Se propone a continuación un breve análisis de los principales modelos para la definición de ontologías. Aspectos metodológicos de la ingeniería del conocimiento semántico son temas de investigación abiertos, de notable complejidad y de absoluto interés científico; un análisis exhaustivo en este sentido se considera fuera de los objetivos del documento.

1.6.1 Estándares para definición de esquemas semánticos

Se propone, a continuación, una breve descripción de unos de los modelos actualmente más relevantes en temas de ingeniería del conocimiento semántico. Dichos modelos son fundamentales desde el punto de vista de la interoperabilidad entre sistemas:

- RDF (Resource Description Framework) y RDF Schema
- OWL (Ontology Web Language)
- SWRL (Semantic Web Rule Language)

1.6.1.1 RDF y RDF Schema

El Marco de Descripción de Recursos (del inglés Resource Description Framework, RDF [Rdf1]) es un framework para metadatos en la World Wide Web (WWW), desarrollado por el World Wide Web Consortium (W3C). El origen de RDF se debe a Ramanathan V. Guha cuando trabajaba en Apple Computer en su forma inicial conocida como MCF, más tarde continuada durante su etapa en Netscape Communications Corporation. Este modelo se basa en la idea de convertir las declaraciones de los recursos en expresiones con la forma sujeto-predicado-objeto (conocidas en términos RDF como tripletes). El sujeto es el recurso, es decir aquello que se está describiendo. El predicado es la propiedad o relación que se desea establecer acerca del recurso. Por último, el objeto es el valor de la propiedad o el otro recurso con el que se establece la relación. La combinación de RDF con otras herramientas como RDF Schema y OWL permite añadir significado a las páginas, y es una de las tecnologías esenciales de la Web semántica. La terminología proviene de la lógica y de la lingüística en las que las estructuras predicativas se utilizan también para dar significado a las representaciones sintácticas.

RDFS o RDF Schema o Esquema RDF [Rdf2] es una extensión semántica de RDF. Un lenguaje básico de ontologías que proporciona los elementos básicos para la descripción de vocabularios. La primera versión fue publicada en abril de 1998 por la W3C, la versión actual de la recomendación fue publicada en febrero de 2004 también por la W3C. Existen actualmente otros lenguajes de ontologías más potentes y expresivos, como puede ser OWL.

1.6.1.2 OWL

OWL [Owl1] es un lenguaje a medida para la definición de ontologías y otros modelos semánticos. OWL pretende ampliar la expresividad de los modelos respecto a otros estándares como XML, RDF, y RDF Schema (RDF-S) proporcionando modelos de datos sobre esquemas semánticos. Más concretamente OWL es un lenguaje de marcado para

publicar y compartir datos usando ontologías en la WWW. OWL tiene como objetivo (principal) facilitar un modelo de marcado construido sobre RDF y codificado en XML.

Tiene como antecedente DAML+OIL, en los cuales se inspiraron los creadores de OWL para crear el lenguaje. Junto al entorno RDF y otros componentes, OWL es una de las referencias en el contexto del actual proyecto de web semántica.

OWL es realmente una familia de soluciones de momento compuesta por tres versiones del lenguaje caracterizadas por expresividad (y complejidad creciente): OWL Lite, OWL DL, y OWL Full.

Estas variantes incorporan diferentes funcionalidades, y en general, OWL Lite es más sencillo que OWL DL, y OWL DL es más sencillo que OWL Full. OWL Lite está construido de tal forma que toda sentencia pueda ser resuelta en tiempo finito, la versión más completa de OWL DL puede contener 'bucles' infinitos.

OWL DL se basa en la lógica descriptiva SHOIN(D). El subconjunto OWL Lite se basa en la lógica menos expresiva SHIF(D).

1.6.1.3 SWRL

SWRL (Semantic Web Rule Language) [Swr1] es una especificación de lenguaje que se propone de formalizar y estandarizar la definición de reglas semánticas entre campos informativos. Combina elementos de OWL y conceptos declarativos derivados de Rule Markup Language (Unary/Binary Datalog).

Las reglas SWRL son, por lo tanto, estructuradas de forma muy parecida a cláusulas declarativas: se componen de un antecedente (body) y un consecuente (head); su interpretación se basa en la relación lógica entre antecedentes y consecuentes.

Fue especificado dentro de W3C en 2004 por el consejo nacional de investigación de Canadá. Es una de los modelos con mayor margen de evolución en tema de representación semántica del conocimiento como testifica su exitoso uso en contextos de gran complejidad [Liu2].

1.6.2 Servicios semánticos

Los servicios semánticos [Sto1] aprovechan las tecnologías semánticas para integrar los típicos descriptores funcionales del servicio con descriptores semánticos. Entre las otras cosas, descriptores de tipo semántico podrían hacer que no requiera de la intervención humana para el análisis de las interfaces y la integración en la aplicación, sino que la aplicación misma pudiera hacer uso de los nuevos servicios cuando fuese necesario. Además, se podrían incrementar, a través de descriptores semánticos, las potencialidades de los servicios web en términos de clasificación y descubrimiento.

La idea a la base de los servicios semánticos es proporcionar el soporte para el desarrollo de una red de servicios distribuidos e interoperables con los que construir de forma automatizable otros servicios más complejos.

Los servicios semánticos podrían asumir un rol de primaria importancia en el contexto de modelos evolutivos del web, como en el caso de *Internet of Services* [Sch2][Rug1]. En este tipo de entorno lógico los aspectos relacionados, directamente o indirectamente, con la interoperabilidad pueden ser determinantes.

1.6.3 Requisitos y limitaciones para tecnologías semánticas

La concreta y efectiva utilización de tecnologías semánticas depende de sistemas concretos por los cuales la relación entre beneficios introducidos y desventajas tiene que ser atentamente analizada.

Tanto los requisitos como las limitaciones se pueden enfocar de acuerdo con una dúplice perspectiva: tecnológica y lógica.

Desde un punto de vista tecnológico, las tecnologías semánticas, como mínimo, implican:

- Utilizo de formatos interoperables que soporten la descripción de información compleja (XML por ejemplo).
- Necesidad de un formalismo para definir ontologías en todas sus componentes (conceptos, propiedades, jerarquía, relaciones semánticas, etc.).

Estos requisitos implican, evidentemente, un intercambio y procesado de información de alto nivel, normalmente a nivel de sistema de información. Para grandes cantidades de datos y/o información especialmente compleja o estructurada, hay un problema concreto de tiempo de procesamiento.

No es objetivo de esta tesis definir los límites de aplicación de las tecnologías semánticas. De todos modos, considerando la complejidad de los modelos informativos y la necesidad de tener que procesar los mismos, dichas tecnologías se consideran, en principio, adecuadas para entornos de aplicación en el cual los aspectos de interoperabilidad son prioritarios. Más concretamente se puede pensar a subsistemas que interactúan, sistemas heterogéneos y aplicaciones sobre larga escala (por ejemplo aplicaciones web-centric).

Otro elemento de interés de las tecnologías semánticas se considera la capacidad de procesado inteligente y automatizado de información heterogénea; conceptos, propiedades y las relaciones entre ellas pueden ser determinantes a la hora de deducir información de forma automatizada.

Una de las limitaciones más significativas para los entornos semánticos es que, aun mejorando cualquier tipo de interacción maquina-maquina, no resultan tan dinámicas para interacciones hombre-máquina. Este tipo de inconveniente puede obviarse de acuerdo con diferentes estrategias, por ejemplo la utilización de modelos híbridos.

1.7 Modelo, Simulación y Evaluación

Todos los aspectos teóricos básicos orientados a proporcionar un soporte para el análisis, a priori o a posteriori, de arquitecturas complejas o para el desarrollo de sistemas de simulación constituyen un aspecto de notable interés en el contexto de investigación corriente. El problema del modelo en redes de sensores puede ser abordado a diferentes niveles (dispositivo [Wan2], red [Gan1], servicio/sistema [Vir1]). La perspectiva con mayor potencial de análisis suele ser la visión a nivel de red.

Modelos concretos pueden referirse a aspectos de carácter general (conectividad/cobertura y otros aspectos implícitos [Gan1]) o, bien, a aspectos o fenómenos más específicos (energy-hole/trafico [Xia1], auto-organización [Wan3], fiabilidad/seguridad [Xin2], clustering [Su1] y enrutamiento [Gui2] por ejemplo).

Especialmente interesantes y creativos se han revelado los modelos basados en sistemas naturales [Cha2] y modelos de cooperación/concurrencia (teoría de juegos [See1], por ejemplo).

Las técnicas de simulación pueden variar significativamente en función del objeto, del objetivo de la simulación (predicción de prestaciones, soporte al diseño y/o desarrollo, etc.) y de los modelos teóricos considerados. Herramientas concretas pueden implementarse en entornos de simulación genéricos (MatLab [Mat1], NS2 [Ns2], OPNet [Opn1], por ejemplo) o como soluciones ad-hoc. La primera clase de soluciones suele garantizar una cierta ventaja para la simulación de sistemas heterogéneos, aprovechando elementos preexistentes; la segunda clase propone todas las ventajas típicas de las soluciones ad-hoc que, implementando solamente los aspectos de interés específico, puede potencialmente garantizar mayor efectividad y prestaciones. Técnicas de simulación avanzadas, basadas en modelos de interacción por eventos discretos o concurrentes, suelen incluir un bien definido modelo para el ambiente circundante (determinante en redes de sensores) y para la interacción del sistema con el mismo (interferencias y obstáculos, condiciones variables en el tiempo, etc.) con el objetivo de proponer, con todas las limitaciones del caso, entornos de simulación lo más parecidos posible a entornos reales.

Modelos específicos para la evaluación de arquitecturas complejas, caracterizadas por varios mecanismos operantes de manera simultánea, constituyen un elemento de especial interés al proporcionar un soporte práctico/teórico para análisis profundizadas orientadas a la comparación de soluciones diferentes en función de la calidad de servicio proporcionada.

La fuerte tendencia a la movilidad también se ve reflejada en temas de modelado, simulación y evaluación [Jin1][Fum1] donde el objetivo fundamental es la evaluación de la robustez de ciertas soluciones y la evaluación específica del impacto de la movilidad sobre las prestaciones de la red.

1.8 Entornos lógicos

En el presente trabajo de tesis, las arquitecturas de redes inalámbricas de sensores se enfocan en un contexto más amplio de lo usual en el cual se pueden identificar, por lo menos, tres diferentes ámbitos de investigación:

- *Redes de sensores*
- *Arquitecturas para middleware*
- *Elementos de ingeniería del conocimiento e inteligencia artificial*

Un enfoque tan amplio pretende responder de forma satisfactoria a las exigencias de los sistemas de última generación que pueden trabajar en diferentes contextos (privado, social, abierto) y de acuerdo con diferentes modelos de negocios.

Para garantizar un diseño de arquitectura cuanto más genérico y flexible posible, en el contexto de este trabajo de tesis se ha considerado oportuno definir la arquitectura en más de un contexto lógico (Figura 1.10). Más concretamente, se consideran perspectivas de diseño coherentes con los entornos lógicos descritos a continuación.

Estos entornos se caracterizan por un nivel de abstracción e interoperabilidad creciente:

- *Recurso físico.* Asume un entorno de trabajo en el cual la arquitectura considerada es un sistema aislado o está empotrada en un sistema autónomo. No hay, por lo tanto, alguna necesidad de interacción con otros sistemas.

- *Recurso lógico o virtual*. Es la extensión más natural del entorno anteriormente descrito en cuanto propone una visión lógica de los recursos que pueden ser recursos compartidos o trabajar en el contexto de organizaciones virtuales.
- *Sensor Web*. Es un concepto intrínsecamente más amplio porque asume un modelo de interacción abierta entre sistemas (*Open World*) a través del entorno web.
- *Semantic Sensor Web*. Es la realización del modelo propuesto por Sensor Web en la web semántica.

Se presenta, a continuación, un breve análisis de los entornos lógicos considerados. En el próximo capítulo, el diseño de los varios componentes de la arquitectura se contextualizará en uno o más de los entornos lógicos descritos en esta sección.

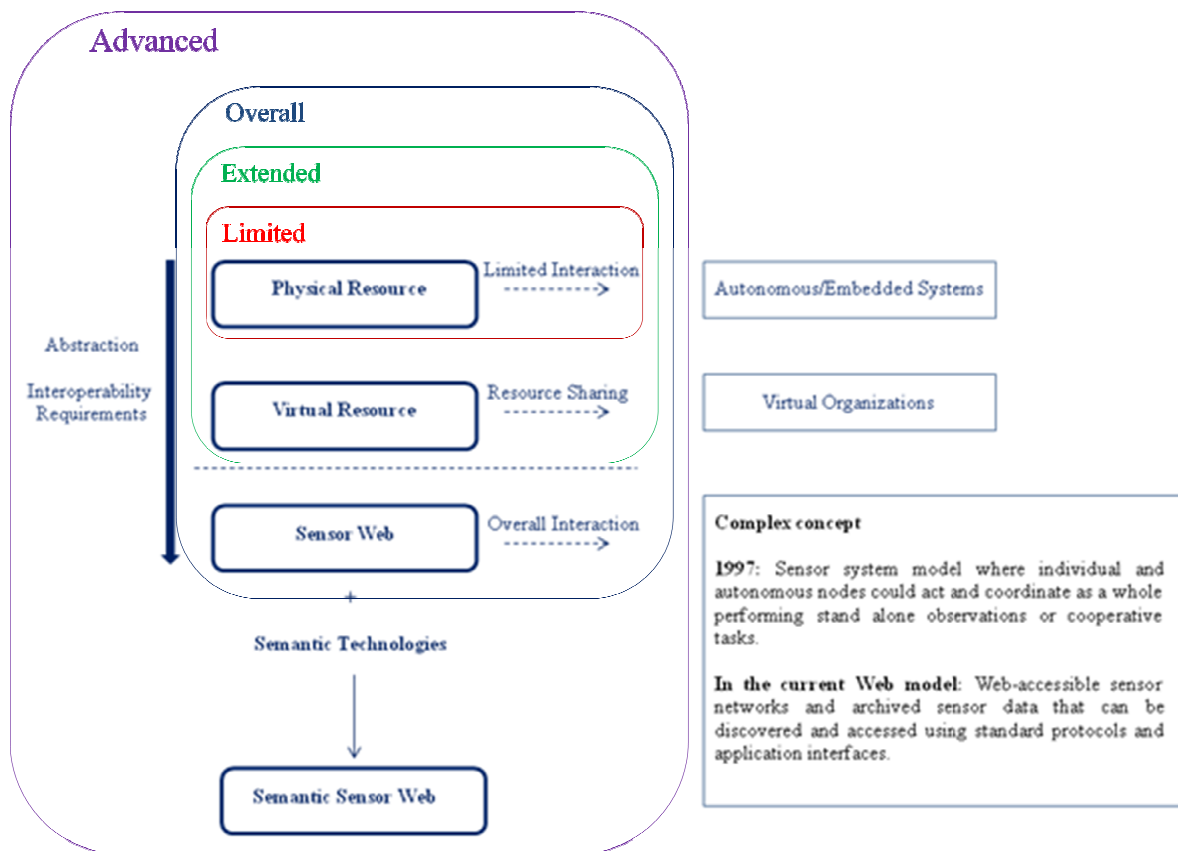


Figura 1.10. Entornos lógicos para arquitecturas de redes inalámbricas de sensores.

1.8.1 Recurso físico

Describir una arquitectura de redes de sensores en este contexto lógico es equivalente a proporcionar una definición técnico/funcional de los componentes de acuerdo con modelos de dispositivos y de red bien definidos.

En práctica se asume un modelo de recurso con limitación a su entorno de trabajo que, por lo tanto, no está contextualizado. La falta de interacción con sistemas externos simplifica notablemente la visión lógica de los recursos. Es, por lo tanto, el modelo más sencillo pero también el más utilizado en redes de sensores.

En este trabajo, esta perspectiva se ha utilizado para definir las funcionalidades de los recursos físicos de la red asumiendo el modelo de nodo sensor de bajo coste y el modelo de red *data-centric* flexible en función de escala y aplicación de referencia.

Al contrario, no se ha utilizado esta perspectiva para definir los componentes con recursos de cálculo avanzados de la red (estaciones base, *gateways*, centro de control) para el cual se prefiere una perspectiva más amplia (recurso virtual).

1.8.2 Recurso lógico o Virtual

Este modelo es la extensión natural del anterior y propone una visión de recurso contextual, intrínsecamente más complejo en cuanto parte de contextos dinámicos heterogéneos. Los recursos ya no se consideran de forma independiente si no como parte de organizaciones virtuales en las cuales suelen actuar con uno o más roles bien definidos o, bien, ser recursos compartidos.

Esta perspectiva de definición es por lo tanto absolutamente contextual en cuanto se basa en la definición de los recursos en función de modelos heterogéneos que incluyen ambiente, actores y relación entre los mismos, definiendo el espacio virtual del recurso considerado.

Un recurso virtual se suele definir de acuerdo con una perspectiva funcional, eventualmente orientada a roles. Los sistemas más modernos proporcionan funcionalidades avanzadas de acceso y gestión remota así como una visión modular de los recursos que suelen interactuar con diferentes sistemas.

Esta perspectiva se considera absolutamente fundamental en el contexto de este trabajo de tesis en cuanto todo el entorno funcional avanzado se proporciona a través de una capa virtual que corresponde a una infraestructura jerárquica de actores inteligentes.

Por otra parte, es práctica común definir un cierto modelo de organización virtual como entorno lógico de referencia para los recursos definidos. Como se describe en detalle en el próximo capítulo, el enfoque generalizado de la arquitectura se refleja en una falta de particularización del modelo de organización virtual de referencia.

1.8.3 Sensor Web

El concepto de *Sensor Web* [Bot1] es relativamente antiguo (1997) y se suele atribuir a Mike Botts and Alex Robin: en su primera definición, este término se utilizó para identificar un modelo innovador de sistema donde nodos sensores autónomos pueden actuar y coordinarse como un único sistema, proporcionando observaciones autónomas o tareas cooperativas.

Evidentemente esta definición fue la respuesta concreta al siempre creciente empleo de sensores en el mundo real y como parte de la vida diaria de individuos e instituciones; la presencia, más o menos masiva de sensores, no implica absolutamente que la información relacionada a estos dispositivos este conectada (o sea conectable), si no exactamente lo contrario: este escenario suele resumirse en grandes cantidades de información y en poco conocimiento.

Una definición de *Sensor Web*, actualmente aceptada por gran parte de la comunidad científica, es: “Redes de sensores y datos procedentes de sensores accesibles vía web que pueden descubrirse y ser utilizados a través de protocolos e interfaces de aplicación estándares”.

Se trata de un concepto absolutamente general que esta progresivamente asumiendo importancia en diferentes contextos de aplicación: sistemas geo-referenciados (GIS [Xia2][Yan2][Liu1]) de largo alcance, *Social Sensors* [Rah1][Rah2] y, más en general, en todos aquellos sistemas que responden a modelos de negocio o organizaciones complejas que asumen el intercambio dinámico de datos sensores heterogéneos desde diferentes fuentes distribuidas.

Como la propia definición deja intuir, se trata de un concepto más amplio de los propuestos hasta ahora en el capítulo tanto en términos de escala (Internet) como de heterogeneidad (recursos y usuarios) y organización lógica potencial (cualquier clase de organización virtual).

1.8.3.1 *Actuales límites del Sensor Web*

La concreta realización de Sensor web en el panorama tecnológico de la web actual se ve caracterizada (y al mismo tiempo limitada) por dos aspectos fundamentales:

- *Modelo de interoperabilidad.* Tanto el modelo convencional de web (conocido como web sintáctica) como el modelo Web 2.0 pueden proporcionar dos modelos distintos y progresivos de interoperabilidad: interoperabilidad básica e interoperabilidad funcional. El primer modelo asume un intercambio de mensajes entre sistemas sin interpretación alguna; el segundo, intercambio de mensajes con interpretación del contexto y campos con una definición estructural común. Evidentemente, este último modelo de interoperabilidad, utilizado por ejemplo por servicios web, puede ser funcional para *Sensor Web* sólo si las interfaces y los modelos de datos están conformes con algún estándar. En caso contrario el nivel de interoperabilidad conseguido permite acceder correctamente a la información en cuestión que no puede interpretarse de forma unívoca. Este aspecto es absolutamente fundamental en entornos de larga escala con información y recursos heterogéneos.
- *Falta de estandarización.* A pesar de los innumerables esfuerzos en esta dirección, la afirmación de estándares de modelos de datos e interfaces de acceso no parece progresar de forma decidida. Además, las diferentes metodologías de estandarización (modelos de propósito general, de contexto específico, etc.) no son demasiado coherentes entre ellas. Un ejemplo de estandarización es SensorML [Sen2], propuesto por Open Geospatial Consortium [Ogc1]. Un análisis exhaustivo de los diferentes estándares se considera fuera de los objetivos primarios de este trabajo de tesis.
- *Modelos de negocio.* Como se ha descrito en las secciones correspondientes al capítulo de introducción de las tesis, las redes de sensores pueden considerarse sistemas de bajo coste relativamente a la tecnología básica; sin embargo siguen siendo soluciones a medida que, además, normalmente necesitan operaciones de mantenimiento más o menos periódicas. Para compartir estos recursos en un entorno de gran tamaño se necesitan modelos de negocio bien definidos que, de momento, no parecen tan claros.

A modo de resumen, el concepto de *Sensor Web* es un concepto progresivo, atractivo desde un punto de vista científico; sin embargo necesita una consolidación a nivel comercial que empuje hacia la formalización de estándares que puedan finalmente determinar una afirmación progresiva del modelo a gran larga escala.

La arquitectura propuesta en esta tesis se ve afectada por los mismos límites generalizados de *Sensor Web*.

1.8.4 Semantic Sensor Web

En la presente tesis se ha estado haciendo referencia, implícitamente o explícitamente, a un modelo de web convencional; la web, como la conocemos actualmente se puede considerar como una Web sintáctica: un lugar donde los ordenadores se encargan únicamente de la tarea de transmisión y presentación de los recursos. Por otro lado, son los usuarios los encargados de realizar las distintas interpretaciones de los recursos presentados y de relacionar los recursos entre sí de forma lógica. Estas tareas que han de realizar los usuarios son costosas sobre todo con la gran cantidad de información disponible actualmente en la web. Se suele denominar a este estadio de la web como Web 1.0.

Incluso considerando una de las posibles evoluciones concretas de la web (Web 2.0), los límites fundamentales, especialmente en términos de interoperabilidad, parecen persistir. Web 2.0 es un término que se extendió a finales del 2004, y que en principio intenta agrupar un conjunto de nuevos servicios y tecnologías que están emergiendo en la Web. Sus bases son la asunción que considera la web la única plataforma, uso de tecnologías RIA (*Rich Internet Application* [Hoo1]) y socialización de la web. Evidentemente el modelo Web 2.0, aunque introduciendo conceptos interesantes, no se puede considerar una real evolución de la web sintáctica desde el punto de vista del modelo de conocimiento e interacción.

Si el Sensor Web sigue haciendo referencia los modelos web corrientes, la única forma de aumentar de forma importante el nivel de interoperabilidad entre sistemas de sensores a gran escala es la estandarización de modelos de datos e interfaces de acceso (como mencionado en la sección anterior).

Sin embargo, existe como mínimo un modelo innovador y alternativo a la web actual es la Web Semántica. La Web Semántica es la "web de los datos interpretados". Se basa en la idea de añadir metadatos semánticos y ontológicos a la Web. Esas informaciones adicionales, que describen el contenido, el significado y la relación de los datos, se deben proporcionar de manera formal, para que así sea posible evaluarlas automáticamente por máquinas de procesamiento. El objetivo es mejorar Internet en su complejo, sus actuales modelos de interacción y computación ampliando la interoperabilidad entre los sistemas informáticos usando actores semánticos capaces de procesar la información de forma automatizada sin intervención humana. La clave de la Web Semántica esta en sus modelos de descripción del conocimiento (ontologías). El término ontología hace referencia a la "formulación de un exhaustivo y riguroso esquema conceptual dentro de uno o varios dominios dados; con la finalidad de facilitar la comunicación y el intercambio de información entre diferentes sistemas y entidades" [Wik1].

La realización del *Sensor Web* en un contexto semántico determina un nuevo concepto conocido como *Semantic Sensor Web* [She1]; *Semantic Sensor Web* es un concepto en constante evolución que introduce una capa semántica en la cual se define formalmente la semántica o el significado de la información. Esta capa debería integrar las infraestructuras de información habituales del web mejorando, de forma sustancial, la capacidad de descubrir, catalogar, clasificar, compartir, manipular y analizar datos procedentes de sensores (o informaciones asociadas); al mismo tiempo, se aseguraría un alto nivel de interoperabilidad entre sistemas a través de interacciones semánticas.

Como cualquier otro entorno semántico, el *Semantic Sensor Web* es el resultado de dos aspectos fundamentales: modelo de interoperabilidad (Interoperabilidad Semántica) e Ingeniería del Conocimiento (semántico en este caso). Estos aspectos y las relativas problemáticas se discuten a continuación.

Las tecnologías semánticas implican el intercambio de datos entre sistemas asumiendo los mismos definidos según un esquema semántico bien definido que incluye tanto el esquema de la información como reglas y relaciones entre los diferentes campos, conceptos y propiedades. En otras palabras, el significado de la información se define en el mismo esquema de manera formal.

Cualquier tecnología semántica asume un modelo de datos de referencia notablemente más complejo que el correspondiente en tecnologías estándares.

En un sistema “semántico” el problema central y más creativo es la ingeniería del conocimiento. La falta de metodologías unificadas en tal sentido se refleja de forma significativa en modelos de datos extendidos (como los modelos semánticos). Una discusión exhaustiva sobre los actuales límites y retos de la ingeniería del conocimiento semántico, aunque de interés absoluto en el panorama científico corriente, se consideran fuera de los objetivos del presente documento.

Tecnologías semánticas comúnmente utilizadas (OWL por ejemplo) suelen utilizar formatos de datos interoperables (XML) sobre los cuales conceptos semánticos, jerarquía relacionada, propiedades, y reglas entre los mismos se definen de acuerdo a esquemas predefinidos (RDF por ejemplo).

Las tecnologías semánticas introducen nuevas perspectivas tanto en los modelos de interacción como en la ingeniería del conocimiento; también aspectos computacionales y, más en general, de inteligencia se pueden ver afectados en ciertos contextos.

Los modelos semánticos se pueden considerar mas propios o aptos para interacción entre maquinas. Evidentemente su aplicación a modelos de interacción hombre-máquina puede resultar “rígida” en ciertos contextos. De todos modos, también la interacción hombre-máquina podría beneficiarse de unas cuantas características de los modelos semánticos y las desventajas podrían ser limitadas de forma significativa a través de la introducción de modelos híbridos. En este último caso es imprescindible disponer de una métrica bien definida para asociar los resultados obtenidos a niveles de fiabilidad diferentes.

1.8.4.1 Interoperabilidad Semántica

La interoperabilidad entre sistemas o sub-sistemas es uno de los temas de mayor interés en el ámbito de los sistemas informáticos modernos. Actualmente la interoperabilidad entre sistemas se realiza de acuerdo a una de los modelos definidos a continuación:

- Interoperabilidad “Básica”: Se intercambian mensajes entre los sistemas, sin interpretarlos.
- Interoperabilidad “Funcional”: Se intercambian mensajes entre los sistemas, se interpreta el contexto de los datos y los campos tienen una definición estructural común.

Con el soporte de las tecnologías semánticas, el modelo de interoperabilidad debería evolucionar de acuerdo a la siguiente definición:

- Interoperabilidad “Semántica”: Se intercambian mensajes entre los sistemas, se

interpreta el significado y el contexto de los datos, y el contenido de los campos de datos se consigue mediante un código común.

Independientemente de tecnologías concretas, la idea fundamental en la arquitectura definida es modelar el intercambio de datos entre sistema de acuerdo con una filosofía diferente respecto al modelo de inteligencia. Normalmente se asume que la información sea “estática” y que las capas inteligentes de un sistema se implementen por los actores que trabajan sobre la información. En modelos convencionales, son los actores los que tiene la responsabilidad de interpretar la información; las tecnologías semánticas pretenden invertir esta visión en cuanto es la información la que implícitamente contiene su propio significado; los actores, aunque puedan implementar capas extra de inteligencia, realmente se comportan de forma más parecida a interpretes.

El modelo de interoperabilidad semántica introduce, por lo tanto, varias ventajas tanto a nivel teórico (modelos innovadores de inteligencias sobre larga escala) como práctico (no es necesario, en principio, asumir ontologías estándar). La desventaja principal del modelo de interoperabilidad semántica puede ser la dificultad de garantizar la consistencia semántica de los datos ya que entornos o sistemas diferentes pueden utilizar conceptos iguales o parecidos con significados diferentes (conceptos contextuales). En este sentido, el nivel de interoperabilidad se puede incrementar, de manera importante, utilizando vocabularios de conceptos compartidos.

1.9 Limitaciones de las Redes Inalámbricas de Sensores

A pesar de los evidentes avances tecnológicos, de una ingeniería en constante evolución y soluciones computacionales altamente avanzadas, las redes de sensores inalámbricos siguen teniendo limitaciones. Algunas se pueden clasificar como intrínsecas o directamente relacionadas con los actuales límites tecnológicos que determinan la calidad del servicio global, otros, de naturaleza más metodológica y relacionadas con la ingeniería, se traducen en un sustancial incremento de los costes que limita notablemente la competitividad comercial de los productos. La clasificación de los límites más relevantes de las redes inalámbricas de sensores en los siguientes puntos:

- *Intrínseca centralidad de las estaciones base y Energy-hole.* La estación base constituye un punto singular de fallo para la red completa. En caso de alimentación a batería, la estación base también constituye un límite superior para el tiempo de vida de la red. La centralidad lógica y física de las estaciones base también determina el fenómeno de “energy-hole”: los nodos más próximos a la estación base soportan el tráfico generado por los nodos menos próximos y, por lo tanto, tienden a tener un tiempo de vida significativamente inferior. El consumo de la batería de los nodos más próximos causa, inevitablemente, el aislamiento de los nodos más lejanos causando una significativa reducción del tiempo de vida global de la red. Además, muchas arquitecturas reales no permiten que la estación base asuma una posición central respecto a la red circundante, determinando rutas de comunicación medianamente más largas y tiempos de vida inferiores de las redes.
- *Elevado coste de producción, mantenimiento y testing.* La fuerte tendencia “ad-hoc” de la ingeniería de redes inalámbricas de sensores y la consecuente falta de soluciones generalizadas determina un coste de desarrollo elevado que se va sumar al coste de testing (intrínsecamente elevado) y de mantenimiento, especialmente crítico en ambientes hostiles.

- *Falta de estándares.* Estándares bien definidos, especialmente en temas de protocolos, contribuirían de forma relevante, a proporcionar un modelo de diseño y desarrollo más abstracto y modular, reduciendo, consecuentemente, los costes.
- *Calidad de servicio no siempre aceptable y no escalable.* Redes de sensores no supervisadas requieren un nivel de robustez, fiabilidad y tolerancia a fallos importante; no siempre las arquitecturas reales cumplen tales requisitos. De la misma forma el tiempo de vida de la red no siempre es considerado aceptable por parte de los usuarios finales. Al crecer la escala, aun considerando técnicas específicas como clustering, la calidad de servicio decrece considerablemente.

1.10 Líneas de investigación activas y futuro de las Redes Inalámbrica de Sensores

Todos los aspectos precedentemente mencionados sobre redes de sensores inalámbricos se consideran temas (o conjuntos de temas) de investigación abiertos. Actualmente los esfuerzos de investigación en el seno a la comunidad científica internacional aparecen centrarse, aun mas en que soluciones altamente innovadoras, en avances en la tecnología de base (que automáticamente se traduciría en un incremento proporcionales de las prestaciones de los productos finales) y en el refinamiento de las soluciones existentes para garantizar parámetros de calidad de servicio al crecer de la escala de las redes.

Las redes de sensores móviles, tanto en entornos militares como civiles (aplicaciones vehiculares por ejemplo), representan un tema de gran atención en términos científicos y comerciales.

Por otra parte, varios proyectos de investigación se enfocan enteramente en la estandarización tanto en temas de protocolos como de modelos de datos.

También la metodología de diseño y desarrollo es al centro de varios estudios orientados básicamente a la reducción de los costes de producción, así como todos los aspectos computacionales se consideran de primordial importancia.

Sistemas de información de gran escala, basados en redes inalámbricas de sensores, parecen orientarse siempre más hacia modelos de computación complejos como *Sensor Web* [Bot1]/*Semantic Sensor Web*[She1] o el modelo de cálculo denominado comúnmente *Internet of Things* que impone la necesidad de infraestructuras middleware aun más flexibles de las actuales [Gui1][Spi1] y de estándares bien definidos en términos de representación de datos y, eventualmente, de tecnologías semánticas interoperables basadas en ontologías compartidas.

CAPÍTULO 2

2 Descripción de la Arquitectura

El objetivo fundamental del capítulo es la descripción de la arquitectura propuesta en su caso más genérico. La misma se compone, realmente, de más subsistemas (red de sensores, middleware, módulos de control y red de acceso). También se consideran partes integrantes de la arquitectura los modelos avanzados de representación de la información y del conocimiento relacionado. De cada componente se proporciona, una descripción técnico-funcional exhaustiva con énfasis mayor sobre la arquitectura protocolar de la red inalámbrica de sensores, en cuanto de interés primario (aunque no exclusivo) del presente trabajo de tesis. La descripción de la arquitectura se articula según una perspectiva lógica que incluye cuatro conceptos progresivos en términos de abstracción y complejidad lógica.

2.1 Introducción

El modelo de arquitectura generalizado propuesto en la presente tesis es el resultado de una metodología de diseño orientada al soporte de aplicaciones de redes de sensores sin asunción alguna sobre objetivos y requisitos de las mismas. Esta metodología de diseño implica, por una parte, la necesidad de contemplar componentes funcionales con múltiples capacidades y posibilidades de interacción; por la otra, la política de diseño/desarrollo extremadamente modular asegura suficiente flexibilidad y dinamismo para adaptar y particularizar el modelo arquitectural general a entorno y aplicaciones concretas.

En el contexto propuesto, el diseño y el desarrollo de una arquitectura completa de redes inalámbricas de sensores se considera principalmente dependiente de los aspectos siguientes:

- *Dominio y escala de la aplicación.* Existe, evidentemente, una gran variedad de aplicaciones basadas en redes inalámbricas de sensores; aunque aplicaciones estructuralmente parecidas presentan patrones relevantes que permiten identificar

dominios aplicativos específicos (adquisición de datos en tiempo real, por ejemplo), diferentes dominios pueden caracterizarse por modos de trabajo e interacción tan diferente entre ellos que requieren estructuras de gestión y soporte totalmente diferentes. Además aplicaciones específicas en el mismo dominio de base pueden proponer nuevos requisitos o privilegiar ciertos aspectos más que otros. Incluso la misma aplicación, operante con escala significativamente diferente, puede presentar requisitos lógico/funcionales conceptualmente diferentes.

En el contexto de la presente tesis el dominio aplicativo se supone absolutamente genérico; la escala del sistema tampoco asume particulares restricciones relativas al número potencial de usuarios y, por lo tanto, de carga de la infraestructura de acceso; al contrario, la escala, en términos de número de nodos sensores por cluster, se supone vinculada a los actuales límites tecnológicos y propone un *tradeoff* de gran importancia con los típicos requisitos no funcionales (fiabilidad, robustez).

- *Características lógicas/físicas de la aplicación y entorno de trabajo.* En el contexto del mismo dominio aplicativo y a paridad de escala, el diseño de arquitecturas concretas debería de tener en cuenta también eventuales restricciones, lógicas o topológicas, y el entorno real de trabajo. Eventuales restricciones lógicas/físicas pueden introducir puntos de fallos o de ineficiencia en los sistemas que suelen añadir un nivel de complejidad añadido normalmente orientado a garantizar los parámetros no funcionales. Sin embargo, entornos de trabajo hostil, caracterizados por condiciones de comunicación potencialmente variables en el tiempo (interferencias y obstáculos) determinan la necesidad de contemplar módulos funcionales de monitorización, gestión y control potencialmente diferentes por peso y complejidad.

La arquitectura propuesta asume un modelo flexible de infraestructura que se basa en la organización dinámica, eventualmente jerárquica, de los actores y de los clusters de forma tal que eventuales restricciones o limitaciones lógicas/físicas simplemente impliquen una diferente configuración del sistema. Por otra parte, las estructuras locales (redes de sensores) proporcionan diferentes posibilidades de trabajo para adaptarse, de forma eficiente, a diferentes requisitos de calidad de servicio en función de las condiciones detectadas.

- *Calidad de servicio o características no funcionales.* El presente trabajo asume un concepto de calidad de servicio directamente relacionado a las características no funcionales que afectan concretamente la aplicabilidad concreta de los sistemas (básicamente fiabilidad, robustez, tolerancia de fallos). Este concepto de aplicabilidad general se suele integrar con requisitos específicos de aplicaciones concretas (tiempo de vida, por ejemplo).
- *Modelo de interacción.* En dependencia de modelos de interacción concretos, la arquitectura tiene que trabajar utilizando modelos de interoperabilidad y de ingeniería del conocimientos diferentes (o profundamente diferentes) entre ellos.

La arquitectura propuesta puede trabajar como sistema autónomo o en contextos lógicos más amplios (Organizaciones Virtuales) eventualmente interoperables de acuerdo a modelos funcionales (*Sensor Web*) o semánticos (*Semantic Sensor Web*).

Tras una descripción general de la arquitectura (sección 2.2), el capítulo se estructura de acuerdo con el esquema representado en Figura 2.1: se considera la aplicación de la arquitectura en entornos lógicos de abstracción creciente.

El contexto lógico a nivel de recurso físico es, posiblemente, el más común; es una visión bastante limitada de las arquitecturas de sensores que se consideran parte de sistemas aislados.

La natural evolución del recurso físico es el recurso lógico que, implícitamente o explícitamente, asume redes de sensores compartidas en el contexto de organizaciones virtuales eventualmente complejas.

El concepto de recurso virtual puede asumir un significado más amplio normalmente conocido como Sensor Web. Finalmente se considera una de las más recientes evoluciones de este modelo, resultante de la convergencia de tecnologías semánticas en el modelo de base..

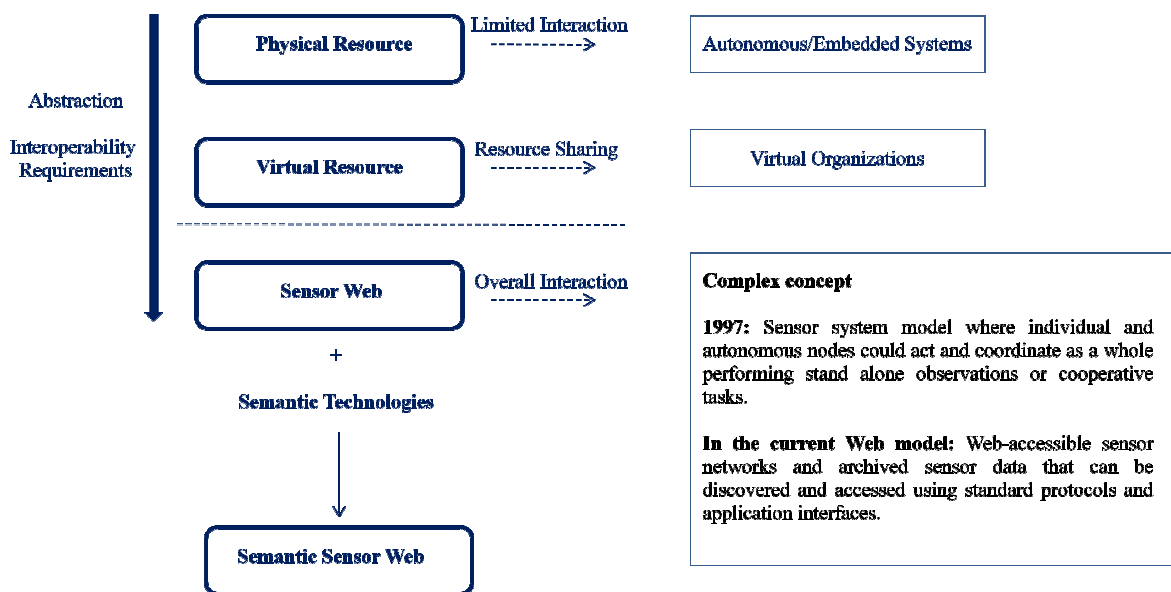


Figura 2.1. Esquema representante los posibles entornos lógicos para una arquitectura de sensores.

2.2 Visión general

Una cantidad variable de nodos sensores se supone desplegada en una cierta área física; cada área física se asocia a un área lógica, que, por lo tanto, puede componerse de diferentes áreas físicas. Las áreas físicas que componen un área lógica pueden ser contiguas o no tener ninguna relación entre ellas. Su agrupación en áreas lógicas es, por lo tanto, una abstracción introducida por el sistema para generar ciertos tipos de jerarquías o relaciones entre las áreas físicas. Cada área física se define red inalámbrica local de sensores.

La arquitectura de gestión propuesta (Figura 2.1.1) pretende actuar a varios niveles para proporcionar:

- *Soporte local*: los nodos sensores pueden trabajar de acuerdo a una topología predefinida o ser desplegados aleatoriamente en el área de referencia así como pueden seguir un patrón de comportamiento estático/nomádico o, bien, móvil. Asimismo, en función de la escala de la aplicación, puede o meno precisarse configuración en clusters (uno por cada estación base). Cada cluster identifica un segmento lógico del área lógico/física. El sistema proporciona tanto una pila de protocolos completa que implementa todas las funcionalidades de los nodos sensores

como la interfaz de control local (estación base).

- *Acceso*: la información generada (o detectada) es accesible a través de las infraestructuras, de acuerdo con la política de acceso a la información misma. El sistema soporta, potencialmente, tanto consulta pasiva como activa (la información es empujada por el sistema de control local) y consulta de datos on-demand (el centro de control solicita la generación de datos).
- *Modelo distribuido o centralizado de monitorización, gestión y control*: el sistema de control puede acceder, a través de la interfaz apropiada, a los sistemas de monitorización, gestión y control de las varias redes de sensores.

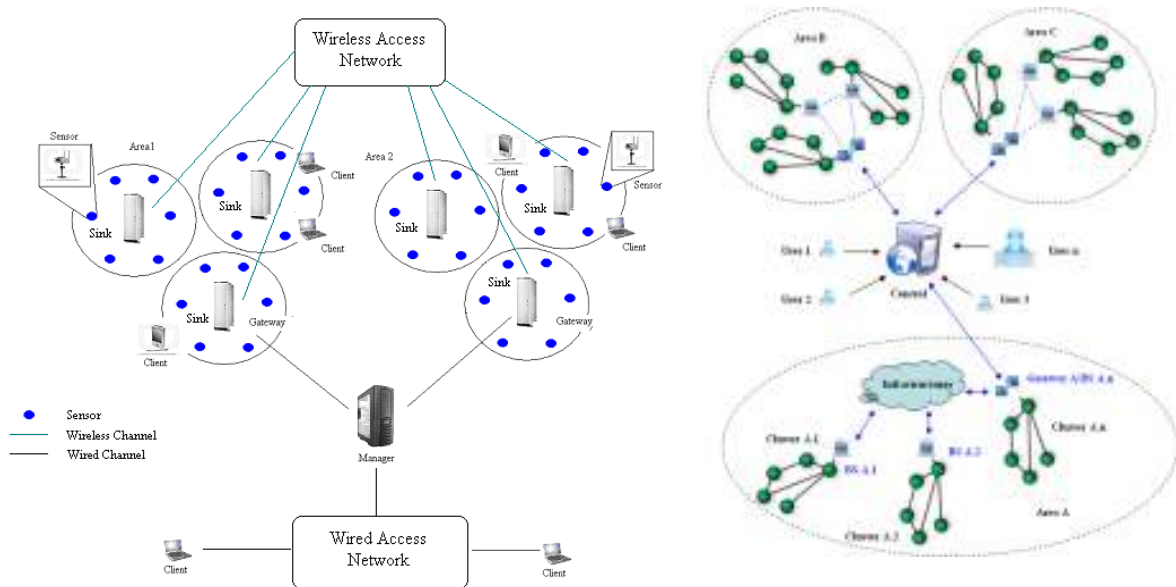


Figura 2.1.1. Izquierda: Visión general de la Arquitectura (Figura original de [Con1]). Derecha: Visión de la Arquitectura basada en clusters (Figura original de [Con4]).

No se hacen asunciones sobre el tipo de aplicación (o aplicaciones) concretamente soportada por la arquitectura; tampoco existe algún vínculo o limitación relacionada con el número de áreas o su organización interna (número o densidad de nodos sensores).

Con referencia la visión esquemática de la arquitectura propuesta (Figura 2.2), podemos distinguir dos tipos de recursos físicos: los nodos de sensores inalámbricos y los actores. Los primeros se identifican con el concepto de nodo sensor inalámbrico de bajo coste más veces mencionado en las secciones anteriores del presente documento; los actores se suponen dispositivos de capacidad notablemente superior en términos de capacidad de procesamiento, memorias y tecnología de comunicación. Los actores son dispositivos multimodales respecto a la capacidad de comunicación: por una parte pueden comunicar con los nodos sensores a través de tecnología inalámbrica a baja potencia; por la otra, comunican con los otros actores o con el sistema de gestión a través de tecnología inalámbrica de potencia superior y/o con comunicación cableada.

A recursos físicos corresponden recursos virtuales. Cada nodo inalámbrico implementa la pila de protocolos desarrollada en el contexto del presente trabajo de tesis; las principales características de la pila de protocolos locales se describen en las secciones siguientes. Se distinguen dos actores lógicos: la Estación Base y el Gateway. El primer actor, con funcionalidades de control local, por una parte proporciona la interfaz de gestión y control

para el cluster que “representa” y, por la otra, implementa funcionalidad de pasarela bidireccional hacia el Gateway. Este último es una extensión de las funcionalidades de la Estación Base que virtualiza todos los recursos de la área lógica asociada que son, por lo tanto, accesibles para el Sistema de Control y Gestión remota.

El número de Estaciones Base puede variar, normalmente en función de la escala de la red en cuestión; como mínimo una de las Estaciones Base que componen cada área lógica tiene que implementar las funcionalidades de Gateway.

Evidentemente los actores pueden interactuar entre ellos según diferentes topologías y modos de interacción (jerárquica o P2P). El conjunto de los actores lógicos constituye la infraestructura middleware del sistema. Normalmente se asume una organización jerárquica de dicha infraestructura con cabeza el Gateway; en caso de múltiples Gateways se puede optimizar la topología de las infraestructuras de forma tal de privilegiar el balanceo de la carga entre los actores. La infraestructura middleware más sencilla asume un único actor que implementa las funcionalidades de Estación Base y Gateway.

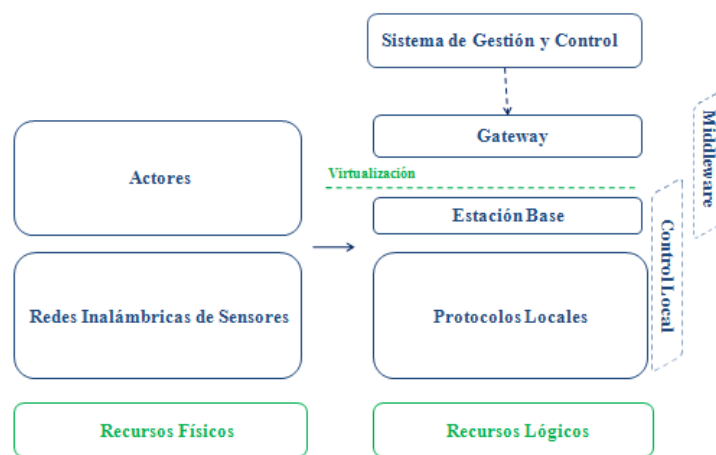


Figura 2.2. Visión esquemática de la arquitectura.

Para que la infraestructura middleware pueda ser efectivamente operativa el sistema tiene que proporcionar un mecanismo de configuración dinámica articulado en dos niveles: partición de la red en clusters, cada uno encabezado por un estación base, y configuración local de cada clusters.

El Sistema de Gestión y Control interactúa directamente con cada Gateway proporcionando un conjunto de servicios directamente accesibles por los usuarios finales. Como descrito en la siguiente sección, el modelo de Gestión y Control generalizado no es el único posible en el contexto de la arquitectura propuesta.

Finalmente, el Sistema de Información, entendido como el conjunto de modelos informativos de alto nivel y los mecanismos que actúan sobre ellos, aunque abordando verticalmente aspectos caracterizados por diferente nivel de abstracción se considera parte integrante de las infraestructuras lógicamente avanzadas y será objetivo de discusión en las secciones siguientes.

2.3 Recursos Físicos

Esta sección presenta dos objetivos fundamentales del trabajo de la presente tesis: definir un modelo de referencia para los recursos físicos y describir la pila de protocolos implementada por los nodos sensores.

El entorno lógico de referencia es el de recurso físico (Figura 2.2.1).

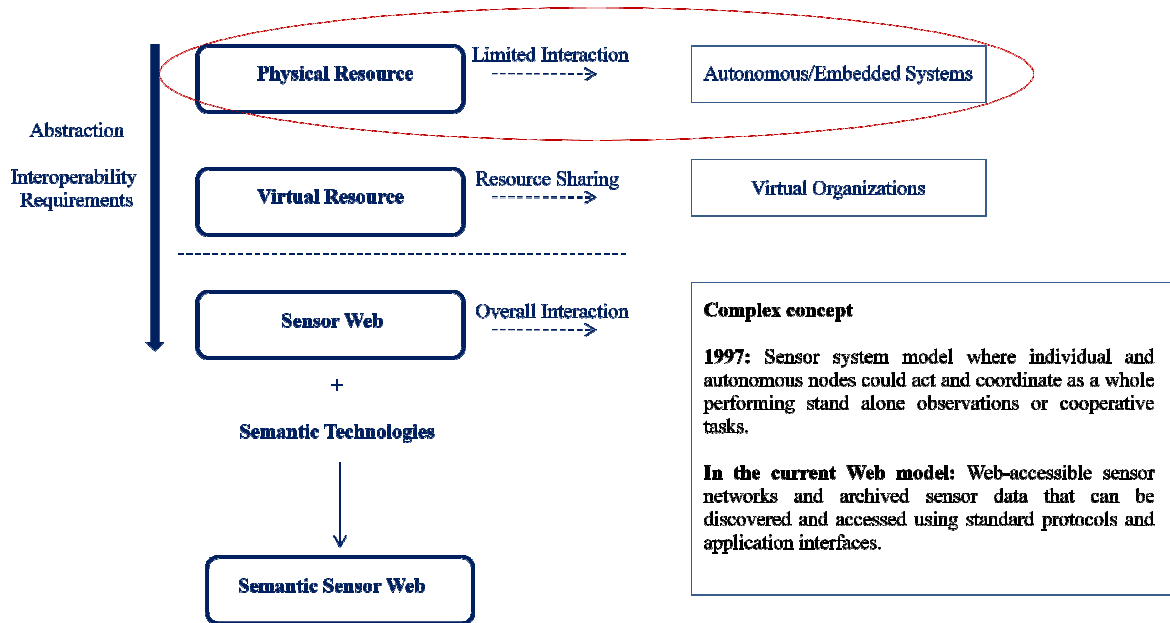


Figura 2.2.1. Entorno lógico: *Physical Resource*.

Definir un modelo de referencia es una necesidad debida a las características intrínsecas de las redes de sensores inalámbricos: las plataformas físicas/lógicas suelen ser soluciones a medida en función del dominio de aplicación.

Adoptar un modelo de referencia para los recursos (siempre y cuando esto sea suficientemente realista) permite enfocar con mayor claridad las características críticas del software proporcionado en función del entorno de aplicación real; además, las características del software proporcionado, aunque teóricamente aplicables utilizando cualquier modelo de plataforma, en la práctica podrían demostrarse poco adecuadas a modelos absolutamente genéricos o poco realistas (por ejemplo nodos sensores con recursos físicos avanzados).

2.3.1 Modelo de referencia de los recursos físicos

En esta sección se describe el modelo generalizado para arquitecturas de sensores. Este modelo es de gran utilidad en el contexto de este trabajo de tesis tanto para evidenciar los límites actuales de las arquitecturas como para definir y caracterizar los modos de funcionamiento e interacción entre los diferentes componentes. Para facilitar el análisis de sistemas complejos, se distingue entre un modelo generalizado de dispositivo y uno de red.

Esta metodología se requiere considerando la gran variedad de plataformas hardware/software actualmente disponibles así como la gran extensión del dominio de

aplicación que incluye arquitecturas profundamente diferentes a soporte de aplicaciones y entornos diferentes.

Aun considerando que el modelo no puede reflejar todas las arquitecturas reales, consideramos las ventajas de trabajar sobre un modelo bien definido significativamente más relevantes de las desventajas introducidas.

2.3.1.1 Modelo de dispositivos

Vista la gran cantidad de aplicaciones que emplean sensores o redes de sensores, identificar un modelo de dispositivo generalizado de nodo sensor.

Una buena aproximación es el dispositivo de bajo coste: capacidad de memoria y de cálculo extremadamente limitada normalmente a soporte de los transductores; eventualmente recursos de comunicación avanzados (inalámbricos por ejemplo).

Dentro de los dispositivos de bajo coste podríamos identificar dos sub-categorías de referencia: dispositivos para entornos controlados y para entornos con despliegue aleatorio. Los primeros suelen ser dispositivos asociados (directamente empujados) en otras máquinas/dispositivos; se caracterizan por una ingeniería de base muy sencilla. Los segundos suelen emplearse en entornos no conocidos a priori, suelen trabajar en red y se suelen alimentar a batería; las arquitecturas, normalmente, están orientadas a la optimización de los recursos (energéticos fundamentalmente) de acuerdo con una ingeniería bastante más avanzada.

La complejidad de las plataformas software de gestión suelen ser “proporcionales” a las capacidades hardware. Hay entornos de gestión derivados de correspondientes entornos genéricos (Linux por ejemplo) y sistemas operativos a medida (TinyOS por ejemplo). La idea fundamental es soportar entornos concurrentes a eventos; esta aproximación garantiza, potencialmente, gestión extremadamente eficiente de los recursos: el dispositivo puede asumir un estado de inactividad caracterizado por consumos no relevantes en el cual pasa gran parte del tiempo; como consecuencia de un evento interno (expiración de un temporizador) o externo (recepción de un mensaje), el dispositivo pasa a un estado de actividad para ejecutar las tareas previstas; al final de dichas tareas, el dispositivo vuelve a estar en estado latente (inactivo).

2.3.1.2 Modelo de red

En los sistemas más modernos es bastante común la utilización de redes o sistemas de sensores más que de dispositivos sensores aislados. Los motivos pueden ser múltiples: necesidad de diferentes puntos de observación, gran área de cobertura, precisión o tareas cooperativas heterogéneas.

Exactamente como en relación a los dispositivos, modelos de redes concretos (topologías, densidades y pilas de protocolos) pueden diferir de forma sustancial entre ellos.

Desde un punto de vista del modelo de red, las redes de sensores se pueden clasificar como redes *data-centric* (aunque esta definición no es unánimemente aceptada) con intrínseca centralidad de la estación base: se suele suponer que los nodos sensores transmitan la información generada a un dispositivo de referencia comúnmente denominado Estación Base. Una Estación Base suele caracterizarse por hardware avanzado y puede implementar tanto funcionalidades básicas como funcionalidades extremadamente avanzadas.

Este modelo centralizado suele tener la gran ventaja de soportar casi exclusivamente tráfico de datos convergentes (desde los nodos sensores hacia las bases); comunicación divergente (desde las bases hasta los nodos) suele limitarse a las tareas de configuración, normalmente en modo broadcast. Poco utilizada la comunicación nodo a nodo. Una excepción significativa es representada por las redes que incluyen actores.

Las principales desventajas se consideran la vulnerabilidad y el fenómeno conocido como *energy-hole*:

- Una red centralizada implícitamente asume un punto de fallo singular en la red (la Estación Base). Un fallo de la estación base determina el fallo de toda la red.
- Una red centralizada con tráfico principalmente convergente determina que los nodos más próximos a la estación base consuman implícitamente más que los nodos más lejanos. Este fenómeno, conocido como *energy-hole* se debe a la comunicación multi salto que obliga a los nodos más próximos a las estaciones base a soportar tráfico proveniente de otros nodos. Este consumo irregular afecta negativamente las prestaciones (en términos de tiempo de vida) de la red en caso de alimentación con baterías: si la batería de los dispositivos más próximos se agota, estos pararán de funcionar y dispositivos más lejanos, aunque disponiendo de recurso energéticos, se quedarán aislados de las estaciones base. Las prestaciones de la red se pueden mejorar a través de políticas de enrutamiento eficientes y a través del uso de múltiples estaciones base que limiten el número de saltos medio en la comunicaciones (*clustering*). Una posible técnica alternativa para limitar el fenómeno de *energy-hole* asume el despliegue de los nodos con densidades superiores en proximidad de las estaciones base y decrecientes respecto a las mismas; dicho tipo de despliegue puede garantizar un tiempo de vida medianamente superior aunque sea más costoso e inaplicable en ciertos casos.

2.3.2 Arquitectura de protocolos de Redes Inalámbricas de Sensores

En esta sección se describe la arquitectura de protocolos diseñada y desarrollada en el contexto del presente trabajo de tesis.

Todos los mecanismos y los protocolos propuestos, aunque de validez general, se consideran asociados a los modelos de recursos físicos anteriormente descritos. Por otra parte no se hace ninguna asunción o restricción respecto a parámetros físicos/topológicos de las redes ni al dominio de aplicación de acuerdo con la metodología propuesta.

La sección se estructura en dos partes que describen, respectivamente, la evolución de la arquitectura a lo largo del trabajo de tesis y para finalmente focalizar el trabajo en la descripción de la arquitectura de protocolos en su versión final.

2.3.2.1 Evolución de la arquitectura

La primera versión de la arquitectura de protocolos propuesta (Figura 2.3) se caracterizaba por una fase de configuración fuertemente centralizada con el objetivo fundamental de repartir óptimamente los recursos en presencia de aplicaciones complejas o múltiples aplicaciones compartiendo los mismos recursos lógicos/físicos de la red.

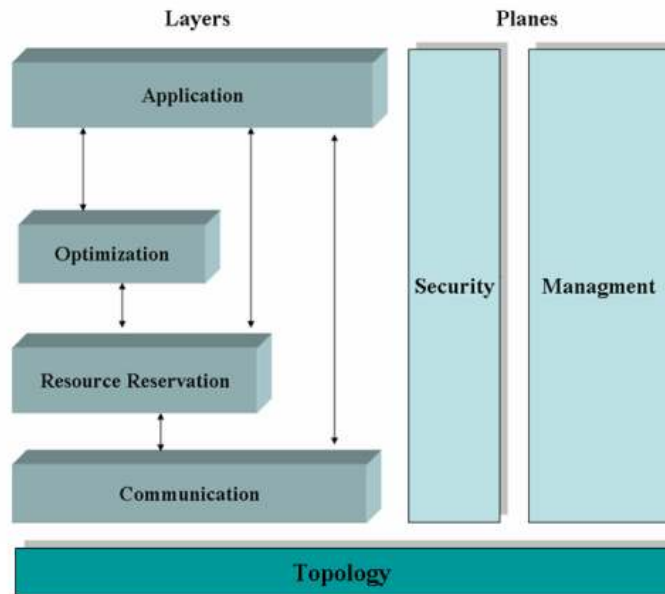


Figura 2.3. Modelo con reserva de recursos. Figura original de [Pos2].

Dicho modelo, denominado “modelo de reserva de recursos”, propone una solución innovadora en el contexto de las redes inalámbricas de sensores; pruebas preliminares efectuadas en prototipos de escala reducida y a través de técnicas de simulación obtuvieron buenas prestaciones en aplicaciones complejas (redes heterogéneas, tareas cooperativas o múltiples aplicaciones/tareas) con clusters de tamaño reducido; las prestaciones se han mantenido aceptables al decrecer de la complejidad de las aplicaciones y al crecer moderado de la escala media de los clusters.

A pesar de la evaluación preliminar en su complejo más que positiva y del enfoque innovador, el modelo con reserva de recursos ha progresivamente evolucionado, en base a los resultados obtenidos en varios experimentos y simulaciones, hacia una arquitectura más clásica y efectiva, aunque caracterizada por ciertas peculiaridades. Esta evolución ha sido determinada por las consideraciones expuestas a continuación:

- *Dominio aplicativo óptimo reducido.* El modelo propuesto trabaja de forma óptima como soporte de aplicaciones poco comunes que implican la necesidad de alocar recursos a múltiples aplicaciones que comparten la misma red física de sensores. Aplicaciones más clásicas suelen ser críticas por densidad, escala o entorno real de trabajo, difícilmente por el tipo de aplicación en si misma.

Uno de los objetivos fundamental del presente trabajo es proporcionar una solución generalizada multi-dominio en el contexto de una política orientada a la reducción de los costes. Evidentemente se pretende soportar óptimamente las aplicaciones mas comunes, asumiendo un degrado asumible de prestaciones en presencia de aplicaciones muy particulares por las cuales, eventualmente, se pueden plantear extensiones (o particularizaciones) significativas de la arquitectura o, incluso, soluciones a medida. Una lógica de desarrollo inversa que privilegia aplicaciones muy complejas, tal vez solamente teóricas, resultaría posiblemente poco productiva.

- *Solución con gran potencial en termino de inteligencia pero difícilmente generalizable en un único framework concreto.* Una solución basada en un mecanismo centralizado de evaluación y asignación de los recursos puede

implementar políticas de gestión extremadamente inteligentes. Independientemente de los recursos necesarios, relevantes especialmente en términos de consumos en un contexto a recursos limitados como las redes inalámbricas de sensores, dichas políticas de gestión suelen ser característica de la aplicación y el contexto de trabajo a soportar.

Aun pudiendo individualizar y definir unos patrones, se considera en el contexto de este trabajo, poco funcional construir la arquitectura completa sobre un mecanismo virtual que implica, de hecho, tener que definir e implementar soluciones concretas a medida, contrariamente a uno de los objetivos primario del trabajo de tesis.

- *Dificultad objetiva para soportar arquitecturas de sensores móviles.* La fase crítica es, con diferencia, la configuración; todos los otros protocolos y mecanismos trabajan en función de los recursos asignados. Para garantizar rutas de comunicaciones consistentes y robustas, el entorno móvil requiere continuamente la reasignación de recursos, implicando prestaciones significativamente inferiores si se comparan con otras soluciones.

Considerando el entorno móvil como de gran interés y perspectiva futura, el trabajo propuesto pretende proporcionar una arquitectura funcional en un contexto de prestaciones aceptables tanto en contextos estáticos como nomádicos y móviles.

- *Problemas de escalabilidad.* Al crecer de la escala la evaluación y la asignación de recursos crece en complejidad de forma relevante. El número de mensajes intercambiado resulta bastante superior al número de mensajes intercambiados en presencia de soluciones más ligeras. las prestaciones, por lo tanto, se consideran aceptables únicamente en presencia de aplicaciones extremadamente complejas que, intrínsecamente, requieren un número importante de mensajes intercambiados.
- *Dificultad en extender o particularizar la arquitectura en función de dominios específicos.* La evaluación y la asignación centralizada de los recursos presentan la gran ventaja de poder repartir óptimamente los recursos de la red considerando el estado global de la misma. Por otra parte, vista la necesidad de fijar los parámetros (o la política) en función de los cuales evaluar y asignar los recursos, el modelo carece de flexibilidad a la hora de tener que integrar nuevos mecanismos en la arquitectura de base o de necesitar políticas de evaluación/asignación de los recursos diferentes de la preestablecida.

Es preferible la utilización de una arquitectura modular, caracterizadas por protocolos de base (como lo de configuración) fácilmente extensibles en un contexto funcional más dinámico y flexible.

Una descripción detallada del modelo de reserva de recursos se considera fuera de los objetivos primarios de esta tesis; en la siguiente sección se presenta la versión final de la arquitectura.

2.3.2.2 Arquitectura modular multi-dominio

La arquitectura de protocolos propuesta (Figura 2.4) se diseñó de acuerdo con una aproximación modular e independiente del dominio de aplicación así como con los otros aspectos metodológicos descritos en los apartados de introducción a la presente tesis.

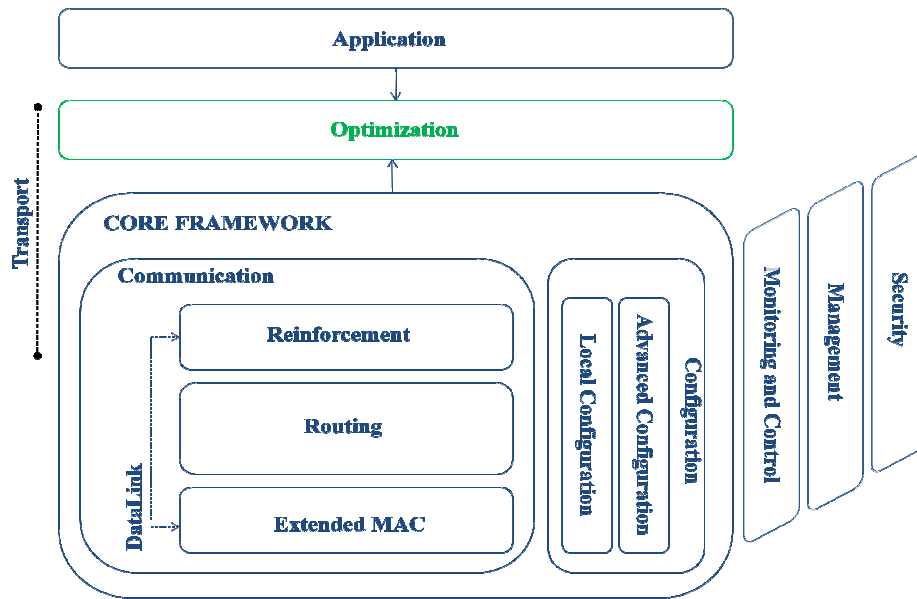


Figura 2.4. Arquitectura modular multi-dominio (Figura original de [Con9]).

Como se puede apreciar, el marco de referencia se compone de una parte central (*CORE*) y de diferentes mecanismos complementarios; todas estas capas de software proporcionan un conjunto abstracto de funcionalidades, básicas y avanzadas, a soporte de los protocolos o mecanismos de nivel aplicación. La meta-capa de optimización actúa a diferentes niveles de la arquitectura para optimizar los recursos, de comunicación normalmente

Se procede a continuación con la descripción de las principales funcionalidades y características de cada una de las componentes de la arquitectura propuesta. Aspectos teóricos (modelo) e de implementación serán objeto específico de los siguientes capítulos.

El módulo *CORE* proporciona un conjunto modular y extensible de funcionalidades que componen la arquitectura de base. Se compone de dos sub-módulos interdependientes:

- *Communication*. Se compone de mecanismos típicos del nivel de enlace de datos y del nivel de red. Este módulo funcional se diseñó con una aproximación estrictamente modular para adaptarse, de forma óptima, a las diferentes características de los dispositivos sensores de bajo coste y a la aplicación a soportar.
- Las funcionalidades de enlace de datos se implementan por dos diferentes protocolos complementarios: *Extended MAC* y *Reinforcement*. El primer protocolo implementa un mecanismo de acceso al medio tipo TDMA, el segundo funcionalidades de réplica de información (envío de información redundante para incrementar la fiabilidad de la comunicación) y/o funciones de *acknowledgment*. Las funcionalidades de nivel de enlace de datos tan solo se requieren en caso de dispositivos que lo requieren (funcionalidades básicas de comunicación extremadamente reducidas). El *Extended MAC* incluye dos protocolos diferentes de acceso al medio (con slot temporal fijo y slot temporal dinámico) a soporte de aplicaciones con características diferentes. El objetivo fundamental de *Reinforcement* es aumentar (si fuera preciso) la fiabilidad del nivel de enlace de datos: puede actuar confirmando la correcta recepción de los mensajes (que pueden reenviarse en caso de ausencia de confirmación en un cierto tiempo) o generando información replicada y redundante, actuando sobre base estadística (modo de funcionamiento preferido en el caso de redes especialmente densas). Desde el punto

de vista del nivel de enlace de datos, la arquitectura propuesta puede trabajar de acuerdo a múltiples modelos: ausencia de funcionalidades de enlace de datos (en general poco común en redes con una cierta densidad), funcionalidades extendidas (acceso al medio simple de acuerdo con uno de los dos modelos propuestos), acceso al medio fiable (integración del mecanismo de réplica de información y/o *acknowledgment*). Las diferentes modalidades de funcionamiento son absolutamente indispensables para garantizar flexibilidad a frente de los diferentes entornos reales que se puedan proponer.

- Si el módulo de comunicación se limitara a las funcionalidades de enlace de datos, la red tan solo podría funcionar en modalidad mono salto ya que las funcionalidades de enrutamiento se ubican a nivel de red. El protocolo de enrutamiento (*Routing*), aproximable a las funcionalidades de nivel de red, proporciona soporte para la comunicación multi-salto. En el presente trabajo se ha tenido en cuenta tanto la comunicación convergente (desde los nodos sensores hacia la base) como comunicación divergente (desde la base hacia los nodos sensores) o punto a punto (comunicación entre nodos sensores) aunque la primera es, con diferencia, la más común. Como evidenciado en el capítulo anterior, el enrutamiento en redes inalámbricas de sensores es un tema de investigación abierto de primario interés. La presente arquitectura implementa diferentes soluciones de enrutamiento (aleatoria, energy-aware, position-aware) que trabajan sobre la misma configuración lógica (descrita en el siguiente capítulo) para soportar óptimamente aplicaciones con diferentes características que pueden trabajar en entornos profundamente diferentes entre ellos respecto a condiciones ambientales, topologías y otros parámetros físicos (densidad, escala). Se compone de dos fases o sub-mecanismos: configuraciones de las rutas (sobre la configuración local) y política de enrutamiento. Detalles al respecto se proporcionan en el Capítulo 4.
- *Configuration*. El módulo de configuración se compone de dos mecanismos (*Local Configuration* y *Advanced Configuration*) que actúan en dos niveles de abstracción distintos: el primer conjunto se preocupa de definir y configurar todos los parámetros básicos para asegurar el correcto funcionamiento tanto de las funcionalidades básicas (comunicación por ejemplo) como avanzadas; el segundo define y configura parámetros lógicamente avanzados o de nivel overlay.
- El protocolo de configuración local es fundamental por cualquier mecanismo de la red en cuestión en cuanto se preocupa de descubrir y configurar correctamente los nodos de la red. La configuración lógica local de un nodo sensor se constituye de diferentes parámetros: identificador de nodo, de grupo, nivel y, eventualmente, sector y cluster asociado. El objetivo fundamental del protocolo de configuración local es configurar dinámicamente los nodos en función del entorno. La configuración dinámica (auto gestión) es un aspecto determinante para asegurar el correcto funcionamiento de la red: en caso de despliegue aleatorio es prácticamente obligatoria; suele resultar de gran utilidad también en caso de topologías predefinidas en cuanto configura los nodos en función de las condiciones efectivamente detectadas que pueden ser influenciadas por obstáculos e interferencias tanto permanentes como temporales. La implementación corriente considera el identificador de cada nodo definido estáticamente (aunque también existen protocolos de asignación dinámica de los identificadores); el grupo puede definirse estáticamente o dinámicamente en función de otros parámetros lógicos; todos los otros parámetros se configuran dinámicamente. En el Capítulo 3 de la tesis se detalla la configuración lógica de un nodo sensor en función de su posición en la

red.

- Los protocolos de configuración avanzada incluyen todos los mecanismos que se proponen configurar dinámicamente parámetros adicionales de la red. Estos parámetros suelen ser lógicamente avanzados (parámetros para definir roles por ejemplo) o de nivel de overlay. Esta última categoría de protocolos es de especial interés en el contexto del presente trabajo de tesis. Más concretamente, se proporciona un mecanismo de configuración dinámica de la red en clusters con estaciones base como cabeza de los mismos. Bajo la asunción de arquitecturas con presencia simultánea de múltiples estaciones base, la partición de la red en cluster constituye una técnica extremadamente eficiente para mejorar tanto las prestaciones de la red (rutas de comunicación medianamente más cortas con la consiguiente limitación del fenómeno de la *energy-hole*) como la robustez de la misma. La configuración o partición de la red de sensores en clusters se puede implementar de acuerdo con diferentes políticas: se puede privilegiar el balanceo del número de los nodos por clusters o la minimización el número de saltos de las rutas de comunicación. La arquitectura propuesta incluye un protocolo de configuración en clusters que minimiza el número de saltos de las rutas de comunicación. Este enfoque puede, en unos casos (redes con densidad no homogénea) determinar clusters no muy balanceados en términos de número de nodos miembros pero, por otra parte, asegura prestaciones superiores.

Un conjunto de mecanismos adicionales se implementa sobre el conjunto de funcionalidades proporcionado por el modulo CORE: *Monitoring and Control, Management y Security*.

El modulo *Monitoring and Control* implementa las más comunes funcionalidades de monitorización (vitalidad, recursos energéticos por ejemplo) y control (arranque y paro de la red, solicitud de datos bajo demanda).

El algoritmo de monitorización de la red se apoya en las funcionalidades de configuración local: estas últimas suelen repetirse de forma periódica para garantizar la fiabilidad de la red frente a condiciones ambientales (interferencias, obstáculos u otros fenómenos temporales que puedan afectar la comunicación) o físicas (cambios de posición, agotamiento de las baterías); evidentemente en aplicaciones que generan datos de forma periódica, también los mensajes de datos se pueden aprovechar para monitorizar la red en un marco de una cierta optimización de recursos. Existe también la posibilidad de solicitar bajo demanda la monitorización de los recursos.

Los mecanismos de gestión de la red (*Management*) suelen ser característicos de aplicaciones y arquitecturas específicas; a título de ejemplo, se ha desarrollado un algoritmo de reconfiguración óptimo de la red a frente de fallos [Con2]. Una descripción exhaustiva de dicho mecanismo se considera fuera de los objetivos primarios de la tesis.

En el modelo de referencia de la Figura 2.4 se menciona un conjunto de funcionalidades (*Security*) dedicadas a mecanismos de seguridad. Aunque de interés en un número relevante de aplicaciones, estos mecanismos no han sido objeto específico de este trabajo.

El nivel de aplicación, en el contexto del presente trabajo, implementa las tareas de nivel de aplicación (adquisición de datos o detección de eventos por ejemplo).

Finalmente, la arquitectura de protocolos propuesta asume la existencia de una meta capa de optimización que asume el rol de optimizar los recursos (principalmente de comunicación) en función de la aplicación considerada. No se trata de mecanismos concretos, si no de

aspectos funcionales que actúan verticalmente a todos los niveles de la arquitectura. El ejemplo más claro es la optimización del *throughput* ya que todos los sistemas están integrados entre ellos para minimizar el número de mensajes intercambiados. Es el caso, por ejemplo, del algoritmo de monitorización que, aunque esté disponible como mecanismo autónomo que se puede solicitar bajo demanda, utiliza mensajes de configuración o aplicación en lugar de mensajes específicos; lo mismo pasa con las tareas de configuración, donde configuración local y global se integran en un único protocolo funcional más eficiente respecto a un protocolo en más fases. Una fase se identifica con una interacción entre base y nodos sensores.

Un breve análisis a posteriori del modelo propuesto evidencia la presencia de funcionalidades comúnmente asociadas al nivel de enlace de datos, otras típicas del nivel de red y, finalmente, al nivel de aplicación. Sin embargo no aparece explícitamente ninguna funcionalidad de nivel de transporte, algo muy habitual en las redes de sensores. La motivación, tal como anticipado en el capítulo anterior, es la fundamental falta de necesidad de este tipo de funcionalidad, a menos de entornos específicos caracterizados por flujos de datos importantes (por ejemplo las redes de sensores a soporte de aplicaciones multimediales). En principio esta última clase de aplicaciones no se considera de interés específico en el contexto de esta tesis. De todos modos, la versión actual del framework utiliza un formato de mensaje que incluye un contador que identifica unívocamente el dato (o mensaje) generado por cada nodo sensor. Si preciso, este campo puede ser utilizado por cualquier mecanismo de transporte que pretenda controlar el flujo de datos integrando las funcionalidades actuales del módulo *CORE*.

2.4 Recursos Lógicos o Virtuales

En la sección anterior la arquitectura global se ha analizado funcionalmente según una perspectiva física: nodos sensores de bajo coste que implementan la aplicación a soportar y un conjunto de infraestructuras funcionales (Estación Base y Gateway) organizadas de forma jerárquica que gestionan el flujo de datos hacia estructuras o sistemas lógicamente superiores (centro de control por ejemplo) así como del flujo de información de retorno.

Esta visión es muy útil a la hora de proporcionar un análisis detallado de las funcionalidades básicas de la arquitectura; sin embargo es una perspectiva limitada a la hora de considerar características abstractas, globales o lógicamente superiores de la arquitectura de redes de sensores.

Esta sección asume el entorno lógico de organización virtual (Figura 2.4.1).

En el contexto del análisis lógico de los recursos es necesario hacer referencia al concepto de organización virtual que se define como un conjunto dinámico de individuos y/o instituciones; la interacción entre los mismos es definida por un conjunto complejo y eventualmente dinámico de reglas y condiciones sobre recursos compartidos. Todas las organizaciones virtuales comparten una estructura de base fundamentalmente compleja; pueden diferir por escala, objetivo, duración y estructura lógico/física.

Una perspectiva lógica de la arquitectura propuesta asume recursos físicos (nodos sensores) asociados a *Recursos Lógicos Locales* (Estaciones Base); en otras palabras podemos asumir que una estación base “representa” su red de sensores asociada o, más precisamente, un cluster de la red a la que pertenece; su actividad o rol se puede enfocar de acuerdo a dos perspectivas diferentes: por una parte gestiona, de forma totalmente transparente, los

recursos físicos asociados y, por la otra, implementa las funcionalidades de enlace a elementos de la arquitecturas lógicamente superiores.

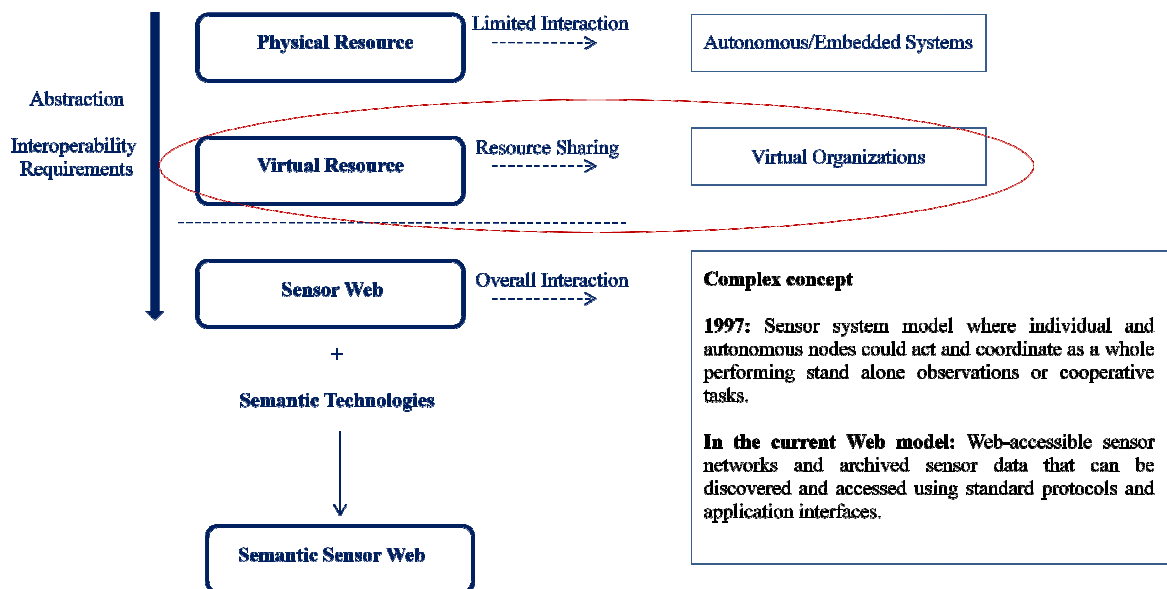


Figura 2.4.1. Entorno lógico: *Virtual Resource*.

El actor desde un punto de vista lógico directamente consecutivo a un recurso lógico local es el *Enabler* que es físicamente coincidente con un Gateway. Este componente es funcionalmente el más avanzado y, por ciertos aspectos críticos, lo es de toda la arquitectura. Un *Enabler* está lógicamente asociado a la entera red de sensores que representa. Del mismo modo que los Recursos Lógicos Locales, implementa funcionalidades tanto para la interacción con elementos de nivel lógico inferior (Recursos Lógicos Locales) como para la interacción con elementos lógicos superiores (centro de control por ejemplo).

El *Enabler* es un elemento lógicamente y funcionalmente crítico en el contexto de cualquier arquitectura: es el (único) elemento visible por parte de cualquier sistema, interno o externo, que pretenda interactuar con los recursos físicos. De forma absolutamente transparente gestiona toda la red que representa proporcionando todas las clases de interfaces de acceso y gestión necesarios en el contexto de la aplicación la que se destina la red de sensores.

Aunque sea posible identificar las funcionalidades de los *enablers* en cualquier sistema, la ingeniería y la complejidad de estos componentes puede ser bastante diferente en función de su contexto lógico y de los objetivos y características de la infraestructura de referencia. Una discusión exhaustiva sobre el rol de los *enablers* en diferentes contextos y modelos se considera fuera de los objetivos principales de este trabajo.

En el contexto propuesto, hemos considerado el rol del *Enabler* bajo un punto de vista funcional y bajo un punto de vista lógico; el primer punto de vista incluye dos entornos de interés específico (sistemas autónomos y sistemas abiertos) que se detallan a continuación. En las siguientes secciones se caracteriza el conjunto de servicios implementado. Por otra parte, las características lógicas de dichos componentes son objeto específico de las siguientes secciones del capítulo donde se privilegia una visión más amplia de los sistemas propuestos.

2.4.1 Sistemas autónomos y Sistemas abiertos

En su más común definición, un Sistema Autónomo es un conjunto de redes y dispositivos router IP que se encuentran administrados por una sola entidad (o en algunas ocasiones varias entidades) que cuentan con una política común de definición de trayectorias para Internet. Esta definición es propia de una visión a nivel de red de una arquitectura. El concepto se puede extender a otros entornos y puede definirse como un conjunto de elementos que interactúan entre sí con el fin de lograr un objetivo específico, sin la necesidad de elementos externos que puedan modificar o influir en el resultado final.

En el contexto del presente trabajo, se define sistema autónomo como una arquitectura distribuida de redes de sensores (recursos físicos) gestionada de forma centralizada. La visión más idónea es la de un modelo que considera el sistema principal como la composición de subsistemas independientes que interactúan entre ellos. Como se ha mencionado, la característica fundamental de una arquitectura autónoma respecto a una abierta es la posibilidad de controlar y gestionar todos los recursos de forma centralizada (Figura 2.5). Esto no significa que el sistema tiene que ser necesariamente diseñado de acuerdo con una política centralizada, si no, simplemente, que existe esta posibilidad concreta: el modelo de organización virtual de referencia es uno de los más sencillos posibles en cuanto asume la presencia de una única entidad de gestión. Evidentemente este tipo de afirmación tiene una validez relativa en cuanto este modelo de organización virtual puede organizarse, internamente, de forma más o menos compleja. La complejidad de una organización virtual, aunque conforme con un modelo de referencia concreto, debería evaluarse en casos concretos.

Como se puede apreciar en Figura 2.5, un conjunto de módulos funcionales de alto nivel, interactúan con los recursos físicos de forma transparente y totalmente abstracta a través de los *enablers*. Independientemente del hecho que dichas funcionalidades se implementen de acuerdo con políticas más o menos centralizadas, este tipo de arquitectura implica que el conjunto de servicios implementados por los *enablers* sea relativamente reducido. La complejidad de dichos servicios crece cuanto el sistema se organiza de forma más distribuida. Un enfoque totalmente centralizado podría reducir las funcionalidades de los *enablers* a simples pasarelas de datos. Contrariamente, un contexto distribuido de la información y de las funcionalidades de alto nivel implica que los *enablers* gestionen autónomamente tanto la información como los recursos asociados que se acceden, gestionan y controlan de forma centralizada.

Un sistema autónomo no es necesariamente menos complejo de uno abierto o cualquier otro tipo de sistema. La complejidad real de los diferentes servicios que componen el sistema depende, únicamente, de la complejidad de las funciones concretas implementadas. Tampoco los sistemas autónomos implican recursos o modelos estáticos a ningún nivel (por ejemplo los recursos lógicos pueden ser descubiertos dinámicamente). Por otra parte, una organización virtual conceptualmente poco articulada asegura posibilidades de gestión y análisis de tipo centralizado.

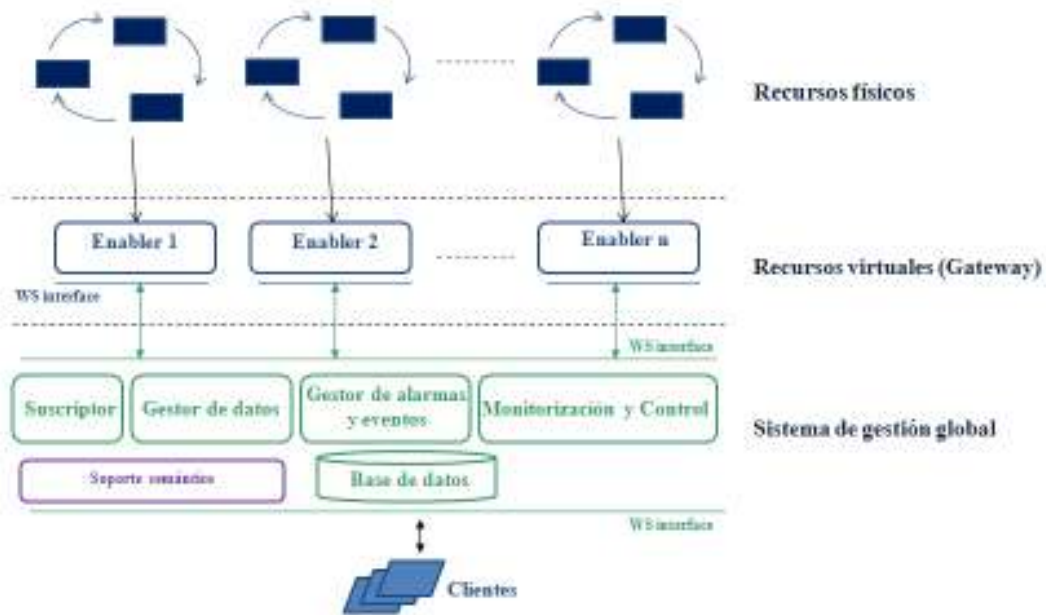


Figura 2.5. Visión de un sistema autónomo con acceso a múltiples recursos físicos.

Una posible definición de sistema abierto es la de un sistema informático que proporciona alguna combinación de interoperabilidad, portabilidad y uso de estándares abiertos. También puede referirse a los sistemas configurados para permitir el acceso sin restricciones por parte de personas y/o otros sistemas. Evidentemente se trata de una definición bastante genérica y abstracta que puede particularizarse dentro de dominios de aplicación concretos.

En nuestro contexto, un sistema abierto (Figura 2.6) asume un entorno lógico más amplio en el cual coexisten diferentes sistemas autónomos; estos sistemas pueden interactuar entre ellos así como con aplicaciones de amplia escala (aplicaciones web-centric por ejemplo).

En este caso, como se puede apreciar en Figura 2.6, la interfaz de acceso a cada sistema autónomo es un *enabler*. Esto equivale a un enfoque completamente distribuido de cada recurso lógico en un sistema autónomo.

La diferencia fundamental entre un sistema abierto y un sistema autónomo totalmente distribuido es, fundamentalmente, el modelo de organización virtual de referencia que determina una diferente perspectiva en la interacción entre recursos lógicos: la interacción dinámica entre sistemas autónomos diferentes necesita un alto nivel de interoperabilidad objetivamente más difícil de alcanzar respecto a sistemas autónomos que suelen trabajar en entornos controlados.

Este tema será objeto de discusión de las partes siguientes del capítulo.

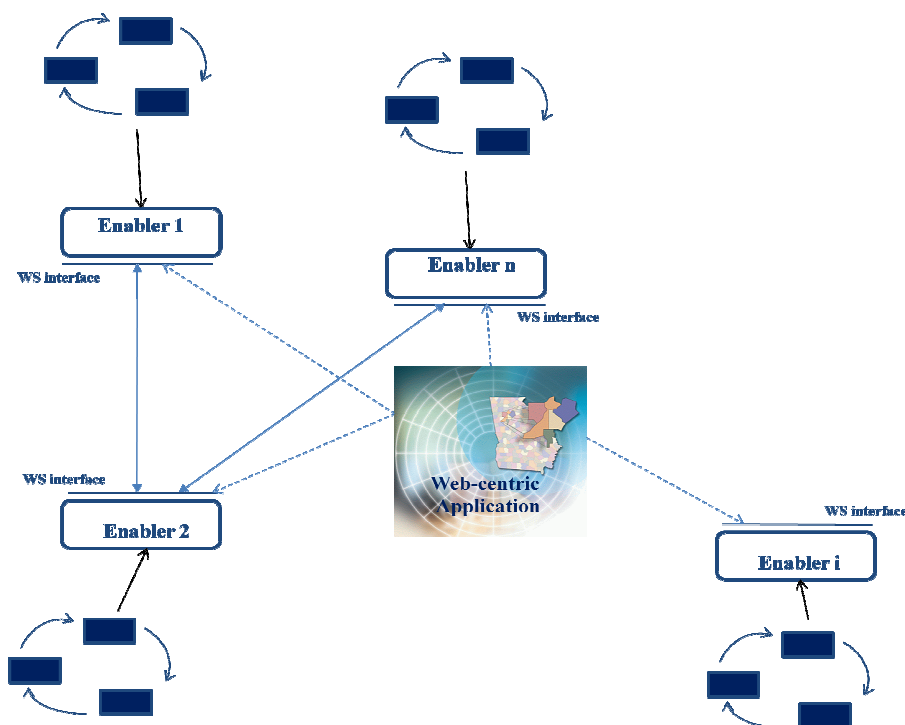


Figura 2.6. Visión de un sistema abierto donde diferentes recursos físicos interactúan entre ellos o con aplicaciones web-centricas.

2.4.2 Enfoque distribuido y Grid Computing

Independientemente de su contexto lógico y de aplicación, las infraestructuras middleware proporcionan un entorno distribuido que se caracteriza por las interacciones entre usuarios y recursos y entre múltiples recursos. Esto implica, por una parte, la interacción entre módulos de control o aplicaciones y los enablers y, por la otra, la interacción transparente entre los componentes de las infraestructuras middleware (estaciones base y enablers).

Vista la gran variedad de aplicaciones que pueden caracterizarse por objetivos, requisitos y complejidad lógica diferentes, se planteó la utilización de un entorno tecnológico flexible, modular y dinámico para infraestructuras middleware.

Tras el análisis de las posibles soluciones existentes descritas en el primer capítulo de la tesis en que se ha descrito el estado del arte, se ha individuado al contexto tecnológico Grid para el desarrollo posterior de la tesis. Más concretamente, todas las implementaciones y bancos de pruebas se han realizado utilizando el paquete de desarrollo Globus Toolkit (GT) en su versión 4 (GT4).

Dicha elección ha sido determinada por diferentes factores; los principales se detallan a continuación:

- *Modularidad y orientación a servicios.* Globus Toolkit implementa un modelo de computación Grid orientado a servicios (web). Este aspecto es absolutamente esencial en cuanto permite modelar e implementar los sistemas de interés de acuerdo con el modelo SOA. La extrema modularidad del modelo orientado a servicios puede contribuir, de forma determinante, al diseño y desarrollo de arquitecturas flexibles y dinámicas en los diferentes entornos de aplicación. Además, la definición de interfaces de acceso en formato interoperable y la clara distinción entre interfaz e implementación del servicio pueden desembocar en un entorno tecnológico efectivo y funcional en contextos abiertos o complejos.

- *Flexibilidad.* El modelo de servicio web comúnmente utilizado asume ausencia de estado (stateless service); esta es una práctica de programación especialmente efectiva en ciertos contextos de aplicación; por otra parte puede afectar negativamente a otras aplicaciones que implícitamente asumen un estado del sistema o recurso con el que están relacionados. Globus Toolkit asume múltiples posibilidades de gestión del estado de un servicio, pudiendo gestionarse cómo un web service estándar (ausencia de estado) o bien como recurso con estado (statefull web service). Además, se proporciona la posibilidad de gestionar de forma dinámica el estado del servicio (por ejemplo mediante temporizadores). En el contexto de la computación Grid se han conseguido importantes resultados en el campo de la computación distribuida de altas prestaciones. Un análisis exhaustivo en este sentido se considera fuera de los objetivos del presente documento.

El enfoque distribuido asume implícitamente que los diferentes componentes del middleware puedan encontrarse en la misma red o en redes distintas. Especialmente en la interacción con los *enablers* por parte de sistemas externos es bastante común la utilización de acceso a los mismos a través de internet.

Considerando que ciertas arquitecturas de sensores podrían incorporar funcionalidades con requisitos (más o menos estrictos) en términos de tiempo de respuesta, se proporciona un breve análisis de las posibles prestaciones de un sistema distribuido implementado en GT4.

2.4.2.1 Evaluación preliminar de GT4

Para conseguir resultados generalizables, con referencia a servicios con estado, se definen dos clases de servicio: la primera incluye operaciones atómicas que no asuman acceso al sistema de ficheros del host; en concreto se consideraran dos operaciones: modificación de una variable de estado (Clase 1) y lectura de la misma (Clase 1(*)); la segunda clase asume operaciones atómicas que interesan al sistema de ficheros; en el banco de pruebas se considera la escritura de un valor numérico en un fichero (Clase 2).

Los resultados obtenidos se han dividido considerando acceso desde red local/intranet (Figura 2.6.1) y acceso desde red externa/extranet (Figura 2.6.2). Todos los resultados obtenidos están directamente relacionados con la latencia media entre las maquinas involucradas.

Como se puede apreciar en Figura 2.6.1, en caso de acceso desde red local (latencia muy pequeña), se nota una fuerte regularidad de los tiempos de respuesta, independientemente de la clase de operación considerada. Se detectan unos picos que indican tiempos de respuesta relativamente altos que no parecen afectar excesivamente las prestaciones medias de los sistemas.

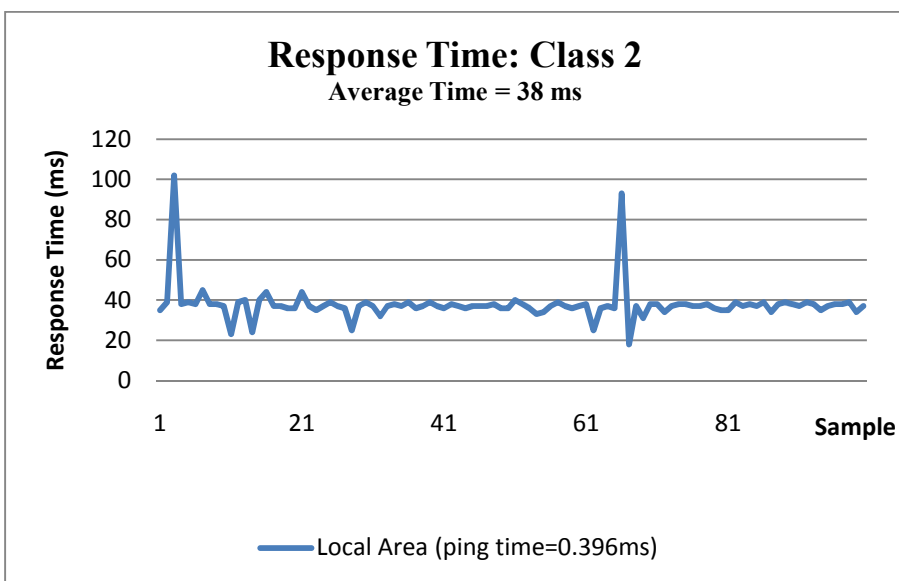
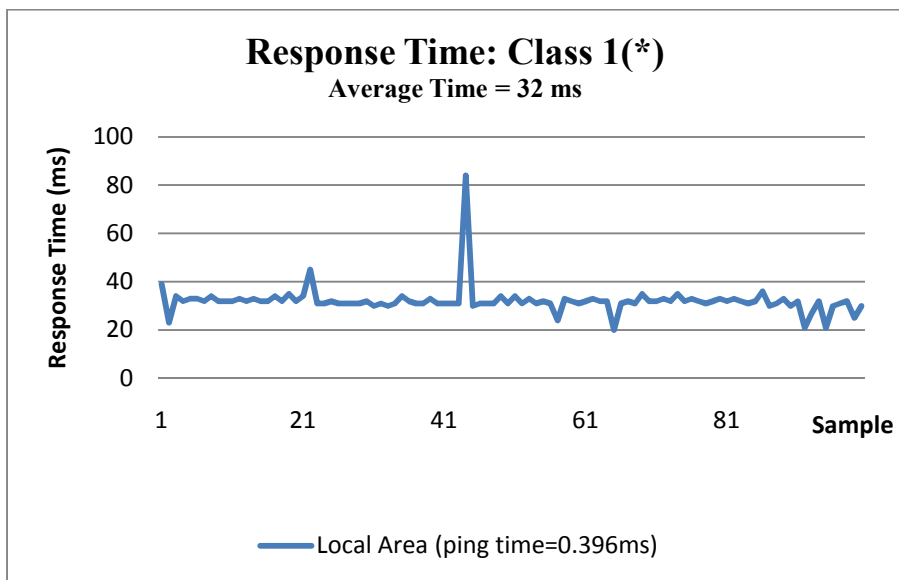
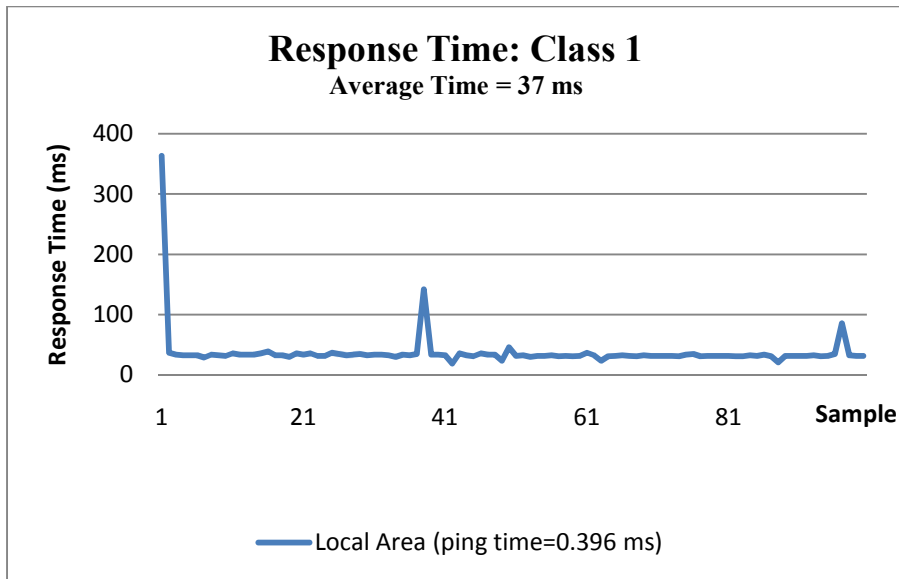
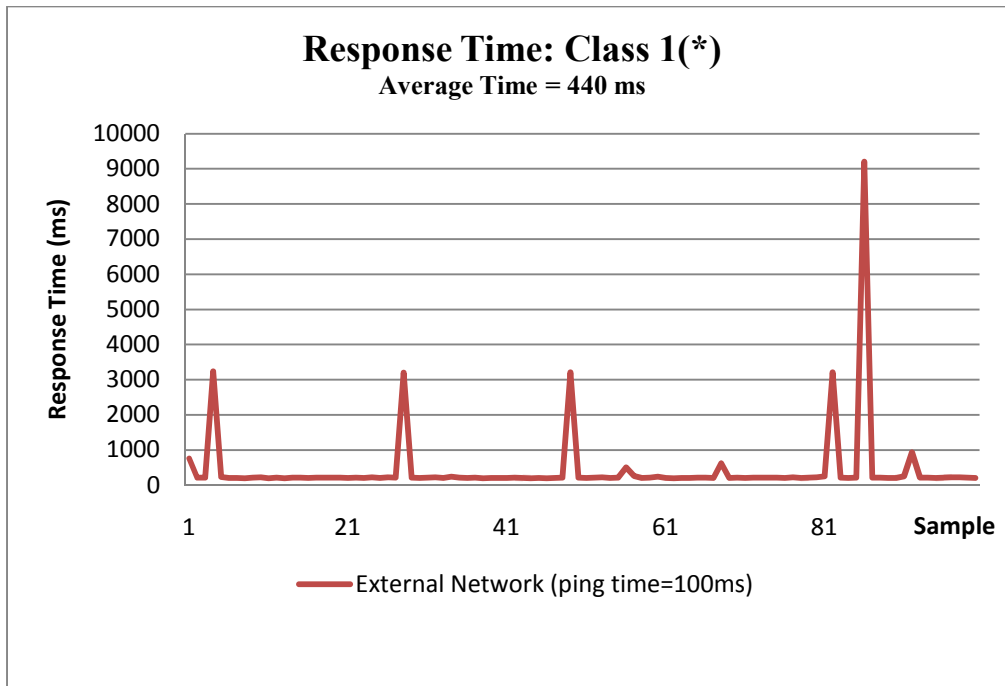
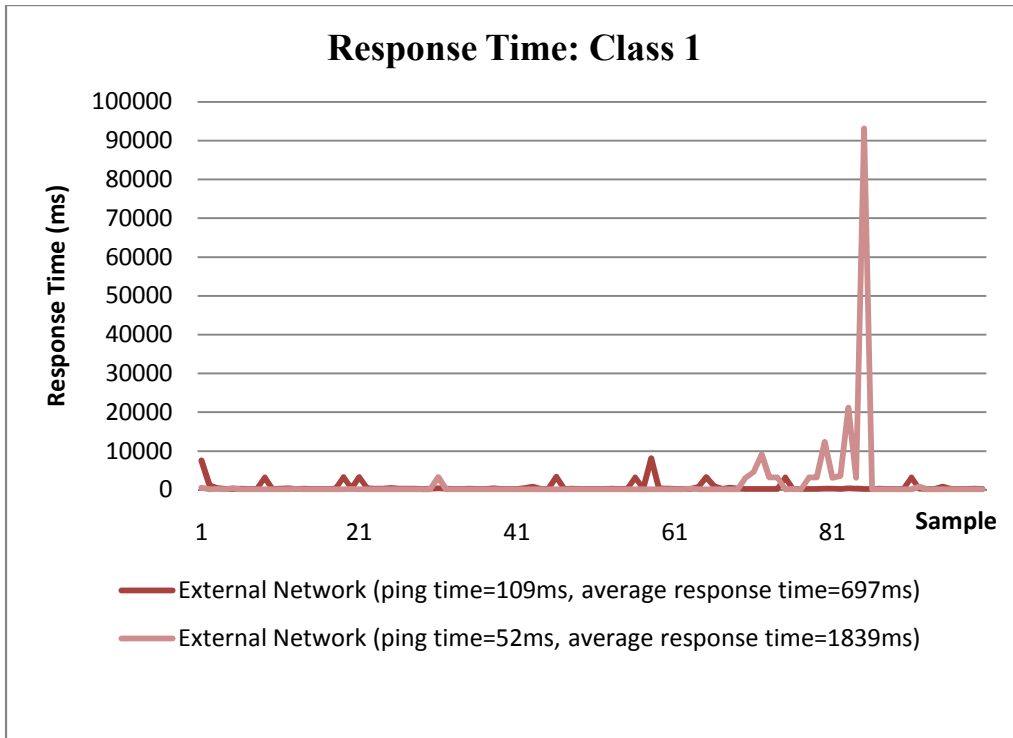


Figura 2.6.1. Tiempos de respuesta de servicios Grid con estado para las diferentes operaciones básicas (red local).

Como esperado, las condiciones irregulares que caracterizan el acceso por red externa se reflejan en los resultados obtenidos (Figura 2.6.2). Los picos detectados podrían afectar, de forma determinante, a las prestaciones en promedio de los sistemas. Evaluaciones en este sentido deberían estudiarse de forma específica para cada aplicación y servicio.



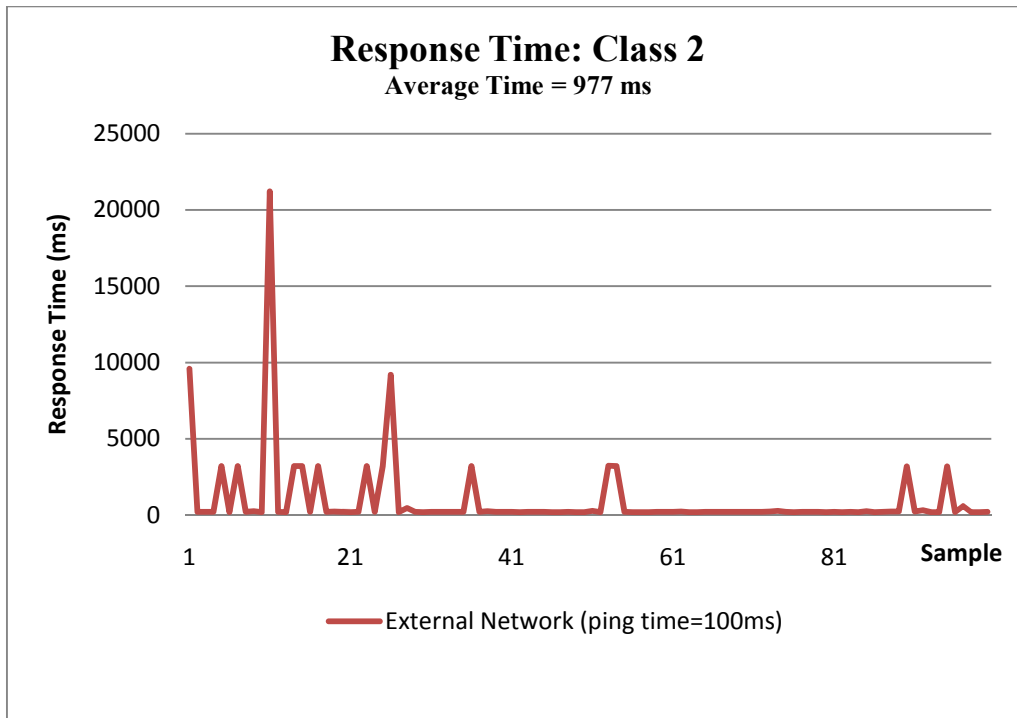


Figura 2.6.2. Tiempos de respuesta de servicios Grid con estado para las diferentes operaciones básicas (red externa).

2.4.2.2 Ejemplo de infraestructura avanzada: monitorización de productos en aeropuertos

Como ejemplo de gestión avanzada del estado de los servicio utilizando tecnologías Grid, se propone la arquitectura para monitorización de productos en aeropuertos representada en Figura 2.6.3. En esta instanciación de la arquitectura diseñada, se pueden apreciar los diferentes componentes funcionales contemplados:

- *Arquitectura de monitorización local.* Se basa en tecnología RFID y se propone detectar e identificar los diferentes productos tanto en los almacenes de los aeropuertos como en aviones físicamente presentes en los aeropuertos. Detalles respecto al sistema de localización e identificación de los productos se consideran fuera de los objetivos primarios del presente trabajo.
- *Sistema de información local.* Cada aeropuerto dispone de su propio sistema de almacenamiento de datos. A través de las infraestructuras de soporte, la información corriente tiene que mantenerse coherente con el estado efectivo de los varios almacenes y aviones. Evidentemente, también se proporciona un historial de la información.
- *Sistema de información global.* Es el sistema constituido por el conjunto de los varios sistemas de información locales.

Para que este sistema pueda funcionar de forma efectiva y eficiente, se necesita un conjunto funcional de infraestructura avanzado para el intercambio de información. El diseño de la arquitectura no puede basarse únicamente sobre una perspectiva funcional porque la organización lógica de la aplicación se modela de acuerdo con un modelo de organización virtual complejo: por una parte, cada aeropuerto se puede considerar de forma individual,

tanto desde un punto de vista de los recursos físicos como desde un punto de vista de la información.

Por otra parte, hay una visión generalizada del sistema que asume la presencia de un sistema de información global. En este escenario pueden intervenir, diferentes actores, interesados en la información relacionada con el conjunto de los aeropuertos, con uno o más aeropuertos, con aviones o, incluso, productos concretos. Las múltiples posibilidades ofrecidas por el contexto tecnológico Grid aseguran el modelado, diseño e implementación de infraestructuras de comunicación avanzadas de acuerdo con un enfoque orientado a servicios extremadamente dinámico: como se puede ver representado en la Figura 2.6.4, los diferentes componentes de la arquitectura constituyen un entorno lógico diferente; más concretamente se utiliza un modelo de servicio convencional (sin estado) para implementar servicios de consulta al sistema de información; las infraestructuras de background, que gestionan el flujo de la información entre los varios componentes funcionales y la consistencia de la información misma, están implementadas por servicios con estado. Una ulterior diferenciación al respecto es necesaria considerando almacenes y aviones. Los almacenes están siempre disponibles y, por lo tanto, se prefiere modelar la información relacionada utilizando servicio con estado. Sin embargo, los servicios relacionados con aviones, aunque bien modelados como recursos con estado, tienen sentido solamente si se asume la presencia física del avión en cuestión en el aeropuerto. Aunque la información de cada avión pasa a ser parte del sistema de información local al aterrizaje, el servicio correspondiente tan solo tiene sentido mientras que el avión permanezca en el aeropuerto; se utiliza, por lo visto, gestión dinámica del estado del servicio: el servicio se crea al aterrizaje del avión y se destruye al despliegue del mismo.

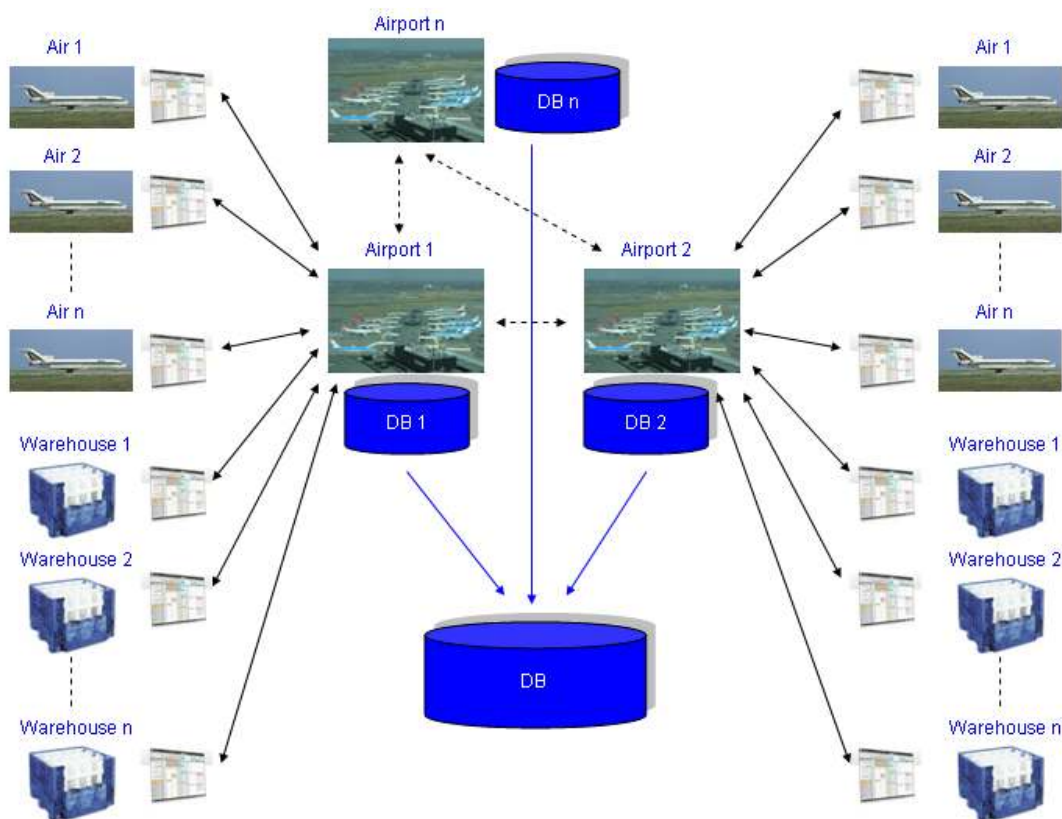


Figura 2.6.3. Visión global de la arquitectura para monitorización de productos en aeropuertos (Figura original de [Con3]).

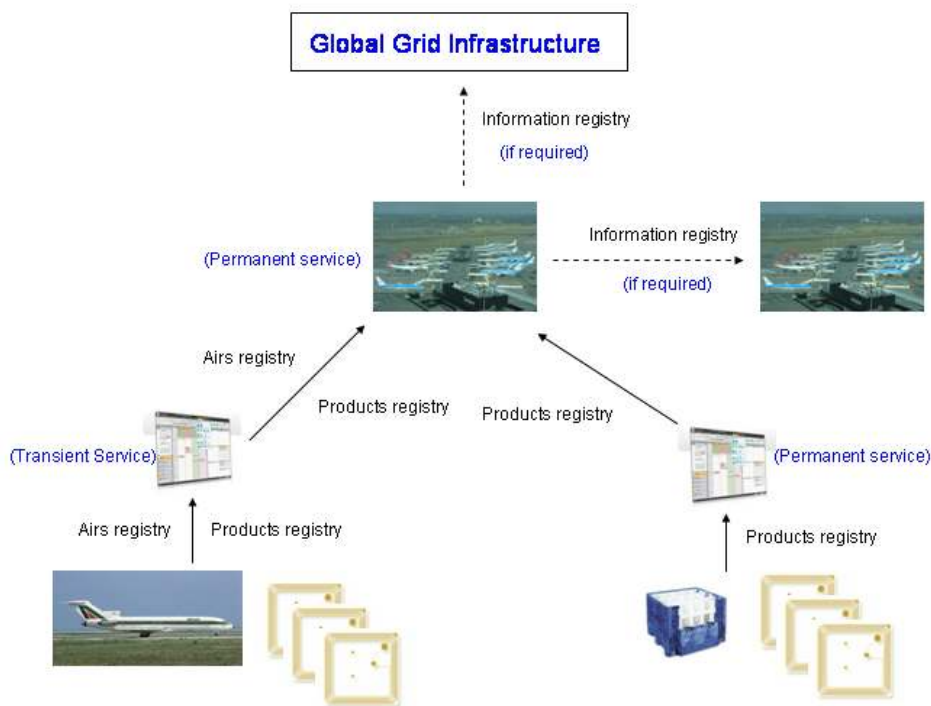


Figura 2.6.4. Gestión del estado de los diferentes servicios de la infraestructura Grid (Figura original de [Con3]).

2.5 Aplicación de la arquitectura en el *Sensor Web*

La presente sección asume el *Sensor Web* como entorno lógico de referencia tal como definido en el primer capítulo de estado del arte.

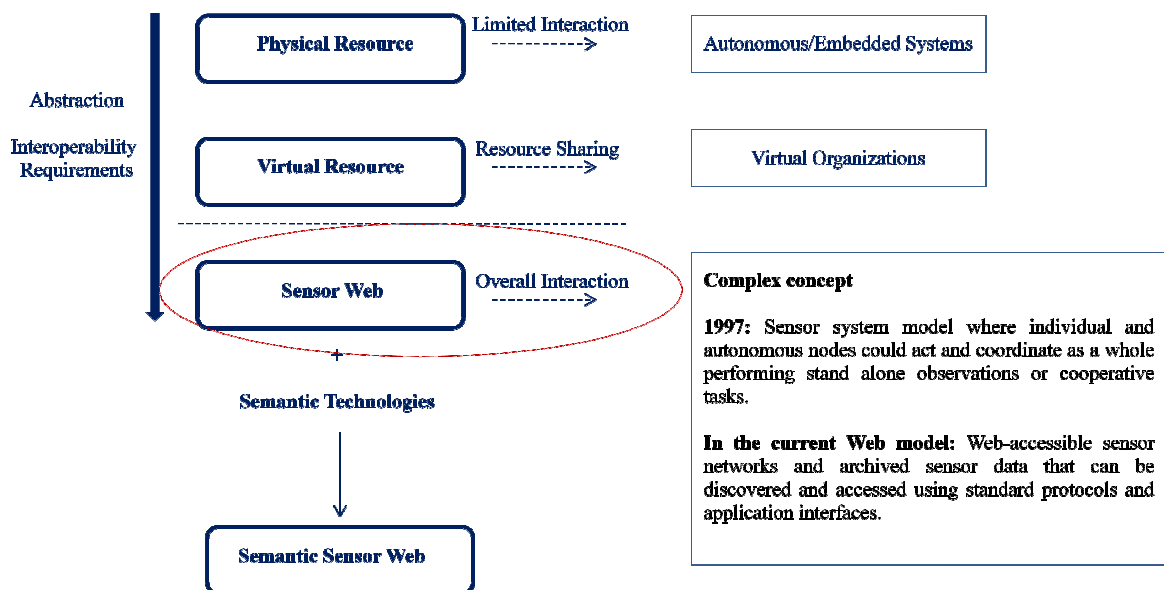


Figura 2.6.5. Entorno lógico: *Sensor Web*.

El concepto, examinado hasta al momento, más próximo al *Sensor Web* es el sistema abierto tal como se ha definido en la sección precedente. El *Sensor Web* pretende ser la generalización de un sistema abierto en cuanto se excluye a priori cualquier tipo de asunción previa sobre los sistemas o aplicaciones a soportar; tales sistemas se conectan a la web que es, por lo tanto, la única plataforma, coherentemente a la visión de Web 2.0 [Mur1].

Paralelamente se requiere un nivel de interoperabilidad implícitamente más alto que lo de cualquier sistema autónomo o conjunto de sistema que interactúan entre ellos (o con sistemas centralizados): exactamente como otros contenidos o servicios web, los recursos sensores deberían poderse buscar, descubrir y ser utilizados para componer servicios o aplicaciones complejas de forma transparente y pervasiva.

Desde un punto de vista lógico, sistemas operantes según el modelo ideal propuesto por *Sensor Web* pueden organizarse de acuerdo con cualquier modelo de organización virtual; respecto a cualquier modelo lógico que refleja una organización propietaria o, de todos modos, controlada se añade la posibilidades de organizaciones o entornos espontáneos.

Como se puede apreciar en Figura 2.7, el modelo a recursos lógicos introducido en las secciones anteriores es perfectamente compatible con el modelo *Sensor Web* bajo todos los puntos de vista. Los *enablers*, de acuerdo con la definición de las secciones anteriores, constituyen el puente lógico/físico hacia recursos físicos como en cualquier otro sistema.

Las actuales limitaciones intrínsecas del concepto del *Sensor Web* han sido objeto de análisis en el capítulo de estado del arte.

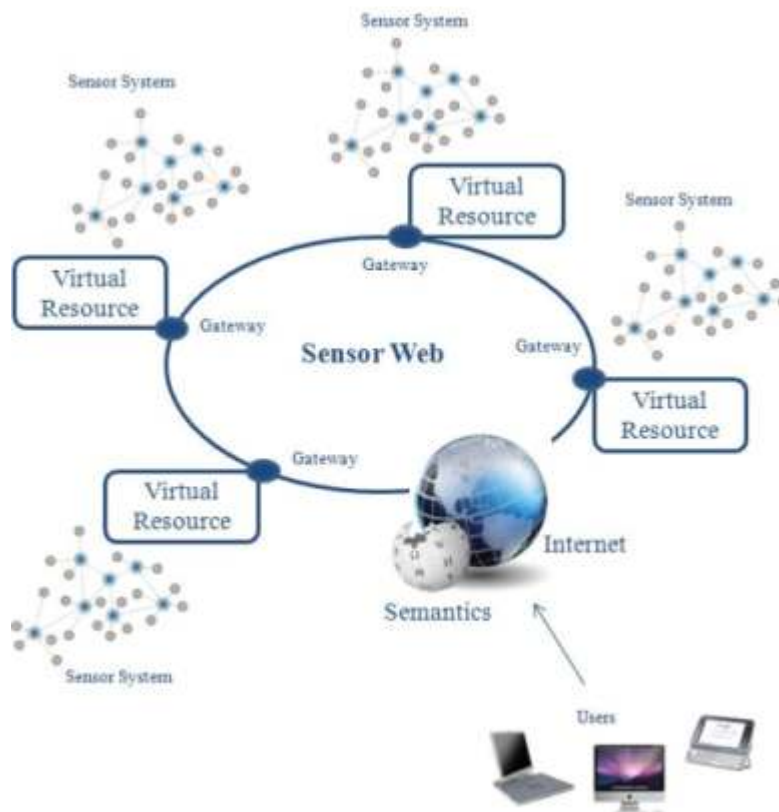


Figura 2.7. Visión de un Sensor Web y posible rol de las tecnologías semánticas (Figura original de [Cha1]).

2.6 Soporte semántico y aplicación en el *Semantic Sensor Web*

Esta sección asume el entorno lógico de referencia más complejo, el *Semantic Sensor Web* de acuerdo con la definición del capítulo anterior.

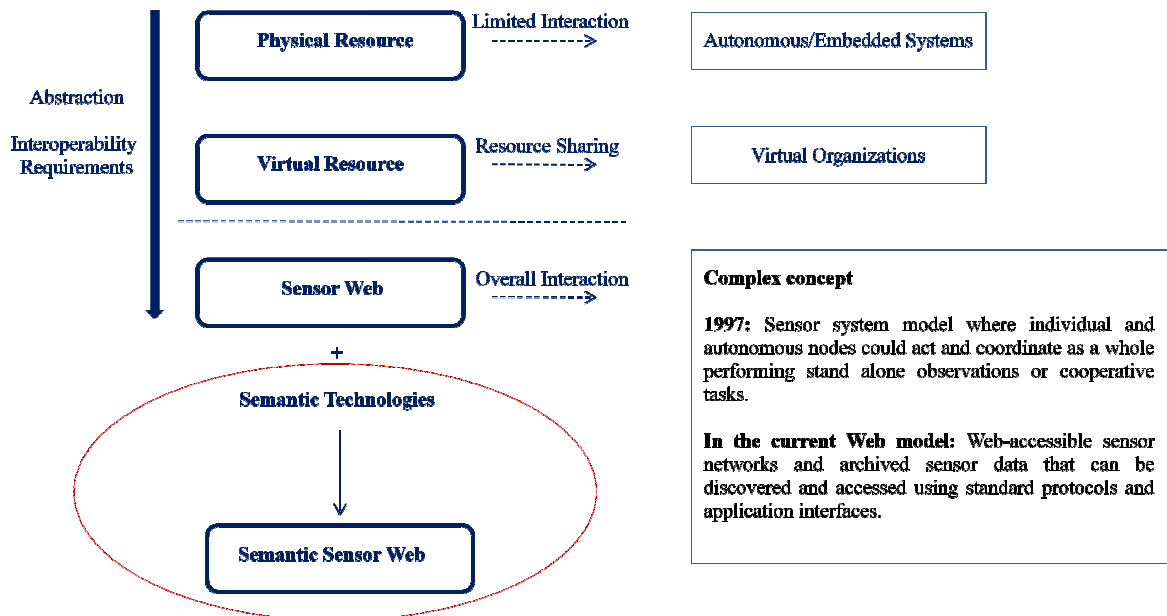


Figura 2.7.1. Entorno lógico: *Semantic Sensor Web*.

Independientemente del concepto de *Semantic Sensor Web*, las tecnologías semánticas (tal como descritas en el Capítulo 1), se aplican, actualmente, en arquitecturas avanzadas de redes de sensores a soporte de diferentes sistemas con diferentes objetivos: soporte avanzado a la descripción y procesamiento de datos [Hua1], gestión avanzada de datos [Lqb1], interoperabilidad avanzada [Ke1], representación dinámica de situaciones y estados de sistemas [Thi1], análisis inteligente de datos [Kaw1] y clasificación [Maj1].

El concepto de *Semantic Sensor Web*, como más veces mencionado, es un concepto muy amplio que asume un entorno tecnológico global de referencia con enfoque semántico (*Semantic Web*).

La arquitectura propuesta se enmarca perfectamente en un contexto de web semántico aunque, como se detallará en las secciones a continuación, utilice un modelo flexible de entorno semántico que permite traer ventaja de las tecnologías semánticas sin necesitar un entorno web completamente semántico.

Las secciones a continuación describen respectivamente:

- Infraestructura para el procesamiento de información semántica y modelo de entorno de conocimiento relacionado (Figura 2.8).
- Ontologías de Dominio y de Procesado de Datos utilizadas para representar el conocimiento en la arquitectura propuesta.

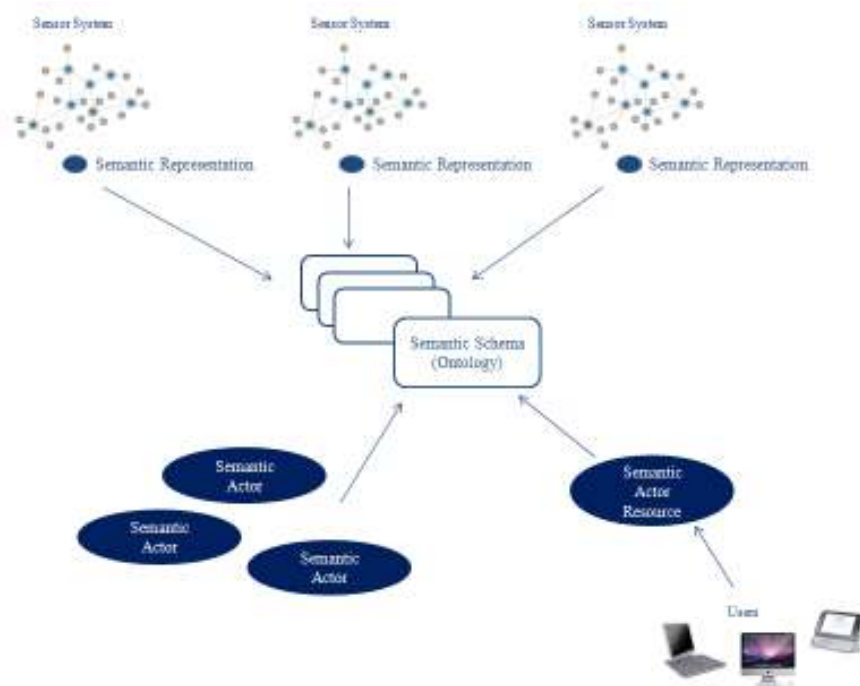


Figura 2.8. Interacción semántica entre redes de sensores (Figura original de [Con11]).

2.6.1 Interoperabilidad Semántica: Entornos de conocimiento y Actores Semánticos

Una aproximación global al *Semantic Sensor Web* (así como al concepto más genérico de *Sensor Web*) asume un único entorno de conocimiento o dominio semántico. Esto implica la posibilidad de generar ambigüedades (y otros problemas relacionados) a la hora de describir diferentes entornos de conocimiento propios en el mismo dominio.

En el contexto del presente trabajo de tesis se asume un modelo reducido de *Semantic Sensor Web* (Figura 2.8) en el cual se asumen enlaces lógicos entre conceptos semánticamente equivalentes de acuerdo con el modelo representado en Figura 2.8.1.

Como se puede apreciar, el nivel de interoperabilidad se puede incrementar, de manera importante, utilizando vocabularios de conceptos compartidos. En este caso, conceptos provenientes de contextos heterogéneos pueden enlazarse lógicamente a vocabularios estándar (Figura 2.8.1) o, simplemente, compartidos para evitar posibles ambigüedades o resultados errados.

La utilización de vocabularios compartidos es similar a la utilización del espacio de nombres en contextos funcionales. Sin embargo, considerando la estructura intrínsecamente más complejas de los modelos semánticos, vocabularios compartidos que representen “islas de conocimientos específicos” se consideran más realistas que vocabularios globales.

También las prestaciones deberían que ser evaluadas atentamente al crecer de la escala de la información.

La disponibilidad de esquemas semánticos (Ontologías), en los cuales tanto la información como su “significado” están formalmente definidos, permite el diseño y la implementación de razonadores estándares capaces de procesar automáticamente los esquemas. Aplicaciones, agentes y toda clase de entidades software pueden interactuar entre ellas a través del intercambio de información semántica automáticamente procesable.

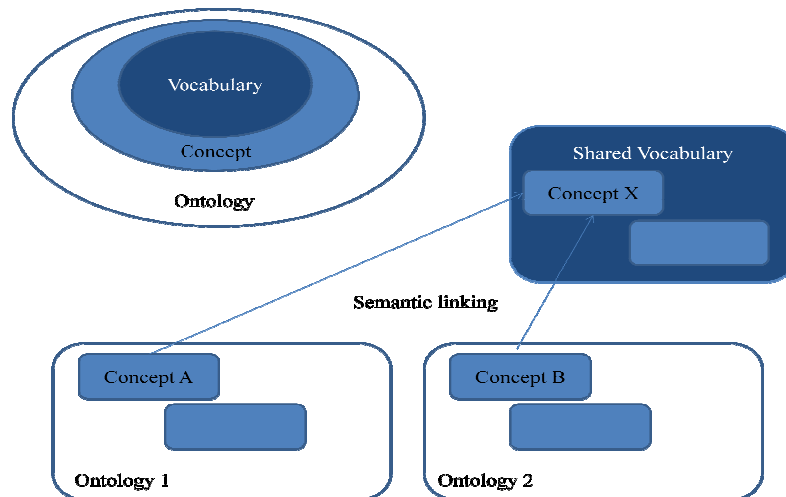


Figura 2.8.1. Enlace lógico entre conceptos semánticos (Figura original de [Con11]).

La estandarización de los lenguajes y de los esquemas para la definición formal de las Ontologías asegura la posibilidad de diseñar e implementar razonadores estándares capaces de procesar de forma automatizadas esquemas semánticos. Dichos razonadores (Jena [Jen1] o Pellet [Sir1] por ejemplo) son la base concreta para la implementación de actores capaces de interactuar semánticamente entre ellos. Actualmente, gracias al soporte de razonadores estándar, la implementación de actores que implementen computación *Ontology-driven* (el actor tan solo interpreta la información contenida en la ontología que únicamente determina el comportamiento del actor) o *Ontology-aware* (el actor es contextual y puede diferenciar su comportamiento en función de la ontología o el dominio de referencia) es bastante más sencilla que previamente. En Figura 2.9 se representa la estructura lógica del actor semántico diseñado y desarrollado en el trabajo de tesis. Como se puede apreciar, todas las funcionalidades se implementan sobre una primera capa de abstracción proporcionada por razonadores estándares. Funcionalidades de complejidad creciente implementan diferentes capas de inteligencia a soporte de diferentes modelos de computación (como los ya mencionados *Ontology-driven* y *Ontology-aware*). Finalmente, la capa más alta implementa dos módulos funcionales de base: funciones específicas del dominio de aplicación (soporte a interrogación compleja) y funcionalidades avanzadas de computación (computación multi-ontología y aprendizaje). Algunos detalles relativos a la implementación y al funcionamiento del actor propuesto se proponen en el Anexo B.

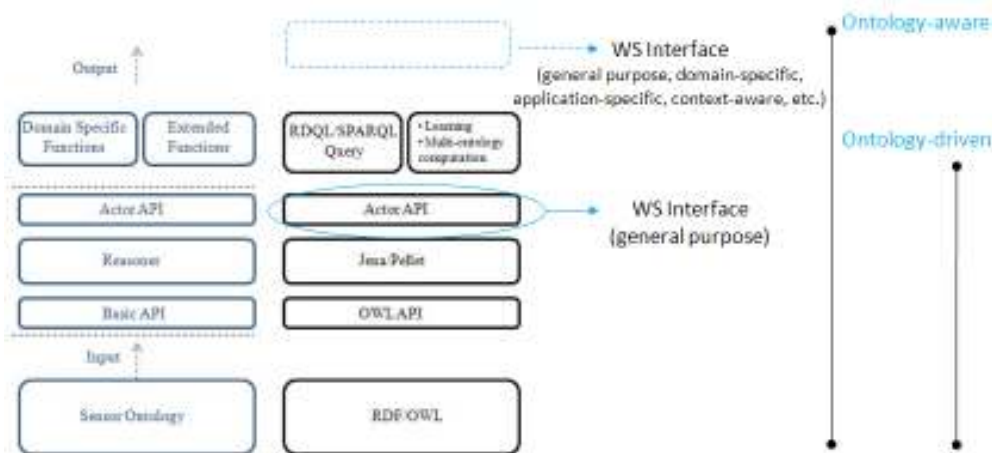


Figura 2.9. Estructura lógica del actor semántico (Figura original de [Con11]).

2.6.2 Ingeniería del conocimiento Semántico: Ontología de Dominio y de Procesado de Datos

En el contexto de las tecnologías semánticas, la ingeniería del conocimiento es, posiblemente, el aspecto más interesante y creativo. Esto se debe, en parte, a la estructura intrínseca de las tecnologías semánticas que asumen modelos de datos complejos caracterizados por la convergencia de estructuras de datos y aspectos de lógica aplicada a los datos mismos. Por otra parte, la falta de metodologías estandarizadas para modelar el conocimiento es más relevante en el ámbito de modelos de conocimiento complejos como se pueden considerar los modelos semánticos. Todos los aspectos metodológicos relacionados con la ingeniería del conocimiento se consideran temas de investigación abiertos de interés primario en el actual panorama científico internacional. Un análisis exhaustivo tanto de las problemáticas como de las posibles soluciones se considera fuera de los objetivos primarios del presente trabajo.

Evidentemente, los modelos de conocimiento suelen relacionarse directamente con sistemas y aplicaciones concretas. Se consideran aspectos de interés específico en el contexto de la tesis la ontología de dominio, la ontología de datos y los entornos de conocimiento para el procesado de información en tiempo real. Estos modelos de datos, por una parte son los más comunes en sistemas de sensores y, por la otra, han sido concretamente utilizados como soporte de los sistemas desarrollados en el ámbito de la tesis.

Una ontología de dominio describe, de forma finita y completa, el dominio de referencia de la aplicación. Teóricamente, cualquier sistema puede asumir la existencia de una ontología de dominio. La ontología de dominio podría resultar especialmente interesante en cuanto se compone de aspectos más propiamente tecnológicos y propios del dominio considerado (sistemas de sensores) y aspectos convergentes desde otros dominios. Sin embargo la coexistencia de estos dos dominios, potencialmente independientes entre ellos, también determina la presencia de un conocimiento “intermedio”; dicho conocimiento debería que relacionar conceptos desde diferente dominios y, al mismo tiempo, definir nuevos conceptos con relación hacia ambos dominios.

La ontología de dominio sería utilizada por todos aquellos mecanismos que pretenden interactuar con componentes específicas del sistema o consultar información de alto nivel referente a cualquier componente del dominio.

Muchos sistemas (o sub-sistemas) suelen “simplemente” intercambiar datos sin necesitar ningún tipo de información respecto a recursos u otros aspectos típicos de las ontologías de dominio. Por ello es habitual proporcionar una ontología para especificar los datos.

Este aspecto es absolutamente fundamental para procesar, de forma avanzada e inteligente, datos heterogéneos generados por diferentes subsistemas. Se incrementa el nivel de interoperabilidad y, además, se proporciona, en el caso de ontologías estructuralmente avanzadas, la posibilidad de describir y consecuentemente analizar los datos de acuerdo con diferentes niveles de abstracción y diferentes perspectivas.

La aplicación de tecnologías semánticas en tiempo real requiere el modelo de ciertas relaciones entre conceptos que representan conocimiento a diferentes niveles de abstracción: por ejemplo la relación entre dato y evento.

Esta área de aplicación de las tecnologías en cuestión puede ser de gran interés para poder analizar los beneficios introducidos en función de las prestaciones de los sistemas, considerando información estructurada de acuerdo con diferentes niveles de complejidad y diferentes cantidades de datos.

2.6.2.1 *Ontología de dominio*

Trabajar con modelos de datos estándar en cualquier entorno de conocimiento proporciona muchas ventajas; sin embargo la estandarización de los modelos de datos no es un proceso fácil o controlable.

El modelo de interoperabilidad semántica permite el desarrollo de entornos de conocimiento propios: las ontologías no tienen necesariamente que ser estándares; aunque la estandarización podría limitar notablemente ciertos aspectos críticos de la ingeniería del conocimiento (ambigüedades por ejemplo), la utilización de modelos que expresen el conocimiento local o personalizado de un cierto sistema podría resultar, en ciertos entornos de aplicación, extremadamente cómodo, modular y eficiente.

En esta sección se propone una ontología de dominio sobre la base de la experiencia obtenida durante la realización de este trabajo de tesis. En este contexto, el objetivo de la ontología de dominio es la descripción, definición y clasificación, de acuerdo con diferentes perspectivas y niveles de abstracción, de las características lógicas y físicas de las redes de sensores consideradas.

Este soporte puede incrementar, de forma significativa, tanto las capacidades de análisis de información y conocimiento asociado como las prestaciones y capacidades de mecanismos que interactúan con el sistema en cuestión (descubrimiento o búsqueda de recursos, interrogación, etc.).

La ontología propuesta se ha diseñado y desarrollado de acuerdo con el esquema lógico representado en Figura 2.10. Como se puede apreciar, el objetivo fundamental es definir un dominio lógico; este se compone de información explícitamente representada e información inferida o deducida. Aunque el dominio definido se considera lógicamente independiente, el mismo se relaciona y, bajo ciertos puntos de vista, se construye sobre conceptos que pertenecen a dominios externos.

Más concretamente, el diseño de la ontología asume la presencia de un concepto (o conjunto de conceptos) central; el conjunto de conceptos centrales tiene que poder ser fácilmente definido en los sistemas interesados y poder relacionarse fácilmente con los conceptos externos de interés.

Además, el conjunto relacional entre conceptos centrales y externos constituye la base lógica para definir conceptos inferidos o deducidos.

Un análisis más abstracto del esquema lógico propuesto evidencia una metodología que asume una doble visión del sistema: una visión “top-down” propone un dominio de conocimiento directamente construido sobre recursos y conceptos físicos con un escaso nivel de abstracción; el punto de vista contrario (visión “bottom-up”) propone una perspectiva del sistema caracterizada por un alto nivel de abstracción, típica de los sistemas altamente avanzados. Evidentemente, las relaciones que asocian conceptos con diferentes niveles de abstracción se consideran la parte más crítica e interesante de todo el esquema.

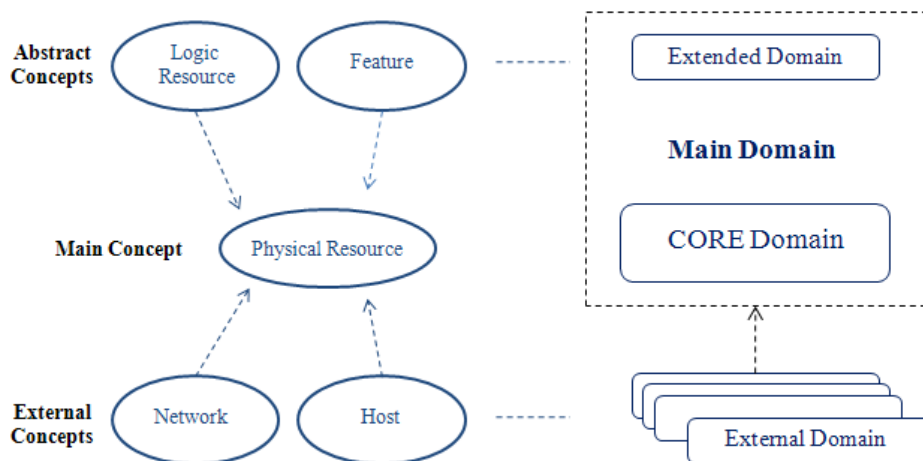


Figura 2.10. Estructura lógica de la ontología de dominio propuesta (Figura original de [Con11]).

En Figura 2.11 se propone una visión genérica de la implementación de la ontología en OWL utilizando Protege [Pro1]. A la izquierda se puede apreciar un desglose limitado al segundo nivel de los conceptos fundamentales; a la derecha se propone el desglose de las características avanzadas (conceptos inferidos).

Un breve análisis del esquema principal, determina la coexistencia de tres diferentes dominios:

- *Main Domain* (o Dominio Principal) que es el dominio que se pretende definir.
- *External Domain* (o Dominio Externo) que es el conjunto de conceptos derivados de dominios externos.
- *Extended Domain* (o Dominio Extendido) que es el conjunto de conceptos inferidos o lógicamente dependientes de los otros dominios.

El dominio principal se compone de conceptos centrales. Estos conceptos se han individuados en los recursos físicos (*Physical Resources*). Este enfoque se debe a múltiples motivos; básicamente los recursos físicos son los únicos conceptos que puedan definirse sobre el mismo esquema lógico sobre la base de parámetros objetivos (físicos) en cualquier sistema; además, es relativamente simple relacionar recursos físicos tanto con conceptos lógicamente superiores (por ejemplo recursos lógicos) como con conceptos de igual/menor grado de abstracción o, incluso, conceptos externos. En otras palabras, el recurso físico ocupa una posición intrínsecamente central en el esquema lógico de un sistema complejo de sensores.

La característica fundamental de los conceptos que componen el dominio externo es que cada uno de estos conceptos tiene un significado autocontenido tanto si se considera independientemente (en su dominio de origen) como relacionado (o contextualizado) en el dominio principal. La ontología definida asume tres dominios externos: *Host*, *Network* y *Supplier*.

En Figura 2.12 se representa la jerarquía de concepto y las relaciones lógicas entre los diferentes componentes del dominio principal, del dominio externo y extendido. Un recurso físico (*Physical Resource*) puede diferenciarse en sensor simple (*Sensor*) y multi-sensor (*Multi-Sensor*). Esta diferenciación tiene una explicación “comercial” (muchas plataformas montan más de un sensor) y una lógica (ciertos sistemas prefieren considerar cada sensor

como un recurso lógico/físico independiente, otros consideran el dispositivo en su conjunto). Ambas explicaciones son perfectamente coherentes entre si. Para garantizar efectividad y la suficiente capacidad de expresión, cada sensor puede asociarse a un multi-sensor, que, por lo tanto, se compone lógicamente de uno o más sensores simples.

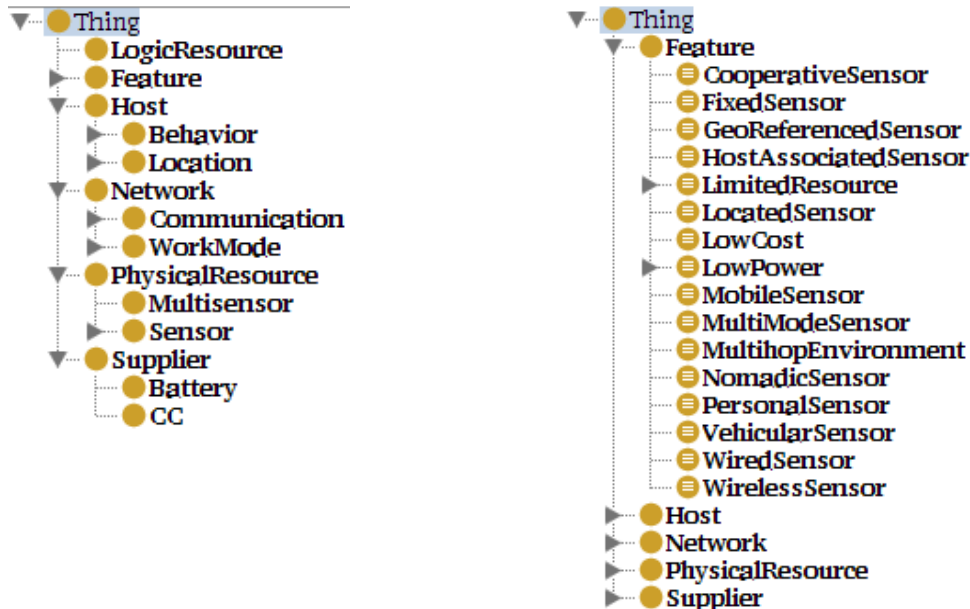


Figura 2.11. Visión la ontología de dominio propuesta empleando Protege (Figura original de [Con10]).

Un recurso físico está relacionado con cada uno de los conceptos externos a través de otras propiedades. Más concretamente, se relacionan recursos físicos con:

- Conceptos que representan las diferentes fuentes de alimentación (*Supplier*). Actualmente, las fuentes de alimentación se clasifican en limitadas (baterías) e ilimitadas (alimentación continua o procedente de un dispositivo externo).
- Conceptos de nivel de dispositivo (*Host*). El objetivo fundamental es relacionar un recurso físico con el comportamiento (*Behaviour*) de su dispositivo respectivo (estático, nomadico, móvil) y con su localización (geo-referenciado, posición asociada a una zona o objeto).
- Conceptos de nivel de red (*Network*). En el caso de redes o sistemas de sensores, se pretende relacionar un recurso físico con la red de pertenencia respecto a las modalidades de comunicación (inalámbrica, cableada, multi-modo) y respecto a su actividad (cooperativa y/o multi-salto, etc.).

El dominio extendido es conceptualmente diferente respecto a los anteriores básicamente por expresar un nivel de conocimiento abstracto y contextual que tiene sentido solo si se considera en el contexto del dominio principal. Se compone de dos categorías de conceptos: *Logic Resources* (o Recursos Lógicos) y *Features* (o Características Avanzadas).

Los recursos lógicos se entienden como la definición de recursos de nivel aplicación; se componen, por lo tanto, de relaciones directas con recursos físicos (utilizando el mismo esquema asociativo que compone multi-sensores a partir de recursos sensores) y de diferentes propiedades para clasificar y especificar la actividad, el rol y otras funcionalidades de alto nivel de los conceptos en cuestión.

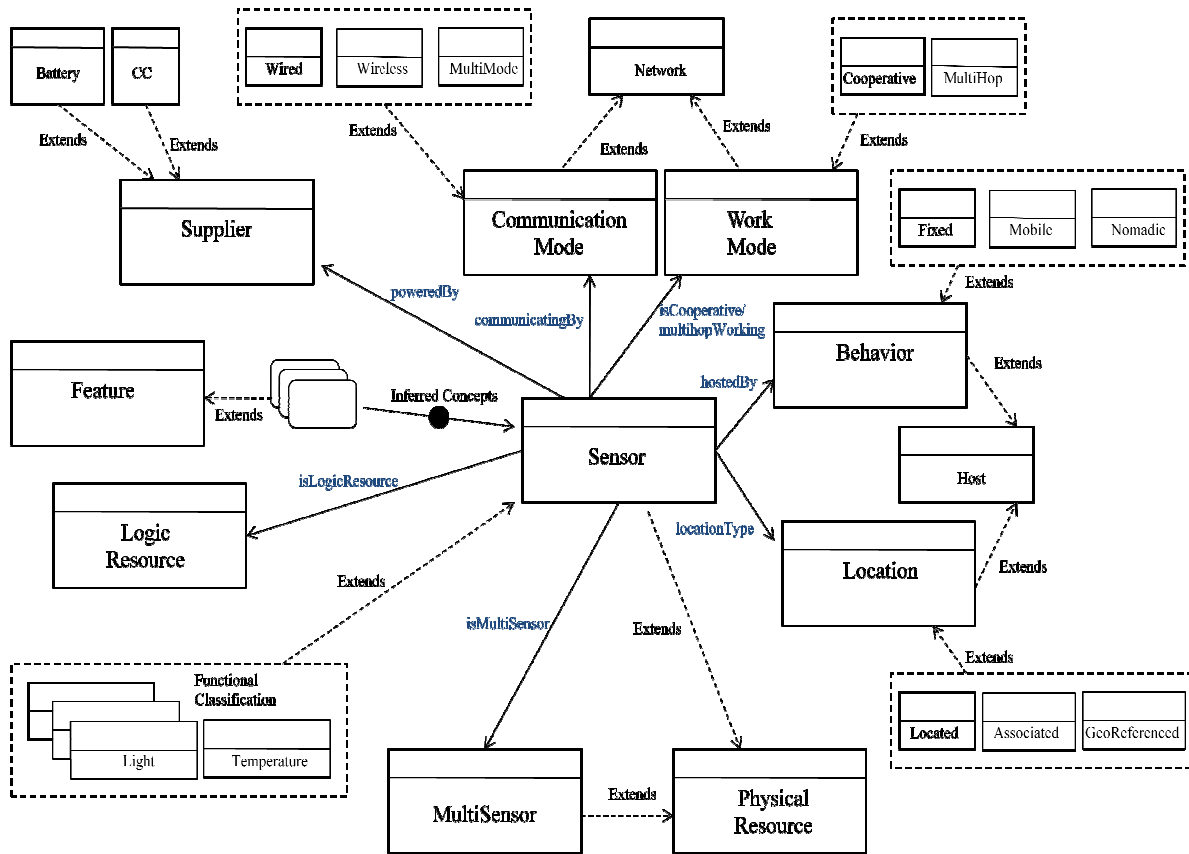


Figura 2.12. Relaciones lógicas fundamentales de la ontología de dominio (Figura original de [Con10]).

Inferred concept	Semantic Rule(s)
Perspective: Network/Communication	
WirelessSensor	Sensor and (communicatingBy some Wireless)
WiredSensor	Sensor and (communicatingBy some Fixed)
MultiModeSensor	Sensor and (communicatingBy some MultiMode)
Perspective: Network/WorkMode	
CooperativeSensor	Sensor and (isCooperative some Cooperative)
MultiHopEnvironment	Sensor and (multiHopWorking some MultiHop)
Perspective: Host/Location	
GeoReferencedSensor	Sensor and (locationType some GeoReferenced)
LocatedSensor	Sensor and (locationType some Located)
HostAssociatedSensor	Sensor and (locationType some Associated)
Perspective: Host/Behavior	
NomadicSensor	Sensor and (hostedBy some Nomadic)
PersonalSensor	Sensor and (hostedBy some Person)
VehicularSensor	Sensor and (hostedBy some Vehicle)
MobileSensor	Mobile or (Sensor and (hostedBy some Mobile))
FixedSensor	Sensor and (hostedBy some Fixed)
Perspective: HighView Feature	
LimitedResource	Sensor and (poweredBy some Battery)
LowCost	LimitedResource and LowPower
LowPower	Sensor and ((communicatingBy some 802.15.1) or (communicatingBy802.15.4))

Tabla 2.1. Conceptos inferidos de la ontología de dominio (Tabla original de [Con10]).

```

-<!--
  file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#CooperativeSensor
-->
-<owl:Class rdf:about="file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#CooperativeSensor">
-<owl:equivalentClass>
-<owl:Class>
  -<owl:intersectionOf rdf:parseType="Collection">
    <rdf:Description rdf:about="file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#Sensor"/>
    -<owl:Restriction>
      <owl:onProperty rdf:resource="file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#isCooperative"/>
      <owl:someValuesFrom rdf:resource="file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#Cooperative"/>
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
</owl:equivalentClass>
<rdf:subClassOf rdf:resource="file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#Feature"/>
</owl:Class>

-<!--
  file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#LowCost
-->
-<owl:Class rdf:about="file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#LowCost">
-<owl:equivalentClass>
-<owl:Class>
  -<owl:intersectionOf rdf:parseType="Collection">
    <rdf:Description rdf:about="file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#LimitedResource"/>
    <rdf:Description rdf:about="file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#LowPower"/>
  </owl:intersectionOf>
</owl:Class>
</owl:equivalentClass>
<rdf:subClassOf rdf:resource="file:///home/flavio/ontologies/SW_draft1/SW_draft1.owl#Feature"/>
</owl:Class>

```

Figura 2.13. Ejemplos de implementación de conceptos semánticos en OWL (Figura original de [Con10]).

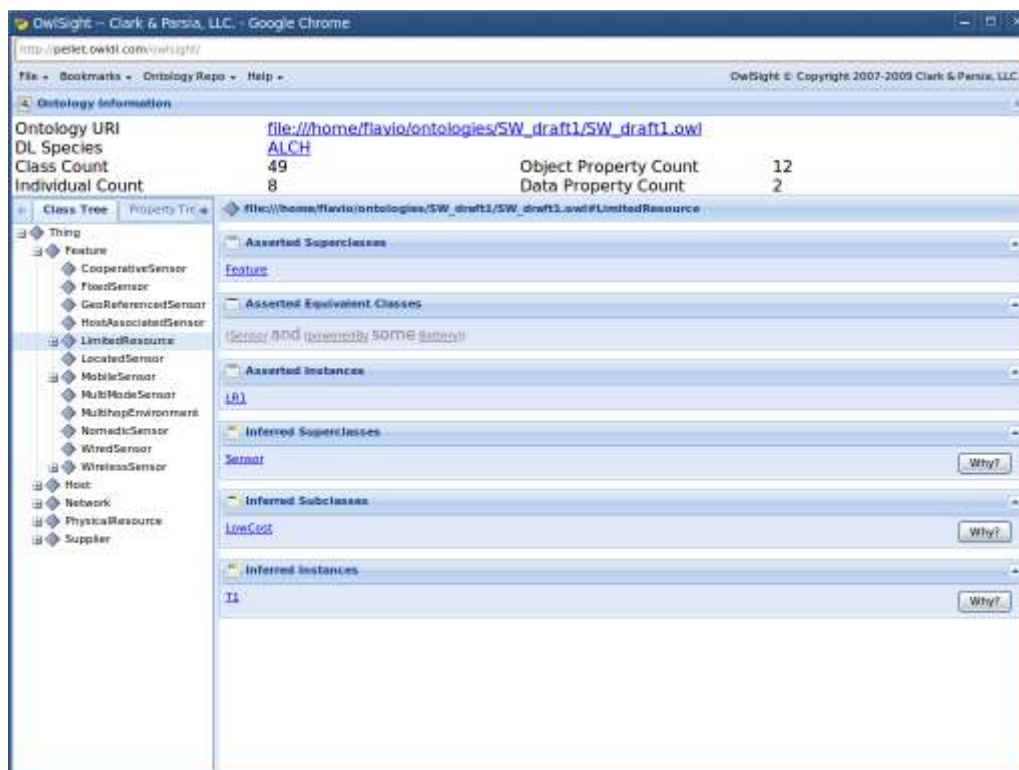


Figura 2.14. Validación de la Ontología utilizando OwlSight (Figura original de [Con10]).

Las características avanzadas (Figura 2.11 derecha) son un conjunto de propiedades contextuales de alto nivel representadas por conceptos inferidos. Cada concepto inferido está definido por una o más reglas semánticas propuestas en Tabla 2.1 según el estilo Protege. En Figura 2.13 se representan ejemplos de implementación en OWL. Las características avanzadas reflejan el mismo esquema lógico de los conceptos a los que hacen referencia: se pueden, por lo tanto, clasificar en función a la perspectiva a la que se refieren (red, dispositivo, etc.).

La ontología se ha desarrollado en OWL utilizando Protege 4.1. La misma se ha sido validada utilizando OwlSight [Sig1] (Figura 2.14).

2.6.2.2 Ambiente semántico orientado a eventos para procesado de datos

Muchos sistemas reales complejos suelen implementar las capas de inteligencia para elaboración y análisis de datos directamente sobre el sistema de información existente. Esto equivale a considerar que tanto el estado del sistema como las acciones correspondientes están determinados por la información disponible.

En el caso de sistemas basados en redes de sensores, la información disponible (datos) puede diferenciarse en datos generados en tiempo real proporcionados por dispositivos sensores mas cualquier tipo de dato disponible en el sistema de información.

Un posible modelo de referencia para la definición de una arquitectura para procesado de datos en tiempo real se representa en Figura 2.14.1. Como se puede apreciar, la infraestructura se compone de diferentes capas funcionales:

- La capa más baja de la arquitectura (*Data Manager*) asume el rol de coleccionar y, eventualmente, sincronizar los datos de entrada. Estos pueden proceder de sensores o, bien, de sistemas de información (internos o externos).
- La información es procesada por un componente inteligente que asume el rol funcionalmente fundamental de convertir la información en conocimiento de alto nivel. Este mecanismo necesita de una representación formal tanto de la información como del conocimiento.
- El conocimiento construido o deducido por la capa de inteligencia es disponible para los sistemas de control, entendidos como las componentes que implementan las funcionalidades más abstractas y avanzadas de los sistemas.

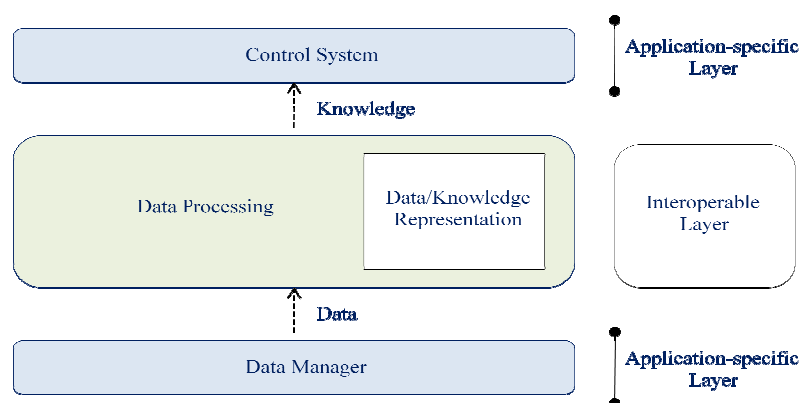


Figura 2.14.1. Capas funcionales que componen un sistema de procesamiento de datos en tiempo real. Figura original de [Con12].

Un ambiente semántico que pretenda soporte análisis complejo orientado a eventos debe poder expresar todas las relaciones lógicas fundamentales en el proceso de generación de datos, identificación de eventos y de eventuales acciones (o cambios de estado). Además, eventuales relaciones con conceptos externos incluidos en el sistema de información tienen que definirse de forma finita. Aunque la gran parte de las relaciones mencionadas puedan considerarse específicas de aplicaciones o dominios concretos, se puede identificar un conjunto de patrones relevante que puede generar un esquema lógico común. A la luz de estas consideraciones, no se pretende proporcionar una ontología concreta, si no una meta ontología que refleje un esquema semántico que puede particularizarse y/o extenderse en función de aplicaciones reales.

La meta-ontología propuesta define un conjunto de conceptos virtuales interrelacionados que definen el esquema lógico de base (Figura 2.15); esto se compone de cuatro conceptos fundamentales: las fuentes de datos o de observación, los datos, los eventos y, finalmente, las acciones. Evidentemente estos conceptos definen una relación lógica secuencial.

Se consideran como fuentes de datos o de observación, datos proporcionados por dispositivos sensores (o redes de sensores) internos al sistema, datos contenidos en el sistema de información o datos provenientes desde bases de datos externas. Los datos asociados a bases de datos, tanto internas como externas, pueden ser de tipo sensor o, bien, de cualquier otro tipo o naturaleza lógica.

Los datos se construyen lógicamente en función del origen; se han distinguido datos de tipo sensores, datos externos y observaciones distribuidas. Esta última categoría define datos complejos compuestos por más observaciones independientes de carácter heterogéneo. Este conjunto de información se define actualmente como conceptos inferidos sobre las fuentes de información.

El evento es un concepto lógicamente más amplio y complejo respecto al dato: los eventos se definen a partir de los datos disponibles sobre la base de razonamientos y relaciones, internas o externas, más o menos complejas. Se consideran conceptos específicos de aplicaciones concretas. El evento es el concepto central del esquema completo: definirlo (o particularizarlo) supone que la meta-ontología propuesta pueda interpretarse como una ontología concreta.

Las acciones se relacionan directamente a los diferentes tipos de eventos que el sistema es capaz de detectar; la implementación corriente distingue entre alarma local, alarma global, alarma personal, notificación y warning. Evidentemente otras clases de acciones y, más en general, de cambios de estado pueden definirse por parte de sistemas concretos.

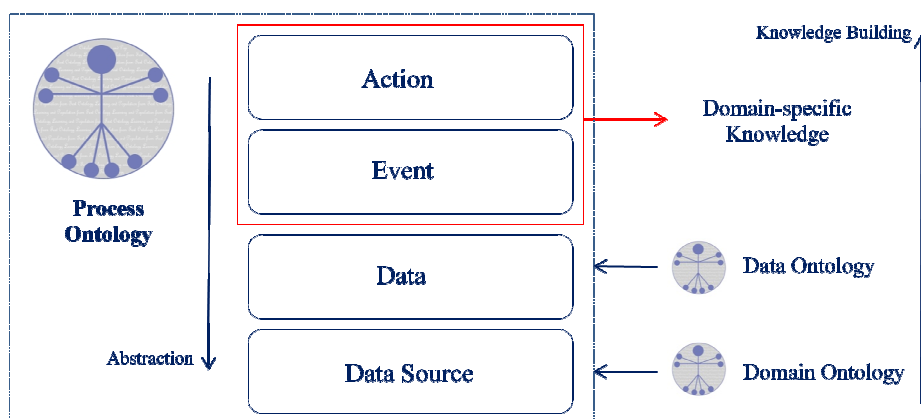


Figura 2.15. Estructura lógica de la Ontología de Proceso. Figura original de [Con12].

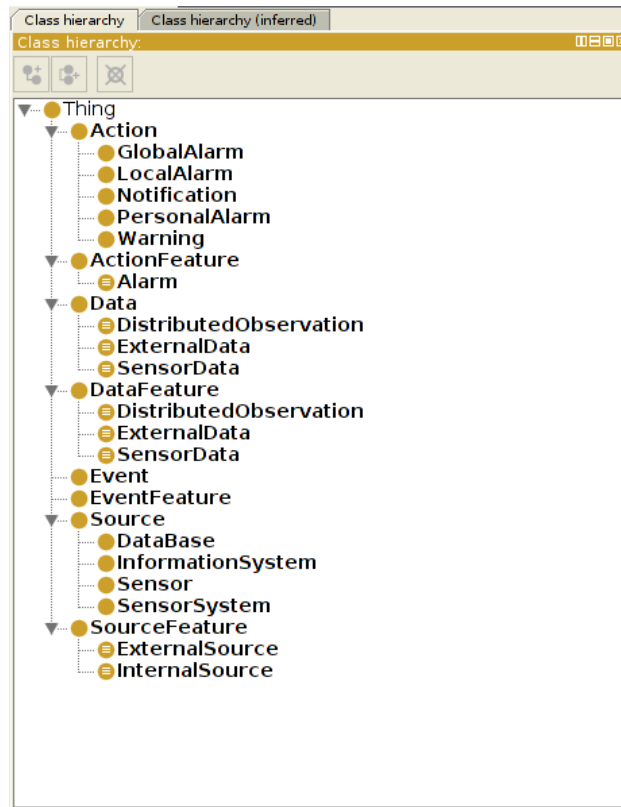


Figura 2.16. Visión en Protege de la jerarquía de conceptos de la Ontología de Proceso. Figura original de [Con12].

Como puede apreciarse en Figura 2.16 (implementación de la Ontología de Proceso en OWL en ambiente Protege), a cada clase de concepto principal se asocia una clase de características avanzadas que se interpretan con el mismo significado y nivel de abstracción vistos en la ontología de dominio. Evidentemente se trata de conceptos asociados a aplicaciones concretas.

La arquitectura para el procesamiento de información (Figura 2.17) se basa, evidentemente, en actores semánticos así como definidos en las secciones anteriores. Sin embargo dicha arquitectura se ha diseñado para poder soportar aplicaciones con diferentes requisitos en términos de escala y prestaciones.

Esta arquitectura, representada en Figura 2.17, esta implementada en tecnología JAVA. Como se puede apreciar, la infraestructura central se basa en razonadores.

Como en la mayoría de las infraestructuras, los datos procedentes de los sensores se reciben en la estación base; el *Data Manager* asume el rol de preprocesador y sincronizador (si fuera necesario) de la información recibida.

Entornos lógicos relativamente simples asumen datos sensores no relacionados (lógicamente independientes); al contrario, en otros entornos podrían proponerse relaciones complejas entre los mismos así como con datos provenientes del sistema de información. Evidentemente, en este estadio, la información manejada siempre se representa de acuerdo con un modelo semántico de referencia.

Los datos de entrada son procesados por la infraestructura de razonadores (actores semánticos) que se compone de un conjunto de Threads JAVA capaces de operar en paralelo en presencia de requisitos estrictos de tiempo real.

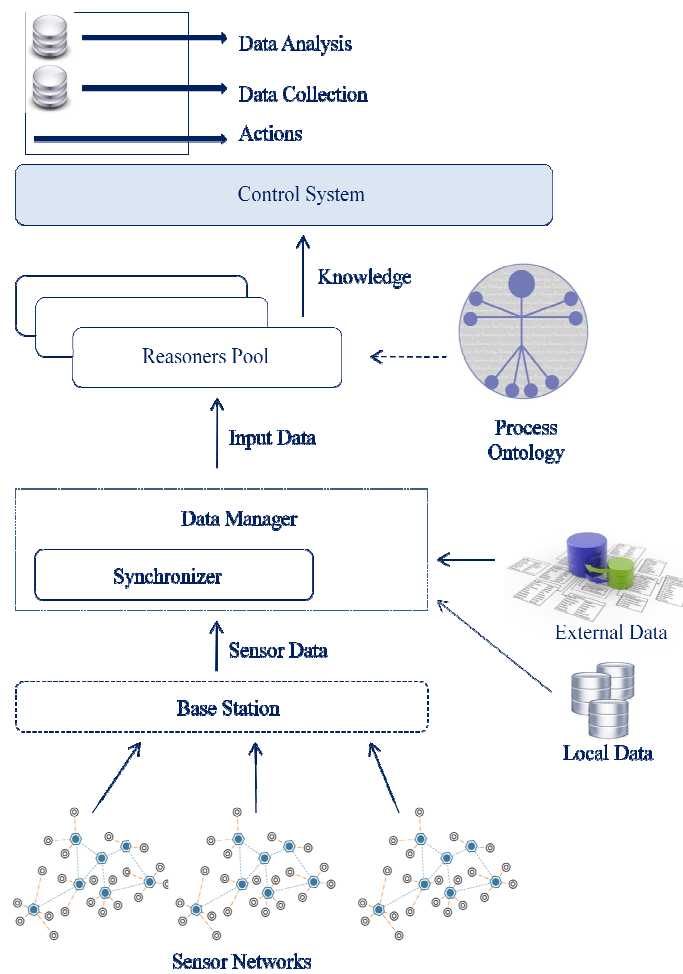


Figura 2.17. Arquitectura para el procesamiento de datos. Figura original de [Con12].

El conocimiento extrapolado por la capa de razonamiento es directamente accesible por el *Control System*, que se considera el elemento lógico visible por las aplicaciones. El conocimiento disponible se utiliza para implementar el motor de gestión orientado a eventos de las acciones en tiempo real. Además, tanto los datos básicos como el conocimiento se almacenan en el sistema de información (local o global) del sistema y pueden ser objeto de análisis avanzadas posteriores.

CAPÍTULO 3

3 Modelo, Simulación y Evaluación de Redes de Sensores Inalámbricas

El presente capítulo se describe todos los modelos teóricos para soportar la definición, especificación y análisis de los mecanismos de mayor interés de las redes inalámbricas de sensores en el ámbito del trabajo de tesis. Los modelos propuestos se utilizan para implementar ambientes de simulación avanzados que permiten el análisis complejo, a diferentes niveles de abstracción, de los aspectos de mayor interés. Al mismo tiempo, los modelos en cuestión se utilizan para la implementación de los mecanismos fundamentales del framework que, por lo tanto, se especifica e implementa sobre la misma base teórica de los ambientes de simulación.

3.1 Introduccion

En este capítulo de la tesis se trata el modelo, la simulación y la evaluación analítica de los recursos físicos de la arquitectura descrita en el capítulo anterior.

El modelo y, consecuentemente, la simulación y la evaluación de las redes inalámbricas de sensores se limitan, en el presente contexto, a aspectos de carácter general relacionados con la comunicación que no dependen de dominios de aplicación concretos.

Asimismo, aunque la mayoría de las arquitecturas asuman la utilización única de nodos estáticos (nomádicos en algunos casos), se consideran también entornos de redes de sensores móviles. Esto considerando que, durante los últimos años, el número creciente de aplicaciones de redes inalámbricas de sensores que asume nodos móviles (Mobile WSNs). Una red de sensores móviles presenta varios puntos de convergencia con los entornos tipo MANET [Kem1][Dha1] aunque las peculiaridades tanto respecto a la tecnología como respecto a las arquitecturas y al dominio de aplicación relacionado aconsejan un análisis específico y diferenciado.

Las prestaciones de una red inalámbricas de sensores dependen de diferentes factores, así como la métrica de evaluación suele ser contextual. Independientemente de la aplicación considerada, se detecta siempre una clara relación entre prestaciones globales de las redes y mecanismos de base (por ejemplo control de topología, enrutamiento) de las mismas. Tanto los mecanismos de base como mecanismos avanzados para garantizar parámetros de calidad de servicios dependen del grado de conectividad de la red. Existen multitud de modelos teóricos que analizan el grado de conectividad de la red en función de la densidad del despliegue de los sensores, y del radio de cobertura de los mismos.

El entorno de red que asumimos en los diferentes escenarios simulados aunque destinado a entornos lo más genérico posible cumple con las siguientes características:

- *Entorno de comunicación multi-salto*, como en la mayoría de los entornos reales, con independencia del protocolo MAC y de nivel de red empleado.
- *Despliegue aleatorio de los nodos*, como sucede en un número importante y siempre creciente de entornos reales. Teniendo en cuenta que la principal diferencia entre los despliegues aleatorios y los despliegues controlados es el grado de conectividad para una misma densidad de despliegue, que es mayor en los despliegues controlados.
- *Configuración lógico/física de la red en clusters*: considerando el típico contexto con recursos limitados, descrito en las secciones anteriores, que caracteriza las redes inalámbricas de sensores, una configuración *overlay* avanzada de la red puede mejorar tanto las prestaciones como la robustez y la calidad de servicio global de las redes de sensores al crecer en escala. La técnica más utilizada para garantizar la escalabilidad es el *clustering*. Esta clase de solución implica la existencia de múltiples estaciones bases, dotadas de mayores capacidad tecnológicas que los nodos sensores, que trabajan simultáneamente y la presencia de los necesarios mecanismos de configuración de la red y gestión de la información. Mediante *clustering* se puede considerar una red de tamaño n como composición de m sub-redes (*clusters*) de tamaño medio n/m . Evidentemente, en presencia de entornos con despliegue aleatorio de los nodos tanto los mecanismos de configuración como de gestión de la información se consideran dinámicos.
- *Modelo de clustering dinámico orientado a las prestaciones*: la configuración en clusters puede efectuarse de acuerdo con diferentes criterios o estrategias. Las soluciones más comunes y relevantes asumen el balanceo de los clusters en tamaño o, bien, minimización de las rutas de comunicación. A menos de requisitos particulares, la segunda solución garantiza mejores prestaciones y mayor robustez siendo este modelo de cluster adoptado en el contexto del trabajo de tesis.
- *Modelo de conexión simple*: un nodo se considera conectado a la red si (y solo si) existe, como mínimo, una ruta de comunicación que lo conecta a una de las estaciones base.
- *Métrica de evaluación generalizada*: la métrica de evaluación considerada está directamente relacionada con el porcentaje de nodos conectados (se acuerdo con la definición anterior) respecto a la densidad de nodos. Esta simple métrica, por una parte, garantiza la posibilidad de evaluar una red de acuerdo con un enfoque generalizado y, por la otra, simplifica análisis posteriores con grado superior de complejidad o abstracción.

La evaluación del impacto de la movilidad de los nodos en las prestaciones de las redes es un tema relevante en muchos dominios de aplicación. También este problema puede enfocarse de acuerdo con múltiples perspectivas en el contexto de diferentes arquitecturas, topologías, tecnologías y entornos ambientales.

La gran mayoría de modelos y técnicas de análisis para entornos de redes móviles hacen referencia, directa o indirecta, a MANET. Normalmente, se intenta enfocar el problema en el contexto de aplicaciones (aplicaciones *group-based*, por ejemplo) o entornos específicos (redes vehiculares, por ejemplo).

Modelos de movilidad en redes inalámbricas de sensores suelen relacionarse directamente con entornos, aplicaciones o dominios concretos: la solución propuesta por [Han1] asume movilidad controlada; el modelo en [Alm2] aplicaciones para adquisición de datos; en [Och1] se consideran aplicaciones *group-based*; en [Gia1] solo unos actores (estaciones bases) son nodos móviles; solo unos cuantos de estos modelos pueden extenderse a entornos de gran escala [Sun3].

El enfoque propuesto respecto a la movilidad es coherente con el enfoque metodológico del capítulo que asume modelos de carácter más genérico. Para que esta metodología sea efectivamente aplicable también en entornos móviles, los modelos correspondientes, tanto relativamente al comportamiento de los nodos como a las métricas de evaluación, tiene que ser abiertos y soportar extensiones verticales orientadas a análisis más específicos.

Finalmente, el capítulo propone un análisis de consumo energético coherente con la metodología global de modelo y evaluación.

3.2 Modelo de Redes Inalámbricas de Sensores

El modelo propuesto a continuación representa los aspectos genéricos e independientes de aplicaciones o dominios de aplicación específicos. Se compone principalmente de un modelo de organización lógica, de ruta de comunicación, de cluster, de comunicación no ideal, de movilidad y de consumo energético.

Un enfoque más global supondría una notable complejidad y podría resultar poco efectivo en términos prácticos.

Dicho modelo se considera extensible tanto en horizontal (otros elementos genéricos como el modelo de consumo explícito que relacione el modelo teórico con tecnologías concretas) como en vertical (aspectos contextuales dependientes de aplicaciones o dominios concretos).

La presente metodología permite, por una parte, concentrarse sobre los aspectos de interés real y, por la otra, no vincular, ni directamente ni indirectamente, los resultados analíticos con dominios concretos. Eventuales extensiones permiten un análisis complejo organizado de acuerdo con niveles de abstracción crecientes.

3.2.1 Configuración lógica

El modelo de configuración se considera el punto de partida para la definición del modelo propuesto y, al mismo tiempo, el elemento central del mismo.

El modelo de configuración lógica se compone de un conjunto de parámetros lógicos que pueden definirse sobre la base de los parámetros físicos o topológicos. Independientemente de la topología considerada, la configuración lógica siempre se define utilizando una

perspectiva *cluster-centric* en cuanto asocia los nodos sensores a su estación base de referencia. Desde un punto de vista teórico, considerar, analizar y configurar un cluster es equivalente a considerar una red de sensores con una única estación base.

Los parámetros lógicos de interés pueden variar notablemente en función de los objetivos a alcanzar, principalmente dependerán de las funcionalidades proporcionadas, de la calidad de servicio que queramos obtener y, finalmente, de la aplicación a la que está destinada la red de sensores.

En el presente contexto, se limita el número de parámetros de acuerdo con las siguientes asunciones:

- La configuración lógica tiene que soportar únicamente la definición, especificación y análisis de mecanismos funcionales de base (configuración y comunicación por ejemplo).
- Todos aquellos parámetros de soporte a mecanismos, genéricos o propios de dominios específicos, que requieren un conocimiento del estado del nodo sensor (estado de carga por ejemplo), de la topología de la red o de su actividad, se consideran objeto de extensión del actual modelo de configuración lógica.

Independientemente de su rol (nodo sensor o estación base), un actor que es parte de una red de sensores se identifica lógicamente en la misma de acuerdo con los parámetros especificados en (eq.3.1). Como se puede apreciar, se proponen dos componentes identificativos (identificador, *ID* y grupo, *Group*), dos parámetros lógicos (*Level* y *Rad*) y, finalmente una referencia a la posición física del sensor (opcional), absoluta o relativa a la estación base de referencia. Los parámetros lógicos asumen valor diferente de cero solo si asociados a nodos sensores. Siempre son nulos para las estaciones bases.

$$\begin{aligned} Node_i &= \{ID_i, Group, Level_i, Rad_i\}U^* \{x_i, y_i\} \\ BS_j &= \{ID_{Base_j}, Group, 0, 0\}U^* \{X_0, Y_0\} \end{aligned} \quad (eq.3.1)$$

Los parámetros de identificación suelen definirse estáticamente en redes reales aunque existan varios algoritmos para diseminación dinámica de los mismos como mencionado en el primer capítulo de la tesis. El identificador de un nodo identifica de forma unívoca el mismo en la red (local o global). El grupo asume la asociación entre sensores de acuerdo con algún aspecto físico, lógico o funcional de la red.

En esta tesis, los identificadores se definen siempre a priori y de forma estática. Por el contrario, el concepto de grupo se asocia a la asociación de un sensor a un cierto cluster (como se ha especificado en las siguientes secciones). Dicha asociación puede efectuarse de forma estática o dinámica. También el posicionamiento del sensor puede asumirse estático o dinámico (geo-referenciado a través de dispositivos GPS por ejemplo). En caso de posicionamiento dinámico, la posición de cada sensor puede representarse con sus coordenadas (posicionamiento absoluto) o respecto a la posición de la estación base de referencia (posicionamiento relativo).

Los parámetros lógicos de nivel (*Level*) y de configuración radial (*Rad*) constituyen la configuración lógica dinámica del cluster representada en la Figura 3.1.

El parámetro de nivel está definido por (eq.3.2). Como se puede apreciar, asume el conocimiento, por parte del nodo en cuestión del valor del parámetro de nivel asociado a los nodos vecinos. Dos nodos se consideran vecinos si pueden comunicar directamente (un solo

salto) entre ellos. El valor asociado a una estación base es siempre nulo; el valor asociado a un nodo sensor se define recursivamente como el valor mínimo asociado a nodos vecinos incrementado en uno. Evidentemente se trata de un valor siempre positivo.

$$Level_i = \min(level_i) + 1 \quad (eq.3.2)$$

$$i \in Neighbors_i$$

$$Level_{Base} = 0 \quad (eq.3.2.1)$$

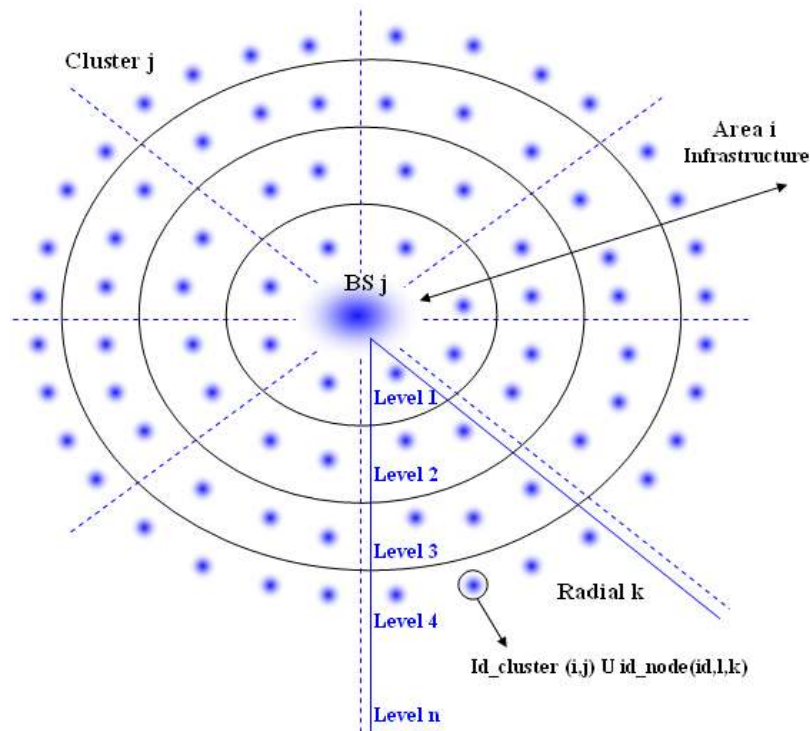


Figura 3.1. Configuración lógica (teórica) de un cluster (Figura original de [Con4]).

El parámetro de nivel equivale al número de saltos de la ruta mínima hacia la base. Evidentemente, este parámetro tan solo expresa la existencia de dicha ruta (óptima bajo muchos puntos de vista); sin embargo no considera ningún factor real que suele definir la configuración de las rutas como la estabilidad de los enlaces y otros factores no ideales.

Con la definición de este parámetro de nivel se pueden definir, de forma finita, diferentes estrategias para comunicación convergente y divergente (modo broadcast) que son los modos de comunicación más difusos. Menos comunes se consideran la comunicación divergente hacia un único nodo, la comunicación divergente multicast y la comunicación nodo a nodo (o punto a punto).

Protocolos de enrutamiento concretos se definen concretizando la estrategia de enrutamiento a frente de múltiples rutas disponibles.

Sin embargo, considerando tan solo el parámetro de nivel, se pueden encontrar varias dificultades a soportar la definición de estrategias de comunicaciones para modalidad divergente (base a nodo o multicast) y nodo a nodo.

Para garantizar un soporte más completo y efectivo se define un segundo parámetro lógico (parámetro radial), definido por las siguientes propiedades:

- Si un cluster C está particionado en grupos de nodos $Rad_1, Rad_2, Rad_3 \dots Rad_n$, entonces $(Rad_1 \cup Rad_2 \cup Rad_3 \dots \cup Rad_n) = C$.
- Si un nodo k se relaciona o asocia a Rad_i , entonces el nodo k no se relaciona o asocia a Rad_j .
- Existe siempre (como mínimo) una ruta que conecta un nodo k , relacionado con Rad_i , a la estación base de referencia que se compone únicamente por nodos compuestos por nodos relacionados por Rad_i .
- Todas las estaciones base asumen valor nulo del parámetro radial.

En Figura 3.2 se representa una configuración ideal de un cluster (izquierda) según el parámetro radial. A la derecha de la misma figura se representa una posible representación en entornos reales. Esta diferencia se debe a las modalidades de funcionamiento de protocolos reales. Detalles al respecto se proporcionan en el Capítulo 4.

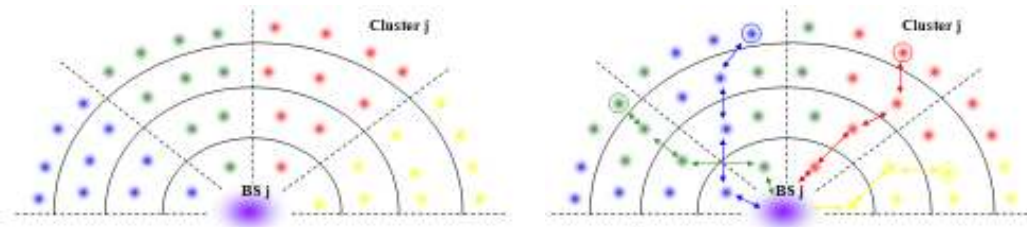


Figura 3.2. Posible configuración en entornos reales (derecha) a frente de la configuración ideal representada a la izquierda (Figura original de [Con4]).

3.2.2 Rutas

El objetivo fundamental del modelo de ruta es definir, diferenciar y clasificar los diferentes tipos de rutas que se pueden configurar en función de la configuración lógica propuesta en la sección anterior.

Una primera clasificación implícita se realiza en función de los modos de configuración; más concretamente se distingue entre comunicación convergente (desde nodos sensores hasta la base), divergente (desde la base hacia los nodos sensores) y nodo a nodo. La mayoría de las aplicaciones basadas en redes de sensores asumen principalmente comunicación convergente.

La comunicación divergente puede clasificarse en comunicación divergente broadcast (muy utilizada especialmente en las tareas de configuración de la red), divergente multicast (desde la base hacia un grupo de nodos) y divergente hacia un único nodo (desde la base hacia un nodo). Estas dos últimas categorías son bastante menos utilizadas en la práctica.

La comunicación nodo a nodo asume comunicación transparente a través de la red entre nodos sensores. Es una funcionalidad no demasiado utilizada por arquitecturas de redes inalámbricas de sensores salvo en topologías que asumen algún tipo de control directo o actor adicional a la estación base. Otro caso de posible interés puede ser la comunicación entre estaciones base a través del soporte físico proporcionado por los nodos sensores;

tampoco es un caso demasiado común porque las estaciones base suelen interconectarse a través de infraestructuras de red con capacidad superior (redes IP normalmente).

Como se puede apreciar en las secciones siguientes, ninguno de los modelos propuestos permite diferenciar entre rutas de la misma especie. Esta diferenciación se efectúa introduciendo un factor de peso a cada enlace. Dicha extensión se analiza en el siguiente capítulo en el contexto de la implementación del framework donde factores de peso pueden identificarse con la fiabilidad del enlace, con parámetros *energy-aware* o *position-aware*.

3.2.2.1 Rutas convergentes

Se definen tres diferentes tipos rutas convergentes que se consideran de interés en el presente contexto. Se trata de las rutas convergentes directas, las rutas paralelas y las rutas indirectas

Más en detalle:

- Las *rutas convergentes directas* se definen de acuerdo con (eq.3.3). Un ejemplo se representa en Figura 3.3. Como se puede apreciar, una ruta convergente directa se compone únicamente de enlaces entre nodos con valor del parámetro de nivel decreciente. Estas rutas son rutas óptimas respecto al número de saltos.

$$\left\{ \begin{array}{l} R_{a \rightarrow Bs}^{(-)} = \bigcup_{i=1}^n r_{j \rightarrow j+1}^i \\ \forall i \in R: Level(j+1) < Level(j) \end{array} \right. \quad (\text{eq.3.3})$$

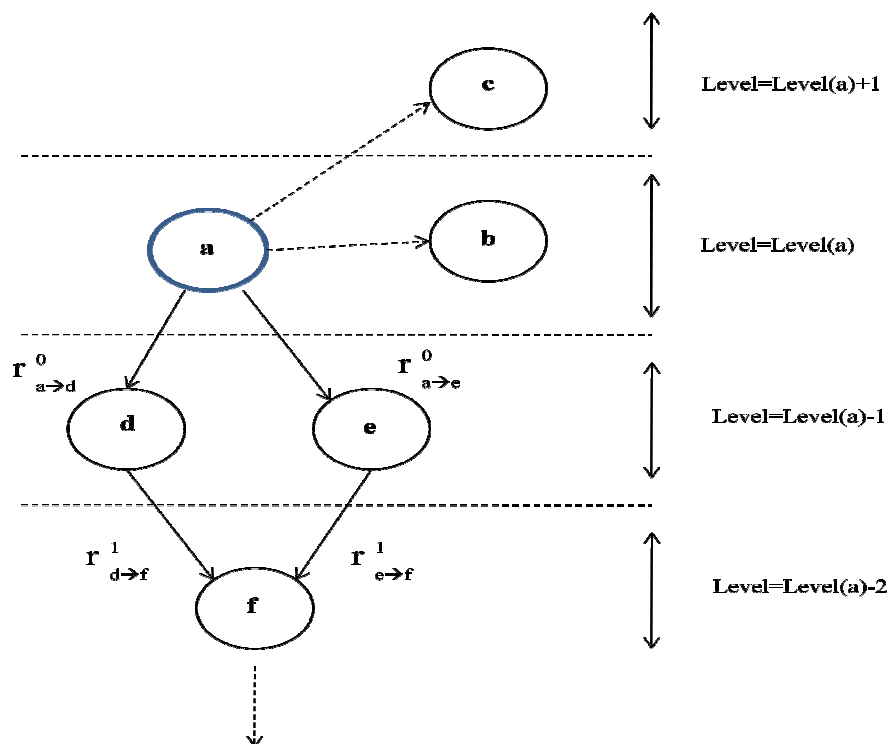


Figura 3.3. Ejemplo de ruta directa.

- Las *rutas convergentes paralelas* están definidas por (eq.3.4). Un ejemplo se representa en Figura 3.4 a la izquierda. La diferencia respecto a las rutas directas es que, al contrario de estas últimas, la rutas paralelas puede incluir enlaces entre nodos con el mismo valor del parámetro de nivel; para evitar ciclos, cada nodo puede entrar en la ruta solo una vez.

$$\left\{ \begin{array}{l} R_{a \rightarrow Bs}^{(=)} = \bigcup_{i=1}^n r_{j \rightarrow j+1}^i \\ \forall i \in R \rightarrow Level(j+1) \leq Level(j) \\ \forall j \in r^i \rightarrow \nexists r_{k \rightarrow j}^m, m > i \end{array} \right. \quad (\text{eq.3.4})$$

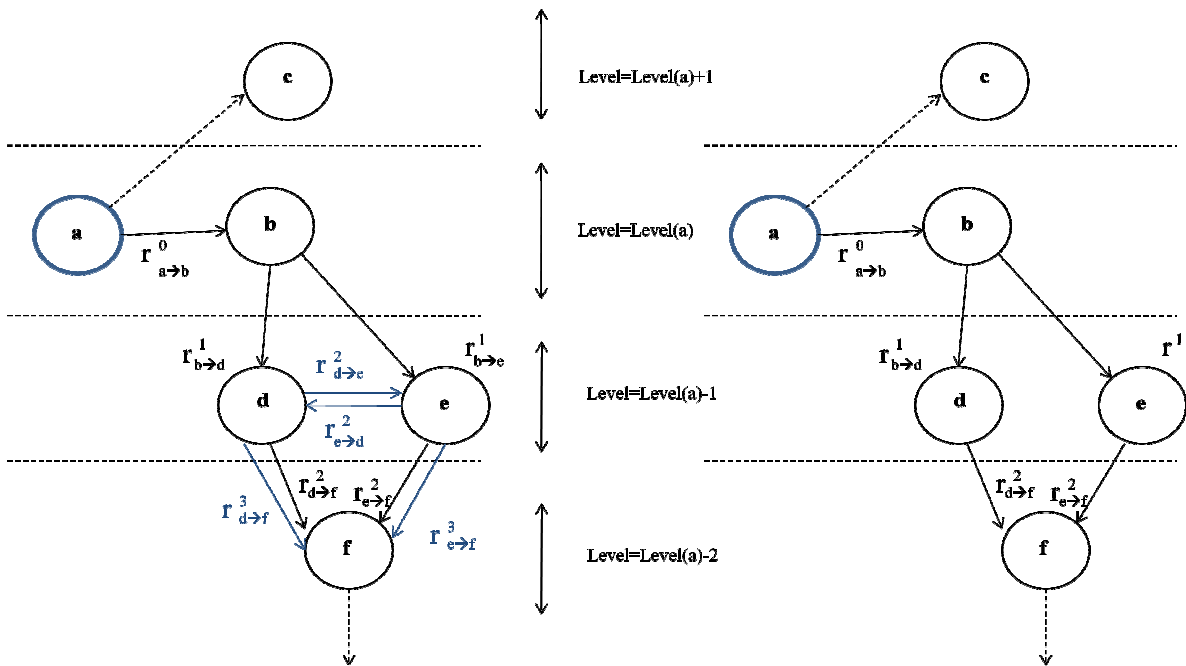


Figura 3.4. Ejemplos de rutas paralelas.

Estas rutas son implícitamente más complejas y, en el caso general, pueden presentar una cierta ineficiencia o, de todos modos, un tamaño medio en saltos superior a las rutas directas.

A efectos prácticos, es importante limitar las rutas paralelas y definir un modelo de ruta intermedio entre ruta directa y ruta paralela; este modelo, llamado *ruta paralela limitada*, está definido por (eq.3.4.1). Un ejemplo se representa en Figura 3.4 a la derecha. Esta ruta admite el enlace entre nodos con el mismo valor del parámetro de nivel tan sólo en el primer salto. Las rutas paralelas limitadas se pueden utilizar para proporcionar a mecanismos concretos rutas alternativas a la directa sin una pérdida excesiva de prestaciones.

$$\left\{ \begin{array}{l} R_{a \rightarrow Bs}^{(=)} = \bigcup_{i=1}^n r_{j \rightarrow j+1}^i \\ r^0 \rightarrow Level(j+1) = Level(j) \\ r^k: k > 0 \rightarrow Level(j+1) < Level(j) \end{array} \right. \quad (\text{eq.3.4.1})$$

- Una *ruta indirecta* (eq.3.5) simplemente puede incluir cualquier enlace entre nodos con la única limitación que un nodo no puede entrar en la ruta más que una vez (ausencia de ciclos). Evidentemente, vista su generalidad, este tipo de ruta es demasiado ineficiente para poderse utilizar en cualquier entorno real.

$$\left\{ \begin{array}{l} R_{a \rightarrow Bs}^{(+)} = \bigcup_{i=1}^n r_{j \rightarrow j+1}^i \\ \forall j \in r^i \rightarrow \nexists r_{k \rightarrow j}^m, m > i \end{array} \right. \quad (\text{eq.3.5})$$

Sin embargo, utilizando la misma aproximación definida para las rutas paralelas, se puede definir la ruta convergente indirecta limitada (eq.3.5.1). Como se puede apreciar en la Figura 3.5, se admite un enlace desde un nodo con valor del parámetro de nivel más bajo a uno más alto tan solo en el primero salto. Este tipo de ruta puede aplicarse en entornos reales con prestaciones parecidas a rutas paralelas limitadas.

$$\left\{ \begin{array}{l} R_{a \rightarrow Bs}^{(+)} = \bigcup_{i=1}^n r_{j \rightarrow j+1}^i \\ r^0 \rightarrow \text{Level}(j+1) > \text{Level}(j) \\ r^1 \rightarrow \text{Level}(j+1) \leq \text{Level}(j) \\ r^k: k > 1 \rightarrow \text{Level}(j+1) < \text{Level}(j) \\ \forall j \in r^i \rightarrow \nexists r_{k \rightarrow j}^m, m > i \end{array} \right. \quad (\text{eq.3.5.1})$$

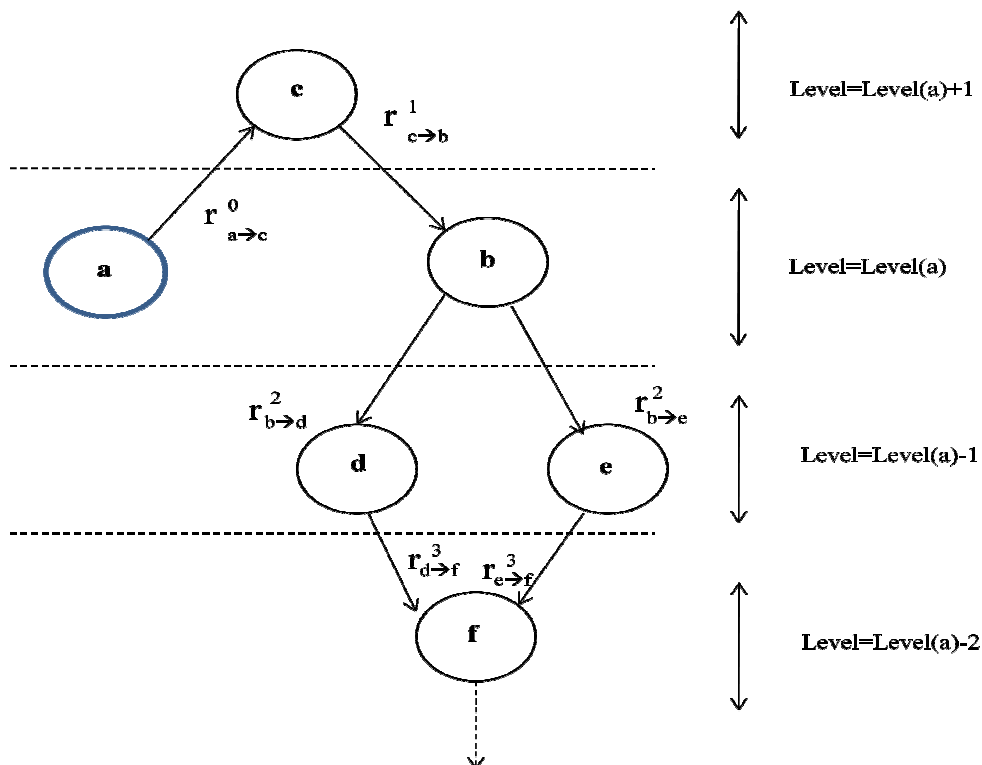


Figura 3.5. Ejemplo de ruta indirecta limitada.

3.2.2.2 Rutas divergentes y nodo a nodo

Las rutas divergentes y nodo a nodo se tratan con un enfoque menos riguroso intentando privilegiar los aspectos de efectivo interés en el contexto del presente trabajo.

Las rutas divergentes están implícitamente definidas por el parámetro radial tal como descrito en las secciones anteriores. Realmente sería posible una clasificación teórica bajo diferentes perspectivas (prestaciones, escalabilidad) de este modelo. Sin embargo, vista la menor utilización en casos prácticos, se prefiere limitar el análisis a implementaciones concretas, como la propuesta en el siguiente capítulo. Como deja clara la propia definición del parámetro radial, las prestaciones reales de los mecanismos de comunicación divergente dependen más de factores topológicos que de la implementación de protocolos concretos.

También un análisis extendido de la comunicación nodo a nodo no se considera necesario. En el framework de referencia (que se describirá en el siguiente capítulo) el mecanismo de comunicación nodo a nodo se implementa aprovechando los parámetros de configuración lógica. Básicamente, se implementa un mecanismo inteligente que es capaz de detectar la proximidad del nodo con el cual se quiere comunicar (sobre la base de los parámetros lógicos). En este caso, la ruta de comunicación (Figura 3.6, izquierda) es prácticamente óptima. En caso contrario, se utiliza la estrategia generalizada (Figura 3.6, centro) que asume el envío del mensaje a la estación base que se encargará de enviarlo al nodo de interés a través del mecanismo de comunicación divergente (base a nodo).

Como se puede apreciar en la Figura 3.6 (derecha), la estrategia de comunicación nodo a nodo generalizado se caracteriza por buenas prestaciones en el caso medio (buena aproximación con la línea recta) y, además, propone la interesante ventaja de notificar implícitamente la estación base.

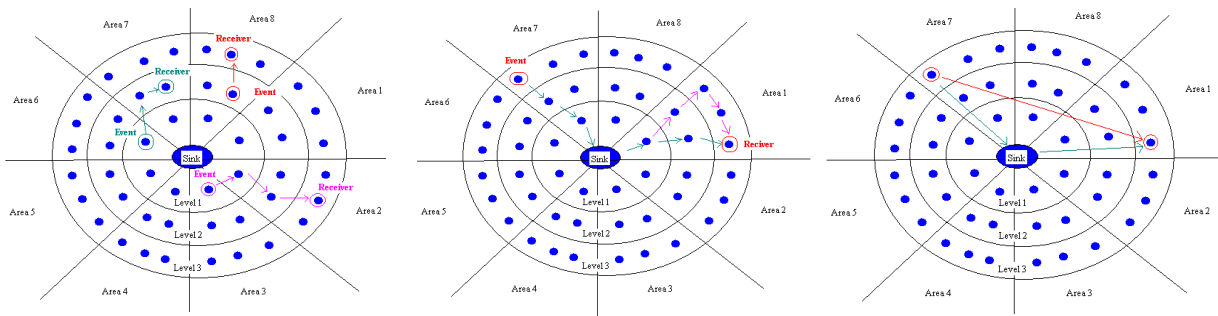


Figura 3.6. Estrategia de enrutamiento nodo a nodo.

3.2.3 Cluster

Como se ha propuesto en el capítulo anterior, el caso más genérico de funcionamiento para las redes inalámbricas de sensores asume la presencia simultánea de múltiples estaciones base. Para que las redes puedan trabajar de forma eficiente y escalable, arquitecturas reales suelen configurar la red para trabajar en *clusters*. En contextos prácticos, por obvios motivos funcionales (y tecnológicos en cierta medida), las estaciones base suelen ser las cabezas de los *clusters*.

La partición de una red en *clusters* permite a una red de tamaño n ser particionada en m redes de tamaño n/m . Considerando el fenómeno del *energy-hole* y los aspectos comerciales directamente o indirectamente relacionados con fiabilidad y robustez en un contexto de recurso limitado, la visión de una red en *cluster* se considera un elemento fundamental para

garantizar la escalabilidad de la arquitectura. Evidentemente, la configuración en *clusters* implica la presencia simultánea de múltiples estaciones base, la necesidad de gestionar la información tanto a nivel local (*cluster*) como global y, en caso de configuración dinámica, de mecanismos de configuración avanzada de la red.

Para garantizar una cierta consistencia entre modelo teórico de referencia y arquitecturas reales, se define, a continuación, una definición analítica de *cluster*. Dicho modelo se utiliza en el framework de referencia como soporte de los mecanismos de configuración avanzada (como definidos en el capítulo anterior).

Un nodo puede asociarse a un solo *cluster* de acuerdo con (eq.3.6), únicamente sobre la base del parámetro de nivel. Como se puede apreciar, de acuerdo con una perspectiva concurrente, un nodo se asocia al cluster que propone la configuración con el mejor valor del parámetro de nivel.

La definición de cluster propuesta, basándose únicamente en los valores del parámetro de nivel, tiende a privilegiar la configuración de cluster con rutas mínimas. Esta definición garantiza buenas prestaciones en términos de comunicación pero puede causar desequilibrios (incluso importantes) en el tamaño de los clusters.

Una alternativa concreta a este modelo es el enfoque contrario que asume configuración en cluster con balanceo del número de nodos. Este modelo, aunque podría resultar efectivo en aplicaciones puntuales, no se ha considerado en el presente trabajo, más que por las prestaciones en sí mismas, por la necesidad de mecanismos de configuración centralizados que no se han considerado en la definición de la arquitectura propuesta en esta tesis..

$$\begin{aligned}
 &Node_i \in Cluster_j \Rightarrow \exists Node_k : k \in Neighbors_i \wedge \\
 &\wedge k \in Cluster_j \wedge \\
 &\wedge k = \min(Level_w), w \in Neighbors_i
 \end{aligned}
 \tag{eq.3.6}$$

De acuerdo con la definición de la (eq.3.6), ciertos nodos con condiciones lógicas límite (Figura 3.7) pueden encontrarse con múltiples posibilidades de configuración en cuanto pueden asociarse con más que un cluster con la misma configuración lógica local.

Esta ambigüedad suele solucionarse a nivel de aplicaciones concretas: puede elegirse a cluster de forma aleatoria, se puede elegir el cluster de acuerdo con un orden preferencial (importante en presencia de arquitecturas jerárquicas en las cuales se intentan privilegiar la asociación con enablers) o, incluso, asociar el nodo con ambos clusters.

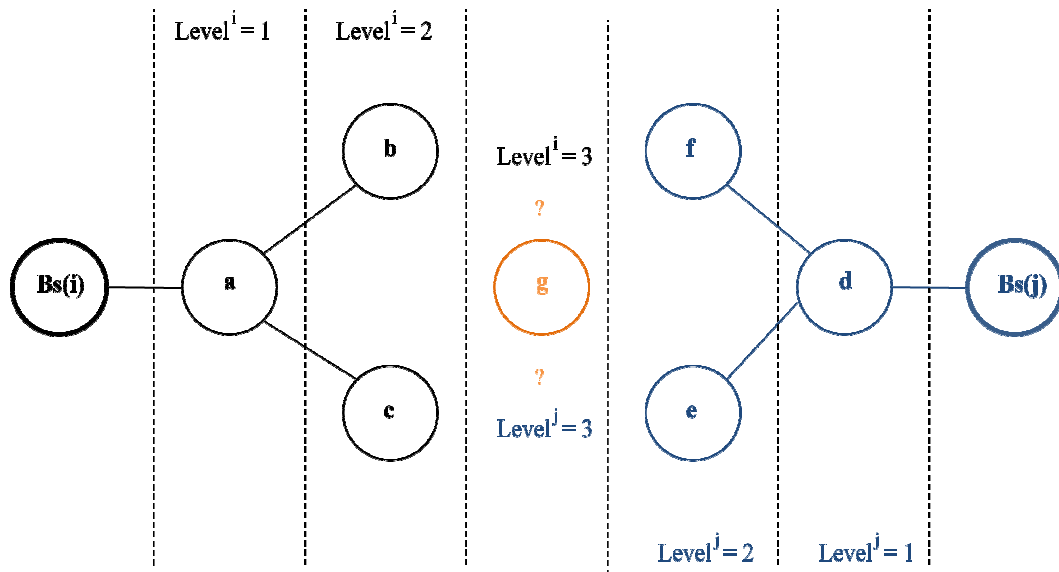


Figura 3.7. Configuración a paridad de condiciones lógicas.

3.2.4 Comunicación no ideal

El modelo de configuración no ideal es necesaria para extender las capacidades de modelos y análisis a situaciones (o condiciones ambientales) diferentes o variables. Un modelo detallado esta fuera de los objetivos del documento.

Sin embargo se proporciona un modelo simple de utilizar que pueda extender las capacidades de análisis frente a condiciones ambientales no ideales y, eventualmente, variables.

Se proporcionan dos modelos distintos, definidos con un alto nivel de abstracción, respectivamente en función de la distancia (eq.3.7) y del tiempo (eq.3.8). El modelo asume que un mensaje intercambiado entre nodos se reciba correctamente con probabilidad (eq.3.7) o (eq.3.8). En caso de utilizar simultáneamente los modelos, la probabilidad es la combinación de las dos probabilidades.

$$ul(z)=f(d) \tag{eq.3.7}$$

$$ul(z)=f(t) \tag{eq.3.8}$$

El primer modelo nace de la constatación experimental que, en condiciones relativamente estables, existe una cierta irregularidad por distancias limites (próximas al radio de comunicación) por las cuales se detecta un porcentaje significativo de pérdida de mensajes. Esta situación, que también puede modelar un cierto tipo de obstáculo o interferencia permanente, puede modelarse a través de una función que asuma un cierto decrecimiento, regular o menos, de la fiabilidad del enlace en función de la distancia (z). En el contexto de este trabajo se ha utilizado una rampa discreta decreciente (eq.3.8.1). Umbrales concretos (X, k1 y k2) son características de entornos ambientales concretos.

$$\begin{cases} ul(z) = k_1, & z \leq X \\ ul(z) = k_2, & z > X \\ k_1, k_2, X \text{ const}, k_2 < k_1 \end{cases} \quad (\text{eq.3.8.1})$$

El modelo de comunicación no ideal en función del tiempo pretende modelar la variación de las condiciones de estabilidad de los enlaces en función del tiempo. Obstáculos e interferencias temporales se consideran los casos más comunes. Particularizaciones típicas de (eq.3.8) son funciones periódicas (sinusoidales como en el caso del ambiente de simulación desarrollado en presente trabajo y descrito en el Anexo C).

3.2.5 Movilidad

En la mayoría de las redes de sensores la movilidad de los nodos se limita a movimientos pasivos determinados por agentes externos (aire, agua, etc.) o a cambios puntuales de topología (redes nomádicas).

Más recientemente, también gracias a la disponibilidad de dispositivos GPS de bajo coste montados sobre nodos sensores, varias aplicaciones, tanto en campo civil como militar, asumen nodos asociados con actores móviles (humanos o vehículos).

Un enfoque bastante común consiste en considerar una red de sensores móviles (Mobile WSN) como un caso particular de MANET [Kem1][Dha1].

Efectivamente Mobile WSN y MANET tienen varios puntos de convergencia. Ambos entornos se caracterizan por proponer diferentes temas de investigación abiertos especialmente en temas de arquitecturas y protocolos. Ambas soluciones suelen asumir entornos de comunicación multi-salto, necesidad de eficiencia energética y mecanismos de autoconfiguración y gestión.

Sin embargo, también presentan diferencias significativas. Las redes de sensores se comportan implícitamente como estructuras data-centric, contrariamente a MANET que suele proponer una orientación user-centric o id-centric. Lo que determina un enfoque diferente para la actividad de red que se estructura en topologías profundamente distintas: aproximaciones totalmente distribuidas para MANET, en algunos casos con organización lógica P2P, y arquitecturas implícitamente centralizadas para redes de sensores (vista la presencia de estaciones base, imprescindibles también en entornos móviles). MANET también suele asumir un equipamiento más avanzado y costoso para el soporte de aplicaciones de alto nivel. Por el contrario, las redes de sensores suelen enmarcarse en un contexto de bajo coste con hardware en gran parte a medida. La interacción con el ambiente es un punto fundamental en una red de sensores; limitada o inexistente in MANET. El tamaño medio (y la densidad) de una red de sensores es, en teoría, bastante superior al de las MANETs. Los mecanismos de comunicación son intrínsecamente más complejos en MANET ya que se suelen emplear tecnologías inalámbricas con capacidades superiores. Una red de sensores asume siempre comunicación de baja potencia (en entornos reales se trataría de básicamente de ZigBee o Bluetooth), MANET propone un abanico de posibilidades más amplio en este sentido, desde pocos mW a decenas de W.

Muchos modelos comúnmente utilizados para MANET [Bai1] pueden extenderse a redes de sensores móviles si se considera la intrínseca centralidad de la estación base. Sin embargo, en el presente trabajo de tesis se pretende un enfoque de los modelos y análisis de los resultados considerando niveles de abstracción creciente. Para que esta metodología sea

efectiva, los resultados tienen que poderse interpretar y analizar de forma independiente a cualquier mecanismo o aplicación específica y, al mismo tiempo, poderse relacionar con comportamientos específicos de los nodos y con condiciones ambientales concretas. Este enfoque permite la generalización de los resultados siempre y cuando exista un modelo de comportamiento para nodos móviles que, como todos los otros modelos propuestos, no dependan o se relacionen directamente con ningún elemento de dominio concreto.

La posición de un nodo en el plano, en el instante genérico de tiempo t , viene definido por las coordenadas (latitud y longitud) como en (eq.3.9).

$$l_t = (lx_t, ly_t) \quad (\text{eq.3.9})$$

En el instante genérico de tiempo t , para cada nodo se define la Matriz de Transición (T) como en (3.10); T se compone de cuatro factores, cada uno representado por una función de probabilidad.

El factor s define, al tiempo t , la probabilidad que el nodo interesado no cambie su posición al instante de tiempo $t+1$. Un concepto similar está expresado por $semS$, definido como la probabilidad que el nodo interesado cambie su posición en el instante $t+1$ aunque tal cambio de posición sea mínimo y, por lo tanto, no relevante en ciertos contextos. Este factor es importante en el contexto de nodos nomádicos.

Los factores x y y representan respectivamente la probabilidad que el nodo interesado cambie su posición en el instante de tiempo $t+1$ en latitud (x) o longitud (y).

$$T(t) = \begin{pmatrix} s(t, p_t(l_{t+1} = l_t)) & x(t, p_t(x_{t+1} \neq x_t)) \\ y(t, p_t(l_{t+1} \neq l_t)) & semS(t, p_t(l_{t+1} \cong l_t)) \end{pmatrix} \quad (\text{eq.3.10})$$

Como mencionado, todos los factores de T se interpretan como funciones de probabilidad; la aproximación más sencilla asume funciones independientes (sin relación) entre ellas.

Por cada nodo, la evaluación del determinante de T (llamado deT) al tiempo t determina la posición del nodo mismo al instante $t+1$ como en (eq.3.11); solamente valores negativos de deT determina cambios de posición significativas.

$$deT(t) = det(T(t)) = s(t, p) \cdot semS(t, p) - x(t, p) \cdot y(t, p) \rightarrow \begin{cases} det(T(t)) \geq 0 \rightarrow l_{t+1} = l_t \\ det(T(t)) < 0 \rightarrow l_{t+1} \neq l_t \end{cases} \quad (\text{eq.3.11})$$

Siempre en función del tiempo, se define mV como en (eq.3.12): se trata de un vector bidimensional; cada uno de sus componentes se define como en (eq.3.12.1); uno de los factores es función de x , el otro de y . Los módulos de alfa y beta pueden asumir diferentes significados; en la gran parte de los casos, representan las componentes de la velocidad media de movimiento de los nodos. Evidentemente, en el tiempo t , mV tiene significado solo si deT , evaluado en el instante de tiempo t , es negativo. En este último caso, mV expresa la entidad del cambio de posición del nodo ($l(t+1)-l(t)$).

$$mV(t) = (k_1\alpha(t) \quad k_2\beta(t)) \quad (\text{eq.3.12})$$

$$\begin{cases} \alpha(t) = |\alpha| \cdot x(t, p) \\ \beta(t) = |\beta| \cdot y(t, p) \end{cases} \quad (\text{eq.3.12.1})$$

En resumen, la posición en el tiempo $t+1$ de cada nodo móvil, dada su posición en el tiempo t , una concretización de T y mV , se determina como en (eq.3.13-eq.3.13.1).

$$l_{t+1} = \begin{cases} deT(t) \cdot mV(t) \\ deT(t) < 0 \end{cases} = \begin{cases} (k_1 \alpha(t) deT(t) \quad k_2 \beta(t) deT(t)) \\ deT(t) < 0 \end{cases} \quad (\text{eq.3.13})$$

$$l_{t+1} = (lx_{t+1} \quad ly_{t+1}) \rightarrow \begin{cases} lx_{t+1} = k_1 \alpha(t) deT(t) \\ ly_{t+1} = k_2 \beta(t) deT(t) \\ deT(t) < 0 \end{cases} \quad (\text{eq.3.13.1})$$

La utilización de T permite una representación compacta de las funciones de probabilidad necesarias. Al mismo tiempo, los operadores básicos consienten la cómoda representación de dependencias y relaciones. Además, el modelo es sencillo para ser empleado en su conjunto y el modelo bidimensional lo relaciona directamente con el plano cartesiano.

Comportamientos concretos pueden definirse particularizando cada factor de T y mV junto a la definición de una topología de salida. El punto de mayor interés del modelo es la posibilidad de definir comportamientos básicos o con complejidad interesante.

En el contexto del presente trabajo, los factores k_1 y k_2 , aunque puedan interpretarse como funciones en el caso más genérico, se definen como factores constantes. Estos factores son contextuales y pueden, por lo tanto, asumir significados diferentes en función de particularizaciones del modelo; por ejemplo pueden modelar, de forma muy sencilla, desplazamientos mono-dimensionales (k_1 o k_2 iguales a cero); en otros contextos, pueden modelar factores externos que pueden modelar factores externos que contribuyen al desplazamiento.

Comportamientos reales pueden necesitar el modelo de patrones, relaciones y dependencias de complejidad importante (por ejemplo comportamientos group-based). Además, los comportamientos podrían cambiar en función de eventos o factores externos (comportamientos dinámicos).

Para poder soportar estas relaciones complejas se requieren dos extensiones del modelo:

- *Definición de comportamientos en función de otros comportamientos* (eq.3.14); en este caso, el comportamiento no se define por un conjunto independiente de reglas si no por funciones que dependen, explícitamente o implícitamente, de uno o más comportamientos de otros nodos. Esta aproximación permite, por ejemplo, definir el líder de un grupo y definir los movimientos de los otros nodos en función del comportamiento del líder. Comportamientos concretos se definen particularizando (eq.3.14).

$$\text{behavior}(i) = f(\text{behavior}(j), \text{behavior}(j+1), \dots, \text{behavior}(j+n)) \quad (\text{eq.3.14})$$

- *Comportamientos con memoria* (eq.3.15); la definición, en el instante de tiempo t , es función del mismo comportamiento evaluado en instantes anteriores. Esta extensión proporciona capacidades añadidas respecto a la definición de reglas dinámicas, en algunos casos en función de eventos externos. Por ejemplo, un nodo móvil puede desplazarse en una cierta dirección y puede seguirla hasta encontrar un obstáculo; detectando el obstáculo, se determina otra dirección entre las disponibles que viene seguida hasta detectar otro obstáculo. Comportamientos concretos se pueden definir particularizando (eq.3.15) y la ventana de memoria relacionada (n).

$$\text{behavior}(i,t)=f(\text{behavior}(i,t-1)\dots\text{behavior}(i,t-n)) \quad (\text{eq.3.15})$$

Evidentemente, ambas extensiones pueden considerarse de forma simultánea, lo que daría una mayor amplitud al estudio del entorno de movilidad. Un análisis más profundizado de tales extensiones se considera fuera de los objetivos principales de este trabajo.

3.2.6 Modelo de consumo energético

Exactamente como en el caso de las comunicaciones, el modelo de consumo energético puede diseñarse considerando múltiples perspectivas y niveles de abstracción. El modelo utilizado se enmarca en el mismo contexto metodológico del modelo general en cuanto se considera independiente respecto a cualquier tecnología o mecanismo específico.

El parámetro de nivel (eq. 3.2) se propone mapear la topología física en un entorno lógico en el cual el parámetro representa, con buena aproximación, la distancia respecto a la base de referencia. Este mismo parámetro permite, por lo tanto, reagrupar los nodos de acuerdo con una configuración lógica coherente con la topología de referencia. Fundamentalmente, desde un punto de vista de los consumos energéticos, el parámetro de nivel asocia los nodos de la red a diferentes clases de tráfico.

Modelos más complejos, para soporte de análisis avanzadas de consumo energético y *energy-hole*, suelen necesitar modelos de tráfico específicos que no son objeto de este trabajo de tesis. Como especificado en el capítulo de estado del arte, existe un número importante de estudios verticales al respecto.

3.3 Evaluación de Redes Inalámbricas de Sensores

La fase de evaluación se ha realizado a partir de la metodología adoptada en la definición de los modelos básicos. En esta sección se describen y analizan los principales resultados analíticos obtenidos.

Antes se consideran entornos estáticos con despliegue aleatorio. Las estadísticas de interés son la conectividad y la configuración en *clusters*. Ambas se consideran en función del número de estaciones base (o *clusters*), de la densidad de nodos sensores y del rango de comunicación medio entre los mismos. Posteriormente se analizan entornos con nodos móviles y se estima el impacto de la movilidad sobre las prestaciones de las redes.

Dichos resultados se han obtenido utilizando las herramientas de simulación descritas en el Anexo C.

3.3.1 Entornos estáticos

El área de simulación es fija (100x100 unidades) y se considera cerrada. El escenario asume el despliegue aleatorio de los nodos sensores en el área de interés. Las estaciones base asumen posición (simétrica) constante (Figura 3.7.1). El número de bases (o *clusters*) puede variar desde un mínimo de una a un máximo de cinco. También se supone comunicación ideal en cuanto no se ha modelado el canal de comunicaciones.

Las estadísticas de conectividad relacionadas (en función de la densidad o número de nodos y del rango de comunicación) se representan en Figura 3.8. Como se puede apreciar, se considera un rango de comunicación entre 20 y 30 unidades. Cada resultado reportado en la figura se refiere al promedio sobre 100 experimentos independientes.

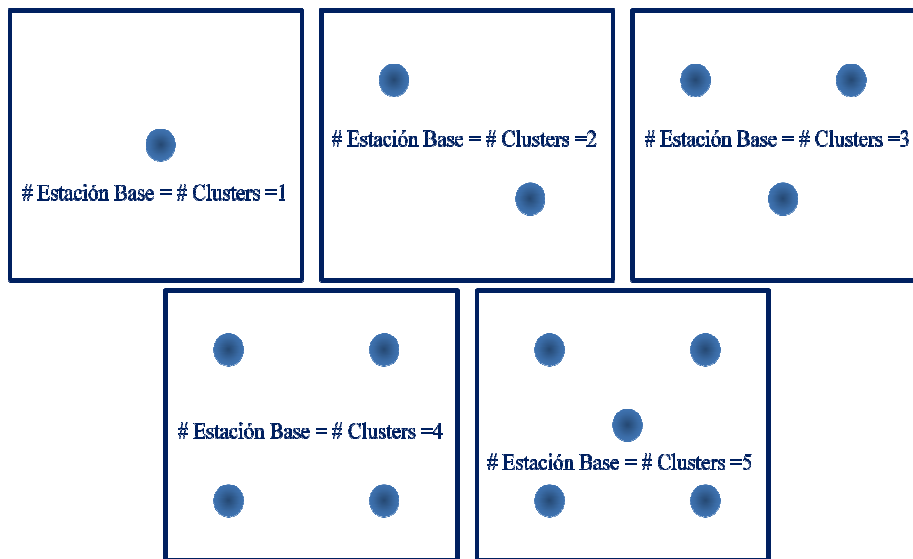


Figura 3.7.1. Disposición simétrica de estación base en un área.

Los diferentes parámetros (numero de *clusters*, densidades, rangos de comunicación, etc) se han elegido intentando reproducir, en los límites del posible, situaciones y topologías reales.

El factor determinante para las prestaciones es el número de estaciones bases desplegado, es decir el numero de cabeceras de *clusters*. La presencia de múltiples estaciones asegura una gran área de cobertura y, al mismo tiempo, altas prestaciones debidas a rutas de comunicaciones medianamente más cortas. Aumentar el número de estaciones que trabajan simultáneamente, en principio, tan solo implica capas middleware más complejas y la utilización de protocolos de configuración avanzada.

El rango de comunicación supone un impacto en las prestaciones parecido al número de estaciones base: valores altos garantizan una mejor conectividad entre nodos. Sin embargo, es importante considerar que una arquitectura multi-cluster no determina, de por sí misma y a paridad de tecnología, un incremento del consumo o decaimiento de otros factores de interés; contrariamente, incrementar el rango de comunicación de los nodos suele ser sinónimo de comunicación con mayor potencia o utilización de tecnología inalámbrica más avanzada y costosa.

Una tercera perspectiva de análisis se proporciona considerando las prestaciones en función de la densidad de nodos. Las prestaciones parecen incrementarse fuertemente en topologías caracterizadas por altas densidad de nodos. Evidentemente, este nivel de análisis tan solo relaciona las prestaciones de la red con la configuración sin tener en cuenta los aspectos,

directamente o indirectamente, relacionados con la comunicación y las característica de las aplicaciones. De hecho, topologías muy densas pueden resultar críticas respecto al acceso al medio generando un gran número de colisiones con pérdida importante de paquetes; este factor puede tener un peso diferente en las aplicaciones que presenten características diferentes (tráfico periódico, impulsivo, etc.).

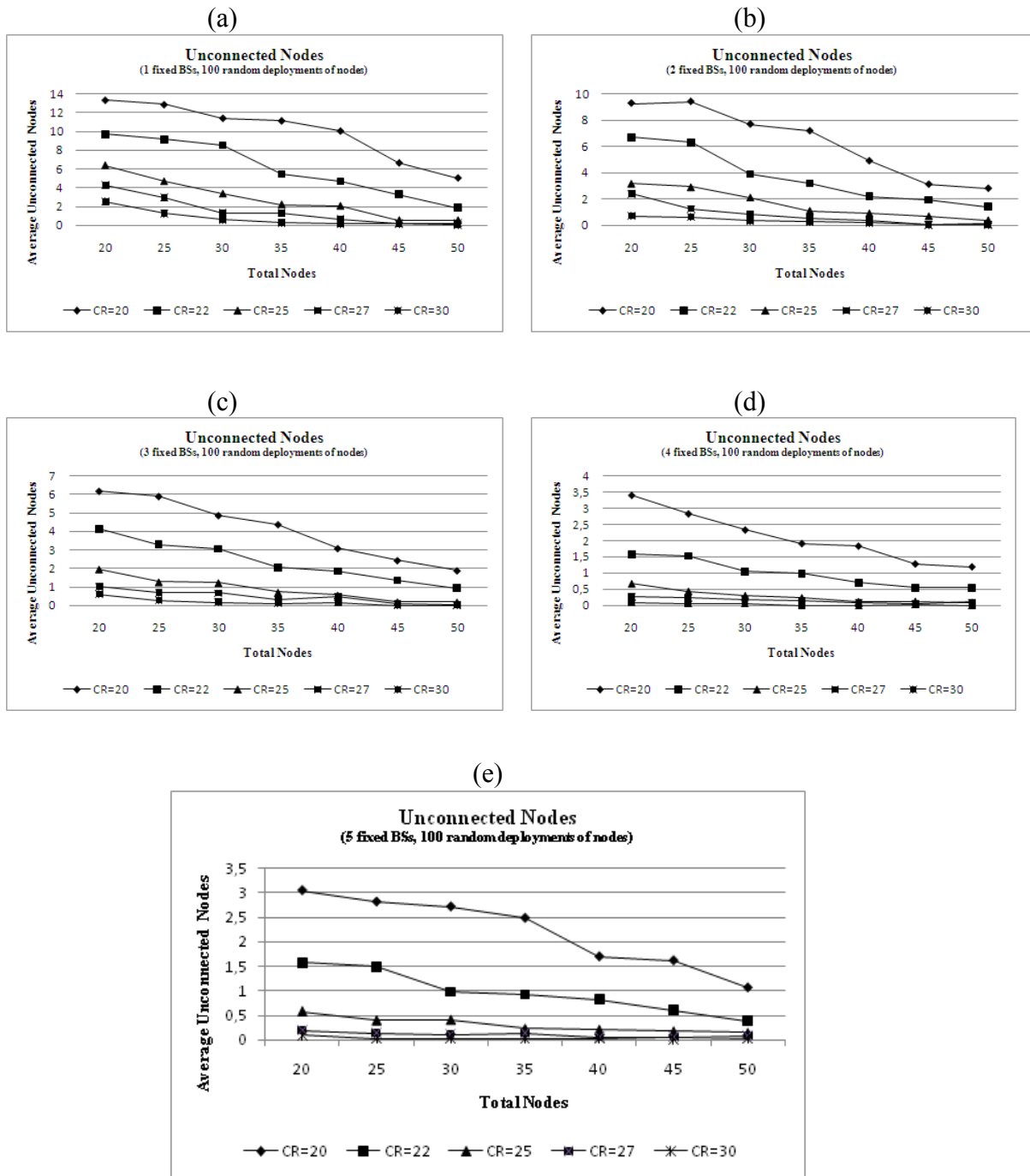


Figura 3.8. Estadísticas de conexión relacionadas con el despliegue aleatorio de nodos estáticos con número variable de estaciones base dispuestas simétricamente. El número de nodos es variable así como el rango de comunicación y el número de estaciones base. Cada valor reportado es el promedio de 100 experimentos independientes.

Finalmente, en la Figura 3.9 se propone una evaluación de la configuración en *cluster*. Como se puede apreciar, aunque el tamaño de los *clusters* tiende a balancearse considerando valores medios sobre más experimentos, despliegues puntuales pueden generar configuraciones muy irregulares. Este aspecto debe considerarse teniendo en cuenta considerando despliegues reales completamente aleatorios.

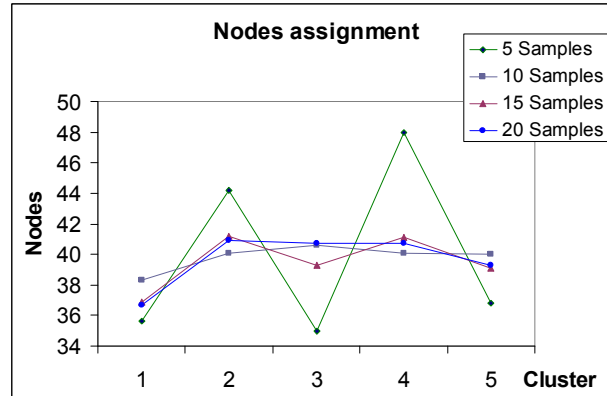


Figura 3.9. Estadísticas de configuración en clusters relacionadas con el escenario estático.

3.3.2 Entornos móviles

Para la evaluación del entorno móvil se ha analizado el impacto de la movilidad de los nodos sobre las prestaciones del sistema. Evidentemente no es posible un análisis completamente independiente de comportamientos concretos.

Considerada la metodología de análisis que se ha seguido hasta al momento, se eligió el modelo de comportamiento de los nodos más sencillo y genérico posible: todos los parámetros de T se definen como funciones aleatorias independientes tanto para los nodos sensores como para las estaciones base.

Bajo un cierto punto de vista, este comportamiento no implementa ningún comportamiento real o tendencia; sin embargo, el movimiento aleatorio en un área cerrada resulta de gran utilidad para evaluaciones de carácter genérico. Como se ha mencionado, el escenario de referencia asume solamente actores móviles y comunicación ideal (los enlaces en el rango de comunicación se suponen fiables al 100%).

Los *clusters* se configuran de acuerdo con el modelo general (enlaces mínimos); a igualdad de condiciones lógicas, un nodo se asocia al *cluster* con identificador menor. Todos los experimentos se efectúan con posición de salida simétrica de las estaciones base y despliegue aleatorio de los nodos sensores.

En Figura 3.10 se representan las estadísticas de conectividad en función del tamaño de la red y del rango de conectividad. Se considera el escenario con cinco estaciones base; la velocidad de cada nodo se considera constante ($\alpha=\beta=100$); cada resultado reportado se refiere a 1000 eventos discretos.

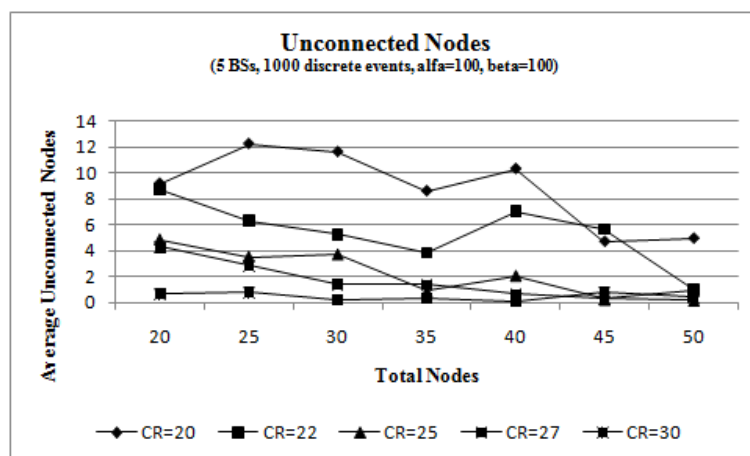


Figura 3.10. Estadísticas de conectividad relacionadas con el escenario móvil. Se considera una topología con 5 estaciones base. Cada valor reportado es el promedio de 100 experimentos independientes.

El escenario natural para relacionar los resultados obtenidos con el entorno estático con el mismo número de clusters (Figura 3.8(e)). Comparando las dos figuras, aparece claro que por rangos limitados de comunicaciones hay un decrecimiento importante de las prestaciones. Además, no se identifica un patrón regular como para entornos estáticos.

La diferencia de prestaciones se reduce al crecer del rango de comunicación, hasta a pasar a no ser relevante para valores máximos. Esto se debe a la mayor disponibilidad teórica de rutas.

Como todos los resultados proporcionados para análisis genéricos, estos resultados teóricos han de ser contextualizados; en el caso específico, la fiabilidad de los enlaces, la tecnología y los protocolos de comunicación pueden afectar las prestaciones de forma importante o determinante.

Analizando la configuración en clusters (Figura 3.11,a) se detecta una situación parecida a la anterior caracterizada por patrones irregulares debida a los bajos radios de comunicación. Valores más altos del rango de comunicación (Figura 3.11,b) determinan patrones más regulares, próximos a la idealidad (de acuerdo con la política de asignación).

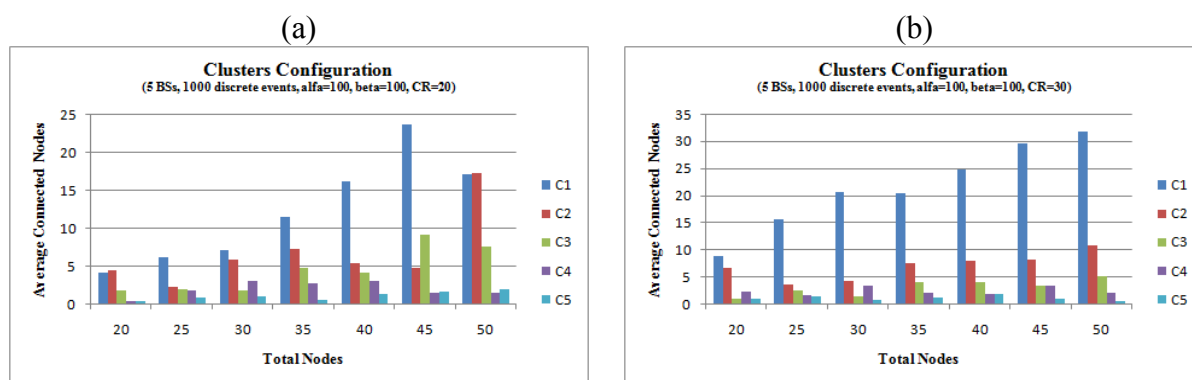


Figura 3.11. Configuración en clusters relacionadas con el entorno móvil. Se consideran rangos de comunicación de 20 metros (a) y de 30 metros (b).

En la Figura 3.12 se presenta la evaluación de las estadísticas de conectividad en función de la velocidad de movimiento de los nodos. Las estadísticas no parecen ser sensibles a la velocidad de movimiento de los nodos.

Este resultado puede considerarse válido tan sólo en un contexto teórico: en arquitecturas reales la velocidad de movimiento de los nodos suele afectar la fiabilidad de las rutas de comunicación. Garantizar una cierta fiabilidad en contextos con movilidad importante puede ser un desafío importante, especialmente considerando comunicación a baja potencia (y radio bajo de comunicación).

Resumiendo, la mayor limitación en el análisis genérico de entornos móviles es su estricta dependencia tanto respecto al patrón de comportamiento de los nodos como de la estabilidad de las rutas. Ambos aspectos se consideran intrínsecamente contextuales.

Por otra parte, las aplicaciones típicas de redes inalámbricas de sensores suelen soportar la generación y la comunicación de pequeñas cantidades de datos de acuerdo con un enfoque data-centric que, en ciertos casos concretos, permiten modelar las prestaciones en función del porcentaje de datos transferidos con éxito. Estos aspectos podrían limitar, de forma importante, el decrecimiento de las prestaciones.

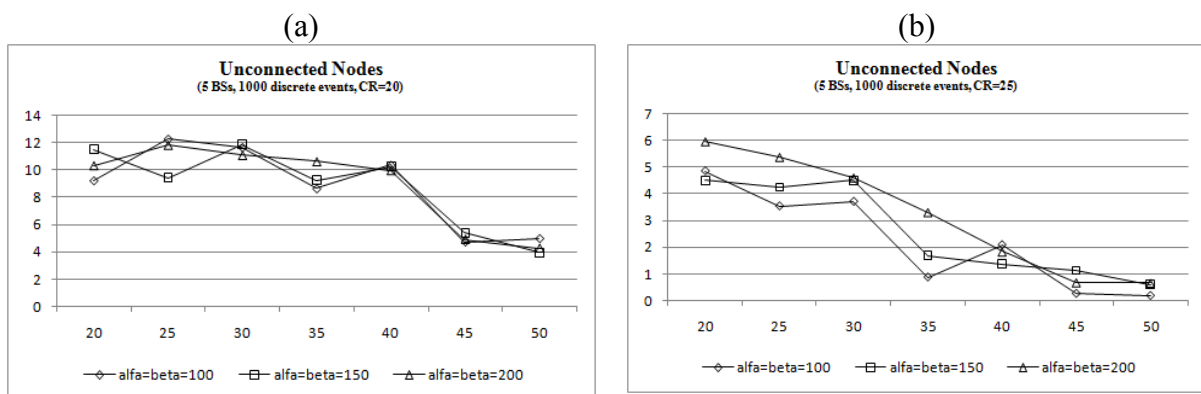


Figura 3.12. Estadísticas de conexión en función de la velocidad de los nodos. Se consideran rangos de comunicación de 20 metros (a) y de 25 metros (b).

A pesar de la objetiva dificultad de relacionar experimentos de redes móviles con entornos reales, el enfoque propuesto también proporciona una serie importantes de ventajas, como la posibilidad de comparar prestaciones relacionadas con diferentes modelos de comportamiento y la posibilidad de estructurar el análisis de las prestaciones en más niveles evaluando el impacto de factores reales sobre resultados teóricos.

3.3.3 Análisis de consumo energético

Para asegurar un análisis de consumo energético coherente con la metodología de modelo utilizada independiente de cualquier mecanismo o tecnología concreta, se ha diseñado un conjunto de pruebas que asumen entornos ideales respecto a topología, tráfico y consumo.

48 nodos sensores se disponen de forma simétrica al alrededor de una estación base con densidad uniforme (Figura 3.13, izquierda). Se disponen en tres diferentes niveles (Figura 3.14, izquierda) según la definición anteriormente formulada en el capítulo (eq. 3.2, eq. 3.2.1). El diagrama de conexión entre los nodos se puede apreciar en Figura 3.13 (derecha).

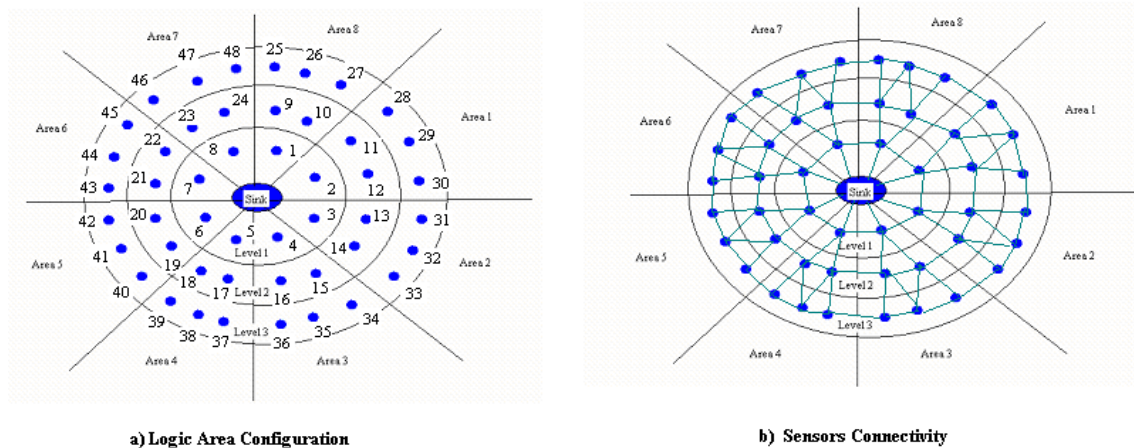


Figura 3.13. Topología y conectividad entre nodos para el análisis de consumo energético. Figura original de [Con1].

Utilizando la topología propuesta se genera un tráfico teórico regular y uniforme en el que cada nodo genera el mismo número de mensajes a transmitir e balancea el tráfico entre las rutas de comunicación disponibles. El destinatario de un mensaje puede ser la estación base o un nodo sensor. Se utilizan solo *rutas de comunicación convergentes directas*, como se han definido anteriormente (eq. 3.3) para comunicación hacia la estación base y rutas óptimas (sección 3.2.2.2) para la comunicación nodo a nodo.

Los resultados obtenidos procesando 5000 eventos determinan un número medio de saltos próximo a cuatro (Figura 3.14).

A causa del más veces mencionado fenómeno de *energy-hole*, la carga de tráfico global soportada por la red no se reparte uniformemente: la intrínseca centralidad de la estación base determina que la carga real sea inversamente proporcional a la distancia con la estación base. Los nodos más próximos a la estación base son, por lo tanto, los que sufren un más mayor consumo de energía.

Este consumo irregular determina realmente el tiempo de vida de la red: un número más o menos importante de nodos puede quedarse aislado respecto al resto de la red aunque de forma individual aun disponga de recursos energéticos, simplemente porque el tiempo de vida de los nodos más cercanos a la estación base es notablemente inferior.

El modelo propuesto permite un análisis más completo y efectivo, organizando los nodos por nivel (Figura 3.15). Como se puede apreciar, incluso en condiciones teóricas de máxima regularidad bajo todos los puntos de vista, el decrecimiento de las prestaciones al crecer del nivel es bastante marcado.

Este aspecto, junto con los más comunes parámetros de calidad de servicio (fiabilidad, robusteza), debería que tenerse en cuenta a la hora de dimensionar las redes.

Aunque el fenómeno de *energy-hole* sea intrínseco en las redes inalámbricas de sensores a causa de la centralidad de la estación base, diseñar topologías y mecanismos (de enrutamiento en particular) capaces de optimizar los consumos puede ser un factor determinante para gestionar de la mejor forma posible el tiempo de vida de la red.

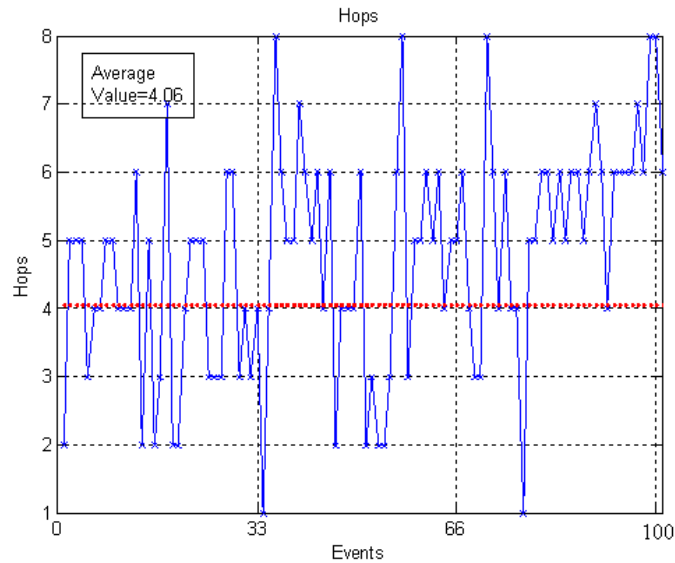


Figura 3.14. Valor medio del número de saltos de las rutas de comunicación. Figura original de [Con1].

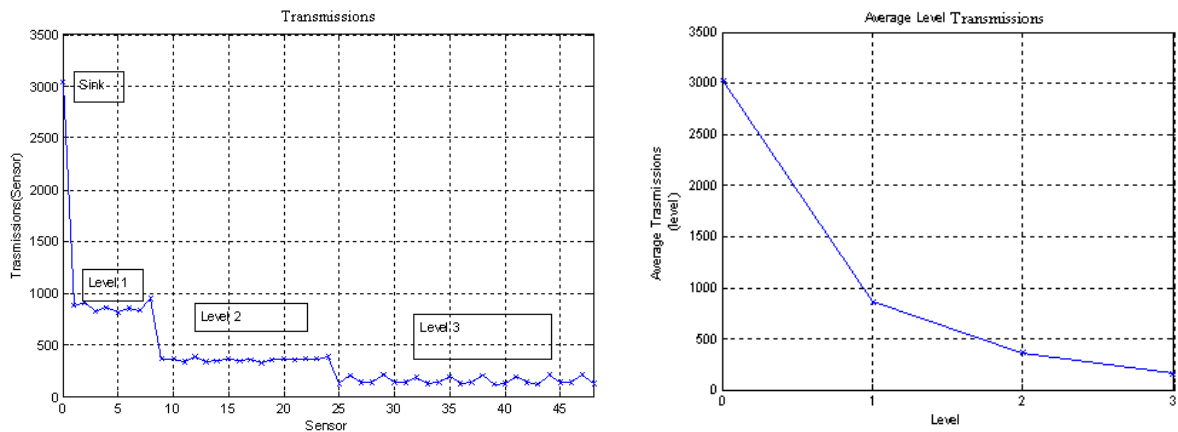


Figura 3.15. Transmisiones por nodo (izquierda) y valor medio de transmisiones por nivel. Figura original de [Con1].

Desde un punto de vista lógico, la solución más escalable y efectiva para diseñar arquitecturas con altas prestaciones es la configuración en *clusters* de la redes, como se ha mostrado tanto en los primeros capítulos de la memoria como en este mismo capítulo.

Desde un punto de vista topológico, para aumentar el tiempo de vida de una red de sensores se deberían disponer los nodos con densidades variables decrecientes con la distancia de las estaciones base. Este tipo de aproximación, que asume alta densidad de nodos en el alrededor de las estaciones base, si soportada por mecanismos dinámicos de gestión del trafico pueden aportar un cierto incremento de las prestaciones globales. Por otra parte, este tipo de técnica no es siempre aplicable, presenta ciertas desventajas (coste mayor, mecanismos de control de topología y de enrutamiento medianamente más complejos) y fuerza la red a trabajar en un contexto de mayor densidad.

CAPÍTULO 4

4 Implementación y Aplicaciones

Este capítulo complementa de forma natural el segundo capítulo en cuanto, por una parte, describe la implementación del framework de base de la arquitectura de protocolos de la red de sensores anteriormente propuesto y, por la otra, propone diferentes particularizaciones de la arquitectura completa en diferentes entornos de aplicación. La descripción de la implementación es relativa al modulo CORE de la arquitectura de protocolos, tal como se ha definido en el segundo capítulo. No se proporcionan detalles sobre la implementación de los componentes semánticos de la arquitectura en cuanto se han ampliamente tratado en el contexto del segundo capítulo así como no se analizan detalles genéricos de la implementación de los componentes middleware en cuanto se consideran infraestructuras a medida. Cada una de las aplicaciones propuestas presenta ciertas peculiaridades en términos de funcionalidades, requisitos o entorno que requieren particularizaciones o extensiones del modelo general.

4.1 Introduccion

En el Capitulo 2 de la presente memoria se ha propuesto un análisis profundo de la arquitectura propuesta con una perspectiva técnico-funcional como en relación a los diferentes entornos lógicos de referencia.

Este capítulo complementa el Capitulo 2 proponiendo:

- Detalles sobre la implementación de los componentes fundamentales de la arquitectura. Más concretamente el análisis interesa el modulo *CORE* de la arquitectura de protocolos tal como descrito en el segundo capítulo de esta memoria.
- Análisis de la aplicación de la arquitectura propuesta en diferentes contextos. Todos los entornos de aplicación considerado se caracterizan por una cierta complejidad respecto a las funcionalidades, a los requisitos o al entorno que requiere particularizaciones o extensiones significativas del modelo de arquitectura general propuesto.

Desde un punto de vista de la implementación, la arquitectura propuesta se puede considerar compuesta por tres diferentes módulos funcionales caracterizados por un nivel de abstracción creciente:

- *Arquitectura de protocolos de las redes de sensores.*
- *Infraestructuras middleware.*
- *Componentes semánticos.*

La primera parte del capítulo está dedicada al análisis de la implementación de la arquitectura de protocolos. Más concretamente se considera el módulo funcional *CORE* en cuanto implementa las funcionalidades genéricas de base del framework. Otros módulos se tratarán con un menor nivel de detalle porque, aunque implementando funcionalidades importantes, se consideran característicos de entornos específicos más que mecanismos generales.

Las infraestructuras middleware, tal como han sido definidas en el contexto del segundo capítulo, también se consideran infraestructuras a medida. Presentan ciertos patrones que podrían permitir la definición de un framework abstracto parecido a lo definido para las redes de sensores. En este trabajo, las infraestructuras middleware se han desarrollado a medida en tecnología GT4. La especificación de un framework de referencia podría ser objeto de trabajos futuros.

Finalmente, las componentes semánticas, tanto a nivel de definición e implementación de los modelos de información como a nivel de las infraestructuras para el procesamiento de los mismos, han sido ampliamente tratados en secciones anteriores.

La segunda parte del capítulo propone una descripción de las siguientes aplicaciones:

- *Plataforma para adquisición de datos.*
- *Multi-mode WSN.*
- *Convergencia entre redes de sensores inalámbricas y tecnología RFID en entornos industriales.*
- *Redes de sensores en entornos vehiculares.*
- *Integración de redes de sensores en sistemas de mando y control.*

4.2 Implementación del Framework: módulo *CORE*

En esta sección se describen los diferentes protocolos y mecanismos que componen el módulo *CORE*. Como se ha presentado en las secciones anteriores y como se puede apreciar en la Figura 4.1, este módulo funcional asume un rol absolutamente primario en el contexto de la arquitectura de protocolos propuesta: por una parte proporciona los mecanismos funcionales básicos y, por la otra, constituye el complejo de funcionalidad sobre el cual se desarrollan los otros mecanismos.

El módulo *CORE* se ha diseñado e implementado como un framework genérico sin asunción alguna sobre la aplicación a soportar. Para que sea efectivo en diferentes dominios y contextos funcionales, las funcionalidades más relevantes ofrecen múltiples modos de funcionamiento con el objetivo de adaptarse de la mejor forma posible a los diferentes entornos de trabajos.

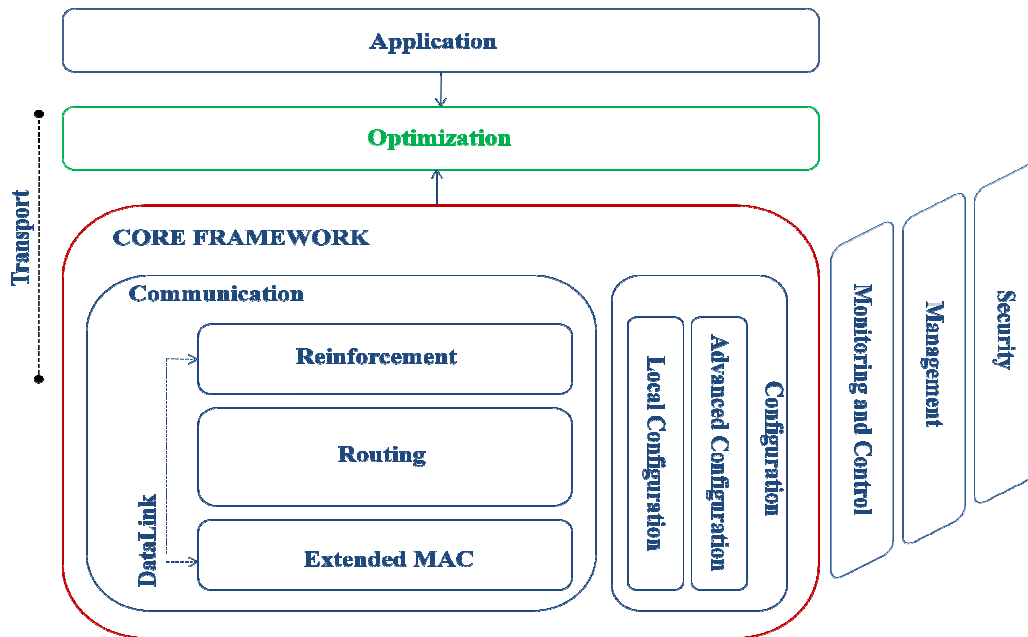


Figura 4.1. Módulo CORE en el modelo de referencia.

A continuación se analizan los diferentes componentes del módulo CORE:

- *Funcionalidades extendidas de acceso al medio.*
- *Protocolo de configuración.*
- *Protocolo de enrutamiento.*
- *Mecanismos de refuerzo (Reinforcement).*
- Se describe además el protocolo de *Monitorización y Gestión* que, aun no forme parte del modulo central, se implementa directamente sobre el protocolo de configuración del cual es un complemento natural.
-

Todos los mecanismos que se describen en esta sección se han implementado como módulos TinyOS en lenguaje nesC.

4.2.1 Funcionalidades extendidas de acceso al medio

Las funcionalidades extendidas de acceso al medio constituyen el mecanismo más básico proporcionado por el framework (Figura 4.2). Como se ha mencionado en el primer y segundo capítulo de la tesis, dicha clase de protocolos se suele adoptar en el caso de dispositivos con funcionalidades limitadas de acceso al medio como en el caso de los dispositivos RFD de ZigBee [Bar1].

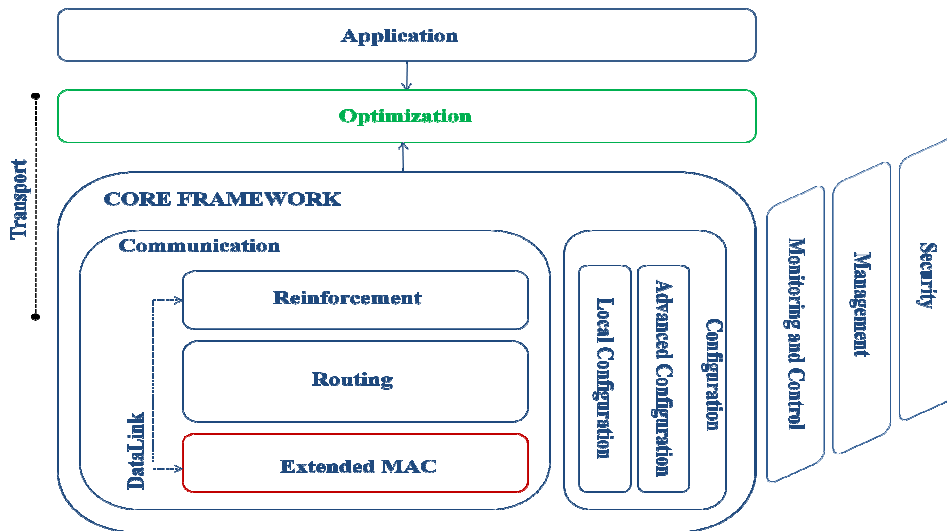


Figura 4.2. Funcionalidades extendidas de acceso al medio en el modelo de referencia.

Las funcionalidades de acceso al medio en la tecnología inalámbrica empleada se proponen limitar el número de colisiones generadas por acceso simultáneo al medio de comunicación. En el caso específico, considerando las características y el modo de funcionamiento habitual de las redes inalámbricas de sensores, las colisiones constituyen un factor absolutamente crítico por lo menos en dos casos muy comunes:

- En presencia de mensajes tipo broadcast que actúan como “sincronizadores”. Como se puede apreciar en la Figura 4.2.1, un mensaje broadcast suele ser recibido en tiempos bastante parecidos por los diferentes nodos interesados. Si estos nodos tienen que enviar un mensaje como reacción al evento de recepción, existe una altísima probabilidad de colisión entre los mensajes enviados por los diferentes nodos. El caso descrito es muy común en la fase de configuración de las redes donde los mensajes broadcast suelen emplearse para el descubrimiento de los nodos y los mensajes de confirmación (unicast normalmente) se envían como reacción a la recepción de un mensaje de descubrimiento.

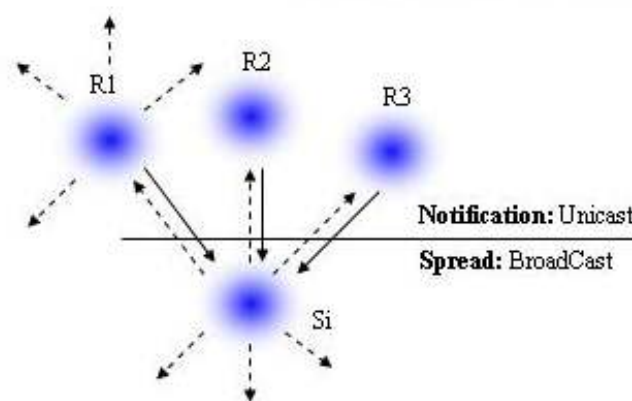


Figura 4.2.1. Típico escenario de colisión de mensajes en WSN. Figura original de [Con5].

$$S(i)=T_0+k*(r/k1+NodeID*k2) \quad (\text{eq. 4.1})$$

$$\left\{ \begin{array}{l} T_0 = \text{tiempo de recepcion de mensaje de spreading} \\ k, k_1, k_2 = \text{factores de escala} \\ r = \text{valor intero aleatorio} \end{array} \right.$$

En la Figura 4.2.2 se representan las colisiones detectadas en presencia de cuatro nodos que reaccionan a mensajes de sincronización enviados por un quinto sensor. El estudio se realiza en función del parámetro k , considerando constantes los otros parámetros de escala (k_1 y k_2). Se han utilizado dispositivos MicaZ.

Como se puede apreciar, el número de colisiones se reduce de forma significativa al crecer del parámetro k . Por un valor de $k=15$ la probabilidad de colisión es muy baja.

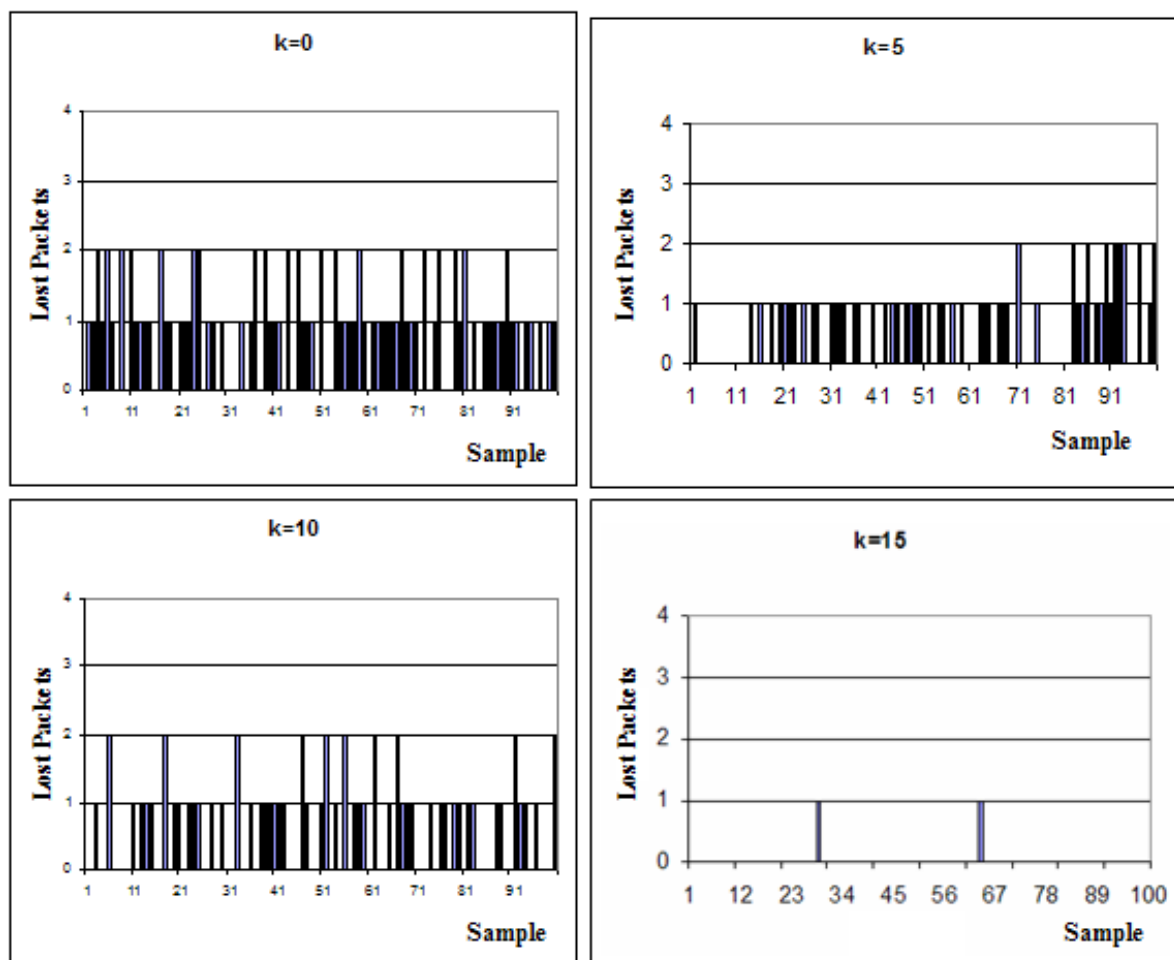


Figura 4.2.2. Colisiones en función de k utilizando el modelo con slot aleatorio y la topología de referencia representada en Figura 4.2.4. Figura original de [Con5].

El modelo con slot aleatorio se considera muy flexible en cuanto no necesita ningún conocimiento previo de la aplicación a soportar o del entorno relacionado (densidad de nodo). Simplemente puede configurarse de forma óptima a través de unos parámetros.

En caso de conocimiento previo de la topología (topologías predefinidas y/o ordenadas) y, por lo tanto, de la densidad de nodos es posible utilizar protocolos basados en el *modelo con slot temporal predefinido* o *PreFixedSlotTDMA* (Figura 4.2.3 considerando 4 nodos, eq.4.2).

$$S(i) = T_0 + C + k * \text{HostID}(i) \quad (\text{eq.4.2})$$

C = valor constante

Como se puede apreciar, al contrario del modelo anterior no hay algún factor aleatorio: el acceso al medio se efectúa de forma ordenada de acuerdo con una cierta prioridad (en este caso definida por el identificador del nodo).

Las prestaciones son parecidas a las del modelo anterior pero este tipo de protocolo es mucho más fácil de configurar para asegurar prestaciones óptimas. El parámetro k determina la duración del slot temporal por el cual se repropone el mismo *tradeoff* entre probabilidad de colisión y reactividad anteriormente analizado.

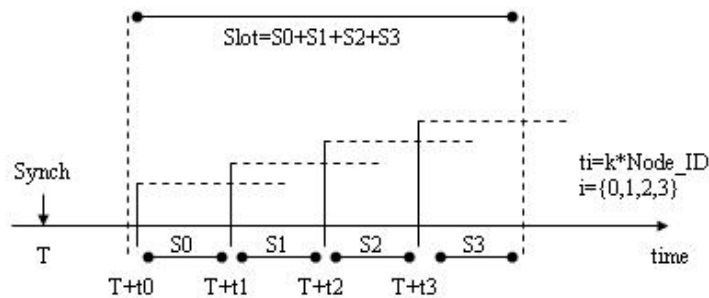


Figura 4.2.3. Modelo con slot temporal predefinido.

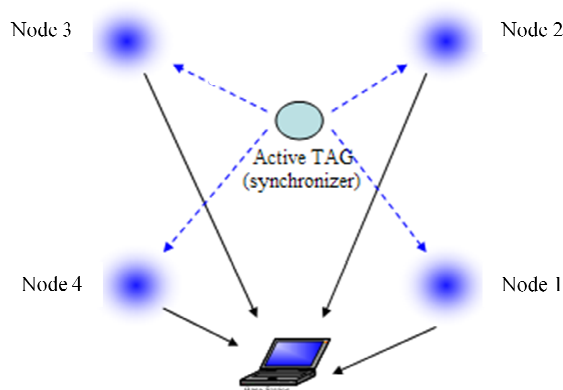


Figura 4.2.4. Topología de referencia para la evaluación de los mecanismos extendidos de acceso al medio.

4.2.2 Protocolo de configuración

El protocolo de configuración se considera uno de los mecanismos fundamental de la arquitectura (Figura 4.3) en cuanto alcanza tres objetivos fundamentales:

- Configura los parámetros lógicos (tal como definidos en el capítulo anterior) de la red.
- Asocia, en el caso de múltiples estaciones base que trabajan simultáneamente, los nodos sensores a una estación base (configuración en clusters).
- Configura las estructuras de datos locales a soporte del protocolo de enrutamiento (tabla de vecinos y, eventualmente, otros parámetros).

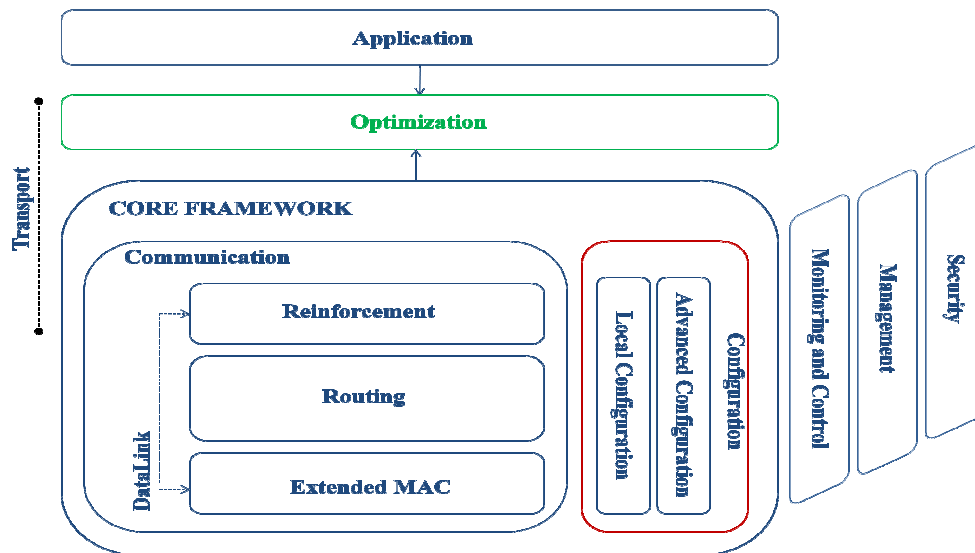


Figura 4.3. Protocolo de configuración en el modelo de referencia.

Las redes locales de sensores distinguen tres diferentes tipos de mensajes:

- Mensajes de tipo *spreading*: se trata de mensajes broadcast que se utilizan para el descubrimiento de los nodos.
- Mensajes de tipo *notification*: se utilizan para notificar la correcta configuración de un nodo.
- Mensajes de nivel de aplicación. Suelen generarse para enviar información generada por los nodos sensores.

Todas las operaciones anteriormente descritas se implementan en único intercambio de mensajes (fase) de acuerdo con los diagramas de actividad descritos en Figura 4.3.1:

- La tarea de configuración siempre se inicia por las estaciones base de forma asíncrona a través del envío de un mensaje de *spreading* que contiene el identificador de la estación base emisora y parámetros lógicos nulos según su definición (Capítulo 3).
- Cuando un nodo recibe un mensaje de *spreading* interpreta el mismo como una propuesta de configuración: intenta configurarse con el valor del parámetro de nivel menor posible. La asociación al cluster correspondiente es implícita: el nodo se asocia al cluster que propone las mejores condiciones. El parámetro radial también se configura de acuerdo con su definición (Figura 4.3.1 a la derecha).

- Se actualizan las estructuras locales, básicamente la tabla de vecinos y el campo *notifier* que es el *sender* del mensaje de *spreading* que ha determinado la configuración corriente.
- Se propaga el mensaje de *spreading* con los campos correspondientes al nodo *sender* actualizados (identificador del nodo y su configuración lógica).
- Solo en el caso de haber modificado la anterior configuración lógica como consecuencia de la recepción de un mensaje de *spreading*, se envía una notificación de la configuración corriente a través del *notifier*.
- Si el nodo recibe un mensaje de notificación proveniente de un nodo con parámetro de nivel superior, este mensaje viene tratado a nivel de red como una notificación propia y, por lo tanto, propagada a través del *notifier* corriente.

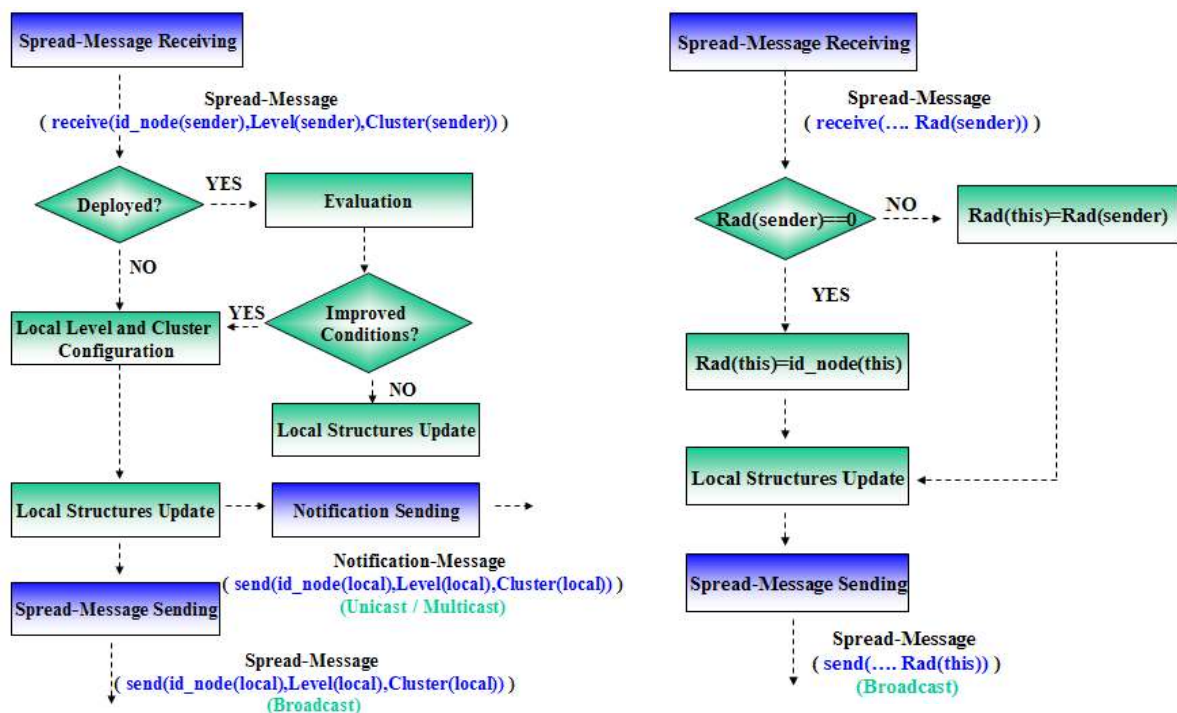


Figura 4.3.1. Diagrama de actividad del protocolo de configuración. Figura original de [Con4].

4.2.3 Protocolo de enrutamiento

El protocolo de enrutamiento trabaja directamente sobre la configuración lógica proporcionada (Figura 4.4) implementando la estrategia de enrutamiento. En esta sección, el análisis se limita al protocolo de enrutamiento para comunicación convergente.

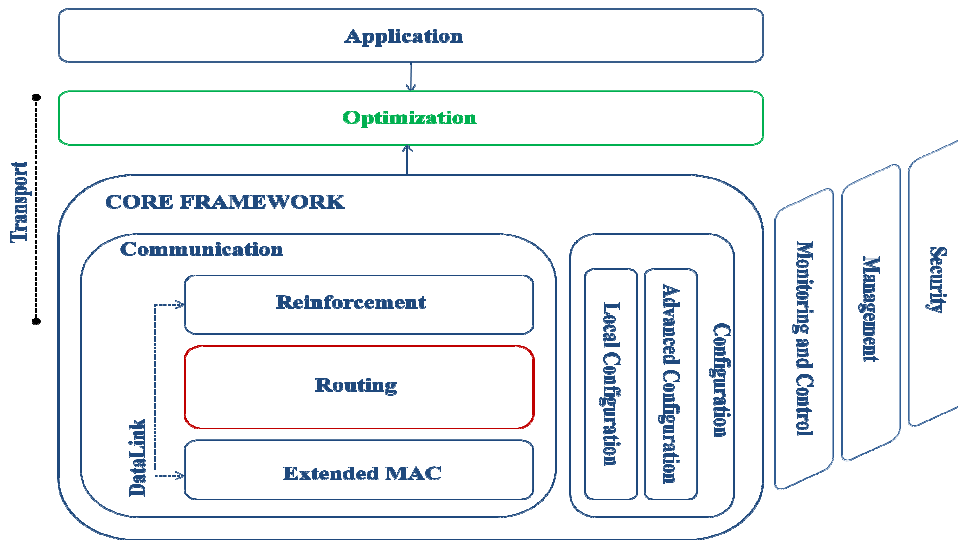


Figura 4.4. Protocolo de enrutamiento en el modelo de referencia.

La configuración lógica proporcionada permite a cada nodo clasificar sus vecinos en función del parámetro de nivel (parámetro de nivel igual, superior o inferior). Por lo tanto, tan solo considerando la configuración lógica, cada nodo puede implícitamente clasificar las rutas disponibles de acuerdo con el modelo propuesto en el Capítulo 3 (rutas directas, paralelas e indirectas). En condiciones normales se utilizan rutas directas (más eficientes).

En el caso de múltiples rutas disponibles (Figura 4.4.1) con características lógicas equivalentes, se precisa un criterio de selección que es implementado por el protocolo de enrutamiento.

El framework incluye cuatro diferentes estrategias de enrutamiento:

- Enrutamiento dinámico (*RandomRouting*): elige la ruta de forma aleatoria entre las disponibles. No requiere ningún tipo de estructura adicional. Suele aplicarse a soporte de aplicaciones y topologías regulares.
- Enrutamiento *energy-aware*: elige la ruta privilegiando nodos con mayor carga energética para maximizar el tiempo de vida de la red. El entorno de conocimiento de cada nodo se limita a sus vecinos. Requiere mecanismos para diseminar el estado energético de cada nodo en su entorno local. Este tipo de protocolo presenta las ventajas y desventajas descritas en el primer capítulo de estado del arte.
- Enrutamiento *location-aware*: es una estrategia de enrutamiento conceptualmente parecida a la anterior, solo que la ruta se elige en función de la posición (se elige el nodo más próximo a la estación base). El entorno de conocimiento se limita a los vecinos y se requieren mecanismos para diseminar la posición de la base a todos los nodos (normalmente en fase de configuración) y de la posición de cada nodo en su entorno de conocimiento local.
- Enrutamiento con flujos de datos: elige la ruta en función de su fiabilidad. Para determinar la ruta más fiable, el nodo envía a sus vecinos unos mensajes de prueba esperando confirmación de correcta recepción. Considerando flujos que interesan un solo salto, esta estrategia de enrutamiento no proporciona un incremento significativo de las prestaciones en relación con los recursos empleados.

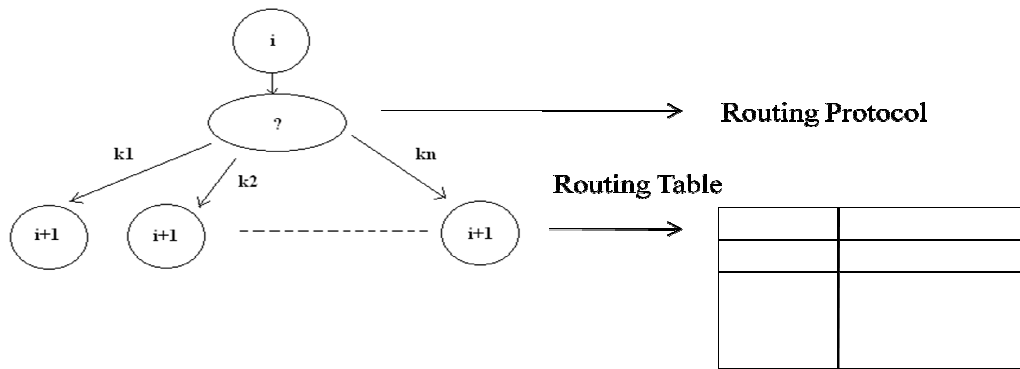


Figura 4.4.1. Protocolos de enrutamiento.

4.2.4 Reinforcement

Los mecanismos de *Reinforcement* actúan en la capa más alta del módulo funcional central (Figura 4.5) para incrementar la fiabilidad de las comunicaciones.

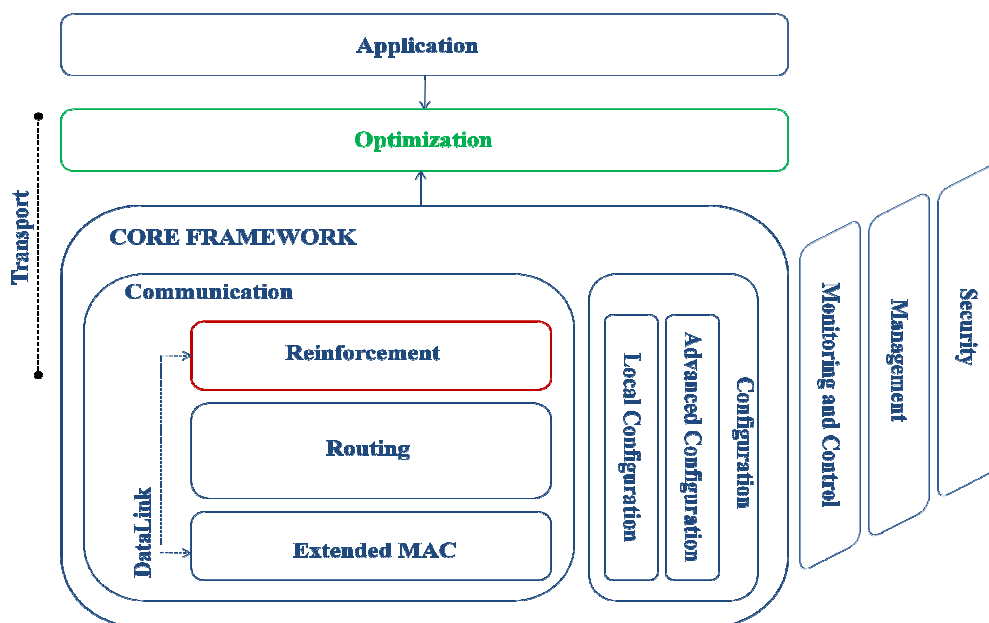


Figura 4.5. Mecanismos de *Reinforcement* en el modelo de referencia.

El framework incluye dos mecanismos diferenciados:

- *Mecanismo de confirmación (ack)*. Es un mecanismo básico de confirmación nodo a nodo. En caso de envío de un mensaje de notificación o aplicación, el nodo que envía se queda en espera de un mensaje de confirmación. Si no lo recibe en el tiempo pre-establecido, vuelve a enviar el mensaje. Evidentemente este mecanismo puede aumentar la robustez y la fiabilidad de las comunicaciones. Sin embargo, no se considera especialmente efectiva en redes con alta densidad de nodos donde los

mensajes de confirmación podrían generar ulteriores colisiones.

El framework no incluye mecanismos de confirmación base a nodo de acuerdo con un modelo tipo transporte fiable.

- *Replica de información.* En redes con densidad importante, una buena alternativa a la confirmación para incrementar la fiabilidad de las comunicación es la réplica de información: el mensaje generado se envía a través de la ruta definida por el protocolo de enrutamiento y por otra que se define privilegiando rutas convergentes paralelas (Capítulo 3) y, a falta de éstas, rutas convergentes directas (Capítulo 3). En caso de falta de disponibilidad de rutas de los tipos anteriormente mencionados, se puede recurrir a rutas indirectas (ineficientes). Como se ha mencionado, los entornos ideales para aplicar este mecanismo son las redes con alta densidad que suelen garantizar un alto nivel de conectividad y, por lo tanto, la disponibilidad de rutas de comunicación alternativas.

Estos dos mecanismos son perfectamente compatibles entre ellos aunque, hasta al momento, no se han utilizado simultáneamente en arquitecturas reales.

4.2.5 Mecanismo de monitorización y control

Este mecanismo (Figura 4.5.1) actúa en colaboración directa con el mecanismo de configuración del cual se puede considerar una extensión. Su principal funcionalidad es monitorizar el estado de la red y controlar, de forma optima, la configuración de la misma.

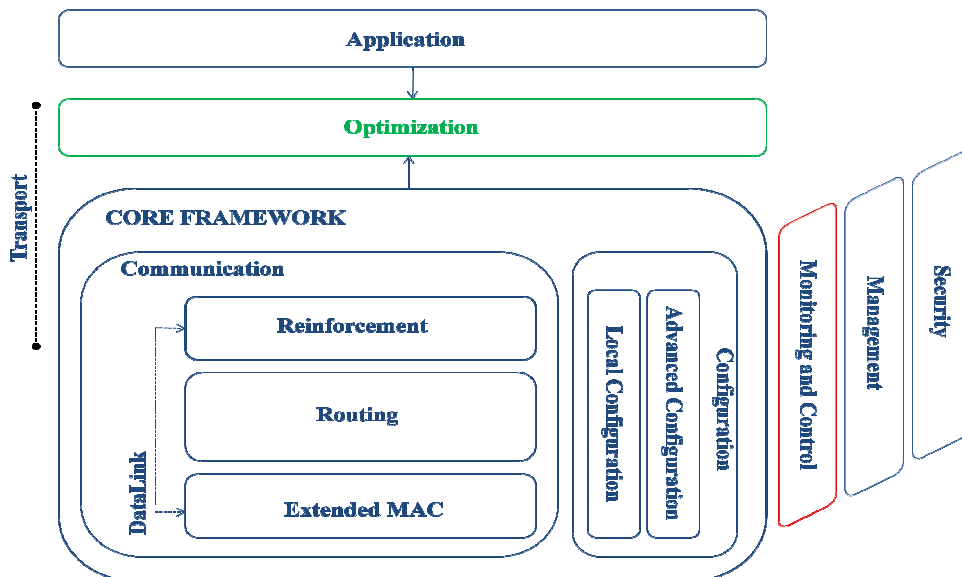


Figura 4.5.1. Mecanismo de monitorización y control en el modelo de referencia.

En condiciones normales de funcionamiento, la red se auto-monitoriza a través de los mensajes de notificación y aplicación a través de los cuales la estación base puede determinar el número de nodos activos, conectados y que funcionan correctamente.

Sin embargo, especialmente considerando que los mensajes de aplicación podrían recibirse con periodos largos (entre la recepción de dos mensajes de nivel aplicación podría pasar un tiempo importante) y que las condiciones del entorno pueden cambiar en el tiempo, las operaciones de configuración suelen repetirse en el tiempo para garantizar un alto nivel de control sobre la red.

La periodicidad de estas operaciones puede garantizar fiabilidad frente a fallos, cambios de topología, agotamiento de las baterías, obstáculos y/o interferencias temporales, etc.

La gestión de la red puede efectuarse de forma estática o dinámica:

- *Gestión estática*: Las operaciones de configuración se repiten periódicamente de acuerdo con el periodo de tiempo preestablecido por el usuario. El periodo de reconfiguración suele relacionarse, en caso de adquisición periódica de datos, con el periodo de la aplicación del cual suele ser múltiplo. Otros tipos de aplicación aconsejan una evaluación a medida.
- *Gestión dinámica (Dynamic Configuration Protocol)*: El sistema de gestión evalúa las condiciones de la red y configura el periodo de re-configuración de acuerdo con el siguiente algoritmo:

```
T0 = tiempo de re-configuración inicial
T = tiempo de re-configuración corriente
Tmin = tiempo de reconfiguración mínimo
Na(t) = numero de nodos activos en el tiempo t
St = constante

T=T0;
.....

Iteración genérica en el tiempo t:

    if ( Na(t) > Na(t-1) ) {
        if (T-St) > Tmin {
            T = T-St;
        }
        else {
            T = Tmin;
        }
    }

    if ( Na(t) < Na(t-1) ) {
        T = T+St;
    }
```

Como se puede apreciar, el periodo de reconfiguración varía en función del número de nodos activos detectados. Este comportamiento refleja una estrategia de gestión que intenta garantizar fiabilidad en un contexto de optimización de recursos: si detecta un número de nodos activos decrecientes, se configura para ejecutar operaciones de configuración más frecuentemente en la esperanza que el número de nodos detectado este decreciendo a causa de fallos de comunicación o de condiciones ambientales temporales. Al contrario, detectar condiciones mejores de las anteriores aconseja el menor utilizo de los recursos con configuraciones menos frecuentes. Esta estrategia se ha demostrado efectiva considerando el dinamismo del protocolo de configuración que se basa en parámetros lógicos que admiten múltiples configuraciones equivalentes.

4.3 Aplicaciones

En esta segunda parte del capítulo se describen las aplicaciones de la arquitectura propuesta. Cada una de estas aplicaciones presenta ciertas peculiaridades respecto a las funcionalidades, los requisitos o el entorno de aplicación. De consecuencia, cada arquitectura se caracteriza por alguna extensión significativa o particularización de modelo general anteriormente propuesto. Las diferentes aplicaciones se han integrados en diferentes proyectos de investigación.

Dada la complejidad de las aplicaciones relacionadas con las redes de sensores, y la alta dependencia de la configuración y funcionamiento de las mismas, no se ha planteado un único caso en el que validar las diferentes características de la arquitectura sino cinco aplicaciones diferentes en las que en cada una se potencia un aspecto diferente y relevante de la arquitectura.

Más concretamente se consideran las siguientes arquitecturas:

- *Plataforma para adquisición de datos.* Es una de las aplicaciones más comunes de las redes inalámbricas de sensores. La solución propuesta se caracteriza por su flexibilidad que les permite trabajar en diferentes contextos lógicos (como sistema autónomo, como sistema empotrado y como sistema distribuido) y físicos (topologías escalables aplicando técnicas de *chuestring*). También se integran unas componentes avanzadas en tecnología semántica orientadas a soportar el procesado avanzado de la información en un contexto de abstracción e interoperabilidad.
- *Multi-mode WSN.* Es un modelo innovador de redes inalámbricas de sensores que aplica el concepto de *Always Best Connected (ABC)* [Abc1] para redes inalámbricas de sensores poco densas. La solución propuesta se compone de nodos inteligentes con capacidades tecnológicas mejoradas. Soluciones concretas pueden caracterizarse por alto nivel de robustez y fiabilidad.
- *Convergencia de redes inalámbricas de sensores con tecnología RFID.* La integración de redes inalámbricas de sensores con tecnologías RFID se caracteriza por un potencial importante a nivel comercial. La solución propuesta se ha diseñado para soportar simultáneamente la detección, identificación y localización de objetos en áreas industriales y la detección de parámetros ambientales y personales en el área considerada.
- *Redes inalámbricas de sensores en entornos vehiculares.* La comunicación de baja potencia no suele adoptarse en entornos vehiculares. Sin embargo varias aplicaciones de redes de sensores en larga escala asumen o pueden asumir nodos sensores montados sobre vehículos. Un enfoque “*green*” de dichas redes, caracterizadas por bajos consumos y alta eficiencia en términos de área de cobertura podrían encajar en diferentes modelos de negocio.
- *Integración en sistemas de mando y control.* Uno de los elementos clave para la efectividad de los sistemas de mando y control frente a emergencias es la capacidad de proporcionar cuanta más información posible respecto a la situación en cuestión (*situation awareness*). Las redes inalámbricas de sensores pueden contribuir, de forma importante, a la evaluación de la situación de emergencia. La plataforma de adquisición de datos se ha integrado en un sistema de mando y control.

4.3.1 Plataforma para adquisición de datos

La adquisición de datos es una aplicación clásica de las redes de inalámbricas de sensores. Dichas plataformas se componen de un conjunto de funcionalidades que habilitan la captura de información por parte de los nodos sensores y la transmisión de la misma hacia el sistema de control local y/o central.

Este tipo de aplicación se caracteriza por una actividad extremadamente regular de los componentes sensores que suelen implementar tareas de captura y transmisión de datos periódicas. Esta aplicación se considera, por lo tanto, de gran interés en el contexto del presente trabajo en cuanto ha permitido tanto la experimentación controlada de los diferentes componentes hardware a disposición (nodos inalámbricos, tarjetas de adquisición de datos) como el análisis del comportamiento de los mismos en diferentes condiciones. Además, la plataforma en cuestión ha asumido un rol primario en la fase de prueba y validación de los diferentes protocolos que componen el framework de referencia.

La integración de mecanismos inteligentes y de representación avanzada de la información se considera un tema de interés añadido.

La plataforma propuesta se ha diseñado e implementado a la luz de las siguientes consideraciones:

- La plataforma está pensada para la monitorización de parámetros ambientales, aunque su campo de aplicación potencial puede fácilmente extenderse a otros entornos.
- La plataforma está pensada para soportar tanto la adquisición de datos en tiempo real bajo demanda como adquisición de datos periódica.
- Puede funcionar de acuerdo con diferentes topologías y entornos. En la presente tesis se consideraran básicamente tres configuraciones: plataforma de adquisición de datos como sistema autónomo, plataforma de adquisición de datos para sistemas empotrados y plataforma de adquisición de datos distribuida.
- La plataforma asume implícitamente despliegue aleatorio de los nodos. Los nodos se suponen fijos o nomádicos. Únicamente la modalidad de funcionamiento bajo demanda puede soportar nodos móviles.
- Soporte de funcionalidades avanzadas (disparadores, reacción a eventos, notificaciones programadas).
- Todos los componentes de la arquitectura se consideran funcionales también en entornos empotrados. La plataforma proporcionada es fácilmente integrable en sistemas más complejos. Más adelante en el capítulo se describe la integración de la plataforma en sistemas de mando y control.
- Fácil integración de mecanismos de inteligencia avanzados basados en tecnologías semánticas (tal como descrito en el Capítulo 2). Ejemplos relevantes pueden ser el procesamiento inteligente de datos en tiempo real y análisis avanzado de datos.

La plataforma de adquisición de datos propuesta y descrita a continuación es una particularización de la arquitectura general descrita en el Capítulo 2.

4.3.1.1 Estructura de la plataforma para adquisición de datos

Las configuraciones lógicas y físicas de interés se representan en Figura 4.6 y 4.7:

- Sistema autónomo de adquisición de datos (Figura 4.6, izquierda)
- Sistema empotrado de adquisición de datos (Figura 4.6, derecha)
- Sistema distribuido de adquisición de datos (Figura 4.7)

Como se puede apreciar en la Figura 4.6 (izquierda), un sistema autónomo no está pensado para integrarse o comunicar con ningún otro sistema y cuenta con control y sistema de información local. Esta topología es extremadamente sencilla y, sin embargo, bastante popular en arquitecturas reales.

En el contexto de la arquitectura general (Capítulo 2), la versión de la arquitectura como sistema autónomo asume las funcionalidades de estación base y de Gateway operativo en el mismo actor. Más concretamente, este actor se compone de cuatro módulos funcionales distintos:

- El *Data Collector* que gestiona el flujo de datos provenientes de la red de sensores.
- El *Local Engine* implementa las funcionalidades de procesamiento local de datos. Recibe en entrada el flujo de datos procedentes de los nodos sensores y reorganizan la información de acuerdo con el modelo de representación local.
- Finalmente el *Gateway* implementa un conjunto de funcionalidades que permiten a los sistemas de nivel de usuario de acceder a la información y a las funcionalidades de las redes de sensores.
- Módulos de nivel de aplicación. Son dos y corresponden a dos diferentes roles o clases de usuario: la interfaz de monitorización y control (Figura 4.8) y la interfaz de usuario. La primera interfaz garantiza el acceso a todas las funcionalidades de administración de la red; la segunda interfaz asegura acceso a los datos de tiempo real y consultas sobre el sistema de información local.

La versión de la arquitectura como sistema empotrado (Figura 4.6, derecha) es conceptualmente parecida a la correspondiente para sistemas autónomos. Sin embargo, el nivel de aplicación no está predefinido como en el caso de la versión para sistemas autónomos.

Como se puede apreciar en la figura, todas las funcionalidades tanto de gestión como de consulta de información son accesibles para sistemas que interactúan con la arquitectura propuesta a través del *Gateway* que, por lo tanto, asume el rol de *Enabler* (tal como descrito en el Capítulo 2).

A pesar de estos parecidos, existe una diferencia fundamental entre las dos arquitecturas: la versión para sistema empotrado interactúa con al menos un sistema externo desconocido a priori. Este aspecto implica que el sistema trabaja en el contexto de una organización virtual más compleja.

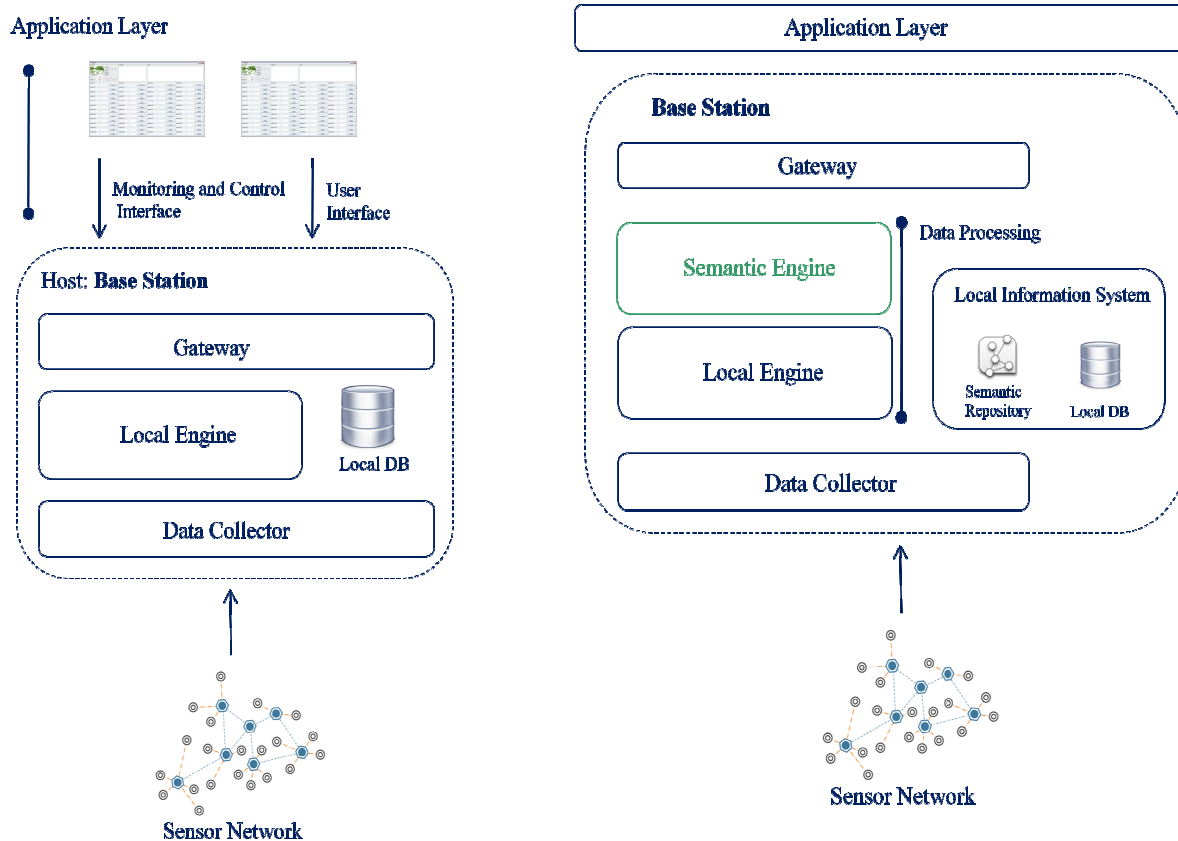


Figura 4.6. Sistema autónomo de adquisición de datos (izquierda) y sistema empotrado de adquisición de datos (derecha).

La solución propuesta integra, respecto a la versión para sistema autónomo, el ambiente semántico para procesado de eventos descrito en la sección 2.6.2.2. Este ambiente proporciona características avanzadas adicionales:

- El sistema de información convencional se extiende con un repositorio semántico. Tanto la información como el conocimiento relacionado puede definirse con un modelo ontológico de referencia. Concretamente se utiliza la ontología de proceso descrita en el Capítulo 2.
- La presencia de modelos semánticos permite un modelo de interacción avanzado en un contexto de interoperabilidad semántica.
- El ambiente semántico para procesado de datos proporciona capacidad extendidas tanto de representación de la información como del análisis de la misma generando modelos de conocimientos extendidos.

Al contrario, la versión a sistema distribuido (Figura 4.7) está pensada para contextos aplicativos intrínsecamente más complejos que pueden incluir control remoto, observaciones distribuidas en múltiples aéreas remotas, acceso y control vía internet, integración e interacción con sistemas complejos. El esquema referencia, como se puede ver en la Figura 4.7, es próximo a la versión general de la arquitectura. No se define ningún componente de nivel de aplicación, por lo cual la entidad más abstracta es el *Gateway/Enabler*. Esto implica, en caso de integración de mecanismos semánticos, que

tanto el repositorio semántico como el procesado de datos a nivel semántico se consideren a nivel de *Gateway*.

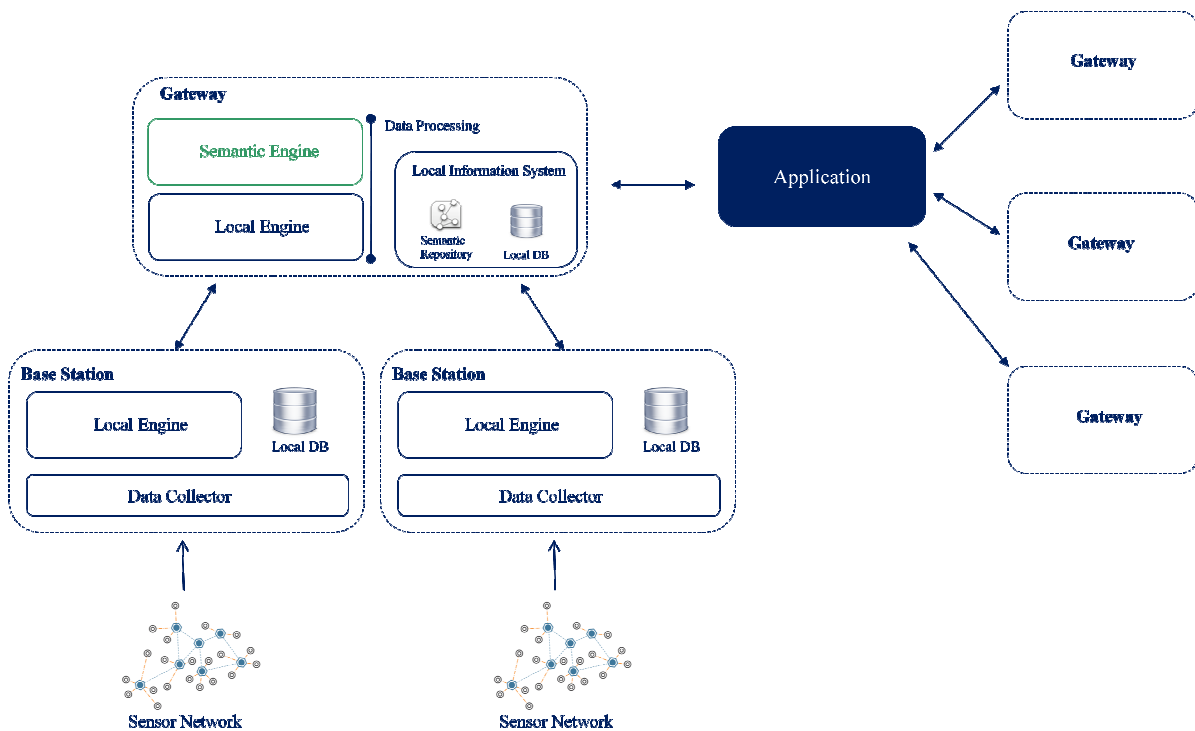


Figura 4.7. Sistema distribuido de adquisición de datos.

La Figura 4.8 representa la interfaz de monitorización y control local implementada en JAVA a través de la cual una red de sensores locales puede ser configurada y monitorizada. El sistema de información de soporte es una base de datos relacional implementada en MySQL [Mys1].

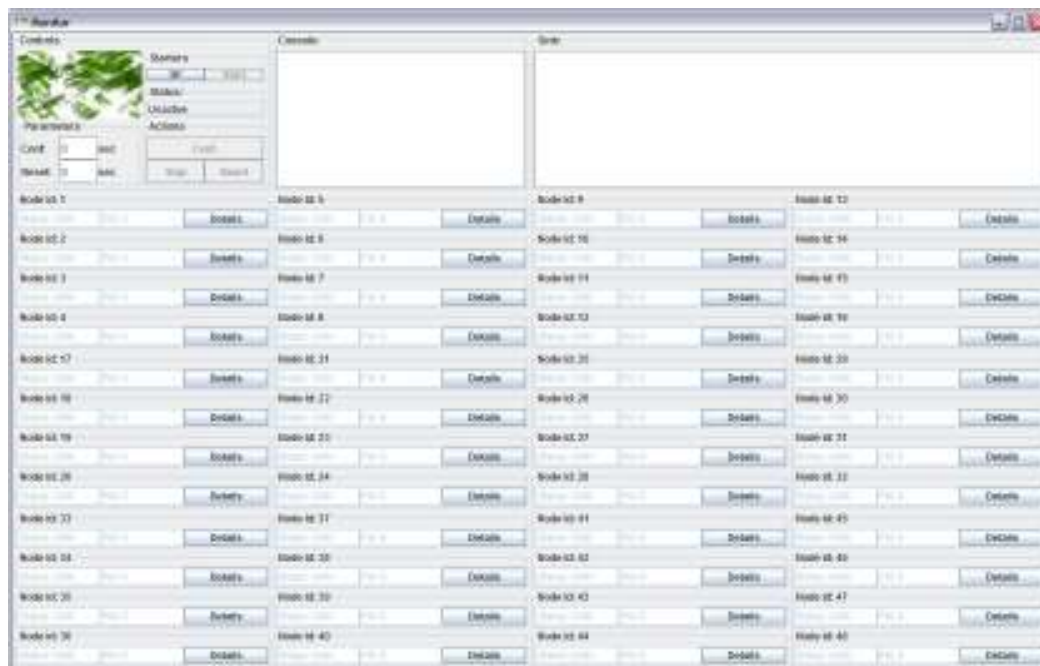


Figura 4.8. Interfaz para monitorización y control local.

4.3.1.2 *Mecanismos avanzados*

Un análisis funcional de la arquitectura propuesta permite distinguir las clases de servicios detalladas a continuación:

- *Servicio de suscripción* (solo en sistemas distribuidos). Este mecanismo añade flexibilidad y dinamismo a la arquitectura en cuanto todos los componentes pueden suscribirse en el entorno lógico correspondiente: los nodos sensores se suscriben en la estación base asociada, las estaciones base se suscriben en el *Gateway* correspondiente y, finalmente, cada *Gateway* se suscribe a la entidad de nivel de aplicación correspondiente. En el caso del *Gateway* se admite suscripción múltiple. La consistencia y la fiabilidad de la información está garantizada por un mecanismo de refresco periódico.
- *Gestor de datos*. Los datos procedentes de los nodos sensores se procesan y gestionan en los diferentes niveles de la arquitectura. El sistema de Gestión de Datos trata tanto el flujo de datos en tiempo real como el almacenamiento de los mismos en las bases de datos que componen el sistema de información.
- *Gestor de eventos*. Los eventos detectables están asociados a umbrales relacionados con los datos procedentes de los nodos sensores. Los eventos se dividen en dos clases: eventos asociados al sistema y eventos de nivel aplicación. La primera clase define *warnings* y alarmas a nivel de administración de sistema (por ejemplo eventos asociados al estado de la batería de los componentes); la segunda incluye eventos asociados con los datos de nivel de aplicación (por ejemplo un dato ambiental crítico). En caso de soporte semántico (Capítulo 2), la relación entre datos y eventos, así como entre eventos y acciones correspondientes, puede definirse de forma dinámica particularizando la ontología de proceso.
- *Monitorización y Control*. Implementa todas las funcionalidades de monitorización y control tanto a nivel local como global.
- *Procesado de datos*. Puede ser estático y asociado a priori a los tipos de datos que están asociados a la aplicación soportada o, bien, integrar el ambiente semántico para procesado de datos que puede añadir capacidades de análisis dinámicas y extendidas (particularizando la ontología de proceso).
- *Sistema de consulta*. Mecanismo de nivel aplicación que soporta el usuario final en la consulta de datos. Puede actuar tanto utilizando un modelo de datos convencional (base de datos relacional con consulta en SQL en este caso) como utilizando un modelo semántico (consulta en RDQL [Rdq1] o SPARQL [Spa1]).

4.3.1.3 *Red de sensores*

La arquitectura de protocolos empleada por la red inalámbrica de sensores se detalla a continuación con referencia el modelo propuesto en el Capítulo 2 y la implementación descrita en la primera parte de este capítulo:

- Las funcionalidades de enlace de datos son las implementadas por *RandomSlotTDMA* (descrito anteriormente en el capítulo). Se ha elegido esta solución en cuanto la arquitectura no trabaja con topologías predefinidas.
- El protocolo de enrutamiento es *RandomRouting* (descrito anteriormente en el capítulo). Esto se debe a las características extremadamente regulares de la

aplicación soportada; esta misma regularidad se refleja en los aspectos de comunicación.

- Protocolo de configuración *DynamicConfigurationProtocol* (anteriormente descrito). Proporciona un cierto dinamismo a frente de condiciones ambientales o topológicas variables.

El hardware utilizado a lo largo de las diferentes pruebas se resume a continuación:

- Componentes inteligentes (Estación base y *Gateway*): se han utilizado PC convencionales en ambiente Linux o Windows.
- Microprocesadores/Transmisores: dispositivos *MicaZ* (Figura 4.9, izquierda) y su versión mejorada *IRIS* (Figura 4.9, derecha).
- Dispositivos para conexión con estaciones base (*Gateway* físico, Figura 4.10): *MIB520*, *MIB600* y *MIB510*, respectivamente con interfaz USB, Ethernet y serie.
- Tarjetas de adquisición de datos (Figura 4.11): *MDA300* para adquisición de datos genéricos; *MTS101* que monta sensores de temperatura, luz, ruido, acelerómetros, magnetómetros; finalmente *MTS420* que proporciona sensores de alta precisión para detectar intensidad de luz, temperatura, humedad, presión barométrica, además de acelerómetros y dispositivo GPS.



Figura 4.9. Dispositivos *MicaZ* (izquierda) e *IRIS* (derecha).



Figura 4.10. Dispositivos para conexión con estaciones base: *MIB520* (izquierda), *MIB600* (centro) y *MIB510* (derecha).



Figura 4.11. Tarjetas de adquisición de datos: *MTS101* (izquierda), *MTS420* (centro) y *MDA300* (derecha).

4.3.2 *Multi-Mode WSN*

Multi-Mode WSN (MM-WSN) es una solución avanzada, robusta y flexible que aplica el concepto (oportunamente revisado) de *Always Best Connected (ABC) Network* [Abc1] a redes inalámbrica de sensores. El concepto de *ABC* suele aplicarse y ser propio de los dispositivos móviles de última generación.

Entornos de trabajos hostiles desde un punto de vista de las comunicaciones, caracterizados por condiciones impredecibles a priori (obstáculos e interferencias temporales o, incluso, permanentes en unos casos), pueden negativamente afectar, tal vez de forma determinante, las prestaciones de las redes. Además, despliegues aleatorios de los nodos pueden generar topologías no siempre óptimas.

Bajo la hipótesis de dispositivos sensores dotados de múltiples capacidades de comunicaciones (*Multi-mode Wireless Sensor Node*), la idea a la base de *MM-WSN* es la capacidad de estos recursos complejos de auto configurarse, de forma inteligente, para optimizar sus recursos en dependencia de las condiciones detectadas.

Evidentemente esta solución asume la utilización de dispositivos intrínsecamente más complejos de los habituales, tanto desde un punto de vista hardware como desde el punto de vista de la plataforma software de gestión.

Como se evidenciará en las partes siguientes de la sección, la *MM-WSN* no se considera una solución generalizada para redes de sensores si no una solución apta para redes no densas. En este entorno de trabajo, la utilización de *MM-WSN* puede incrementar, de forma considerable, la calidad de servicio proporcionada por las redes en términos de flexibilidad, prestaciones, robustez y tolerancia de fallos.

Antes de analizar en detalle las características de la plataforma hardware-software desarrollada en el contexto de la tesis, se propone un breve análisis de las posibilidad concretas de aplicación del concepto de *ABC* en redes inalámbricas de sensores. Dicho análisis quiere poner en clara evidencia las posibles ventajas que se pueden aportar en relación con las actuales limitaciones tecnológicas.

4.3.2.1 *AC y ABC en Redes inalámbricas de Sensores*

Como descrito en el Capítulo 1 de estado del arte, las plataformas para nodos sensores inalámbricos se caracterizan por proporcionar capacidades avanzadas en un contexto de recursos limitados [Aky1].

Dichas plataformas se desarrollan de acuerdo con una ingeniería de base explícitamente diseñada para soportar eficientemente el trabajo en red y la comunicación y generación impulsiva de datos. Este enfoque asegura altas prestaciones en términos de consumo gracias a la capacidad de los nodos inalámbricos de quedarse en estado latente (caracterizado por un consumo mínimo) durante los periodos de inactividad.

Considerando el radio de comunicación limitado y para asegurar grandes áreas de cobertura mediante un alto número de puntos de observación, se asume un entorno de comunicación multi-salto con tecnología orientada a soportar óptimamente comunicación impulsiva de pequeñas cantidades de datos [Bar1].

La última generación de dispositivos comerciales proporciona una solución flexible, efectiva y eficiente tanto para aplicaciones que requieren adquisición de datos como para

aplicaciones reactivas. Además, la integración de dispositivos GPS permite de asociar directamente los datos generados a posiciones geográficas concretas.

Esta eficiencia a nivel de hardware se refleja en la ingeniería de las arquitecturas de protocolos que implementan mecanismos complejos en un contexto de optimización de recursos.

Un número creciente de arquitecturas diseñadas para el mundo real no asumen una topología fija o predefinida si no despliegue aleatorio. El despliegue aleatorio óptimo es una condición imprescindible para asegurar la eficiencia y, en algunos casos, incluso la efectividad de las redes. Despliegues inadecuados suelen limitar el área de cobertura y/o el tiempo de vida de la red. Una cierta densidad de nodos puede ser indispensable para garantizar cierta calidad de servicio. Una descripción exhaustiva de los temas de interés relacionados con el despliegue aleatorio se considera fuera de los objetivos de la presente tesis.

En la Figura 4.12 se representan dos ejemplos de despliegue aleatorio del mismo número de nodos en la misma área. A la derecha se puede apreciar un despliegue relativamente eficiente; sin embargo, la topología de la izquierda se puede considerar muy poco efectiva a causa de la escasa densidad de nodos en proximidad de la estación base. A causa de la intrínseca centralidad lógica de la base (que recibe las observaciones generadas por todos los nodos sensores), la topología de la derecha asegura un tiempo de vida y, posiblemente, un nivel de viabilidad bastante mayor.

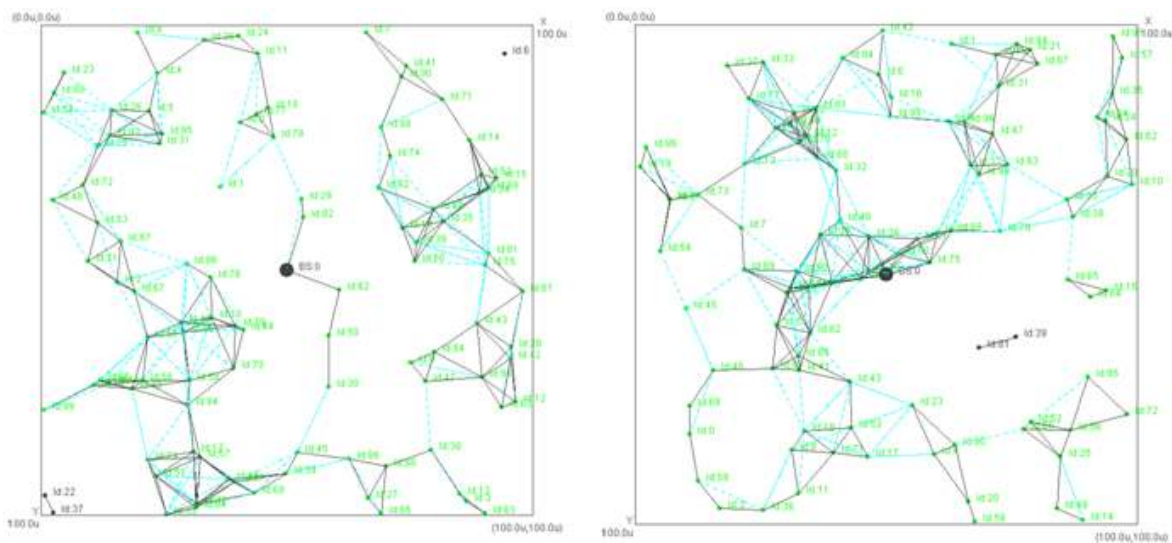


Figura 4.12. Dos ejemplos de despliegue aleatorio con diferente grado de conectividad respecto a la estación base (Figura original de [Con9]).

Las redes inalámbricas de sensores trabajan en el mundo real en presencia de condiciones ambientales cambiantes, muchas veces irregulares e impredecibles. La consecuencia más directa de entornos hostiles en las comunicaciones es la reducción del rango de los enlaces de comunicación y de la fiabilidad de los mismos. Los despliegues aleatorios pueden ser afectados de forma considerable y resultar menos eficientes en entornos hostiles.

La intervención humana es en general muy costosa, en unos casos poco práctica y en otros incluso imposible. De hecho los mecanismos de auto gestión dinámica de las redes [Soh1] son uno de los requisitos comerciales más comunes.

La aplicación del concepto (oportunamente contextualizado) de *Always Best Connected (ABC)* [Abc1] a redes inalámbricas de sensores podría incrementar de forma importante las prestaciones en despliegues aleatorios de redes a baja densidad. El concepto clave de una arquitectura *ABC* es la capacidad de dispositivos multi-modo (dispositivos con múltiples capacidades de comunicación) de auto-configurarse de forma óptima, en función de las condiciones detectadas.

El modelo *ABC* difiere claramente del más simple concepto de *Always Connected (AC)* por su claro enfoque a la optimización de los recursos: *AC* tan solo asume que el dispositivo se mantenga siempre conectado a través de uno de los modos de comunicación disponibles; *ABC* extiende *AC* en cuanto busca la mejor opción entre las disponibles de acuerdo con una cierta política de preferencia.

Aplicaciones muy complejas pueden, incluso, asumir la utilización simultánea de más que un modo de comunicación [Jou1].

Para que el modelo *ABC* sea aplicable al entorno de las redes inalámbricas de sensores se requiere:

- Un nodo sensor inalámbrico multi-modo. La ingeniería de base del nodo, anteriormente descrita, no debe alterarse. En caso contrario el dispositivo sería poco funcional en arquitecturas reales.
- Una capa de gestión interna de los diferentes modos de comunicación.
- Una capa de inteligencia añadida que permita configurar el nodo para trabajar con el modo de comunicación más apropiado en función de las condiciones detectadas.

En Figura 4.13 se representan dos posibles escenarios (a y b) en los cuales, a causa respectivamente del fallo de un nodo (o agotamiento de la batería del mismo) o de un despliegue ineficiente, una parte importante de la red, aunque en pleno funcionamiento, se encuentra aislada sin la posibilidad de conectarse a la red. El escenario c representa un posible intento de reconfiguración de un dispositivo multi-modo.

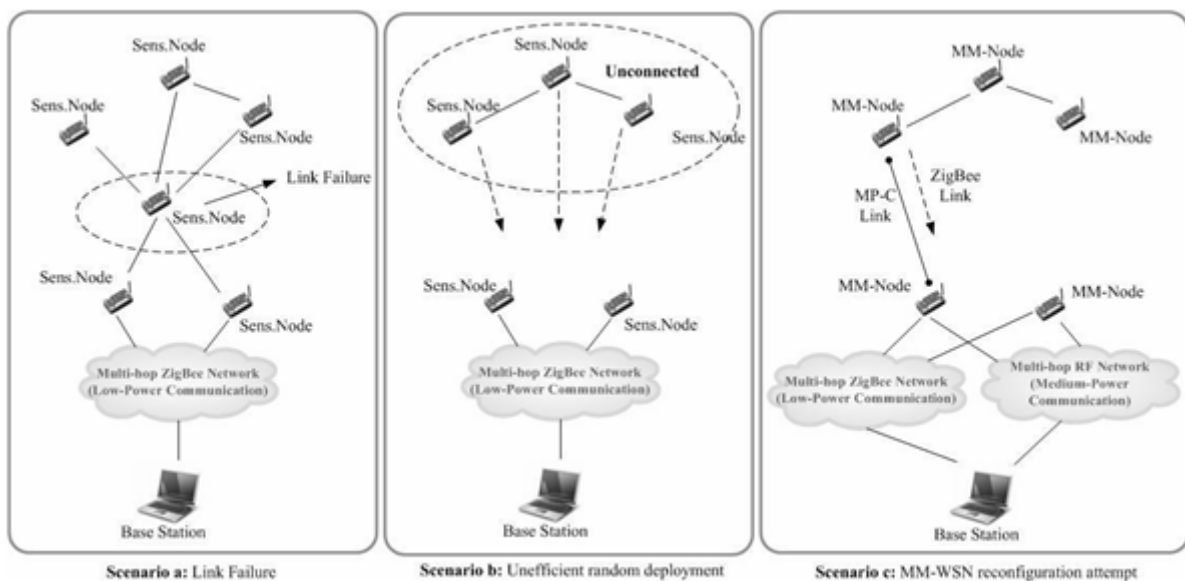


Figura 4.13. Fallo de un enlace (Escenario a); despliegue aleatorio ineficiente (Escenario b); intento de reconfiguración de un modo multi-modo inteligente (Escenario c). Figura original de [Con9].

Las soluciones propuestas por cada uno de estos temas se analizan a continuación.

Considerando la popularidad de los despliegues aleatorios de redes inalámbricas de sensores en entornos reales y la importancia, a nivel comercial, de los aspectos relacionados con la calidad de servicio en términos de eficiencia y robustez, un modelo de arquitectura basado en ABC puede constituir una solución altamente avanzada e innovadora.

4.3.2.2 Multi-Mode Wireless Sensor Node

La solución multi-modo propuesta difiere conceptualmente respecto a los modelos de dispositivo adoptados tanto en el campo específico de las redes de sensores como en el entorno de los dispositivos móviles. Esto es porque *Multimode Wireless Sensor Node (MM-Node)* está compuesto por dos componentes activos (microprocesadores) independientes y potencialmente cooperantes. Cada uno de los componentes activos controla sus propios recursos en términos de memorias, transductores y comunicación.

La idea fundamental es que los dos componentes activos tengan capacidades parecidas respecto a recursos de cálculo y memorias; al contrario las capacidades de comunicación sean distintas relativamente a tecnología de base, potencia de transmisión y/o frecuencia. Evidentemente, más que un único dispositivo se está definiendo una clase de dispositivos. En el contexto del presente trabajo, interesan modos de comunicación que aseguren radios de comunicación distintos: uno de baja potencia de transmisión (y radio limitado) y uno con mayor potencia (y rango).

La presencia de dos microprocesadores separadamente programados que implementan comportamientos y funcionalidades potencialmente independientes y la capacidad de los mismos de intercambiar información (cooperación) proporcionan una plataforma compleja que abre interesantes perspectivas en el campo de la computación paralela, de la inteligencia y de la réplica de información y/o recursos.

Aunque esta sección está enfocada a aspectos relacionados con las comunicaciones entre nodos, la solución propuesta es mucho más amplia y abre interesantes perspectivas respecto a otros parámetros típicos de calidad de servicio como la tolerancia de fallos tanto en su fase de detección como de gestión. Tanto la detección como la gestión de fallos, en un contexto de red convencional, es un problema que se aborda a nivel de red: el fallo de un nodo sensor tiene que ser detectado por la red circunstante y eventuales mecanismos de gestión pueden diseñarse de acuerdo con enfoques centralizados o distribuidos.

Asumiendo que los dos dispositivos básicos que componen el nodo sean cada uno la réplica del otro tanto desde el punto de vista software (funcionalidades) que hardware (transductores), el nodo podría proporcionar funcionalidades de tolerancia a fallos extendidas: en su modo de funcionamiento normal, solo uno de los dos dispositivos de base está activo respecto a la red; el otro dispositivo de base comprueba periódicamente el correcto funcionamiento de su asociado; si detecta una situación de fallo, sustituye el dispositivo que no funciona pasando en estado activo.

Evidentemente la tolerancia de fallos es un problema que tiene que ser abordado para casos concretos (fallos parciales, totales, etc.).

La plataforma propuesta, como se describirá con más detalle en las secciones siguientes, proporciona funcionalidades avanzadas que permiten a cada componente activo de acceder, de forma transparente, a las funcionalidades de su componente gemelo. Esta flexibilidad permite el diseño de comportamientos complejos caracterizados por un notable nivel de inteligencia.

La arquitectura de MM-Node (Figura 4.14) es conceptualmente más compleja que cualquier arquitectura para nodos sensores tradicionales.

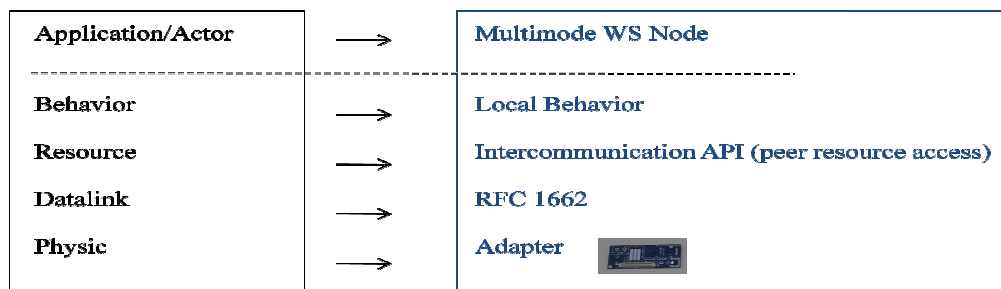


Figura 4.14. Arquitectura de MM-Node.

El nivel físico (*Physic*) está representado por un componente hardware (*Adapter*) que proporciona la conexión física entre los dispositivos. El nivel de enlace de datos (*Datalink*) define un protocolo de nivel básico para la comunicación entre los componentes (se ha utilizado el estándar RFC 1662). El nivel de recurso (*Resource*) proporciona una interfaz de acceso a las funcionalidades del componente activo; se construye directamente sobre el nivel de enlace de datos del cual es una abstracción. El nivel de *Bahaviour* o de comportamiento implementa funcionalidades a nivel de nodo. Finalmente el nivel de aplicación es la capa más alta de la arquitectura.

4.3.2.3 La plataforma hardware

La plataforma hardware utilizada (Figura 4.15, izquierda) se describe a continuación. El componente activo principal (*Host*) proporciona el modo de comunicación de baja potencia además de los típicos recursos de un nodo sensor inalámbricos en términos de capacidad de procesamiento y almacenamientos. Es un dispositivo MicaZ.

El segundo componente (*R-device* o *Coordinator*, Figura 4.15 en bajo a la derecha) tiene características similares al componente principal en términos de recursos de cálculo y almacenamiento pero un modo de comunicación con alcance notablemente superior.

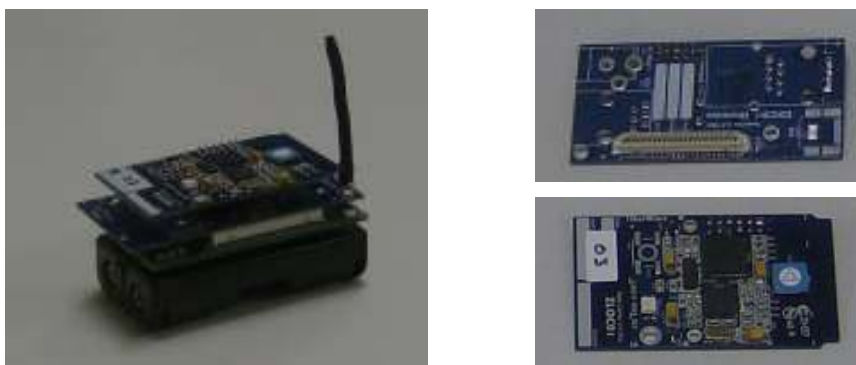


Figura 4.15. Componentes hardware de MM-Node (a la izquierda). *Adapter* (en alto a la derecha) y *R-device* (en bajo a la derecha).

Los dos dispositivos comunican físicamente a través de un componente pasivo de soporte (*Adapter*, Figura 4.15 en alto a la derecha).

El dispositivo supone un modelo de gestión de recursos diferente respecto al correspondiente para dispositivos convencionales. Como se puede apreciar en Figura 4.16, hay más que un estado que implica consumos significativos. El estado latente (*Sleep*) implica ambos dispositivos en estado latente. Si ambos dispositivos no están en estado latente, el consumo es significativo. Se distingue en el estado *Full Operations* (ambos componentes operativos) y dos estados (*Host_op* y *Coordinator_op*) que asumen, respectivamente, el dispositivo *Host* activo y el dispositivo *R-device* operativos.

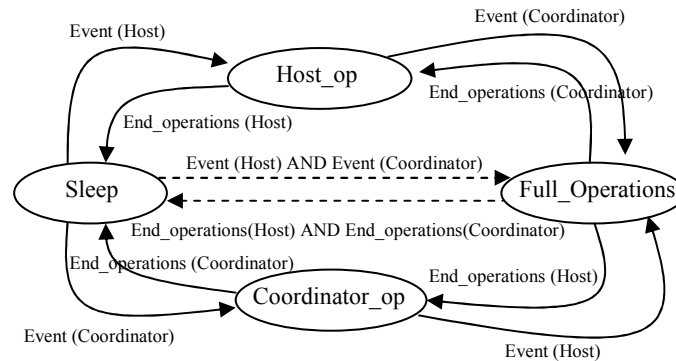


Figura 4.16. Modelo teórico de gestión de recursos a eventos de MM-Node (Figura original de [Con8]).

Con referencia a la Figura 4.16, las transiciones directas de estado entre estado *Sleep* y estado *Full_Operations* son puramente teóricas en cuanto las actividades de los dispositivos son completamente independientes entre sí.

4.3.2.4 La interfaz de programación

Los dos componentes activos que caracterizan *MM-Node* son independientes el uno del otro bajo todos puntos de vista en cuanto se programan separadamente. De todos modos, la idea a la base asume que los dos dispositivos cooperantes de acuerdo con un esquema master-slave o peer-to-peer para implementar comportamientos complejos de los nodos. Desde un punto de vista funcional, cada dispositivo implementa su propio comportamiento; sin embargo, la definición de comportamientos complejos de los nodos a través de la cooperación entre los dispositivos es el tema de mayor interés.

A la luz de estas consideración, en el presente trabajo se ha considerado tema primordial facilitar una interfaz avanzada que permita la definición e implementación de tareas cooperativas con alto nivel de abstracción.

Los dos componentes activos de la plataforma comunican físicamente a través del *Adapter* (interfaz física) utilizando un protocolo serie de enlace de datos (RCF 1662, Tabla 4.1). El nivel de enlace de datos habilita el intercambio de mensajes entre dispositivos. El tamaño de dichos mensajes, en entorno TinyOS, está limitado a 36 bytes.

Si la interfaz de comunicación entre los dispositivos se limitara a este nivel tan básico (también denominado nivel de *Packet/Frame*), la definición y la implementación de mecanismos cooperativos complejos sería ardua y posiblemente poco efectiva.

TABLE I
RFC 1662 STANDARD RESUME

Byte#	Field	Description/Value
0	Synchronization byte	Always 0x7E
1	Packet type	There are 5 admitted values (types): <ul style="list-style-type: none"> • P_PACKET_NO_ACK (0x42): packet with no ACK required • P_PACKET_ACK (0x41): packet with ACK required; it includes a prefix byte; receiver has to send a P_ACK response with prefix byte as contents. • P_ACK (0x40): ACK response that includes prefix byte. • P_UNKNOWN (0xFF): Unknown or generic packet.
2...n-1	Payload data	Data message
n	Synchronization byte	Always 0x7E

Tabla 4.1. RCF 1662 (Tabla original de [Con8]).

Para facilitar el diseño y el desarrollo de comportamientos cooperativos complejos se ha proporcionado una API (Tabla 4.2) que implementa un conjunto extensible de funcionalidades de alto nivel que, a través de mensajes activos entre los componentes, asegura el acceso a los recursos del componente activo asociado en un contexto de abstracción.

TABLE II
API SPECIFICATION

Code	Function
Generic	
0	uint8_t test (Resource resource)
1	uint8_t store (int8_t[] data, uint8_t id)
2	int8_t[] getStored (uint8_t id)
3	uint8_t deleteStored (uint8_t id)
4	void setPolicy (Policy policy)
5	Policy getCurrPolicy ()
Communication Resource	
6	uint8_t send (TOS_Msg msg)
7	uint8_t notifyMsg (TOS_Msg msg)
Sensing Resource	
8	uint16_t getADC (uint8_t adc)

Tabla 4.2. API para el acceso a las funcionalidades del componente activo gemelo en *MM-Node* (Tabla original de [Con9]).

En otras palabras, la API propuesta proporciona a cada componente activo una visión virtual de su componente gemelo. Dicha API se implementa directamente sobre el nivel de enlace de datos (o *Packet/Frame*) como módulo TinyOS y proporciona tanto el soporte para interacción genérica entre los componentes como el soporte para operaciones específicas.

Este conjunto de funcionalidades se ha diseñado e implementado de acuerdo con los más comunes principios de la ingeniería del software: la interfaz de comunicación es bien distinta de su implementación; la interfaz define los mensajes activos que los dispositivos pueden intercambiar; a la interfaz está asociado un motor que actúa en *background* que gestiona el evento disparado para la recepción de un mensaje activo.

Este modelo de computación se considera especialmente flexible tanto relativamente a la utilización de la API (el programador no tiene necesariamente conocer las características de la API si no simplemente su interfaz) como relativamente a su extensión (nuevos eventos o funcionalidades implican una extensión de la interfaz y la asociación del disparador correspondiente en el motor de background).

La API, en su versión actual, está explícitamente diseñada para soportar el desarrollo de mecanismos cooperativos complejos a nivel de nodo o red, así como funcionalidades avanzadas para detección y gestión de fallos locales.

La interfaz propuesta en la Tabla 4.2 propone un conjunto de métodos concretos que no dependen del hardware concretamente utilizado y de un conjunto de métodos virtuales. Por ejemplo, un componente puede comprobar que su gemelo está funcionando enviando un mensaje activo de test (con el código correspondiente). En este caso no hay necesidad de utilizar un driver concreto o ninguna funcionalidad de bajo nivel que caracteriza un dispositivo en concreto. Otros métodos se han definido como funciones abstractas en cuanto tienen que ser concretizados en el contexto de plataformas hardware reales, normalmente mapeando recursos lógicos con los correspondientes recursos físicos. Es el caso de los transductores que se acceden con la función genérica *getADC*. El código en ingreso (*adc*) tiene que asociarse a un transductor concreto.

Los recursos de almacenamiento también son accesibles por el dispositivo asociado a través de un sistema de direccionamiento basado en claves. También se pueden utilizar estructuras de gestión predefinidas (pilas, colas, etc.) cada una caracterizada por su modo de gestión (*First in First out, Last in First out, etc.*).

Las fases de debug, validación y prueba de la API propuesta son muy difíciles de efectuar directamente en dispositivos multi-modo. Dichas pruebas podrían haberse efectuado con el auxilio de simuladores convencionales de TinyOS (TOSSIM [Tos1][Asi1] por ejemplo) que pero proporcionan un entorno muy limitado.

Para poder efectuar pruebas complejas en entornos próximos a los reales se ha desarrollado un emulador en tecnología JAVA que se ejecuta en un PC y actúa como uno de los componentes activos. Este actor emulado comunica con el otro actor, efectivamente ejecutado en un nodo sensor, a través de un Gateway físico (Figura 4.17). Esta aproximación permite la interacción entre un actor real y uno emulado.



Figura 4.17. Emulador para prueba y validación de la API desarrollada.

4.3.2.5 La plataforma software

La plataforma software proporciona mecanismos y protocolos que habilitan el *MM-Node* para trabajar en un contexto de red (funcionalidades externas), al contrario de la interfaz de comunicación que implementa funcionalidades a nivel de nodo (internas).

Adaptive Communication Protocol (ACP) se considera una extensión de las arquitecturas de

protocolos convencionales que proporciona la capacidad de trabajar en red como nodo multi-modo. Fundamentalmente permite detectar las condiciones externas, evaluarlas y auto-configurarse en consecuencia.

ACP asume nodos multi-modo: comunicación de bajo alcance (*LM-Communication*) y de largo alcance (*HM-Communication*). El primer modo se considera el modo convencional de comunicación de los nodos sensores inalámbricos: bajo consumo y baja potencia de transmisión que, en casos reales, suelen asociarse a rangos de comunicación entre 20 y 40 metros. El modo de largo alcance simplemente asume un rango de comunicación significativamente más elevado a coste de mayores consumos energéticos.

Evidentemente, desde un punto de vista de los consumos, *LM-Communication* es el modo de comunicación preferible para todos los nodos de la red.

Respecto a la red circunstante, el nodo puede asumir diferentes estados:

- *Unconfigured*: el nodo esta desconectado.
- *LM-Configured*: el nodo está conectado a la red a través del modo de comunicación *LM-Communication*.
- *HM-Configured*: el nodo está conectado a la red a través de su modo de comunicación *HM-Communication*.

ACP (Figura 4.18) detecta la imposibilidad de conectar el nodo a la red a través del modo de comunicación preferido (*LM-Communication*) e intenta descubrir una ruta de comunicación válida para conectarse a la red utilizando el modo de comunicación a largo alcance (*HM-Communication*). Considerando el mayor rango de comunicación, su probabilidad de éxito aumenta notablemente.

Para optimizar los recursos e implementar efectivamente una estrategia de gestión *ABC*, un nodo en estado *HM-Configured* que descubre una ruta para conectarse a través del modo de comunicación preferido cambia su estado en *LM-Configured*.

El mayor punto de fuerza de *ACP* es que no depende de ningún protocolo de enrutamiento concreto, así como se considera independiente de cualquier otro mecanismo de configuración y gestión.

Su mayor desventaja es la necesidad de estructuras locales más complejas. Por ejemplo, la tabla de vecinos, típicas en la mayoría de los protocolos de enrutamiento, tiene que asociarse a cada uno de los modos de comunicación. Es decir que cada modo de comunicación tiene su propia tabla de vecinos.

Un análisis más amplio y conceptual propone el enfoque distribuido con sus puntos fuertes y débiles: por una parte, limitar el entorno de conocimiento del nodo a sus vecinos (enfoque distribuido) asegura mecanismos ligeros y eficientes; por la otra, dicho enfoque, no permite una visión extendida o global de la red por lo cual *ACP* determina su condición de trabajo optima en relación a su entorno de conocimiento limitado; esto no es garantía de optimización de los recursos globales de la red. Un enfoque centralizado de *ACP* añadiría un factor de inteligencia importante con el coste de la necesidad de alto intercambio de mensajes entre los nodos y de algoritmos de cálculo centralizados de complejidad importante. Es un compromiso actualmente en fase de análisis que podría ser objeto de trabajos futuros.

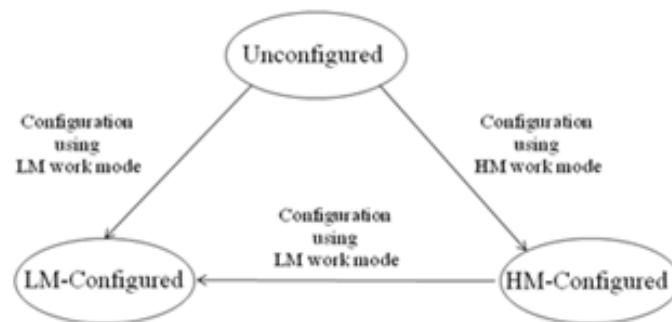


Figura 4.18. Diagrama de actividad simplificado de ACP. Figura original de [Con9].

4.3.2.6 Evaluación preliminar

El prototipo de la arquitectura proporcionada ha sido objeto de varias pruebas tanto relativamente a sus componentes software como a sus componentes hardware. En el contexto de este trabajo la evaluación de la plataforma propuesta se efectúa en un contexto cualitativo en cuanto no se ha efectuado una experimentación en un contexto de red con escala significativa.

Las ventajas introducidas se pueden resumir en los puntos a continuación:

- *MM-Node* puede mejorar las prestaciones de una red de sensores en un contexto de baja densidad de nodos actuando como dispositivo multi-modo.
- También puede emplearse para mejorar la fiabilidad de las arquitecturas proponiendo replica de recursos que permiten la implementación de estrategias de detección y gestión efectiva de fallos locales.
- Facilita la integración y la convergencia entre tecnologías convergentes.
- Permite incrementar de forma significativa la inteligencia de los dispositivos a nivel de red a través de la definición de tareas cooperativas entre los componentes activos.

Estas ventajas deberían considerarse en el contexto de las siguientes consideraciones:

- Se requiere un hardware implícitamente más complejo y costoso caracterizado por consumos mediamente más elevados.
- Se requiere una arquitectura de gestión más compleja.
- La definición y la implementación de tareas cooperativas entre dispositivos asociados es intrínsecamente compleja.

4.3.3 Convergencia entre WSN y tecnología RFID: aplicaciones industriales

La convergencia entre redes inalámbricas de sensores y tecnologías RFID [Jup1][Yan3] es un tema de notable interés, especialmente considerando el gran potencial comercial de arquitecturas integradas. Dicha convergencia puede resultar en el desarrollo de arquitecturas complejas donde la coexistencia de múltiples tecnologías cooperantes puede representar una solución avanzada y competitiva para problemas complejos en diferentes entornos.

La arquitectura propuesta en el contexto del presente trabajo se enfocan principalmente a

entornos industriales, más concretamente al soporte simultáneo de aplicaciones para detección, identificación y localización de objetos, además de monitorización de parámetros ambientales y personales.

Realmente, diferentes temas de investigación distintos (optimización de recursos, comunicación inalámbrica a baja potencia en entornos hostiles, localización indoor, etc.), cada uno caracterizado por sus propios problemas y retos, confluyen en la arquitectura propuesta.

Tanto desde un punto de vista conceptual como de acuerdo con perspectivas más concretas, se pueden identificar los siguientes enfoques para arquitecturas integradas de este tipo:

- *Coexistencia* de múltiples tecnologías, especialmente tecnologías inalámbricas en este caso.
- *Cooperación* entre actores inteligentes que utilizan tecnologías distintas para alcanzar objetivos comunes o desarrollar tareas cooperativas.
- *Competición* entre actores inteligentes que, de acuerdo con un enfoque adaptivo, son capaces de auto-configurar su rol y comportamiento en función de las condiciones y, eventualmente, de las restricciones detectadas. Estos aspectos pueden ser especialmente relevantes, tanto a nivel de modelo como de soluciones reales, en ambientes hostiles.
- *Optimización* de recursos considerando un entorno de trabajo con recursos limitados en algunos casos y un enfoque “green” de los sistemas.

En este contexto, la solución propuesta se ha diseñado para garantizar efectividad y altas prestaciones en entornos con requisitos variables, en términos de robustez y fiabilidad.

La sección, en primer lugar, introduce las tecnologías RFID tanto en su versión con componentes activos [Jup1] como en su versión con componentes pasivos [Yan3]; a continuación se detallan las características de la arquitectura propuesta y su aplicación en entornos industriales; finalmente, en relación con la aplicación para localización indoor, se propone un breve análisis de las pruebas realizadas en un entorno industrial concreto.

4.3.3.1 Tecnología RFID y su convergencia con redes inalámbricas de sensores

Las oportunidades comerciales que interesan las tecnologías RFID, tanto en su versión con dispositivos activos como pasivos, son importantes y están en constante crecimiento.

La aplicación típica de la tecnología RFID pasiva [Yan3] es la detección e identificación de objetos: una etiqueta pasiva puede ser detectada por un lector a una cierta distancia; a distintas etiquetas pueden asociarse códigos distintos para diferenciar los objetos asociados. Un análisis exhaustivo de los modos de funcionamiento de la tecnología RFID pasiva se considera fuera de los objetivos primario de este trabajo.

Durante los últimos años, la tecnología RFID con componentes pasivos (TAG pasivos, Figura 4.19) se ha extendido gracias a sus principales ventajas:

- Los TAG pasivos no requieren ningún tipo de alimentación.
- Los dispositivos pasivos actualmente en el mercado se caracterizan por tamaños reducidos o, incluso, extremadamente reducidos.
- Suelen tener un tiempo de vida largo.

- Se pueden considerar dispositivos de muy bajo coste.
- No necesitan ningún mantenimiento particular.
- Las etiquetas más modernas son bastante robustas.

Por el contrario, el campo de aplicación de la tecnología RFID pasiva actual es parcialmente limitado por los siguientes aspectos:

- Rango de detección reducido (unos metros en el mejor de los casos), menor en algunos casos.
- Modo de trabajo peculiar: un TAG pasivo se detecta y permanece detectado por un tiempo largo también después que el mismo está fuera del alcance del lector. Además un TAG pasivo se detecta múltiples veces en el tiempo de exposición al lector.
- Imposibilidad de montar en un TAG pasivo cualquier elemento hardware activo (un sensor por ejemplo) que requiere alimentación para poder funcionar.



Figura 4.19. Ejemplo de etiquetas pasivas (Figura original de Con[8]).

La tecnología RFID de componentes activos [Jup1] se basa en componentes activos que requieren alimentación; pueden detectarse a grandes distancias y disponen de las capacidades típicas de los elementos activos (por ejemplo pueden iniciar una comunicación) en una red. Evidentemente, la tecnología RFID activa es funcionalmente más avanzada que la correspondiente tecnología pasiva. Sin embargo emplea dispositivos de tamaño superior, es más costosa y se puede comparar, en términos de mantenimiento e comportamiento en red, a un nodo sensor inalámbrico de bajo coste con funcionalidades limitadas. La evolución de la tecnología RFID activa se considera intrínsecamente relacionada a los avances tecnológicos de los diferentes mecanismos de alimentación que, de todos modos, se consideran actualmente más que satisfactorios en la mayoría de las aplicaciones comerciales. Actualmente, respecto al modo de aplicación, se distingue entre dispositivos activos con alimentación propia y alimentación externa.

Una parte considerable de arquitecturas de última generación y de las oportunidades a nivel comercial, así como las perspectivas de próximas generación, parecen relacionadas con la convergencia entre tecnología RFID y redes de sensores [Jup1]. La creatividad de los ingenieros, en este sentido, podría asumir un rol determinante.

La arquitectura propuesta a continuación se coloca naturalmente en el marco de investigación mencionado en cuanto propone la coexistencia y la cooperación entre tecnología RFID (activa y pasiva) y redes inalámbricas de sensores.

En principio, la arquitectura no está diseñada para un entorno concreto, en el sentido tanto las infraestructuras de soporte como su aplicación no están directamente enfocadas o relacionadas con un dominio concreto. Sin embargo, los componentes a medida de la arquitectura han sido desarrollados para funcionar en entornos industriales. También todas las pruebas relacionadas se han efectuado en entornos industriales.

La arquitectura se ha diseñado sin hacer hipótesis alguna sobre la topología de referencia o, bien, sobre la escala o densidad de las redes desplegadas. La arquitectura podría funcionar de acuerdo con topologías muy sencillas para cobertura de áreas limitadas en un contexto de baja densidad o, bien, en un contexto de alta densidad para garantizar alta precisión.

La arquitectura se ha diseñado considerando la imprevisibilidad de las condiciones ambientales respecto a las comunicaciones: ciertos entornos industriales pueden proponer condiciones regulares, sin embargo otros representar entornos especialmente hostiles con interferencias y obstáculos temporales o permanentes.

4.3.3.2 *Arquitectura para identificación y localización de objetos en entornos industriales*

Áreas industriales de gran tamaño implican un gran número de trabajadores especializados empeñados a desarrollar tareas específicas y, en algunos casos, interdependientes. En ciertas fabricas, el entorno puede ser frenético e, incluso, caótico (nivel de ruido importante, transito de maquinas y personas, etc.)

El entorno de referencia en el contexto de este trabajo es el área industrial de AERNOVA (Vitoria) donde se producen piezas de aviones. Como representado en la Figura 4.20, el proceso de producción se organiza en sectores cada uno reservado a ciertas tareas. Unas cuantas de estas áreas de trabajo pueden considerarse fijas, otras pueden variar en el tiempo respecto a posición, dimensión e, incluso, tarea a desarrollar.

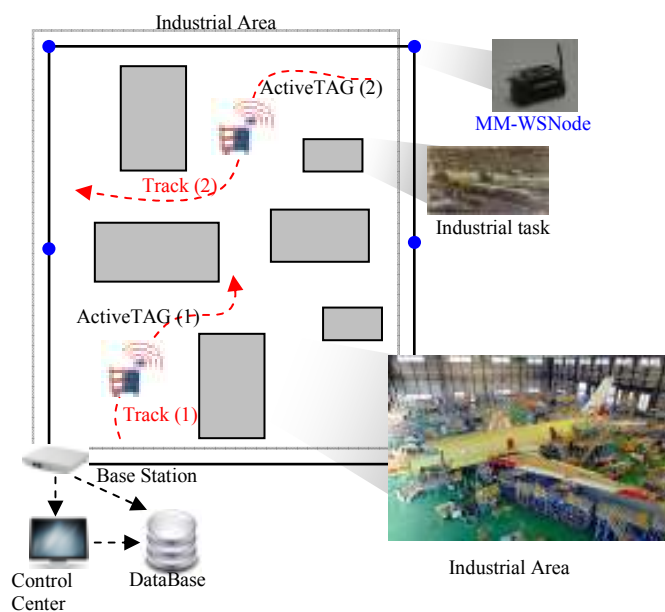


Figura 4.20. Visión esquemática de la área industrial de AERNOVA (Figura original de Con[8]).

El proceso productivo interesa un gran número de objetos de diferente tamaño (muchos de los cuales muy pequeños) que son utilizados por personas diferentes en diferentes áreas de trabajo.

La presencia en el área de trabajo de dichos objetos puede generar los siguientes problemas:

- *Problemas de seguridad e ineficiencia en la fase de control de calidad.* Objetos que por error se queden en algún componente podrían suponer graves riesgos para la seguridad de los productos finales. Por lo tanto la fase de control de calidad suele resultar especialmente importante pero, al mismo tiempo, no demasiado eficiente a falta del soporte tecnológico adecuado.
- *Gran ineficiencia de la reorganización del entorno de trabajo a los cambios de turno así como de los inventarios.* Los objetos de la área de trabajo suelen asociarse a un sector concreto o a algún repositorio. En cada turno los objetos deberían que colocarse en el sitio al que pertenecen. Esta tarea no suele resultar especialmente rápida sin un soporte tecnológico específico, así como los inventarios.

Además, como anteriormente se ha mencionado, el ambiente de trabajo podría necesitar la monitorización de parámetros ambientales o, incluso, personales de los trabajadores para garantizar un ambiente lo suficientemente saludable.

Un soporte tecnológico integrado que sea capaz de monitorizar el ambiente (respecto a los parámetros críticos) y, eventualmente, el personal (por ejemplo a través de sensores biomédicos) y, al mismo tiempo, proporcionar un sistema para detectar, identificar y localizar objetos podría resultar de extrema utilidad en este tipo de área industrial.

La solución propuesta (Figura 4.20) implica la coexistencia de redes de sensores, TAG RFID activos y pasivos. Como se puede apreciar en la Figura 4.20, en la área industrial están unos cuantos carros que suelen pasar gran parte del tiempo parados en proximidades de las áreas de trabajo o moverse lentamente (empujados por trabajadores). Cada carro monta una antena para detección de etiquetas RFID pasivas asociados a los objetos que se pretende detectar e identificar. El detector de etiquetas pasiva se asocia a la interfaz de un TAG inteligente activo capaz siempre asociado al carro en el que está montado. El TAG activo transmite la información detectada por la antena a la red de sensores circunstante. Los nodos sensores están posicionados de forma estratégica en la área industrial; por una parte reciben la información enviada por los TAGs activos y, por la otra, monitorizan el ambiente. La información procedente de los TAG activos se integra con el (*Received Strenght Signal Indicator*) RSSI detectado. De tal forma, el centro de control al cual la red de sensores envía toda la información disponible puede también localizar el carro asociado en el área y, con buena aproximación, los objetos detectados por el mismo carro.

Una visión más detallada de la arquitectura se representa en la Figura 4.21. Como se puede apreciar, los nodos sensores fijos (*Anchor Nodes*) son nodos multi-modo (descritos en la sección anterior). El modo de comunicación de bajo consumo y alcance se reserva para la comunicación entre sensores y, por lo tanto, hacia el centro de control (estación base). El otro modo (a largo alcance) se reserva para las comunicaciones con los dispositivos RFID activos.

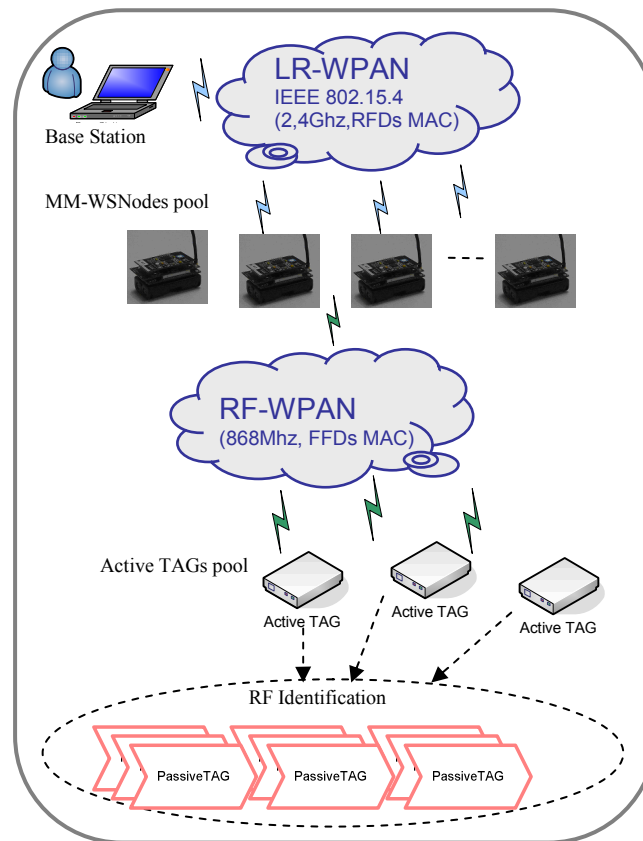


Figura 4.21. Componentes de la arquitectura (Figura original de Con[8]).

El número y la posición concreta de los nodos sensores dependen fundamentalmente del área que se quiere cubrir y del número de puntos de detección del RSSI que el algoritmo de localización necesita para garantizar la precisión deseada. Un algoritmo de localización proporciona cierta elaboración matemática de los RSSI detectados en los puntos prefijados. El algoritmo de localización no ha sido objeto específico de este trabajo.

Evidentemente, la arquitectura propuesta es el resultado de la integración optimizada entre redes de sensores y tecnologías RFID. Un análisis de la arquitectura desde un punto de vista funcional permite distinguir los siguientes módulos o aplicaciones:

- *Monitorización ambiental* (temperatura, humedad, ruido, etc.). Esta implementada por nodos fijos que, por lo tanto, no se consideran críticos en términos de consumo.
- *Sensores personales* (sensores biomédicos por ejemplo). Todos los sensores y dispositivos asociados a personas para monitorizar la actividad o el estado de salud de los trabajadores. Son actores intrínsecamente móviles.
- *Detección Identificación, localización y seguimiento* de los carros y de los objetos marcados con etiquetas pasivas.

La estructura del sistema de captura está representada en la Figura 4.21. Como se puede ver, el sistema por una parte actúa como un red inalámbrica de sensores convencionales (generación de datos y envío de los mismos a la estación base); por la otra constituye la infraestructura de comunicación para todos los datos procedentes de los TAGs activos que se reciben, integran y se envían hacia el centro de control.

El prototipo actual se compone de dispositivos *MM-Node*, un conjunto de lectores RFID capaz de detectar etiquetas pasivas y, finalmente, los TAGs RFID activos que comunican a través de WPAN (RF 868Mhz/FFDs MAC) con los dispositivos *MM-Node*. El otro modo de comunicación de los dispositivos *MM-Node* utiliza tecnología ZigBee (IEEE 802.15.4, 2,4Ghz, RFDs MAC) para proporcionar la infraestructura de comunicación hacia la estación base.

En el caso de integración de sensores personales, estos se comportarían en red de la misma forma que los TAGs activos.

La Figura 4.22 ofrece una visión modular de las diferentes funcionalidades del sistema de detección, identificación y localización en relación con el actor físico que las implementa.

Se pueden distinguir dos capas principales (*Service/Application* y *Local Infrastructure*) de alto nivel. *Service/Application* se compone de las funcionalidades de nivel de servicio y aplicación; *Local Infrastructure* implementa las funcionalidades locales de base. Cada capa se compone de dos diferentes módulos funcionales:

- *Service-level infrastructure*. Soporte, eventualmente Context-aware, para la integración de la arquitectura en un contexto de organización virtual multi-rol.
- *Control System*. Plataforma a soporte de los usuarios para monitorización y control de la área industrial. También proporciona herramientas de administración para configuración y gestión.
- *Information System*. Sistema de información de acuerdo con su definición común.
- *Localization Engine*. Módulo funcional que implementa el algoritmo de localización sobre la base de la información detectada.

Los nodos inalámbricos implementan dos diferentes módulos funcionales:

- *Coordination Module*: se compone de dos diferentes conjuntos funcionales. El primero, *ID-Management*, está implementado por el TAG activo y permite la detección e identificación de las etiquetas pasivas. Esta información se integra con la información asociada al TAG activo receptor y se envía a los dispositivos *MM-Node*. El segundo conjunto de funcionalidades, *RSSI-Management*, está implementado por el *MM-Node* y se preocupa de detectar y gestionar oportunamente el RSSI. Este módulo funcional no implementa funcionalidades de comunicación complejas.
- *Local Communication Infrastructure*: este módulo está implementado integralmente en el *MM-Node* y gestiona la aplicación de nivel de aplicación. Su rol fundamental es garantizar la comunicación fiable de toda la información disponible. Evidentemente su rol en la arquitectura es absolutamente fundamental y un fallo eventual podría inutilizar la arquitectura completa.

Como anteriormente se ha mencionado la topología no es predefinida aunque interesa el algoritmo de cálculo así que, posiblemente, la utilización de una topología organizada de acuerdo con algún tipo de geometría regular sería lo ideal. En la arquitectura de prueba se ha considerado una topología flexible organizada de acuerdo con la composición de celdas básicas (Figura 4.23) siendo el área de trabajo considerada bastante más larga que ancha y con altura significativa. Eventuales constricciones en el despliegue efectivo de los nodos deberían que considerarse. En caso de múltiples estaciones base trabajando simultáneamente, la infraestructura de comunicación puede configurarse en clusters para incrementar la robustez y la eficiencia de las rutas de comunicación.

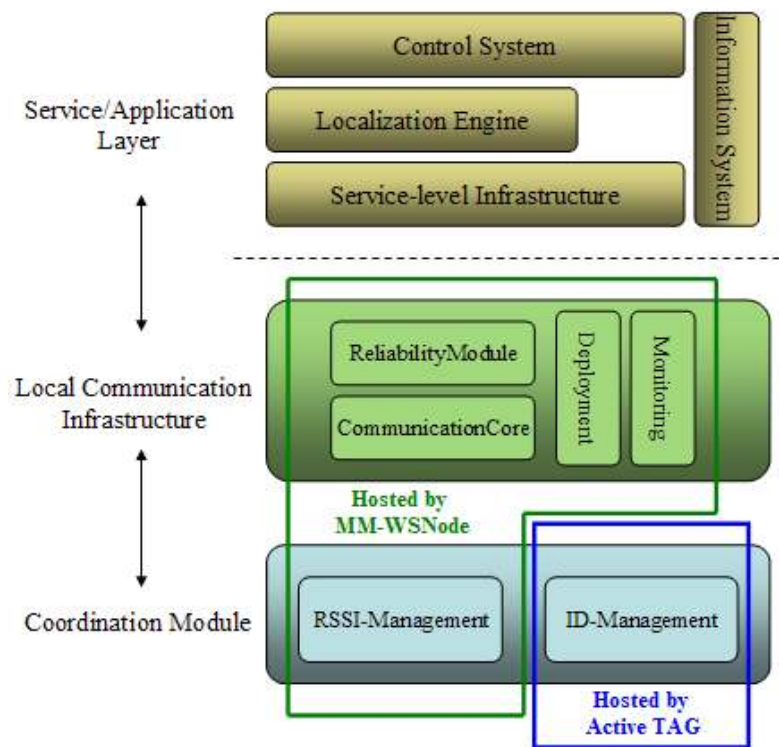


Figura 4.22. Estructura lógica de la arquitectura (Figura original de Con[8]).

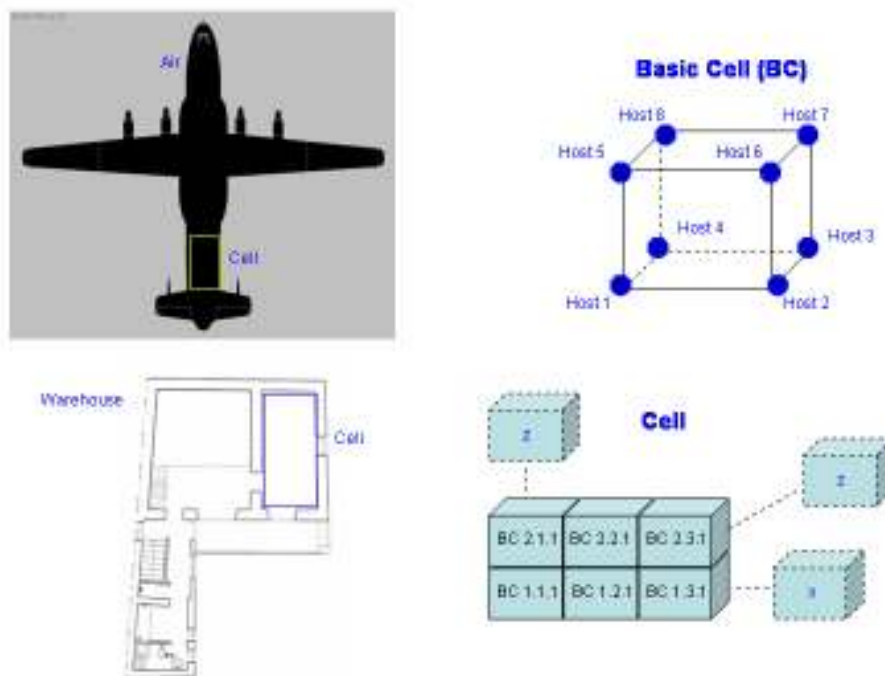


Figura 4.23. Composición de celdas básicas. Figura original de [Con3].

4.3.3.3 Pruebas de detección de RSSI

La precisión del sistema de localización depende de varios factores que caracterizan entornos concretos entre los cuales se encuentran factores ambientales y sensibilidad del hardware. En el caso de detección ideal, la potencia detectada (RSSI) debería que decrecer regularmente en función de la distancia.

Las pruebas de detección del RSSI (*Received Signal Strength Indicator*) se han efectuado en la banda de frecuencia de 2.4 Ghz utilizando el modo de comunicación de alta capacidad de *MM-Node*. Las pruebas indoor se han efectuado en la nave industrial de AERNOVA.

Como se puede apreciar en la Figura 4.24, los valores medios de RSSI detectados aparecen relativamente regulares para el entorno outdoor; por el contrario bastante lejanos del resultado esperado por el entorno indoor, incluso considerando la media sobre muchas muestras (cada valor reportado en figura es la media sobre 100 muestras). Los resultados obtenidos se demuestran coherentes con estudios recientes realizados en la misma banda de frecuencia [Jan2].

La precisión de los sistemas de localización outdoor basados en radiofrecuencia suele ser mayor de los correspondientes en indoor, incluso utilizando técnicas de localización relativamente sencillas [Sto2].

La precisión de los sistemas indoor puede mejorarse a través de:

- *Técnicas de cálculo avanzadas*. Suelen emplear algoritmos de cálculo complejos que suelen trabajar sobre modelos analíticos o, incluso, estadísticos. La eficiencia de estas técnicas crece, en condiciones normales, al crecer del número de puntos de observaciones disponibles en cuanto suelen auto-calibrarse.
- *Mecanismos de calibración previos*. Asumen la identificación de unos puntos de referencia en el área considerada. En dichos puntos de referencia se efectúan pruebas preliminares de calibración. El posicionamiento se efectúa comparando los valores detectados con valores que caracterizan los puntos de referencia. Estos mecanismos no se consideran demasiado fiables en entornos caracterizados por condiciones variables.
- *Mecanismos con memoria*. Oportunamente utilizados pueden limitar el rango de errores para actores que se mueven lentamente en el tiempo (como en el caso del escenario considerado).
- *Topologías con alta densidad de puntos de observación*.

Sin embargo, los resultados representados en Figura 4.24 pueden proporcionar un punto de vista demasiado simplificado del problema de la localización con tecnología a RF.

Con el objetivo de ampliar el entorno de conocimiento se describen a continuación los resultados obtenidos en otro banco de prueba.

Las pruebas se han efectuado en una área outdoor de aproximadamente 20x20 metros en plena ciudad. Se han detectado, al momento de las pruebas, numerosas fuentes de potencial interferencia (redes inalámbricas WiFi, dispositivos Bluetooth, etc.). Todas las pruebas se han efectuado en ausencia de obstáculos.

Los puntos de observación se han dispuesto a los ángulos del perímetro del área considerada y se han tomado 16 posiciones de referencia de acuerdo con la estructura mallada representada en Figura 4.24.1. También se representan los valores de RSSI detectados en una de las pruebas.

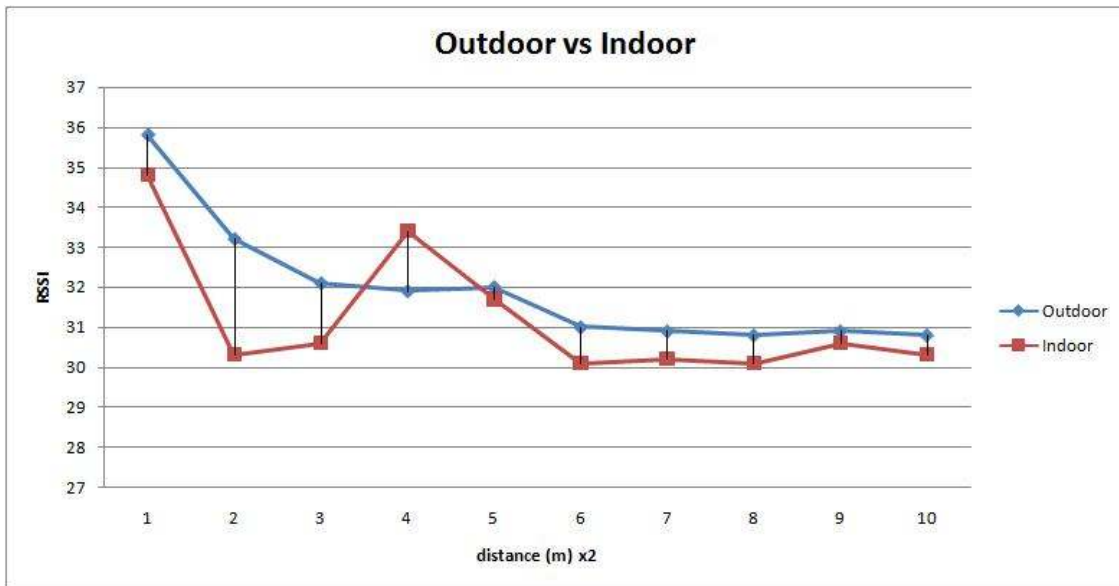


Figura 4.24. Valores medios de RSSI detectados en entornos indoor y outdoor. Figura original de [Con8].

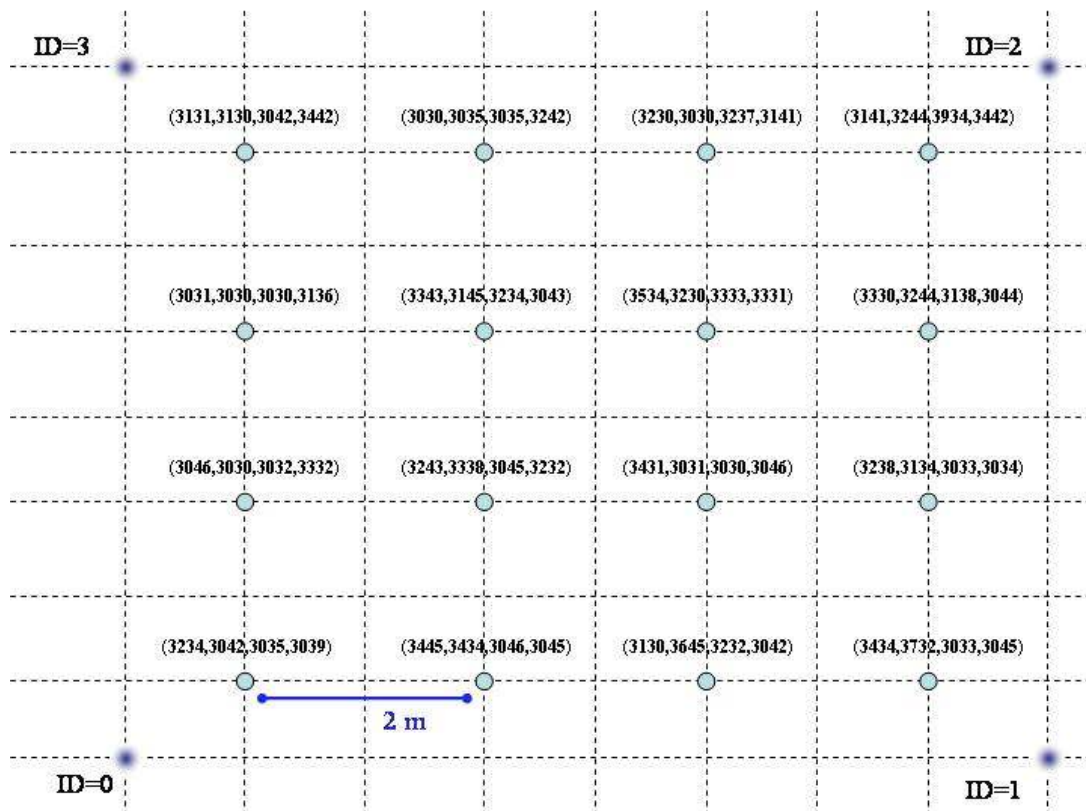


Figura 4.24.1. Posiciones de referencia para la detección de RSSI en áreas regulares.

Por cada posición se reporta el valor de RSSI detectado (eq. 4.3) por cada uno de los puntos de observación según la topología indicada:

$$h_k = (RSSI_0, RSSI_1, RSSI_2, RSSI_3) \quad (\text{eq. 4.3})$$

Las condiciones de la prueba se pueden considerar bastante variables visto el entorno en que los nodos trabajan.

En Figura 4.24.2 se proporciona el valor de RSSI detectado por cada uno de los puntos de observación en función de la distancia media del TAG activo. El valor esperado de la curva debería que presentar un comportamiento marcadamente decreciente.

Con referencia a la muestra considerada, se puede apreciar un comportamiento satisfactorio relativamente a los valores correspondientes a los puntos de observación con identificador 1, 2 y 3; el dispositivo con identificador 0 propone, en este caso, una curva de difícil interpretación. Esta irregularidad se debe, seguramente, a interferencias y es un ejemplo relevante de problemas que afectan las arquitecturas en el mundo real.

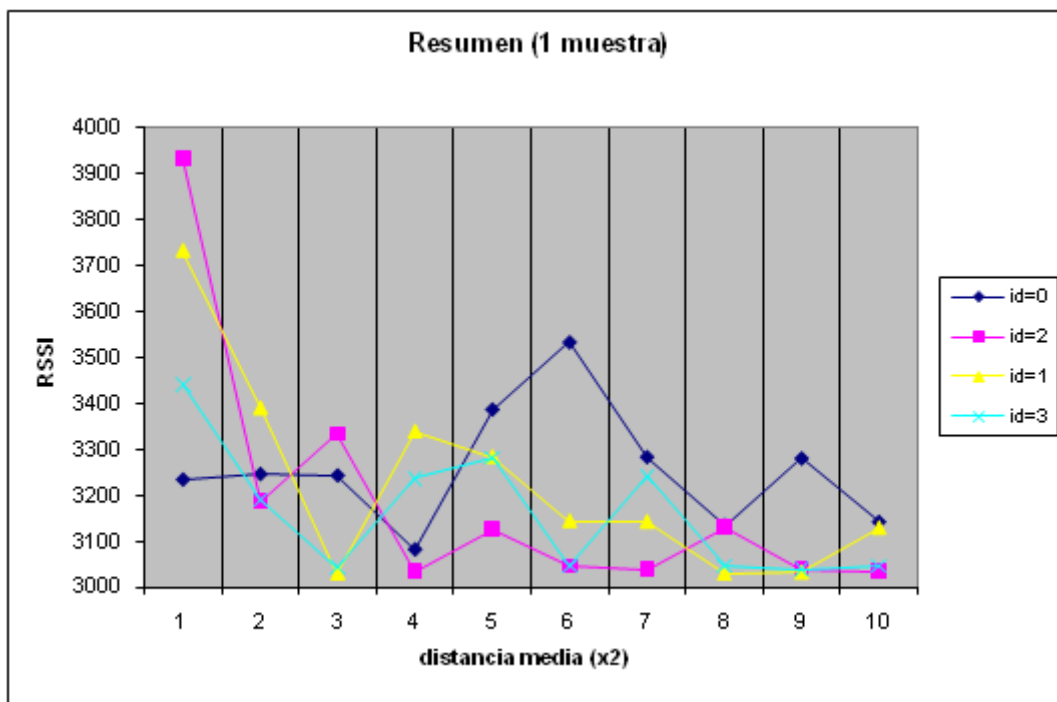


Figura 4.24.2. RSSI en función de la distancia media.

Repitiendo las pruebas y considerando valores medios se puede proporcionar una referencia más robusta respecto a la relación entre RSSI y distancia.

En Figura 4.24.3 se representan los valores medios relativos a un número creciente de muestras. Como se puede notar, la curva tiende a ser decreciente aunque no de forma muy marcada por menos de 10 muestras.

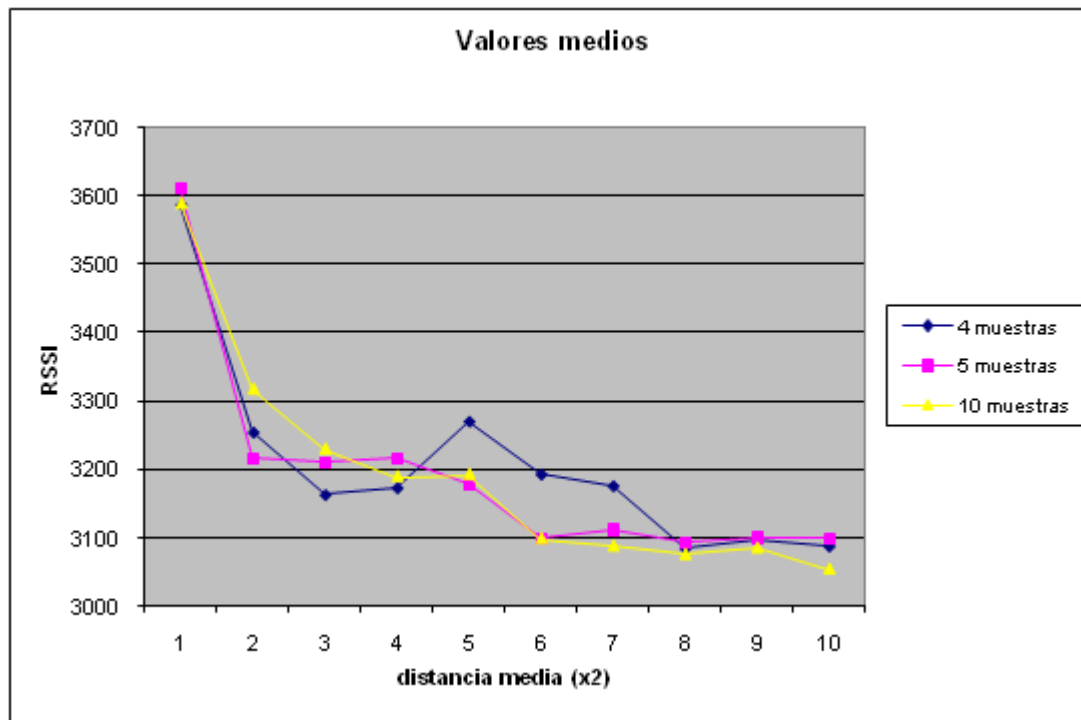


Figura 4.24.3. Valores medios de la relación entre RSSI y distancia.

4.3.4 Redes inalámbricas de sensores móviles: entornos vehiculares

Durante los últimos años, las redes inalámbricas en entornos vehiculares (Vehicular Ad-hoc Networks, VANETs [Sun4]) se han progresivamente afirmado conjuntamente a los avances de la tecnología inalámbrica correspondiente que proporciona entornos tecnológicos siempre más avanzados y flexibles tanto para la comunicación entre vehículos como entre vehículos y estaciones bases (*roadsides*).

Las VANETs se caracterizan por amplias posibilidades en términos de aplicación y posibilidades de negocio relacionadas. De momento VANET suele utilizar tecnologías para comunicación inalámbrica de medio alcance, normalmente tipo WiFi. Las aplicaciones más comunes que operan sobre infraestructuras MANETs incluyen servicios de varios tipos entre los cuales servicios *position-based*, sistemas para transporte inteligente, *hot-spots* WiFi, comunicación de datos dinámicos, diagnóstico sobre vehículos, control remoto y aplicaciones para diversión.

La gran parte de los temas de investigación abiertos se enfocan a la robustez de las comunicaciones de mensajes críticos, a la calidad de servicio de aplicaciones *user-centric* y a la utilización de componentes inteligentes (dispositivos multi-mode, agentes, etc.). El desarrollo de VANETs en un contexto avanzado, escalable y seguro es uno de los desafíos más relevantes para la comunidad científica internacional en el contexto de comunicación inalámbrica.

En la presente sección se considera la aplicación de redes de sensores en entornos vehiculares: en primer lugar se analizan las potencialidades tanto en términos tecnológicos como comerciales; luego se analiza una posible arquitectura para el soporte de aplicaciones para adquisición dinámica de datos en áreas metropolitanas y, finalmente, se procede a la descripción de la particularización del framework y de las limitaciones actuales.

4.3.4.1 Redes de sensores inalámbricos en entornos vehiculares: potencialidades y modelos de negocios

La aplicación de redes de sensores en entornos vehiculares (Wireless Vehicular Sensor Network, WWSN) es un campo de estudio relativamente reciente. Aunque en constante evolución, existe todavía una marcada diferencia entre el modelo de recursos empleado en redes de sensores (nodo inalámbrico de bajo coste) y el correspondiente utilizado en contextos VANET (plataformas normalmente más avanzadas respecto a los recursos de comunicación).

A pesar de estas importantes diferenciaciones entre los dos entornos inalámbricos, la aplicación de redes de sensores en entornos vehiculares no se puede considerar una novedad absoluta: en el pasado reciente, nodos sensores inalámbricos se han empleado en entornos vehiculares para solucionar problemas complejos [Dje1] o proporcionar servicios avanzados [Kar3].

La convergencia entre VANET y WSN puede proporcionar un interesante contexto tecnológico de bajo coste para aplicaciones de largo alcance en las cuales el número de puntos de observación es un parámetro determinante.

El escenario ideal que se considera en el presente trabajo de tesis es una aplicación que necesita monitorizar áreas importantes (áreas metropolitanas por ejemplo). La métrica de evaluación para la aplicación es la cantidad de puntos de observación y la cantidad de datos que el centro de control recibe. Aplicaciones reales conformes con estas características podrían interesar diferentes entornos e instituciones (administraciones públicas u organizaciones científicas interesadas en la monitorización de ciertos parámetros ambientales, por ejemplo).

Soluciones convencionales a este tipo de problema pueden implicar ciertas desventajas relevantes especialmente relacionadas con la escalabilidad; el problema fundamental es que garantizar la cobertura de una área de tamaño metropolitano implica el despliegue de un gran número de sensores dichos aspectos pueden determinar una pérdida de competitividad.

Una solución basada en WWSN (Figura 4.25) puede aprovechar la movilidad de un cierto número de vehículos en las áreas metropolitanas. La posibilidad de montar dispositivos GPS puede incrementar la flexibilidad de la solución: los datos generados se asocian dinámicamente a la posición correspondiente (con buena aproximación) al vehículo en el momento de la generación. La aplicación se basa, por lo tanto, en puntos de observaciones no prefijados si no determinados por la ruta de cada vehículo.

Un factor añadido podría ser el enfoque *green* de las redes: empleando tecnología de bajo coste y consumo, dispositivos sensores podrían montarse sobre cualquier tipo de vehículo (incluso vehículos eléctricos o bicis).

Como ya se ha mencionado, uno de los puntos fuertes de este tipo de arquitectura podría ser su modelo de negocio (normalmente el punto débil de las aplicaciones de adquisición de datos en gran escala).

Se podrían plantear múltiples escenarios considerando la enorme área de cobertura que podrían garantizar indirectamente ciertas empresas o instituciones que suelen disponer de vehículos para rutas (policía por ejemplo) o distribución de productos.

Una administración pública podría proponer a los diferentes conductores de montar los sensores a cambio de una reducción del impuesto de circulación por ejemplo. Solucionado el despliegue de sensores, las infraestructuras middleware podrían ser un coste asumible especialmente considerando que la arquitectura a disposición podría ser compartida con

otras instituciones (científicas por ejemplo). El mantenimiento ordinario sería muy limitado, prácticamente inexistente (los sensores se alimentarían a través del vehículo correspondiente).

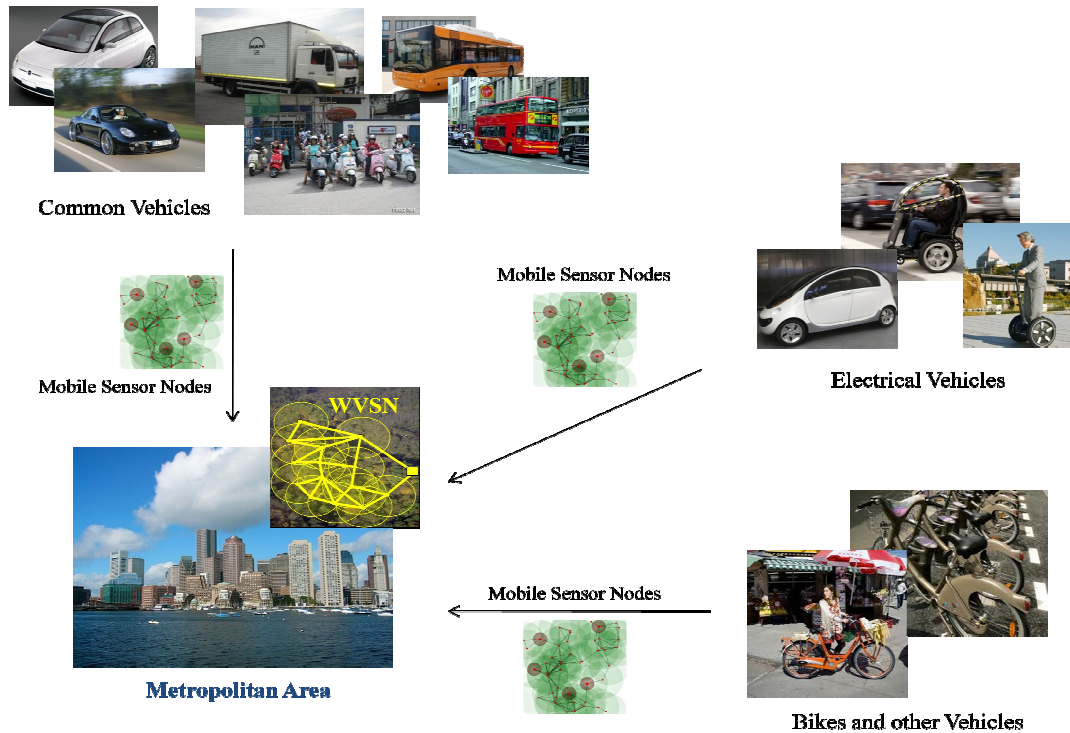


Figura 4.25. Redes inalámbricas de sensores en entornos vehiculares.

Un aspecto crítico que puede afectar de forma determinante las prestaciones reales de los sistemas es el despliegue estratégico de las diferentes estaciones base (fijas o móviles) para la transmisión de la información generada hacia el centro de control.

4.3.4.2 Arquitectura de redes de sensores para entornos vehiculares

La arquitectura propuesta para aplicación de redes de sensores en entornos vehiculares se representa en la Figura 4.26. Como se puede apreciar, tanto desde un punto de vista funcional como topológico se puede considerar una particularización del modelo de arquitectura general propuesta en el Capítulo 2.

La arquitectura se compone de los siguientes componentes:

- *WWSN*: es el conjunto de los nodos sensores móviles. Implementa tanto las funcionalidades de adquisición de datos como un mecanismo cooperativo para la transmisión de la información a las estaciones base. Detalles al respecto se describen en la siguiente sección.
- *Middleware Infrastructure*: tal como se ha descrito en el modelo general. El concepto de *Roadside* es equivalente al concepto de *Gateway*.
- *Knowledge Engine*: sistema para procesado de datos. Tal como se ha especificado en el contexto del modelo general, puede al menos incluir estructuras y modelos de

representación semánticos.

- *Application*: nivel de aplicación.

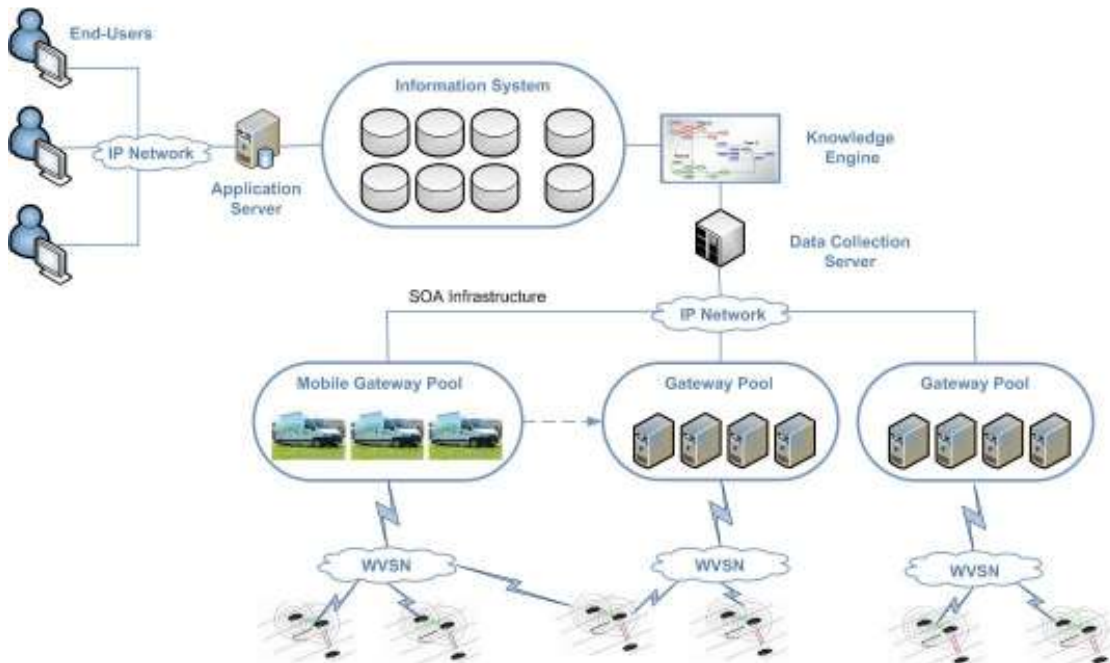


Figura 4.26. Arquitectura para redes inalámbricas de sensores en entornos vehiculares. Figura original de [Con5].

4.3.4.3 Particularización del framework

La arquitectura de redes inalámbrica de sensores se extiende y particulariza para soportar, de forma óptima, las siguientes funcionalidades:

- *Generación de datos*. Se asume que cada nodo sensor tenga capacidad de almacenamiento bastante limitada. La generación de datos o detección de eventos puede efectuarse de acuerdo con diferentes modalidades: la modalidad *on-demand* genera datos puntuales como consecuencia de una petición explícita por parte de una estación base; la modalidad *single-value* asume la captura periódica de datos pero tan solo el último dato generado se guarda por lo cual este dato es el único que se transmite a la estación base cuando el nodo este en su alcance; finalmente la modalidad *data-flow* asume captura periódica de datos que se almacenan en un buffer con una cierta política de reemplazo. La primera modalidad prácticamente no requiere estructuras de almacenamiento, la segunda estructuras mínimas, la tercera estructuras más significativas y flujos de datos importantes hacia la estación base.
- *Asociación entre datos y posiciones*. Puede efectuarse de forma explícita (con la utilización de dispositivos GPS) o aproximada. Esta última modalidad suele generar datos en modalidad *single-value* y asociar al dato la posición de la estación base

receptora. En ambos caso pueden surgir problemas de privacidad.

- *Sistema de transmisión de datos.* Es equivalente al mecanismo de comunicación multi-salto en una red de sensores. Considerando que el tiempo medio de exposición (conexión potencial con una estación base) podría variar enormemente en dependencia del posicionamiento de las estaciones base y del comportamiento de los vehículos (por ejemplo un semáforo en verde o rojo), se ha configurado el sistema de adquisición de datos a funcionar de la forma más reactiva posible (bajo demanda). Como se puede apreciar en Figura 4.27, las estaciones base están comprobando la presencia de nodos sensores en su alrededor. Si un nodo viene descubierto directamente, intenta transmitir los datos disponibles utilizando transmisión mono-salto y, al mismo tiempo, intenta descubrir nuevos nodos creando un mecanismo de comunicación multi-salto. Evidentemente la métrica de evaluación de este mecanismo es el porcentaje de mensajes recibidos por una estación base en un cierto tiempo y con una cierta densidad de vehículos.

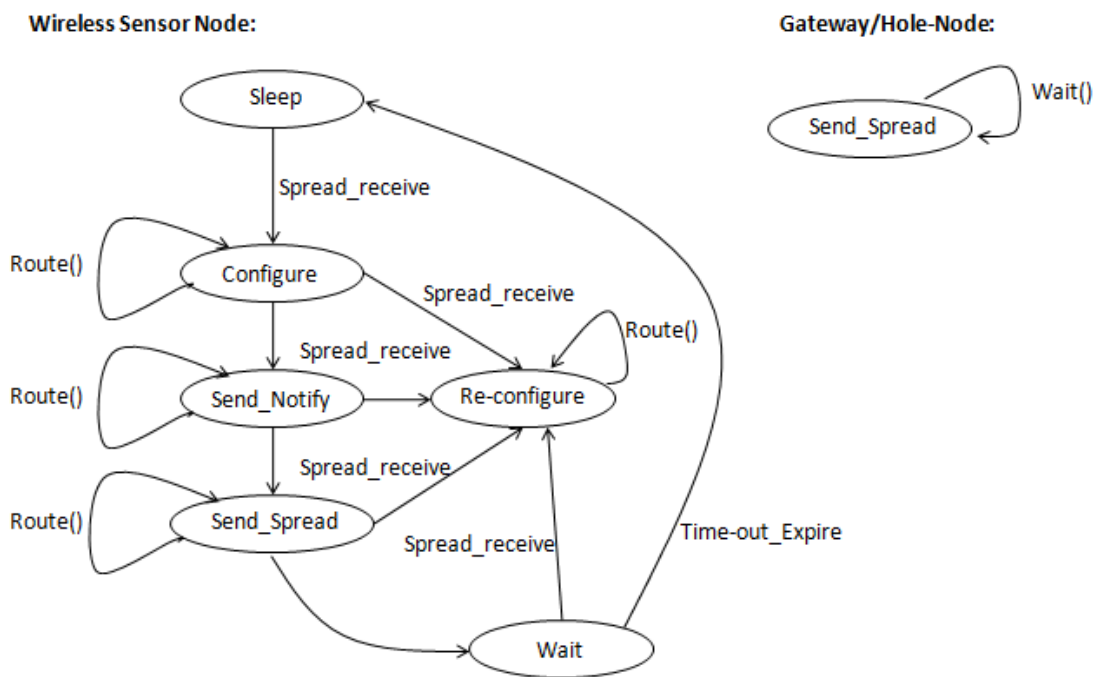


Figura 4.27. Particularización del framework (Figura original de [Con5]).

4.3.4.4 Limitaciones actuales

La plataforma propuesta ha sido objeto de diferentes experimentos de simulación (Figura 4.28) con un número variable de estaciones base tanto fijas que móviles. Los resultados se consideran en línea con los resultados de carácter general presentados en el Capítulo 3.

No se han efectuado pruebas de comunicación sobre vehículos reales que, posiblemente, requiere hardware específico a parte de implicar un cierto coste a escala significativa.

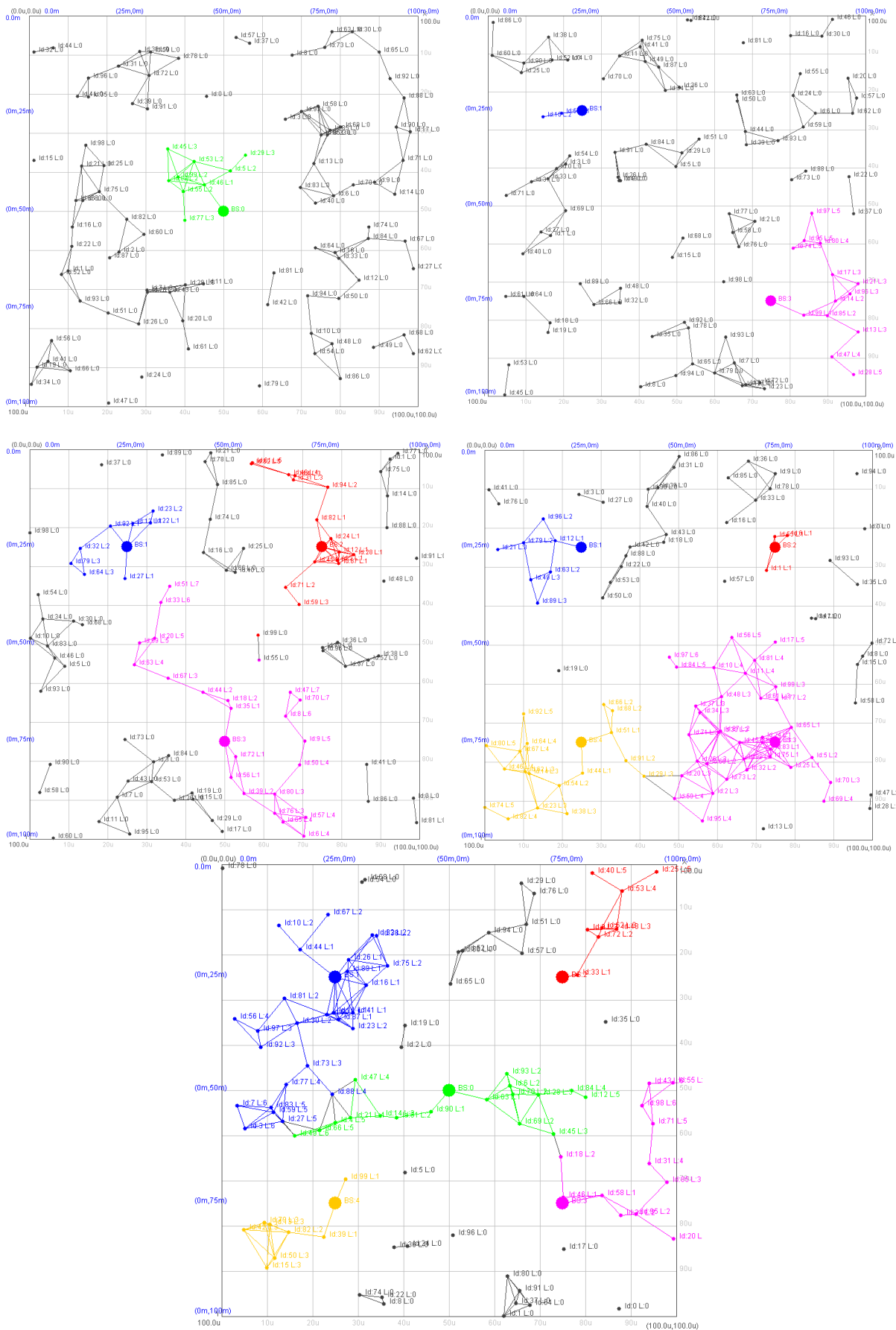


Figura 4.28. Entornos de simulación (Figura original de [Con5]).

Pruebas futuras que deberían proporcionar un conocimiento profundizado de la plataforma en entornos vehiculares se consideran las siguientes:

- Caracterizar el entorno tecnológico, especialmente en términos de comunicación inalámbrica.
- Evaluar el tiempo medio de transmisión en función de la densidad de vehículos y a la velocidad de movimiento de los mismos.

4.3.5 Integración de sensores en sistemas de mando y control

El sistema de adquisición de datos descrito en una de las secciones anteriores se ha integrado en el sistema de mando y control desarrollado por el grupo de investigación para entornos civiles [Isr1].

En esta sección se describen:

- El aporte potencial de las redes inalámbricas de sensores en los sistemas de mando y control, especialmente en relación a la fase de evaluación de la situación.
- Las modalidades de integración de redes de sensores inalámbricos como sistema empujado.

4.3.5.1 Sistemas de mando y control: *situation awareness*

Los sistemas de mando y control son sistemas de comunicación complejos que pretenden soportar equipos especializadas en la gestión de situaciones de emergencia.

Una emergencia suele caracterizarse por situaciones caóticas en las cuales hace falta evaluar con efectividad la situación, coordinar las diferentes instituciones que hacen frente a la situación de emergencia y actuar de forma apropiada.

La utilización de soportes tecnológicos avanzados, tanto para los equipos de primer auxilio como para las diferentes jerarquías de mando, puede aumentar de forma importante la efectividad de las operaciones de socorro y la probabilidad de planear operaciones con altas probabilidades de éxito.

Los sistemas de mando y control (Figura 4.29) se caracterizan por estructuras dinámicas a medida, topologías lógicas con una jerarquía predefinida (que normalmente refleja las jerarquías de mando) y topologías físicas difíciles o imposibles de predecir.

Uno de los problemas más relevantes en el diseño de los sistemas de mando y control es la interoperabilidad entre los sistemas ya que, tanto en un contexto militar que civil, las operaciones de socorro suelen involucrar diferentes instituciones con carácter heterogéneo respecto a cultura, competencia y rol. Dichas instituciones o cuerpos deberían actuar de forma coordinada y sincronizada para garantizar la efectividad de las operaciones.

Las tecnologías inalámbricas de última generación (gran alcance y ancho de banda) permiten soluciones innovadoras y el uso simultáneo de múltiples tecnologías avanzadas.

La información procedente de sensores puede jugar un rol importante, especialmente en la fase de evaluación de la situación donde pueden aportar informaciones adicionales, tal vez importantes. Las más modernas soluciones, caracterizadas por dispositivos compactos que se auto-organizan dinámicamente en red pueden ser bastante funcionales tanto en la

monitorización de parámetros ambientales como dispositivos asociados a personas o para alcanzar objetivos específicos (detección de presencia por ejemplo).

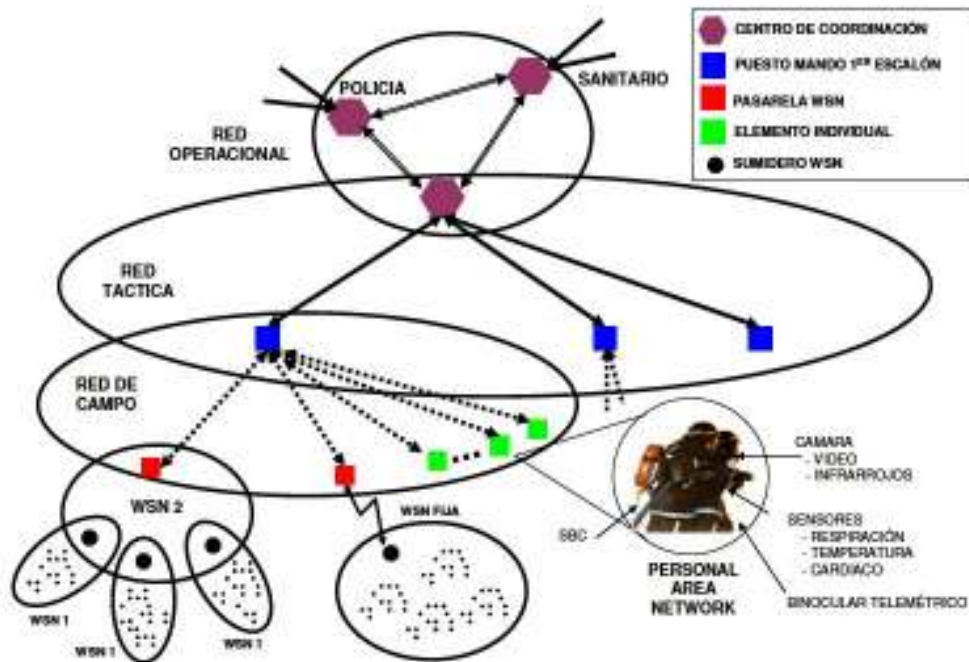


Figura 4.29. Posible arquitectura de un sistema de mando y control (Figura original de [Isr1]).

4.3.5.2 Integración de redes inalámbricas de sensores

El sistema de adquisición de datos descrito en las secciones anteriores se ha integrado en el sistema de mando y control considerado actuando como sistema empotrado en los puestos de mandos locales (Figura 4.30). En otras palabras, la estación base es parte del puesto de mando local.

Para la integración del sistema de adquisición de datos se ha actuado a diferentes niveles:

- *Información.* La información procedente de los nodos sensores y recibidos por la estación base se insertan directamente en el sistema de información (local) del sistema de mando y control. El formato de la información se uniforma al formato de datos de los otros datos en el sistema de mando y control. Se añade el concepto de nodo sensor que puede componerse de un cierto número de generador de datos (sensores). Cada nodo sensor se identifica unívocamente en el sistema a través del identificador junto con el puesto de mando al cual está asociado. Los datos sensores se asocian directamente al nodo sensor de referencia e, indirectamente, al puesto de mando correspondiente.
- *Configuración.* Aunque todos los mecanismos integrados de las redes inalámbricas de sensores se auto-gestionan y auto-configuran, los nodos tiene previamente que darse de alta en el sistema de mando y control.
- *Infraestructura de comunicación.* El sistema de mando y control considerado dispone de un mecanismo de réplica de la información generada a nivel local, de

acuerdo con la jerarquía especificada. Gracias al soporte de este mecanismo general de comunicación en el sistema, no se han tenido que desarrollar mecanismos auxiliares de comunicación [Isr1].

- *Representación de la información*, La representación de la información disponible es una funcionalidad muy importante a la hora de evaluar la situación de emergencia y las consecuentes estrategias de gestión. En la siguiente sección se describen unos detalles al respecto.

Se ha utilizado el siguiente hardware:

- Dispositivo IRIS como microprocesador/transmisor inalámbrico.
- MTS101/MTS420 como tarjetas de adquisición de datos respectivamente para la versión no geo-posicionada y con dispositivo GPS integrado.
- Gateway físico con diferentes interfaces (serie, USB o ethernet) tal como se ha descrito en secciones anteriores del capítulo.

En la Figura 4.31 se representan las dos versiones del nodo sensor inalámbrico. Solo una de ellas (izquierda) monta dispositivo GPS integrado para geo-posicionamiento.



Figura 4.31. Nodo sensor geo-referenciado (izquierda) y nodo sensor no geo-referenciado (derecha).

4.3.5.3 El nivel de presentación

Como se ha descrito en la sección anterior, el sistema de adquisición de datos actúa como sistema empotrado y la información generada o capturada se inserta en el sistema de información local. El protocolo de replica que actúa en cada puesto de mando y control se preocupa de replicar la información de acuerdo con la topología definida.

Sin embargo, la representación de la información de tiempo real es la funcionalidad más importante en un sistema de mando y control que opera en situaciones de emergencia. Por lo tanto se ha extendido la capa de representación de datos en tiempo real para que también los datos procedentes por los nodos sensores se representen juntos con los demás.

Se han desarrollado dos diferentes formas de representar los datos en tiempo real:

- *Información geo-referenciada* (Figura 4.32). La utilización de información geo-referenciada implícitamente asocia las observaciones a la posición del nodo que las genera. Por lo tanto, esta información se considera global y se representa en el mapa geo-referenciado con el icono y el valor corriente en la posición que corresponde. Tratándose de un punto de observación geo-referenciado, las observaciones no se asocian lógicamente al puesto de mando y control si no a la posición de referencia. En este modo de funcionamiento, también existe la posibilidad de geo-referenciar, a través de un nodo sensor, un puntos genéricos de interés.
- *Información no geo-referenciada* (Figura 4.33). Este modo de funcionamiento asume que la información generada no se asocia a una posición concreta. La información esta, por lo tanto, asociada al puesto de mando y control considerado. En caso de más puntos de observación del mismo tipo, se representan valores medios de las medidas generadas. Este modo de representación también se utiliza para dispositivos geo-posicionados en el caso de falta de cobertura GPS (ambientes indoor por ejemplo).



Figura 4.30. Sistema de adquisición de datos y puesto de mando local (estación base).



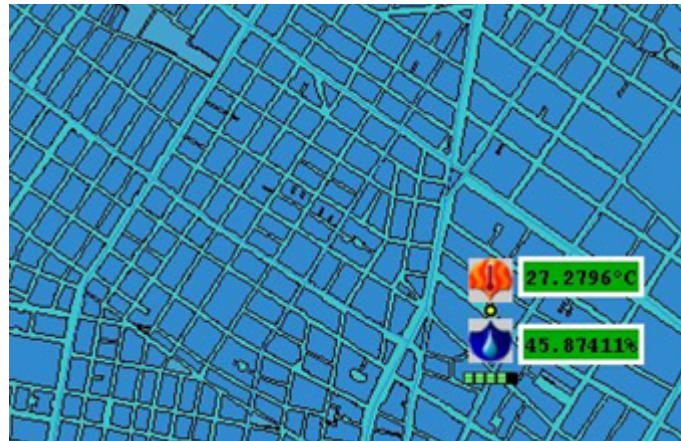


Figura 4.32. Ejemplos de representación de datos geo-referenciados.

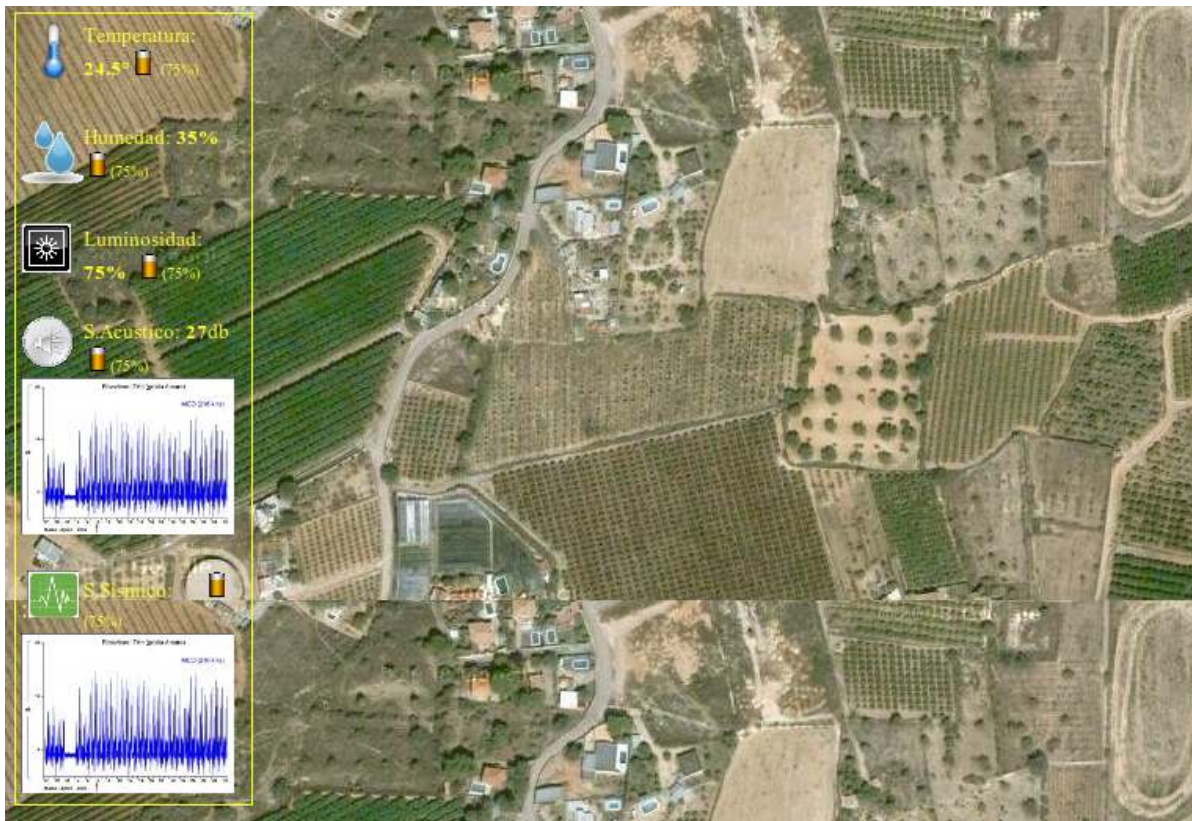


Figura 4.33. Ejemplos de representación de datos no geo-referenciados asociados a una unidad de mando y control local.

Conclusiones y Trabajo Futuro

El presente trabajo de tesis ha presentado una arquitectura de un entorno de redes de sensores inalámbricos. Las redes de sensores inalámbricos es un área de trabajo emergente en la que intersectan diferentes áreas de conocimiento: procesado de señal, comunicaciones a nivel físico, gestión del acceso al medio, enrutamiento, gestión de recursos, algorítmica, semántica, planificación gestión de la calidad de servicio y seguridad entre otros. Las redes de sensores inalámbricas han tenido una gran dependencia de la aplicación para la que se han sido diseñadas. Por lo que las generalizaciones en muchos casos son complicadas. Existen algunos marcos de trabajos y arquitecturas generales con la finalidad de poder reutilizar conceptos y modelos pero por lo general no son extrapolables los resultados de un entorno a otro. En el primer capítulo se ha realizado un estudio exhaustivo de las principales tecnologías relacionadas con las redes de sensores, dada la amplitud del tema principal de la tesis, se ha trabajado con una mayor profundidad aquellos elementos más relacionados con los sucesivos capítulos de la tesis.

La primera de las aportaciones de la tesis ha sido la arquitectura propuesta propiamente dicha, en que se ha realizado una revisión de todos los niveles lógicos, físicos, virtuales, entornos lógico/funcionales y complejos proporcionando un modelo de referencia portable y efectivo.

Una vez definida la arquitectura el trabajo se ha articulado en tres fases lógicamente interdependientes que han proporcionado, respectivamente, un soporte teórico (Capítulo 3) orientado al análisis a diferentes niveles de abstracción, el desarrollo en un contexto modular y flexible (Capítulo 4, sección 4.1) y, finalmente, la experimentación (Capítulo 4) de la propia arquitectura en diferentes dominios de aplicación.

Vista la metodología horizontal que ha caracterizado el trabajo de tesis, los resultados obtenidos se pueden analizar según varios puntos de vista y se pueden enmarcar, básicamente, en tres principales contextos de investigación:

- Ingeniería de protocolos para redes inalámbricas de sensores.
- Servicios middleware avanzados para sistemas de sensores.
- Ingeniería avanzada del conocimiento.

Con referencia al contexto tecnológico proporcionado por el actual concepto de nodo sensor inalámbrico de bajo coste, respecto a las soluciones corrientes (Figura C.1, izquierda), el conjunto de protocolos proporcionado se caracteriza, a paridad de funcionalidades proporcionadas (auto-organización y gestión, soporte a despliegue aleatorio, *clustering* etc.), por mayor modularidad y flexibilidad que determina un contexto aplicativo bastante más extendido (incluido el soporte para redes con nodos sensores móviles) y, consecuentemente, una potencial reducción de los costes globales de desarrollo. Además, el sólido fundamento teórico proporcionado por los modelos analíticos de base, juntos con los ambientes avanzados de simulación, amplían las posibilidades de análisis, tanto a priori como a posteriori, soportando al diseñador en todas las fases de diseño y desarrollo de arquitecturas concretas.

Por otra parte, soluciones generalizadas multi-dominio como la arquitectura propuesta pueden, en ciertos contextos, caracterizarse por prestaciones inferiores a las prestaciones proporcionadas por soluciones a medida. La solución propuesta, ha demostrado, en el contexto de su experimentación, un potencial interesante y comparable, en términos de prestaciones, con soluciones a medida; debiéndose en gran medida a la metodología modular de diseño de la pila de protocolos que proporciona un framework de base fácilmente extensible y relativamente cómodo de particularizar en función de exigencias concretas.

Otros aspectos de gran importancia científico/comercial (fiabilidad y robustez tras el crecimiento de la escala) se mantienen en línea con las otras soluciones orientadas a *clusters*. Una vez más, la modularidad y las posibilidades de extensión/particularización que caracteriza al framework permiten, frente a requisitos no funcionales extremadamente estrictos, la integración de mecanismos a medida.

Los resultados conseguidos relativos a servicios middleware (Figura C.1, derecha) se deben interpretar de forma diferente en cuanto han sido diseñados y desarrollados en función de tecnologías fuertemente consolidadas en otros contextos, aunque emergentes y con amplio potencial de mejora y expansión en el próximo futuro en el ámbito de las redes de sensores inalámbricas.

Las infraestructuras desarrolladas en tecnología Grid se presentan más flexibles, eficientes y dinámicas, en un contexto interoperable y pervasivo, respecto a otras clases de solución. El contexto infraestructural propuesto se ha demostrado extremadamente funcional y eficiente tanto en el contexto de servicios complejos (*content-aware* o *context-aware*) dentro de organizaciones virtuales como de entornos emergentes y lógicamente más complejos como *Internet of Things* o *Sensor Web*; soluciones corrientes que parecen caracterizarse por una menor flexibilidad que determina ciertas carencias en uno o más de los contextos aplicativos de referencia.

Independientemente del modelo de referencia y de la tecnología de desarrollo, las infraestructuras se consideran soluciones a medida; aunque proporcionando un conjunto de

servicios virtuales de base, el coste de desarrollo no varía significativamente, así como su complejidad.



Figura C.1. Resumen de los resultados obtenidos.

La evolución de los modelos de datos y de interacción hacia modelos innovadores y en gran ascenso (modelos semánticos) determina un contexto tecnológico avanzado y con gran potencial en perspectiva futura. Una ingeniería del conocimiento innovadora determina, en este caso, un contexto de interoperabilidad notablemente más avanzado con múltiples potencialidad en términos de computación; además, los modelos de inteligencia pueden rápidamente evolucionar, ofreciendo novedosas y, por gran parte inexploradas, capacidad de evolución. La utilización de modelos de conocimiento complejos facilita la definición de múltiples perspectivas, tanto para la representación como para el análisis del conocimiento mismo, caracterizadas por diferentes niveles de abstracción y enfoques.

En su conjunto, el presente trabajo ha dado lugar a las contribuciones listadas en el Anexo A que se resumen en una publicación indexada JCR; dos capítulos de libro; doce ponencias en congresos (una de ellas con el premio al mejor artículo del congreso) y dos posters en congreso de investigación. Además, se han organizado dos eventos científicos internacionales, respectivamente sobre *Sensor Web* (SSW 2010, celebrado en Valencia en Octubre de 2010) e interoperabilidad semántica (IWSI 2011, celebrado en Roma en Enero de 2011).

Las líneas de trabajo futuro se pueden, de momento, resumir como a continuación:

- *Extender la experimentación de la plataforma*, extendiendo las diferentes aplicaciones en las que se puede utilizar así como la escala del sistema.
- *Evaluación experimental de las prestaciones de redes de sensores móviles/vehiculares* (como caso particular de redes de sensores móviles) y análisis extendido de sus posibles modelos de explotación.
- *Refinamiento*, si fuera necesario, de los modelos propuestos. Integración de los mismos con aspectos de dominios concretos de interés general (modelos de consumo relacionados con tecnologías concretas por ejemplo). Estudio de la posibilidad de adoptar modelos declarativos.
- *Ampliación de las funcionalidades de los ambientes de simulación*, especialmente integrando aspectos característicos de dominios concretos (por ejemplos los varios algoritmos de enrutamiento proporcionados por el framework). Funcionalidades para la evaluación de los consumos y del *energy-hole* en función del *duty-cycle* de las

aplicaciones se consideran de interés potencial.

- *Estudio de las posibilidades de integración de estándares en modelo propuesto.* La idea fundamental es utilizar interfaces de acceso estandarizadas para todos los componentes middleware.
- *Extensión de las comunicaciones* mediante la utilización del estándar *6LOWPAN*, a efectos de tener una continuidad IPv6 desde el sensor hasta el centro de monitorización.
- *Experimentación de modelos de computación avanzados en tecnologías semánticas.* Más concretamente, se consideran de interés el aprendizaje de actores semánticos y la computación multi-ontología. Para ambos se requiere una métrica para determinar el nivel de fiabilidad de los resultados obtenidos.
- *Consolidación* de los resultados obtenidos tanto desde un punto de vista teórico (representación rigurosa de los diferentes modelos y arquitecturas de referencia) que practico (definición e implementación de un único framework de referencia).

Anexo A: Contribuciones

➤ Artículos en Revistas

[Jou1] Benjamin Molina, *Salvatore F. Pileggi*, Manuel Esteve, Carlos E. Palau, **A Negotiation Framework for Content Distribution in Mobile Transient Networks**, Journal of Network and Computer Applications, Volume 32, Issue 5 (Special issue on Next Generation Content Networks), pp. 1000-1011, September 2009.

➤ Capítulos de libro

[Cha1] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **Enabling Wireless Sensor Network within Virtual Organizations**, Advances in Next Generation Services and Service Architectures, Eds. A. Prasad, J. Buford, V. Gurbani, River Publishers. to Appear 2011.

[Cha2] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **Analysis techniques and models for resource optimization in Wireless Sensor/Actuator Network environment**, 2007, in IFIP International Federation for Information Processing, Volume 248, Wireless Sensor and Actor Networks, eds. L.Orozco-Barbosa, Olivares T., Casado R., Bermudez A., (Boston: Springer), pp 23-34.

➤ Artículos en actas de conferencia

[Con1] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **Grid Sensor/Actuator Network Architecture**, Proceedings of International Conference on Wireless and Mobile Communications, ICWMC'06, July 29-31, 2006, Bucharest, Romania.

[Artículo premiado con Best Paper Award]

[Con2] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **An adaptive and flexible fault tolerance mechanism designed on multi-behavior agents for Wireless Sensor/Actuator Network**, Proceedings of International Conference on Sensor Technologies and Applications (SENSORCOMM 2007), October 14-20, 2007, Valencia, Spain.

[Con3] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **Integrating Wireless Sensor Network with Grid Computing and RFID technology for products monitoring into airports**, Proceedings of IASTED International Conference on Communication Systems and Networks (CSN2007), August 29-31, 2007, Palma de Mallorca, Spain.

[Con4] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **Cluster Partitioning in logic-organized Wireless Sensor/Actor Network**, Proceedings of the seventh IASTED International Conference on Communication Systems and Networks (CSN2008), September 1-3, 2008, Palma de Mallorca, Spain.

[Con5] *Salvatore F. Pileggi*, **A multi-domain framework for Wireless Vehicular Sensor Network**, Proceedings of International Conference on Ultra Modern Telecommunications (ICUMT2009), October 12-14, 2009, St. Petersburg, Russia.

[Con6] Benjamin Molina, *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **A Social Framework for Content Distribution in Mobile Transient Networks**, Third Workshop on the Use of P2P, GRID and Agents for the Development of Content Network(UPGRADE-CN'08), June 23-27, 2008, Boston, MA, USA.

[Con7] *Salvatore F. Pileggi*, Benjamin Molina, Carlos E. Palau, Manuel Esteve, **Movie for All (M4ALL): A live J2ME Content Exchange Facility for Mobile Environments**, Proceedings of IEEE International Conference on Multimedia and Expo 2009 (ICME2009), June 28-July 3, 2009, New York, USA.

[Con8] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **On the convergence between Wireless Sensor Network and RFID: industrial environment**, Proceedings of the 8th International Symposium on Modeling and Optimization in Mobile, Ad-hoc and Wireless Networks and Workshops (WiOPT2010), May 31st-June 4th, 2010, Avignon, France.

[Con9] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **Multimode WSN: Improving Robustness, Fault Tolerance and Performance of Randomly Deployed Wireless Sensor Network**, Proceedings of the Second International Conference on Computational Intelligence, Communication Systems and Networks (CicSyn 2010), 28-30 July, 2010, Liverpool, UK.

[Con10] *Salvatore F. Pileggi*, **A Novel Domain Ontology for Sensor Networks**, Proceedings of the Second International Conference on Computational Intelligence, Modelling and Simulation (CimSim2010), 28-30 September 2010, Bali.

[Con11] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **Building Semantic Sensor Web: Knowledge and Interoperability**, Proceedings of the International Workshop on Semantic Sensor Web (SSW2010), October 2010, Valencia, Spain.

[Con12] *Salvatore F. Pileggi*, **A Semantic Environment for Data Processing in Embedded Sensor Networks**, Proceedings of the International Workshop on Semantic Interoperability (IWSI2011), January 2011, Rome, Italy.

➤ Posters

[Pos1] *Salvatore F. Pileggi*, C.E. Palau, Manuel Esteve, **A declarative approach for Hierarchical-organized Wireless Sensor/Actor Network**, Proceedings of Work-in-Progress, Poster and Demo papers of 4th IEEE International Conference on Distributed Computing in Sensor Systems 2008 (DCOSS'08), June 11-14, 2008, Santorini Island, Greece.

[Pos2] *Salvatore F. Pileggi*, Carlos E. Palau, Manuel Esteve, **A Global Resource Management Model (GRMM) for Wireless Sensor/Actuator Network**, Adjunct poster/demo proceedings of 4th European Conference on Wireless Sensor Networks (EWSN2007) , January 29-31, 2007, Delft, Netherlands.

Anexo B: Implementación de un Actor Semántico

El Actor Semántico, diseñado de acuerdo al modelo propuesto en el Capítulo 2, pretende proporcionar un conjunto de funciones de alto nivel de abstracción a soporte de sistemas y aplicaciones que pretenden interactuar entre ellos a través de modelos semánticos.

Más concretamente, se ha desarrollado una capa de abstracción directamente sobre *Pellet 1.3* [Sir1] en lenguaje Java. Dicha capa implementa una interfaz de acceso bien definida que puede accederse a través de una interfaz *web-service*, a través de una interfaz de agente o ser parte de cualquier otro tipo de aplicación.

Para garantizar la suficiente flexibilidad, la interfaz de acceso principal se ha dividido en múltiples sub-interfaces.

En Figura AB.1 se representa la sub-interfaz principal que soporta la interacción de aplicaciones según una perspectiva orientada al concepto: el concepto semántico (o una característica(s)) es siempre el input; el output corresponde a las instancias (o individuos) que cumplen los requisitos del input. Esta perspectiva es, con diferencia, la más utilizada en casos reales. Como se puede apreciar, los resultados se devuelven en un esquema XML con diferentes niveles de detalle.

```
boolean validateOntology(XMLSchema Ontology)  
XMLSchema getVocabulary(XMLSchema Ontology)  
  
XMLSchema getIndividuals(XMLSchema Ontology, String concept)  
XMLSchema getIndividuals(XMLSchema Ontology, String concept, int detailDegree)  
XMLSchema getIndividuals(XMLSchema Ontology, XMLSchema systemData, String concept)  
XMLSchema getIndividuals(XMLSchema Ontology, XMLSchema systemData, String concept, int detailDegree)
```

Figura AB.1. Métodos principales de la interfaz proporcionada.

La sub-interfaz dual propone una inversión de perspectiva: un individuo es una entrada, la salida la construyen los conceptos del cual el individuo es una instancia. También en este caso, la interfaz permite la posibilidad de considerar resultados con diferentes niveles de detalle o abstracción.

Las funcionalidades implementadas por la interfaz principal se consideran el corazón de un framework para soportar tanto modelos de computación relativamente sencillos (computación *ontology-driven* por ejemplo) como modelos más complejos (*ontology-aware*) o avanzados (por ejemplo aprendizaje, computación multi-ontología).

Como aplicación de ejemplo se ha desarrollado un mecanismo de búsqueda con interacción humana. En figura AB.2 se propone la GUI implementada para interacción humana. Esta aplicación, prácticamente, actúa como un motor de búsqueda sobre modelos semánticos. Esta aplicación permite de cargar una ontología (y los datos relacionados), reconstruye el árbol de conocimiento y admite búsqueda sobre la base del vocabulario implementado por la ontología.

Desde un punto de vista computacional, la aplicación actúa según una lógica *ontology-driven* en cuanto se limita a “interpretar” la semántica de la ontología en cuestión sin implementar alguna lógica añadida.

La aplicación también incluye un módulo para representación gráfica del conocimiento asociado a los resultados. Dicho módulo es un ejemplo muy sencillo de computación *ontology-aware* en cuanto un conjunto de preferencias (iconos, tamaños, etc.) están disponibles para ontologías conocidas. Por otra parte, si la ontología procesada no se conoce a priori, se buscan, por cada concepto, las preferencias para representación gráfica. Si no existen, se utilizan preferencias genéricas.

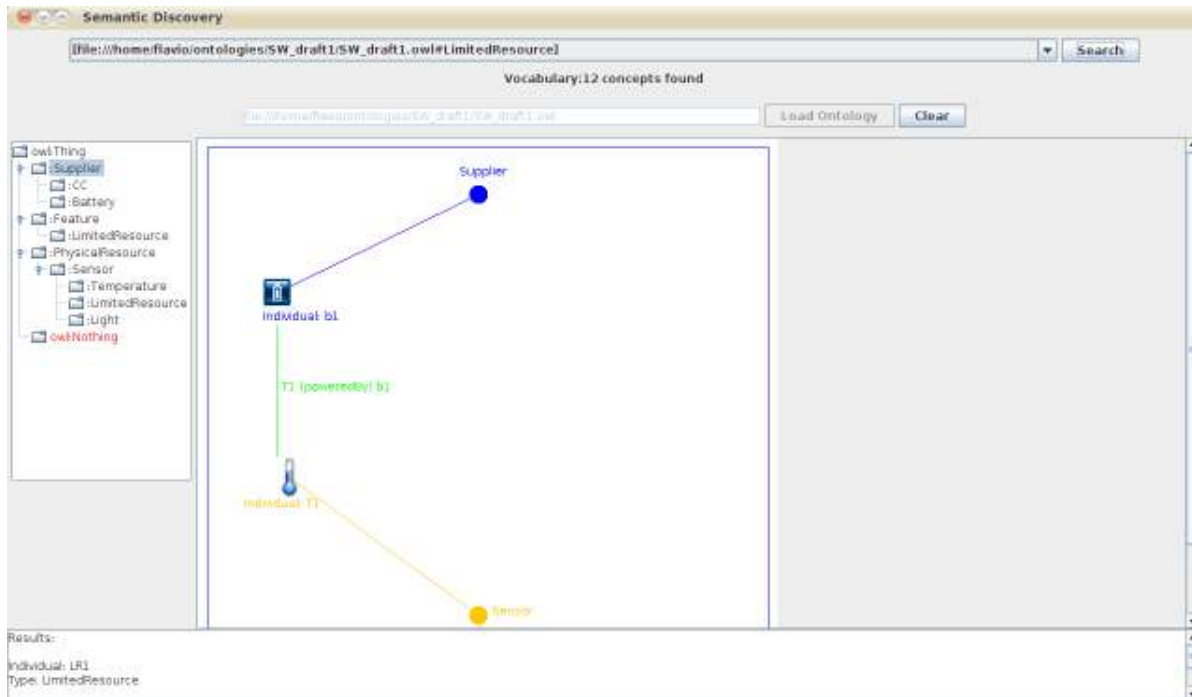


Figura AB.2. Interfaz usuario.

Aspectos computacionalmente más avanzados (por ejemplo computación multi-ontología y aprendizaje), aunque temas de primaria importancia en computación avanzada e inteligencia artificial, no se consideran aspectos de interés específico de este trabajo.

Actualmente, las funcionalidades de aprendizaje habilitan la composición de un entorno personalizado de conocimiento altamente dinámico y extensible, en el cual conceptos y relaciones pueden ser importados por el actor desde diferentes dominios y modelos, por lo general de tipo heterogéneo.

El funcionalidades de calculo multi-ontología asumen la coexistencia de mas modelos ontológicos independientes y potencialmente heterogéneos que se consideran un único modelo de conocimiento. Objetivo fundamental es buscar y definir relaciones potenciales entre conceptos desde diferentes ontologías.

Estos últimos dos módulos funcionales son actualmente en fase de experimentación; en esta fase es de primaria importancia definir una métrica para determinar el nivel de fiabilidad de los resultados obtenidos.

Anexo C: Ambientes de Simulación

Los ambientes de simulación propuestos implementan los modelos teóricos descritos en el contexto del Capítulo 3 para permitir una evaluación preliminar de las diferentes topologías de acuerdo con una metodología de trabajo que asume, en primera análisis, la ausencia de cualquier aspecto característico de aplicaciones o dominios concretos.

Se ha desarrollado un ambiente a medida para implementar únicamente los aspectos de real interés en el contexto del trabajo. Utilizando una perspectiva de nivel de red para evaluar teóricamente tanto topologías predefinidas como despliegues aleatorios con comportamiento estático, nomadico o móvil de los diferentes actores involucrados.

Considerando el tipo de simulación en cuestión, se ha optado por un modelo discreto a eventos implementado por un conjunto de módulos funcionales representados en Figura AC.1. Como se puede apreciar, un componente central efectúa los cálculos necesarios sobre la base recibe los varios inputs y un generador de eventos.

Los principales inputs se consideran los siguientes:

- *Topología.* Incluye el tamaño de la red (numero de bases, máximo 5, y numero de nodos sensores) y la posición de cada nodo respecto a la área de referencia. Dicha posición puede ser predefinida (proporcionada por un fichero) o, bien, generarse de forma aleatoria (despliegue aleatorio). Ejemplos de despliegues aleatorios se representan en Figura AC.4. En el caso de entornos estáticos, dicha posición es el input para los cálculos relativos a las varias configuraciones; en entornos nomadico o móviles se interpreta como topología de salida.
- *Parámetros de comunicación.* Se articula en dos fases. En primer lugar se define el radio medio de comunicación. En segunda instancia se pueden considerar entornos de comunicación non ideales de acuerdo con los modelos relacionados propuestos en el Capitulo 3.
- *Comportamiento de los actores.* Por defecto, un actor se considera estático. Si se desea que cambie de posición en el tiempo, se tiene que proporcionar el comportamiento (de acuerdo con el modelo propuesto en el Capitulo 3). En la versión corriente del simulador, cada actor tiene una posición de salida distinta y un generador asociado independiente aunque todos los actores del mismo tipo tienen el mismo comportamiento. En otras palabras, se define un modelo de comportamiento para las bases y uno para los nodos sensores; la independencia de los generadores de eventos y de las posiciones de salida garantiza comportamientos independientes y diferentes.
- *Ambiente.* Plantillas concretas, con objetos y obstáculos de interés. No s ha utilizado este modulo en el presente trabajo.
- *Otros parámetros.* El radio del área de cobertura de los nodos sensores entre otros.

También el generador de eventos puede considerarse, bajo ciertos puntos de vista, una entrada en cuanto proporciona la generación de valores aleatorios bajo demanda del componente de control. Sobre la base de estos valores se generan los varios eventos que determinan un cambio de estado de la red.

En Figura AC.2 se describen las interfaces de usuario principales (para entornos estáticos y móviles) que permiten gestionar las entradas, iniciar la simulación y acceder a

funcionalidades avanzadas. Mediante el panel principal se puede acceder a los otros paneles: el *Display*, el *ToolBar*, el Panel de estadísticas y el *Tracer*.

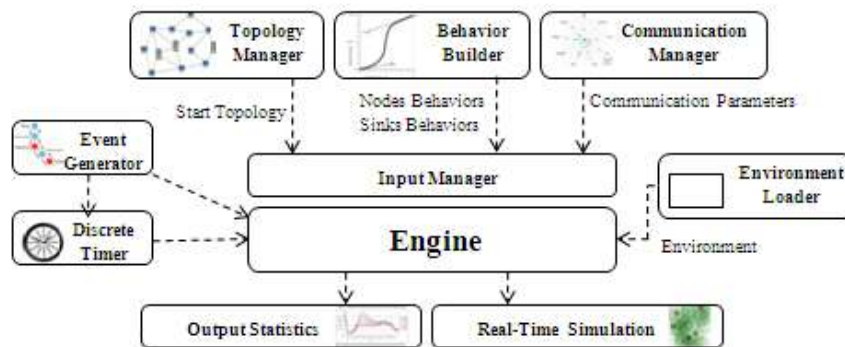


Figura AC.1. Módulos funcionales que componen los ambientes de simulación.

El *Display* es el panel que visualiza la topología en análisis; se trata de una imagen fija en el caso de entornos estáticos mientras es un panel animado en el caso de entornos móviles. El *Display* se controla a través del *Toolbar* (Figura AC.3) que permite gestionar tanto parámetros de configuración del área de referencia como aspectos visuales (el zoom por ejemplo).

Esta herramienta permite un enfoque de la red considerando múltiples perspectivas (Figura AC.5): visión global (se considera la red como una única entidad lógica), visión a *clusters* (a cada estación base se asocia un *cluster*), área de cobertura, etc.

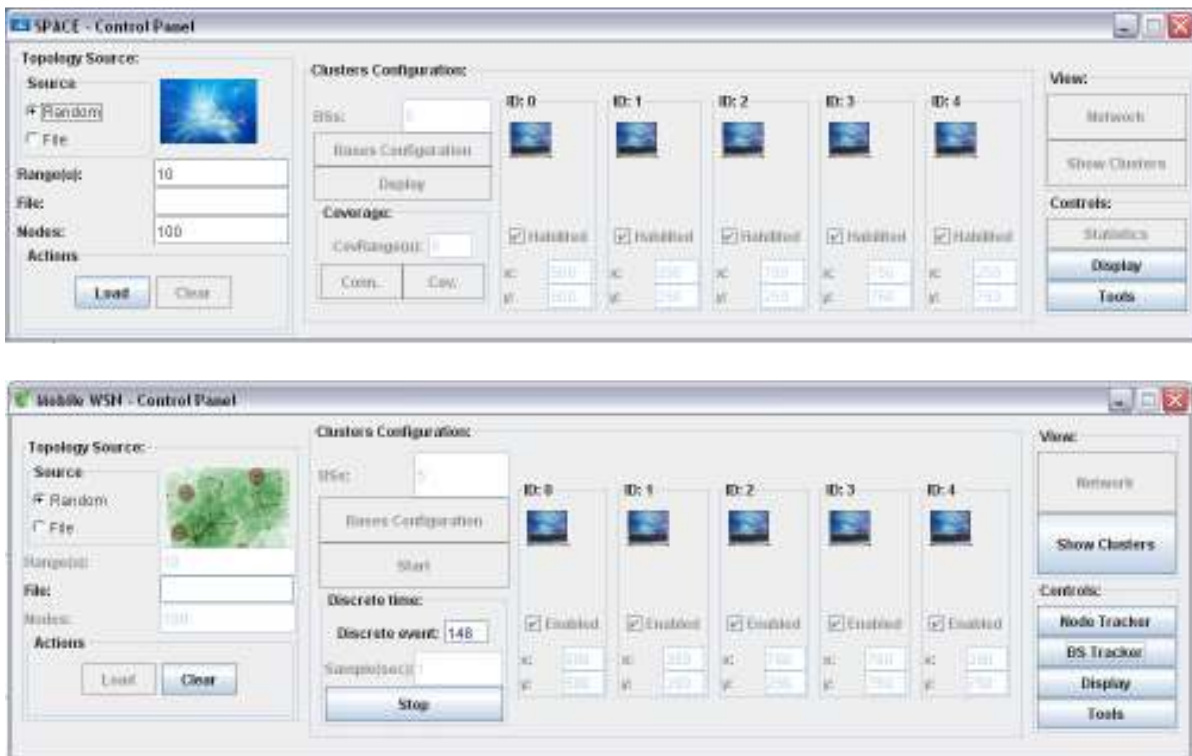


Figura AC.2. Interfaces de usuario principales para el ambiente con actores estáticos (arriba) y móvil (abajo).



Figura AC.3. ToolBar.

El Panel de estadísticas proporciona un resumen de las estadísticas de interés: nodos conectados, estadísticas sobre la configuración de cada *cluster* y sobre la configuración lógica de la red, etc.

Los entornos móviles son intrínsecamente más complejos de analizar por dos motivos principales:

- Entornos móviles requieren un análisis de los resultados en función del comportamiento de los actores. Para soportar dicha análisis se ha dotado la aplicación de una herramienta adicional (*Tracer*, Figura AC.7) para poder monitorizar el comportamiento de un máximo de cinco actores (estaciones base o nodos sensores).
- En ciertos casos, el análisis se limita a resultados finales. Las estadísticas de interés suelen ser las mismas que las correspondientes en entornos estáticos. Por otra parte, en presencia de comportamientos peculiares de los nodos, puede interesar la evolución del estado de la red o el comportamiento en ciertos periodos concretos del intervalo de simulación.

También se pueden considerar entornos híbridos respecto a la movilidad en los cuales unos actores son estáticos y otros móviles. Casos típicos se consideran topologías con estaciones base fijas y nodos sensores móviles o con estaciones base móviles y nodos sensores fijos.

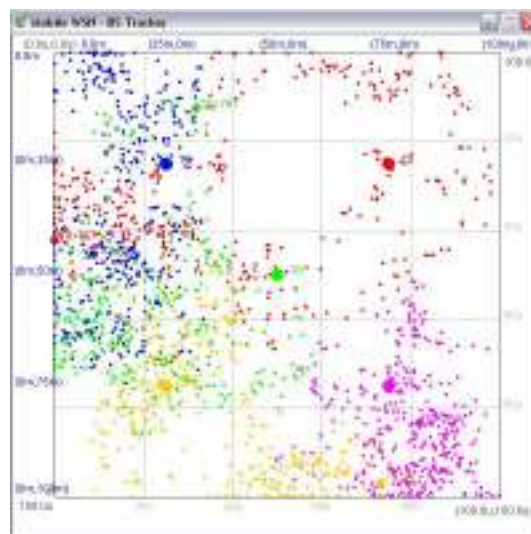


Figura AC.7. Tracer.

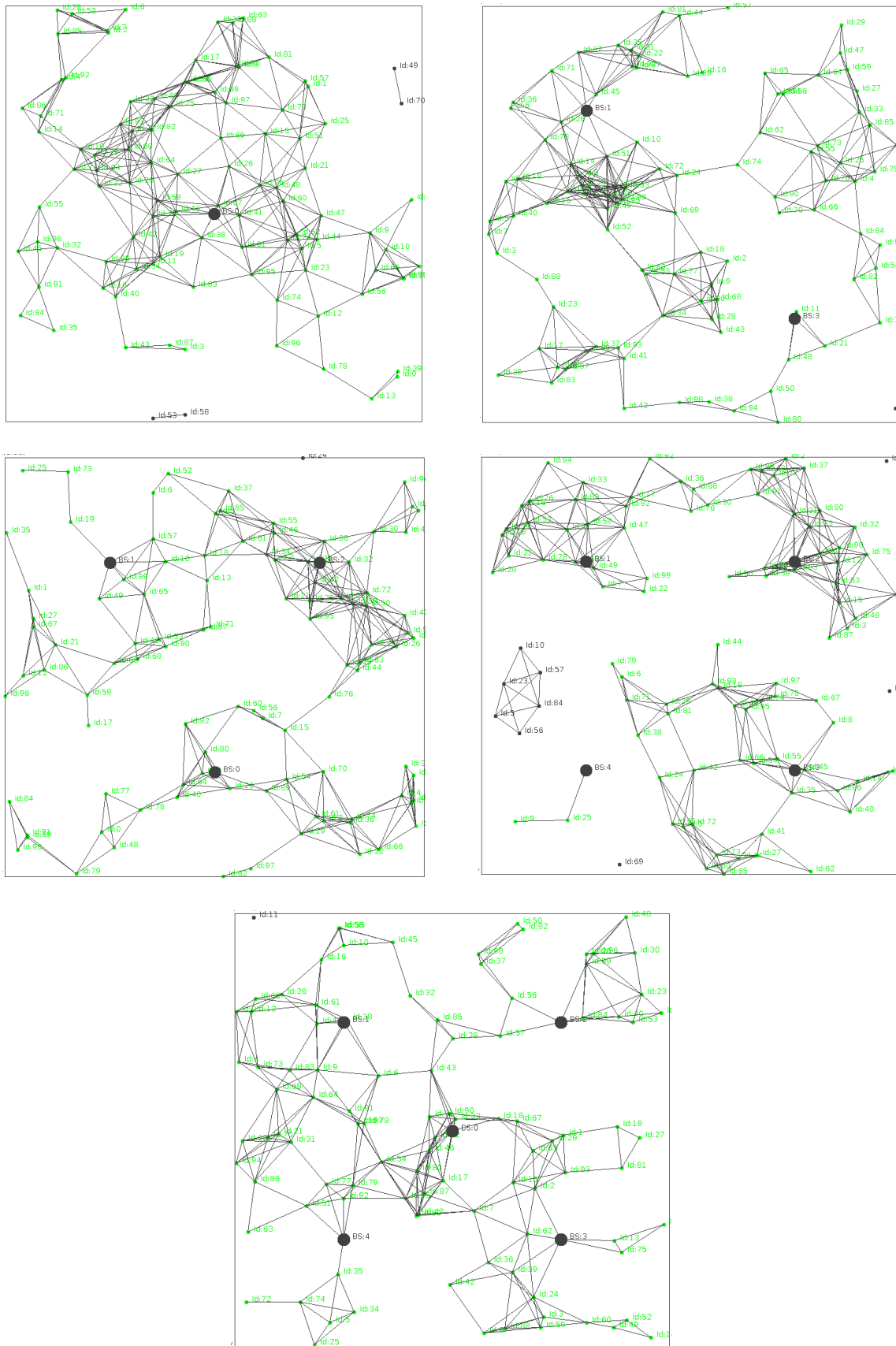


Figura AC.4. Ejemplo de despliegue aleatorio con diferentes números de estaciones base dispuestas simétricamente.

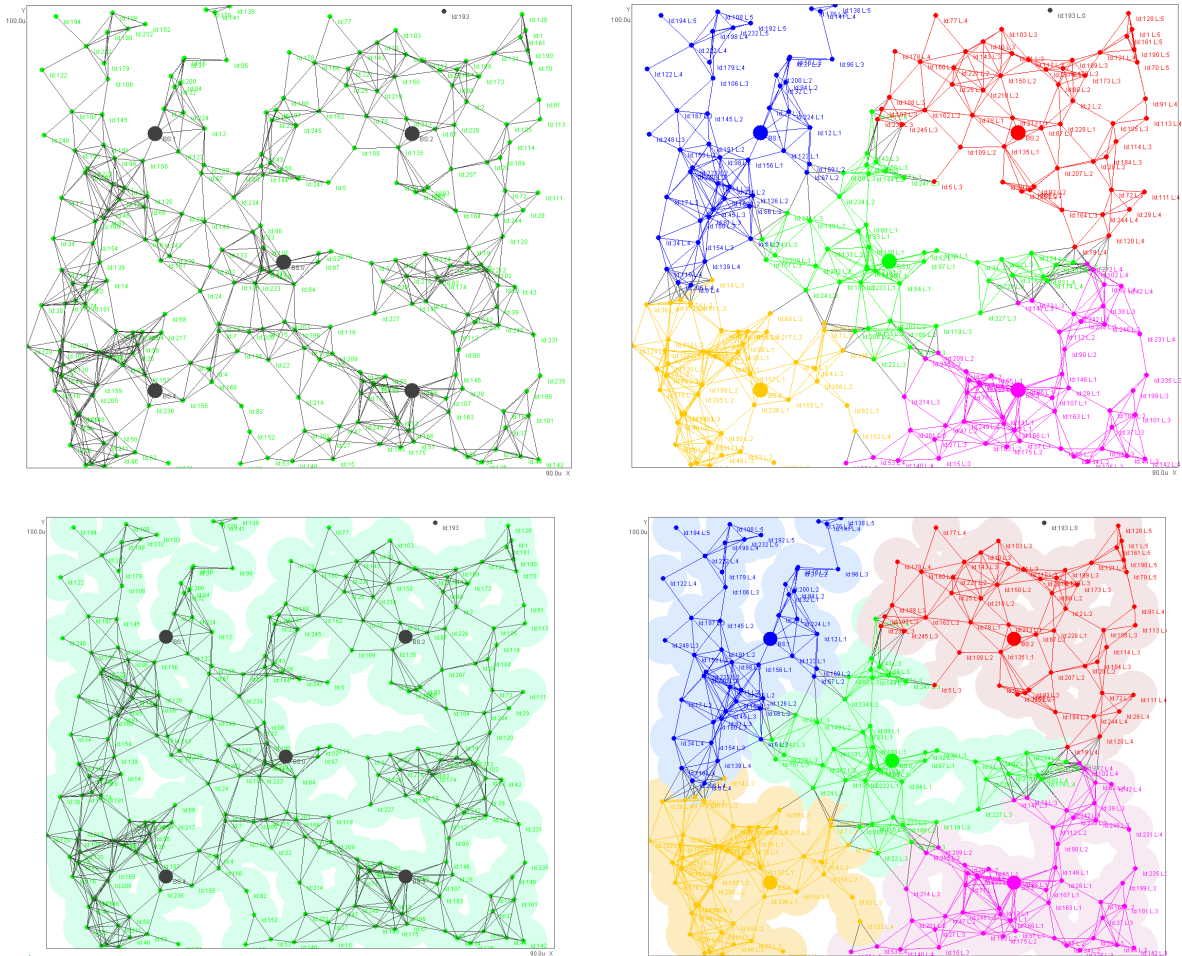


Figura AC.5. Diferentes puntos de vista de la red.

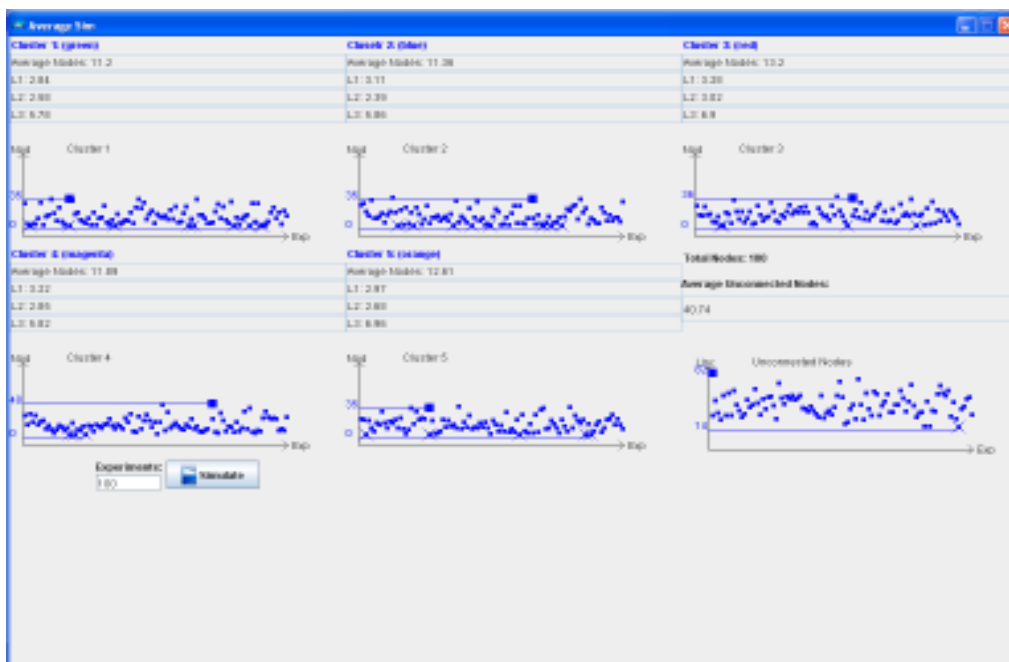


Figura AC.6. Panel de estadísticas.

Referencias

- [Abb1] Ameer Ahmed Abbasi, Mohamed Younis, *A survey on clustering algorithms for wireless sensor networks*, Computer Communications, Volume 30 , Issue 14-15 (October 2007), Pages 2826-2841.
- [Abc1] Gustafsson, E.; Jonsson, A.; Always best connected, *Wireless Communications*, IEEE, Volume 10, Issue 1, Feb. 2003. Page(s): 49-55.
- [Akk1] Kemal Akkaya and Mohamed Younis, *A Survey on Routing Protocols for Wireless Sensor Networks*, Ad Hoc Networks 3(3):325-349, May 2005.
- [Aky1] Akyildiz, I.F.; Weilian Su; Sankarasubramaniam, Y.; Cayirci, E.;, *A survey on Sensor Network*, IEEE Communication Magazine, August, 2002, Volume 40, Issue 8, Aug. 2002 Page(s):102 - 114.
- [Aky2] Akyildiz, I.F. Melodia, T. Chowdury, K.R. , *Wireless multimedia sensor networks: A survey*, IEEE Wireless Communications, December 2007, Volume 14, Issue 6, page(s): 32-39.
- [Alm1] Almeida, J.; Grilo, A.; Pereira, P.R.; , *Multimedia Data Transport for Wireless Sensor Networks*, Next Generation Internet Networks, 2009.
- [Alm2] Almasaeid, H.M.; Kamal, A.E.; , *Modeling Mobility-Assisted Data Collection in Wireless Sensor Networks*, IEEE Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008.
- [Asi1] Asikainen, M.; Haataja, K.; Honkanen, R.; Toivanen, P.; , *Designing and Simulating a Sensor Network of a Virtual Intelligent Home Using TOSSIM Simulator*, Fifth International Conference on Wireless and Mobile Communications, 2009. ICWMC '09.
- [Bai1] F. Bai, A. Helmy, *A Survey of Mobility Modeling and Analysis in Wireless Ad-hoc Networks*, Wireless Ad Hoc and Sensor Networks, Kluwer Academic Publishers, 2004.
- [Bar1] Paolo Baronti Prashant Pillai Vince Chook Stefano Chessa Alberto Gotta Y. Fun Hu, *Wireless Sensor Networks: a Survey on the State of the Art and the 802.15.4 and ZigBee Standards*, Computer Communications, 2006, pp. 1-30.
- [Bar2] Barisic, D.; Krogmann, M.; Stromberg, G.; Schramm, P.; *Making Embedded Software Development More Efficient with SOA*, 21st International Conference on Advanced Information Networking and Applications Workshops, 2007, AINAW '07.
- [Ben1] Hock Beng Lim, Yong Meng Teo, Mukherjee P., Vinh The Lam, Weng Fai Wong, See, S., *Sensor grid: integration of wireless sensor networks and the grid*, The IEEE Conference on Local Computer Networks, 2005.
- [Blu1] *BlueTooth*, www.bluetooth.com.
- [Btn1] *BTNode*, www.btnode.ethz.ch.
- [Bot1] Mike Botts and Alex Robin. *Bringing the sensor web together*. Geosciences, pages, 46–53, 1997.

[Bou1] Bougard, B., F. Catthoor, D. C. Daly, A. Chandrakasan and W. Dehaene, *Energy Efficiency of the IEEE 802.15.4 Standard in Dense Wireless Microsensor Networks: Modeling and Improvement Perspectives*, Proc. Design, Automation, and Test in Europe (DATE), March 2005.

[Bou2] Boukerche, A.; Yan Du; Jing Feng; Pazzi, R.; , *A reliable synchronous transport protocol for wireless image sensor networks*, Computers and Communications, 2008. ISCC 2008. IEEE Symposium on

[Cam1] S. A. Camtepe, and B. Yener, *Key Distribution Mechanisms for Wireless Sensor Networks: a Survey*, RPI Technical Report TR-05-07, Rensselaer Polytechnic Institute, March 23, 2005.

[Cha1] H.Chan, A.Perrig and D.Song. *Random key predistribution schemes for sensor networks*. In Proceedings of the 2003 IEEE Symposium on Security and Privacy, May 2003.

[Cha2] Charalambous, C.; Shuguang Cui; , *A biologically inspired networking model for wireless sensor networks*, Network, IEEE , vol.24, no.3, pp.6-13, May-June 2010

[Che1] Chen Wang, Li Xiao, *Improving Event-to-Sink Throughput in Wireless Sensor Networks*, Proceedings of Work-in-Progress, Poster and Demo papers of 4th IEEE International Conference on Distributed Computing in Sensor Systems 2008 (DCOSS'08), June 11-14, 2008, Santorini Island, Greece.

[Chg1] Ching-Ju Lin, Po-Lin Chou, Cheng-Fu Chou, *HCDD: Hierarchical Cluster-based Data Dissemination in Wireless Sensor Networks with Mobile Sink*, IWCMC'06, July 3–6, 2006, Vancouver, British Columbia, Canada.

[Chi1] David Chiyuan Chu, Lucian Popa, Arsalan Tavakoli, Joseph M. Hellerstein, Philip Levis, Scott Shenker and Ion Stoica. *The Design and Implementation of A Declarative Sensor Network System*. The 5th ACM Conference on Embedded networked Sensor Systems (SenSys 2007), Sydney, Australia, 6-9 Nov, 2007.

[Cio1] Victor Cionca, Thomas Newe, Vasile Dad?rlat, *TDMA Protocol Requirements for Wireless Sensor Networks*, Second International Conference on Sensor Technologies and Applications (SENSORCOMM2008), 2008.

[Cor1] CORBA, <http://www.corba.org/>.

[Cor2] CORBA/E, <http://www.corba.org/corba-e/index.htm>.

[Cro1] *Crossbow Technology Inc.*, <http://www.xbow.com>.

[Das1] K. Dasgupta, K. Kalpakis and P. Namjoshi, *An Efficient Clustering-based Heuristic for Data Gathering and Aggregation in Sensor Networks*, IEEE WCNC, 2003.

[Daz1] Dazhi Chen and Pramod K. Varshney, *QoS Support in Wireless Sensor Networks: A Survey*, International Conference on Wireless Sensor Network 2004, Las Vegas, Nevada, USA.

[Dha1] S. Dhar, *MANET: Applications, Issues and Challenges for the Future*, International Journal of Business Data Communications and Networking, Vol. 1, No. 2, April - June 2005, Pages 66-92.

[Dje1] Djenouri, D., *Preventing vehicle crashes through a wireless vehicular sensor network*, 2008 24th Biennial Symposium on Communications, 24-26 June 2008.

[Dpw1] *OASIS, Devices Profile for Web Services (DPWS)*, <http://docs.oasis-open.org/ws-dd/ns/dpws/2009/01>.

[Du1] W. Du, J. Deng, Y. S. Han and P. K. Varshney, *A Witness-Based Approach for Data Fusion Assurance in Wireless Sensor Networks*, In Proceedings of GLOBECOM, 2003.

[Emb1] *Ember*, www.ember.com.

[Fer1] R.Fernandez, J.Merè, F. Martinez, A.Marcos, F.Alba, R.Lostado, A.Pernià, *Redes inalámbricas de sensores: teoría y aplicación práctica*.

[Fin1] T.Finin, R.Fritzon, D.McKay, R.McEntire, *KQML as an agent communication language*, Proceedings of the third international conference on Information and knowledge management, Gaithersburg, Maryland, United States, 1994.

[Fip1] *FIPA*, <http://www.fipa.org/>.

[Fip2] *FIPA Agent Communication Language Specifications*, <http://www.fipa.org/repository/aclspecs.html>.

[Fip3] *FIPA-OS*. <http://fipa-os.sourceforge.net/>, 2001.

[Fos1] I.Foster, *What is the Grid? A Three Point Checklist*, GRIDToday, July 20, 2002.

[Fos2] I.Foster, C.Kesselman, S.Tuecke, *The Anatomy of the Grid: Enabling Scalable Virtual Organizations*, International J. Supercomputer Applications, 15(3), 2001.

[Fos3] I. Foster. *Globus Toolkit Version 4: Software for Service-Oriented Systems*, FIP International Conference on Network and Parallel Computing, Springer-Verlag LNCS 3779, pp 2-13, 2006.

[Fum1] Fummi, F.; Quaglia, D.; Ricciato, F.; Turolla, F.; , *Modeling and simulation of mobile gateways interacting with wireless sensor networks*, Design, Automation and Test in Europe, 2006. DATE '06.

[Gan1] Gandomi, A.; Tafti, V.A.; Ghasemzadeh, H.; , *Wireless Sensor Networks Modeling and Simulation in Visualsense*, 2010 Second International Conference on Computer Research and Development, 7-10 May 2010.

[Gan2] Ganesh, S.; Amutha, R.; , *Energy efficient transport protocol for wireless sensor networks*, 2nd IEEE International Conference on Computer Science and Information Technology, 2009. ICCSIT 2009.

[Gay1] M.Gaynor, S.Moulton, M.Welsh, E.LaCombe, A.Rowan, J.Wynne, *Integrating Wireless Sensor Network with the Grid*, IEEE Computer Society, 2004.

[Gia1] Giannakos, A.; Karagiorgos, G.; Stavrakakis, I.; , *A Message-Optimal Sink Mobility Model for Wireless Sensor Networks*, Eighth International Conference on Networks, 2009. ICN '09.

[Glo1] *Globus Alliance*, www.globus.org.

[Gri1] *GridGain*, <http://www.gridgain.com/>

[Gui1] Guinard, Dominique; Trifa, Vlad; Karnouskos, Stamatis; Spiess, Patrik; Savio, Domnic; , *Interacting with the SOA-Based Internet of Things: Discovery, Query, Selection, and On-Demand*

Provisioning of Web Services, Services Computing, IEEE Transactions on , vol.3, no.3, pp.223-235, July-Sept. 2010.

[Gui2] Gui, N.; Jianbin Hu; Zhong Chen; , *A novel wireless sensor network model and its application in multi-path routing*, ISECS International Colloquium on Computing, Communication, Control, and Management, 2009. CCCM 2009.

[Gun1] Gungor, V.C.; Hancke, G.P.; , *Industrial Wireless Sensor Networks: Challenges, Design Principles, and Technical Approaches*, , IEEE Transactions on Industrial Electronics, vol.56, no.10, pp.4258-4265, Oct. 2009.

[Han1] Hanoun, S.; Creighton, D.; Nahavandi, S.; , *Decentralized mobility models for data collection in wireless sensor networks*, IEEE International Conference on Robotics and Automation, 2008. ICRA 2008.

[Has1] H.Hassanein, Jing Luo, *Reliable Energy Aware Routing In Wireless Sensor Networks*, Second IEEE Workshop on Dependability and Security in Sensor Networks and Systems, 2006.

[Hog1] Lingyan Hong, Shijue Zheng, Xiangtao Qi, *Research on Task Assignment Algorithm of Mobile Learning System for Wireless Sensor Network*, etc, vol. 3, pp.177-180, 2009 First International Workshop on Education Technology and Computer Science, 2009.

[Hon1] Yul Pyo Hong , Young-Jae Choi , Tae Goo Kim , Yun Sik Kwak , Ki Hyung Joe , Hyung Il Kang , Seok Il Song, *Energy Efficient Data Centric Storage for Sensor Networks Employing Multilevel Grid Techniques*, 2008 Second International Conference on Future Generation Communication and Networking, December 2008, pp. 133-136.

[Hoo1] Hoon Kim; Young-Jun Jeon; Seung-Ho Shin; , *Platform of Rich Internet Application for Wireless Sensor Network*, 2008 International Conference on Computer Science and Software Engineering, 2008.

[Hua1] Vincent Huang and Muhammad Kashif Javed. *Semantic sensor information description and processing*. In Second International Conference on Sensor Technologies and Applications, 2008.

[Hwa1] Hwang, J. and Kim, Y. *Revisiting random key pre-distribution for sensor networks*. In ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN 04), 2004.

[Iee1] *IEEE 802.15.4*, <http://ieee802.org/15/pub/TG4.html>.

[Iee2] Institute of Electrical and Electronics Engineers, Inc., *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)*, New York, IEEE Press. October 1, 2003.

[Iee3] *IEEE 802.15 WPAN Task Group 1 (TG1)*, <http://www.ieee802.org/15/pub/TG1.html>.

[Iee4] *IEEE 802.11TM WIRELESS LOCAL AREA NETWORKS*, <http://www.ieee802.org/11/>.

[Ily1] M. Ilyas, and I. Mahgoub, *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, CRC Press, 2004.

[Isa1] *ISA100, Wireless Systems for Automation*, <http://www.isa.org/MSTemplate.cfm?MicrositeID=1134&CommitteeID=6891>.

[Isr1] Israel Pérez Llopis, *Arquitectura de un sistema C4ISR para pequeñas unidades*, Tesis Doctoral.

[Jad1] *JAVA Agent Development Framework*, <http://www.fipa.org/repository/aclspecs.html>.

[Jan1] Janev, V.; Vranes, S.; , *Semantic Web Technologies: Ready for Adoption?*, IT Professional, Sept.-Oct. 2009.

[Jan2] Jan Blumenthal, Ralf Grossmann, Frank Glatowski, Dirk Timmermann, *Weighted Centroid Localization in Zigbee-based Sensor Networks*, IEEE International Symposium on Intelligent Signal Processing (WISP2007), 2007.

[Jbi1] *Java Technology for Business Intelligence*, <http://java.sun.com/products/jmi/pres/guide.html>.

[Jey1] M.K. Jeya Kumar, R.S. Rajesh, *A Survey of MANET Routing Protocols in Mobility Models*, International Journal of soft Computing 4 (3),: 136-141, 2009.

[Jen1] *Jena – A Semantic Web Framework for Java*, <http://jena.sourceforge.net/>.

[Jin1] Jin Zhu; Papavassiliou, S.; Sheng Xu; , *Modeling and analyzing the dynamics of mobile wireless sensor networking infrastructures*, 2002 IEEE 56th Vehicular Technology Conference, 2002.

[Jup1] Ju-Peng Chen; Tsung-Han Lin; Polly Huang, *On the potential of sensor-enhanced active RFIDs*, Emerging Information Technology Conference, 2005.

[Kal1] K. Kalpakis, K. Dasgupta and P. Namjoshi, *Maximum Lifetime Data Gathering and Aggregation in Wireless Sensor Networks*, IEEE ICN, 2002.

[Kar1] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*, Wiley, 2005.

[Kar2] Al-Karaki, Kamal, *Routing Techniques in Wireless Sensor Networks: a Survey*, IEEE Wireless Communications, December 2004.

[Kar3] Kar-Keung D. Young, Yong Quan Ou, Jun Tao Feng, Zhi, Liang Ou, Lun Hui Cai, Ken Kin Man Cheng, Jason Kam On, Ho, and Timmy T M Tsang, *Automated navigation and mobile vehicle control using wireless sensor network technology*, Proceedings IEEE International Conference on Industrial Technology, 2008. ICIT 2008. Chengdu, China, 21-24 April, 2008.

[Kaw1] Kawakami, T.; Bich Lam Ngoc Ly; Takeuchi, S.; Teranishi, Y.; Harumoto, K.; Nishio, S.; *Distributed Sensor Information Management Architecture Based on Semantic Analysis of Sensing Data*, International Symposium on Applications and the Internet, 2008.

[Ke1] Wang Ke, Salma Abu Ayyash, Thomas Little, *Semantic Internetworking of Sensor Systems*, 2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems.

[Kem1] M.K. Jeya Kumar, R.S. Rajesh, *A Survey of MANET Routing Protocols in Mobility Models*, International Journal of soft Computing 4 (3), pp. 136-141, 2009.

[Kha1] Khattak, A.M.; Pervez, Z.; Sarkar, A.M.J.; Young-Koo Lee; , *Service Level Semantic Interoperability*, 10th IEEE/IPSJ International Symposium on Applications and the Internet (SAINT), 2010, 19-23 July 2010.

[Kou1] Koushanfar, F. Potkonjak, M. Sangiovanni-Vincentelli, A., *Fault tolerance techniques for wireless ad hoc sensor networks*, Sensors, 2002. Proceedings of IEEE.

[Lin1] Longbi Lin, Ness B. Shroff, R. Srikant, *Energy-aware routing in sensor networks: A large system approach*, Ad Hoc Networks, Volume 5, Issue 6 (August 2007), Pages: 818-831.

[Liu1] Jen-Chu Liu; Kuo-Yu Chuang; , *WASP: An innovative sensor observation service with web-/GIS-based architecture*, 2009 17th International Conference on Geoinformatics, Aug. 2009.

[Liu2] Liu, C.-H.; Chang, K.-L.; Chen, Jason J.-Y.; Hung, S.-C.; , *Ontology-Based Context Representation and Reasoning Using OWL and SWRL*, 2010 Eighth Annual Communication Networks and Services Research Conference (CNSR), 11-14 May 2010.

[Lop1] Lopes, F.; Novais, A.Q.; Mamede, N.; Coelho, H.; , *Negotiation Among Autonomous Agents: Experimental Evaluation of Integrative Strategies*, portuguese conference on Artificial intelligence, 2005. epia 2005., 5-8 Dec. 2005.

[Lqb1] M.Lqbal, H.B.Lim, W.Wang, Y.Yao, *A Service-Oriented Model for Semantics-based Data Management in Wireless Sensor Networks*, 2009 International Conference on Advanced Information networking and Applications Workshops.

[Mad1] Madureira, A.; Santo, J.; Gomes, N.; , *Hybrid Multi-agent System for Cooperative Dynamic Scheduling Through Meta-Heuristics*, Seventh International Conference on Intelligent Systems Design and Applications, 2007. ISDA 2007, 20-24 Oct. 2007.

[Maj1] Majavu, W.; van Zyl, T.; Marwala, T.; *Classification of web resident sensor resources using Latent Semantic Indexing and ontologies*, IEEE International Conference on Systems, Man and Cybernetics, 2008.

[Man1] Mansouri, V.S.; Afsari, B.; Shahmansouri, H.; , *A Simple Transport Protocol for Wireless Sensor Networks*, *Intelligent Sensors*, Proceedings of the 2005 International Conference on Sensor Networks and Information Processing Conference, 2005.

[Mat1] *MATLAB – The Language of Technical Computing*, www.mathworks.com/products/matlab.

[Mic1] *.NET Micro Framework*, <http://www.microsoft.com/netmf/default.aspx>.

[Mur1] Murugesan, S.; , *Understanding Web 2.0*, IT Professional, vol.9, no.4, pp.34-41, July-Aug. 2007.

[Mys1] *MySQL*, <http://www.mysql.com/>.

[Nan1] *Nano-RK*, www.nanork.org.

[Nes1] David Gay, Philip Levis, Robert von Behren, Matt Welsh, Eric Brewer, and David Culler, *The nesC Language: A Holistic Approach to Networked Embedded Systems*, Proc. ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2003), pp. 1-11, San Diego, CA, USA.

[Net1] *June 2003..NET Micro Framework*, <http://www.microsoft.com/netmf/default.aspx>.

[Ns2] *The Network Simulator - ns-2*, <http://www.isi.edu/nsnam/ns/>.

[Och1] Ochirsuren, E.; Indrusiak, L.S.; Glesner, M.; , *An Actor-Oriented Group Mobility Model for Wireless Ad Hoc Sensor Networks*, 28th International Conference on Distributed Computing Systems Workshops, 2008. ICDCS '08.

[Ogc1] *Open Geospatial Consortium*, <http://www.opengeospatial.org/>.

[Ogs1] *OGSi Alliance*, <http://www.osgi.org/Main/HomePage>.

[Olf1] Olfati-Saber, R.; Fax, J.A.; Murray, R.M.; , *Consensus and Cooperation in Networked Multi-Agent Systems*, *Proceedings of the IEEE* , vol.95, no.1, pp.215-233, Jan. 2007.

[Onw1] *Wireless Sensor Networks: Growing Markets, Accelerating Demand*, 2005, Mareca Hatler and Charlie Chi, OnWorld.

[Opn1] *OPNET*, <http://www.opnet.com>.

[Ort1] Antonio Ortiz, Teresa Olivares, Luis Orozco-Barbosa, *A Heterogeneous Role-based Sensor Network*, 2nd ACM International Workshop on Performance Monitoring, Measurement, and Evaluation of Heterogeneous Wireless and Wired Networks, PM2HW2N'07, October 26, 2007, Chania, Crete Island, Greece.

[Oul1] ElMoustapha Ould-Ahmed-Vall and George F. Riley and Bonnie S. Heck, *A Geometric-Based Approach to Fault-Tolerance in Distributed Detection Using Wireless Sensor Network* , *Information Processing in Sensor Network (IPSN2006)*, Nashville, TN, USA, April 19-21, 2006.

[Owl1] *OWL Web Ontology Language Overview*, <http://www.w3.org/TR/owl-features/>.

[Pat1] Al-Sakib Khan Pathan, Hyung-Woo Lee, Choong Seon Hong, *Security in Wireless Sensor Networks: Issues and Challenges*, 16th International Conference on Artificial Reality and Telexistence (ICAT2006), November 29- December 1st, 2006.

[Per1] Adrian Perrig, John Stankovic, David Wagner, *Security in wireless sensor networks*, *Communications of the ACM*, Volume 47, Issue 6 (June 2004).

[Pro1] *Protégé*, <http://protege.stanford.edu/>.

[Pyo1] Yul Pyo Hong , Young-Jae Choi , Tae Goo Kim , Yun Sik Kwak , Ki Hyung Joe , Hyung Il Kang , Seok Il Song, *Energy Efficient Data Centric Storage for Sensor Networks Employing Multilevel Grid Techniques*, 2008 Second International Conference on Future Generation Communication and Networking, December 2008, pp. 133-136.

[Rah1] Rahman, M.A.; El Saddik, A.; Gueaieb, W.; , *Data visualization: From body sensor network to social networks*, *IEEE International Workshop on Robotic and Sensors Environments*, 2009.

[Rah2] Rahman, M.A.; El Saddik, A.; Gueaieb, W.; , *Building Dynamic Social Network From Sensory Data Feed*, *IEEE Transactions on Instrumentation and Measurement* , vol.59, no.5, pp.1327-1341, May 2010.

[Rao1] Ananth Rao, Sylvia Ratnasamy, Christos Papadimitriou, Scott Shenker, and Ion Stoica, *Geographic Routing without Location Information*, *Proc. 9th International Conference on Mobile Computing and Networking (MobiCom 2003)*, pp. 96-108, San Diego, CA, USA, September 2003.

[Rdf1] *Resource Description Framework (RDF)*, <http://www.w3.org/RDF/>.

[Rdf2] *RDF Vocabulary Description Language 1.0: RDF Schema*, <http://www.w3.org/TR/rdf-schema/>.

[Rdq1] *RDQL - A Query Language for RDF*, <http://www.w3.org/Submission/RDQL/>.

[Rom1] K.Romer, *Wireless Sensor Network: from Science to Reality*, Tutorial, SENSORCOM2007.

[Rom2] Kay Römer, Christian Frank, Pedro José Marrón, Christian Becker, *Generic role assignment for wireless sensor networks*, Proceedings of the 11th workshop on ACM SIGOPS European workshop, 2004.

[Rug1] Ruggaber, R.; , *Internet of Services SAP Research Vision*, 16th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2007. WETICE 2007., 18-20 June 2007.

[Rui1] Linnyer Beatrys Ruiz, Isabela G. Siquiera, Leonardo B. e Oliveira, Hao Chi Wong, Jos'e Marcos S.Nogueira, Antonio A. F. Loureiro, *Fault management in event-driven wireless sensor networks*, Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems, Venice (Italy), 2004.

[San1] P. Santi, *Topology Control in Wireless Ad Hoc and Sensor Networks*, ACM Computing Surveys 37 (2):164-194, June 2005.

[Sat1] Santos, R.A.; Edwards, A.; Alvarez, O.; Gonzalez, A.; Verduzco, A. *A Geographic Routing Algorithm for Wireless Sensor Networks*, Electronics, Robotics and Automotive Mechanics Conference, 2006. Volume 1, Issue , 26-29 Sept. 2006 Page(s):64 – 69.

[Saz1] David Sánchez Sánchez, Heribert Baldus, *A Deterministic Pairwise Key Pre-distribution Scheme for Mobile Sensor Networks*, First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM'05), Athens, Greece, 5 - 9 September, 2005.

[Sca1]

	<i>Service</i>	<i>Component</i>	<i>Architecture</i> ,
http://www.ibm.com/developerworks/library/specification/ws-sca/ .			

[Sch1] Scholz, A.; Gaponova, I.; Sommer, S.; Kemper, A.; Knoll, A.; Buckl, C.; Heuer, J.; Schmitt, A.; *ESOA - Service Oriented Architectures adapted for embedded networks*, 7th IEEE International Conference on Industrial Informatics, 2009. INDIN 2009.

[Sch2] Schroth, Christoph; , *The internet of services: Global industrialization of information intensive services*, 2nd International Conference on Digital Information Management, 2007. ICDIM '07. , 28-31 Oct. 2007.

[See1] See-Kee Ng; Seah, W.K.G.; , *Game-Theoretic Model for Collaborative Protocols in Selfish, Tariff-Free, Multihop Wireless Networks*, INFOCOM 2008. The 27th Conference on Computer Communications. April 2008.

[Sen1] *SENTILLA*, www.sentilla.com.

[Sen2] *Sensor Model Language (SensorML)*, <http://www.opengeospatial.org/standards/sensorml>.

[She1] Amit Sheth, Cory Henson, and Satya S. Sahoo. *Semantic sensor web*. IEEE Internet Computing, 12(4):78–83, 2008.

[Shi1] Elaine Shi, Adrian Perrig, *Designing Secure Sensor Networks*, IEEE Wireless Communications, pp. 38-43, December 2004.

[Sho1] *SHOCKFISH*, www.shockfish.com.

[Shu1] Lei Shu, Yan Zhang, Laurence T. Yang, Yu Wang, Manfred Hauswirth, *Geographic Routing in Wireless Multimedia Sensor Networks*, 2008 Second International Conference on Future Generation Communication and Networking.

[Sig1] *OwlSight*, <http://pellet.owldl.com/ontology-browser/>.

[Sir1] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, Yarden Katz, *Pellet: A practical OWL-DL reasoner*, Web Semantics: Science, Services and Agents on the World Wide Web, Volume 5 Issue 2, 2007.

[Soh1] K.Sohrabi, J.Gao, V.Ailawadhi and G.Pottie, *Protocols for Self-Organization of a Wireless Sensor Network*, IEEE Personal Communications, pp. 16-27, October 2000.

[Spa1] *SPARQL Query Language for RDF*, <http://www.w3.org/TR/rdf-sparql-query/>.

[Spi1] Patrik Spiess, Stamatis Karnouskos, Dominique Guinard, Domnic Savio, Oliver Baecker, Luciana Moreira S´ de Souza , and Vlad Trifa, *SOA-based Integration of the Internet of Things in Enterprise Services*, 2009 IEEE International Conference on Web Services.

[Sto1] Stollberg, M.; Haller, A.; , *Semantic Web services tutorial*, 2005 IEEE International Conference on Services Computing, 11-15 July 2005.

[Sto2] Radu Stoleru, Tian He, John A. Stankovic, *Localization System for Outdoor Wireless Sensor Networks*, Proceedings of the 3rd international conference on Embedded networked sensor systems, San Diego, California, USA, 2005.

[Su1] Su, H.; Zhang, X.; , *Energy-Efficient Clustering System Model and Reconfiguration Schemes for Wireless Sensor Networks*, 2006 40th Annual Conference on Information Sciences and Systems, March 2006.

[Sun1] *Sun*, www.sun.com.

[Sun2] *Sun SPOT*, www.sunspotworld.com.

[Sun3] Sunho Lim; Chansu Yu; Das, C.R.; , *Clustered Mobility Model for Scale-Free Wireless Networks*, 31st IEEE Conference on Local Computer Networks, 2006.

[Sun4] Sun Xi, Xia-Miao Li, Study of the Feasibility of VANET and its Routing Protocols, 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008 (WiCOM '08), 12-14 Oct. 2008 Page(s):1 – 4.

[Swr1] SWRL: A Semantic Web Rule Language; Combining OWL and RuleML <http://www.w3.org/Submission/SWRL/>.

[Tav1] Arsalan Tavakoli, David Chu, Joseph Hellerstein, Philip Levis, and Scott Shenker. *A Declarative Sensornet Architecture*. International Workshop on Wireless Sensor Network Architecture (WWSNA 2007), Cambridge, Massachusetts, April 25-27, 2007.

[**Thi1**] Thirunarayan, K.; Henson, C.A.; Sheth, A.P.; , *Situation awareness via abductive reasoning from Semantic Sensor data: A preliminary report*, International Symposium on Collaborative Technologies and Systems, 2009.

[**Thr1**] K. C. Thramboulidis ,G. Doukas, G. Koumoutsos, *A SOA-based embedded systems development for industrial automation*, EURASIP Journal on Embedded Systems, Volume 2008 (April 2008).

[**Til1**] S. Tilak, N. B. Abu-Ghazaleh and W. Heinzelman, *Infrastructure Tradeoffs for Sensor Networks*, WSNA02, September 2002, Atlanta, Georgia, USA.

[**Tin1**] *TinyOS*, www.tinyos.org.

[**Tin2**] *TinyOS 2.0*, Proceedings of the 6th ACM conference on Embedded network sensor systems, 2005.

[**Tin3**] *TinyOS 2.1 adding threads and memory protection to TinyOS*, Proceedings of the 6th ACM conference on Embedded network sensor systems, 2008.

[**Tos1**] *Simulating TinyOS Networks*, <http://www.cs.berkeley.edu/~pal/research/tossim.html>.

[**Tse1**] Vassileios Tsetsos, George Alyfantis, Tilemahos Hasiotis, Odysseas Sekkas, and Stathes Hadjiefthymiades, *Commercial Wireless Sensor Networks: Technical and Business Issues*, Second Annual Conference on Wireless On-demand Network Systems and Services (WONS'05).

[**Vid1**] R.Vidhyapriya, P.T.Vanathi, *Energy Aware Routing for Wireless Sensor Networks*, International Conference on Signal Processing, Communications and Networking, 2007 (ICSCN '07).

[**Vir1**] Virk, K.; Hansen, K.; Madsen, J.; , *System-level Modeling of Wireless Integrated Sensor Networks*, 2005 International Symposium on System-on-Chip, Nov. 2005.

[**W3c1**] *W3C, World Wide Web Consortium*, www.w3c.com.

[**W3c2**] *Web Services Description Language (WSDL) 1.1*, <http://www.w3.org/TR/wsdl>.

[**W3c3**] *SOAP Version 1.2*, <http://www.w3.org/TR/soap/>.

[**Wad1**] *WADL*, <https://wadl.dev.java.net/>.

[**Wan1**] Yuanli Wang, Xianghui Liu, Jianping Yin, *Requirements of Quality of Service in Wireless Sensor Network*, International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06).

[**Wan2**] Qin Wang; Hempstead, M.; Yang, W.; , *A Realistic Power Consumption Model for Wireless Sensor Network Devices*, 2006 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks, 2006.

[**Wan3**] Wang-Yan; , *Study on Model and Architecture of Self-Organization Wireless Sensor Network*, International Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08.

[**Wcf1**] *Windows Communication Foundation*, <http://msdn.microsoft.com/en-us/netframework/aa663324.aspx>.

[Wen1] Wen-Bin Hu; Shao-Mei Wang; , *Research on the negotiation mechanism of multi-agent system based on game theory*, Proceedings of the Ninth International Conference on Computer Supported Cooperative Work in Design, 2005, 24-26 May 2005.

[Wif1] *WiFi Alliance*, <http://www.wi-fi.org/>.

[Wik1] *Ontologia (Informatica)*, Wikipedia – La enciclopedia libre, [http://es.wikipedia.org/wiki/Ontologia_\(informatica\)](http://es.wikipedia.org/wiki/Ontologia_(informatica)).

[Wir1] *WirelessHART*, www.hartcomm.org.

[Xia1] Xiao Laisheng; Peng Xiaohong; Wang Zhengxia; Xu Bing; Hong Pengzhi; , *Research on Traffic Monitoring Network and Its Traffic Flow Forecast and Congestion Control Model Based on Wireless Sensor Networks*, International Conference on Measuring Technology and Mechatronics Automation, 2009. ICMTMA '09.

[Xia2] Xiaolin Lu; , *Develop Web GIS Based Intelligent Transportation Application Systems with Web Service Technology*, 2006 6th International Conference on ITS Telecommunications, June 2006.

[Xin1] Xinyang Feng; Jianjing Shen; Ying Fan; , *REST: An alternative to RPC for Web services architecture*, International Conference on Future Information Networks, 2009. ICFIN 2009.

[Xin2] Xing, L.; Michel, H.E.; , *Integrated modeling for wireless sensor networks reliability and security*, Annual Reliability and Maintainability Symposium, 2006. RAMS '06.

[Xue1] Hua Xue; Chennian Di, and J.A.G. Jess, *A Net-Oriented Method for Realistic Fault Analysis*, Proceedings of the 1993 IEEE/ACM international conference on Computer-aided design, Santa Clara, California, United States, 1993.

[Yan1] Yu Yang, Zhang Xiaomin, Peng BO, Fu Yujing, *Design of Sensor Nodes In Underwater Sensor Networks*, The 4th IEEE Conference on Industrial Electronics and Applications (ICIEA 2009), 25-27 May, 2009.

[Yan2] Yandong Wang; You He; Jianya Gong; Jizhen Sheng; , *A Framework of Spatial Sensor Web*, Third International IEEE Conference on Signal-Image Technologies and Internet-Based System, 2007.

[Yan3] Yang Hong; Chi Fat Chan; Jianping Guo; Yuen Sum Ng; Weiwei Shi; Ho, Marco; Lai Kan Leung; Ka Nang Leung; Chiu Sing Choy; Kong Pang Pun; , *Design and challenges of passive UHF RFID tag in 90nm CMOS technology*, IEEE International Conference on Electron Devices and Solid-State Circuits, 2008. EDSSC 2008.

[You1] Younghwan Choi; Soochang Park; Donghun Lee; Euisin Lee; Fucai Yu; Sang-Ha Kim, *User mobility model and data dissemination scheme for wireless sensor networks*, Asia-Pacific Conference on Communications, 2007. APCC 2007. Volume , Issue , 18-20 Oct. 2007 Page(s):369 – 372.

[Yua1] Wei Yuan, Srikanth V. Krishnamurthy, and Satish K. Tripathi, *Synchronization of Multiple Levels of Data Fusion in Wireless Sensor Networks*, In Proceedings of GLOBECOM, 2003.

[Zeu1] Nwana et al., *ZEUS: A tool-kit for building distributed multi-agent systems*, Applied Artificial Intelligence Journal. v13 i1. 129-186, 1999.

[Zhe1] J. Zheng and M.J. Lee, *Will IEEE 802.15.4 Make Ubiquitous Networking a Reality?*, IEEE Communication Mag., Jun 2004.

[Zig1] *ZigBee Alliance*, www.zigbee.org.

[Zou1] Zouganeli, E.; Svinnset, I.E.; , *Connected objects and the Internet of things — A paradigm shift*, International Conference on Photonics in Switching, 2009. PS '09.

Resumen

Síntesis

Durante los últimos años las redes de sensores inalámbricas han sido objeto, como consecuencia de un creciente interés comercial, de una intensa actividad de investigación que ha determinado relevantes avances tanto en la tecnología base como en los aspectos de ingeniería a todos los niveles.

Las redes de sensores inalámbricas se basan en el concepto de nodo sensor autónomo de bajo coste que proporciona recursos limitados en términos de cálculo y capacidad de almacenamiento de información, baja potencia de transmisión y sensorica avanzada. Se caracterizan por el tamaño extremadamente reducido y una ingeniería orientada a la eficiencia energética.

A pesar de la disponibilidad de soluciones altamente avanzadas, caracterizadas por la eficiencia y la flexibilidad, la difusión comercial masiva se ha planteado más veces como hipótesis plausible y además parece tardar en concretarse de forma definitiva. Las principales causas están relacionadas, directamente o indirectamente, con dos factores: coste elevado y falta de suficiente fiabilidad/robustez.

Una de las consecuencias del desarrollo de arquitecturas “ad-hoc” que caracteriza actualmente las redes de sensores inalámbricas es la de garantizar una gran cantidad de óptimos locales siendo la causa principal de una preocupante ausencia de estándares tanto en términos de protocolos de comunicación como en términos de organización y representación de información. También nuevos modelos de negocio y de explotación dentro de las organizaciones virtuales de última generación son actualmente temas de atención en el seno de la comunidad científica internacional.

Objetivos

Este trabajo se sitúa en el marco de las últimas líneas de investigación orientadas a conciliar soluciones avanzadas, caracterizadas por una ingeniería innovadora, con su aplicación efectiva en el mundo real siguiendo principios de escalabilidad y manteniendo siempre el principio de bajo coste. Considerando la aplicación de la arquitectura, en términos no solamente tecnológicos si no también comerciales, como elemento primario y determinante, la solución propuesta en la presente tesis se caracteriza por su independencia respecto a un dominio aplicativo predefinido al proporcionar un marco flexible que puede ser particularizado en función del entorno, de la aplicación y relacionarlo con requisitos funcionales y no funcionales. La arquitectura propuesta, al contrario de la mayoría de las soluciones existentes, sería operativa tanto para entornos estáticos (o semi-estáticos) como para entornos caracterizados por la presencia de nodos y actores móviles.

Resultados logrados

Desde un punto de vista metodológico, se considera de gran importancia proporcionar un modelo teórico de referencia para diferentes aspectos de la arquitectura propuesta: modelo de conectividad-cobertura, clustering dinámico y comportamiento de los actores móviles. El objetivo principal de la modelización es proporcionar una sólida base teórica, independiente de implementaciones concretas, para el desarrollo de ambientes de simulación avanzados orientados a soportar tanto la fase de diseño previo como la fase de evaluación preliminar con especial atención a la escalabilidad.

Paralelamente, la implementación de la plataforma y la experimentación de la misma en entornos de trabajo reales proporcionan un feedback vital para el refinado de los modelos y de los ambientes de simulación (caracterizando y validando entornos,

plataformas y aplicaciones mediante datos reales). Al mismo tiempo permite comparar las prestaciones reales con las esperadas evaluar el impacto de fenómenos impredecibles (interferencias temporáneas o permanentes, obstáculos, etc.).

Elementos de la metodología a destacar

Considerando el estado del arte global de las arquitecturas de redes sensores inalámbricos, los cinco ejes del desarrollo de la presente tesis serían:

- Método “horizontal” en contraposición al tradicional método de estudio “vertical” que caracteriza la mayoría de los proyectos sobre redes de sensores inalámbricas.
- Ingeniería orientada a la reducción de los costes a través de soluciones flexibles multidominio.
- Elaboración de aspectos teóricos con sucesiva evaluación experimental de los modelos propuestos en varios entornos de aplicación.
- Soporte para entornos estáticos, nomádicos y con nodos sensores/actores móviles.
- Diseño de una arquitectura flexible y dinámica apta a soportar diferentes modelos de interacción e interoperabilidad en entornos aplicativos de diferente escala y complejidad.

Estructura de la tesis

El trabajo propuesto se estructura en tres partes estrechamente relacionadas y que conforman la tesis doctoral desarrollada:

- La primera parte describe el diseño de la arquitectura propuesta. Desde un punto de vista estructural, dicha arquitectura consta de tres componentes fundamentales: redes de sensores, infraestructuras middleware y sistema de información. Desde un punto de vista conceptual, se estructura de acuerdo a cuatro conceptos progresivos: recursos físicos, recursos virtuales, Sensor web y, finalmente, Semantic sensor web.
- La segunda parte, con enfoque fundamentalmente teórico, se centra en la elaboración de un modelo analítico, tanto para entornos estáticos como nomádicos y móviles, de la arquitectura definida. El conjunto de los modelos propuestos se caracteriza por su independencia respecto a aspectos específicos de dominios de aplicación concretos; este aspecto se considera absolutamente fundamental para proporcionar un enfoque general a los varios análisis y resultados relacionados. Dicho modelo es el fundamento teórico para el entorno de simulación propuesto. También se proporciona una introducción a los posibles modelos de análisis y evaluación de arquitecturas reales complejas.
- La tercera parte del trabajo de tesis, con carácter más aplicativo y experimental se centra en el desarrollo del framework de referencia propuesto y en su particularización e integración para soporte de aplicaciones concretas. Los dominios aplicativos objeto de interés en el ámbito del trabajo de tesis se caracterizan por requisitos funcionales complejos que determinan soluciones innovadoras y conceptualmente avanzadas. La arquitectura se ha implementado como se presentará en el capítulo correspondiente mediante la utilización de los nodos MicaZ de Crossbow.

Resum

Síntesi

Durant els últims anys les xarxes de sensors sense fils han sigut objecte, com a conseqüència d'un creixent interès comercial, d'una intensa activitat d'investigació que ha esdevingut en rellevants avanços tant en la tecnologia base com en els aspectes d'enginyeria a tots els nivells.

Les xarxes de sensors sense fils es basen en el concepte de “node sensor autònom de baix cost” que proporciona recursos limitats en termes de càlcul i capacitat d'emmagatzemament d'informació, baixa potència de transmissió i sensorica avançada. Es caracteritzen per la grandària extremadament reduïda i una enginyeria orientada a l'eficiència energètica.

A pesar de la disponibilitat de solucions altament avançades, caracteritzades per l'eficiència i la flexibilitat, la difusió comercial massiva s'ha plantejat més vegades com a hipòtesi plausible i a més pareix tardar a concretar-se de forma definitiva. Les principals causes estan relacionades, directament o indirectament, amb dos factors: cost elevat i falta de suficient fiabilitat/robustesa.

Una de les conseqüències del desenvolupament d'arquitectures “ad-hoc” que caracteritza actualment les xarxes de sensors sense fils és la de garantir una gran quantitat d'òptims locals éssent la causa principal d'una preocupant absència d'estàndards tant en termes de protocols de comunicació com en termes d'organització i representació d'informació. També els nous models de negoci i d'explotació dins de les organitzacions virtuals d'última generació són actualment temes d'atenció en el si de la comunitat científica internacional.

Objectius

Este treball se situa en el marc de les últimes línies d'investigació orientades a conciliar solucions avançades, caracteritzades per una enginyeria innovadora, amb la seua aplicació efectiva en el món real seguint principis d'escalabilidad i mantenint sempre el principi de baix cost. Considerant l'aplicació de l'arquitectura, en termes no sols tecnològics si no també comercials, com a element primari i determinant, la solució proposada en la present tesi es caracteritza per la seua independència respecte a un domini aplicatiu predefinit, en proporcionar un marc flexible que pot ser particularitzat en funció de l'entorn, de l'aplicació i relacionar-ho amb requisits funcionals i no funcionals. L'arquitectura proposada, al contrari de la majoria de les solucions existents, seria operativa tant per a entorns estàtics (o semi-estàtics) com per a entorns caracteritzats per la presència de nodes i actors mòbils.

Resultats aconseguits

Des d'un punt de vista metodològic, es considera d'una gran importància proporcionar un model teòric de referència per a diferents aspectes de l'arquitectura proposada: model de connectivitat-cobertura, clustering dinàmic i comportament dels actors mòbils. L'objectiu principal de la modelització és proporcionar una sòlida base teòrica, independent d'implementacions concretes, per al desenrotllament d'ambients de simulació avançats orientats a suportar tant la fase de disseny previ com la fase d'avaluació preliminar amb especial atenció a l'escalabilidad.

Paral·lelament, la implementació de la plataforma i l'experimentació de la mateixa en entorns de treball reals proporcionen un feedback vital per al refinat dels models i dels ambients de simulació (caracteritzant i validant entorns, plataformes i aplicacions per mitjà de dades reals). Al mateix temps permet comparar les prestacions reals amb les esperades i avaluar l'impacte de fenòmens impredecibles (interferències temporànees o permanents, obstacles, etc.).

Elements de la metodologia a destacar

Considerant l'estat de l'art global de les arquitectures de xarxes sensors sense fils, els cinc eixos del desenvolupament de la present tesi serien:

- Mètode “horitzontal” en contraposició al tradicional mètode d'estudi “vertical” que caracteritza la majoria dels projectes sobre xarxes de sensors sense fils.
- Enginyeria orientada a la reducció dels costos a través de solucions flexibles multi-domini.
- Elaboració d'aspectes teòrics amb successiva avaluació experimental dels models proposats en diversos entorns d'aplicació.
- Suport per a entorns estàtics, nomádics i amb nodes sensors/actors mòbils.
- Disseny d'una arquitectura flexible i dinàmica apta a suportar diferents models d'interacció i interoperabilitat en entorns aplicatius de diferent escala i complexitat.

Estructura de la tesi

El treball proposat s'estructura en tres parts estretament relacionades i que conformen la tesi doctoral desenvolupada:

- La primera part descriu el disseny de l'arquitectura proposada. Des d'un punt de vista estructural, la dita arquitectura consta de tres components fonamentals: xarxes de sensors, infraestructures middleware i sistema d'informació. Des d'un punt de vista conceptual, s'estructura d'acord amb quatre conceptes progressius: recursos físics, recursos virtuals, Sensor web i, finalment, Semantic sensor web.
- La segona part, amb enfocament fonamentalment teòric, se centra en l'elaboració d'un model analític, tant per a entorns estàtics com nomádics i mòbils, de l'arquitectura definida. El conjunt dels models proposats es caracteritza per la seua independència respecte a aspectes específics de dominis d'aplicació concrets; este aspecte es considera absolutament fonamental per a proporcionar un enfocament general a les unes quantes anàlisis i resultades relacionades. El dit model és el fonament teòric per a l'entorn de simulació proposat. També es proporciona una introducció als possibles models d'anàlisi i avaluació d'arquitectures reals complexes.
- La tercera part del treball de tesi, amb caràcter més aplicatiu i experimental se centra en el desenrotllament del framework de referència proposat i en la seua particularització i integració per a suport d'aplicacions concretes. Els dominis aplicatius objecte d'interés en l'àmbit del treball de tesi es caracteritzen per requisits funcionals complexos que determinen solucions innovadores i conceptualment avançades. L'arquitectura s'ha implementat com es presentarà en el capítol corresponent per mitjà de la utilització dels nodes MicaZ de Crossbow.

Abstract

Summary

During last years, Wireless Sensor Network has been object of an intense research activity as consequence of an increasing commercial interest. This activity has determined relevant advances for both the base technology and the final architectures.

Wireless Sensor Network (WSN) is based on the concept of low cost autonomous node that provides limited resource in terms of data processing and storage, low-power transmission and advanced sensors. This class of sensor device is featured by reduced sizes and by an engineering oriented to the power optimization.

Even if advanced, efficient and flexible solutions are currently available, the massive commercialization of WSN, normally considered as next to be a fact, is not appening. The reasons are related, directly or indirectly, to two main factors: high overall cost and inadequate level of reliability/robustness.

One of the most relevant consequences of the ad-hoc approach, that at the moment characterizes the development of WSNs, is the definition of several local optimums for the problems but, at the same time, a fundamental lack of estandardiation and generalization of the solutions. In this sense, innovative explotation models for WSN are object of interest for the scientific community.

Goals

This work was planned according to the last research lines that propose innovative and advanced solutions able to work in the real word in a context of scalability, flexibility and economic sustainability.

In order to assure a certain competitiveness, the proposed architecture is designed without any dependence or relation with concrete applications or technologies. This approach deterimines a flexibility of the reference architecture that can be particularized in function of concrete applications and in function of related functional/non functional requirements.

The proposed architecture is also independent of the nodes behaviour that can be static, nomadic or mobile.

Results

By methodological point of view, it is important to design a reference model for the different aspects of the network (connectivity model, dynamic clustering, mobile behaviours). The model can assure a solid theoretical ground, independent of any concrete application domain or technology, that can be useful for the different phases of the development of concrete architectures (design, simulation, preliminarily evaluation, refination).

At the same time, the implementation and the experimentation of the platform within real environments provide a vital feedback for the refination of the models and of the framework.

Methodology

The methodology adopted is summarized by the following points:

- Horizontal approach as contradistinction of the tradictional vertical approach that characterizes the great part of similar works.
- Overall approach oriented to cost reduction using multi-domain flexible solutions.
- Elaboration of theoretical models refined using the information provided by the experimental evaluation in several application domains.

- Support for static, nomadic and mobile nodes and actors.
- Development of a reference architecture able to work within several logic environments according to different interaction and interoperability models, as well as different scale and complexity.

Structure of the Work

The work is structured in three interdependent logic sections:

- The first one describes the design of the reference architecture. By a functional perspective, three different layers can be identified: sensor network, middleware infrastructure and information system. By a conceptual point of view, the architecture is designed and defined according to four progressive concepts: Physical Resource, Virtual Resource, Sensor Web and Semantic Sensor Web.
- The second logic section is entirely focused on WSN and it has a theoretical focus. Several related models are proposed in order to have an analytic tool that should support the designer during the different phases of the deployment (design, simulation, implementation, preliminary evaluation).
- The last logic section describes the implementation of the reference architecture and its particularization and experimentation in several application domains.

