

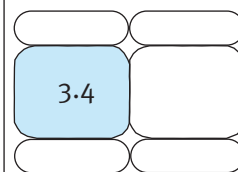


# Tecnologie alla ribalta

## RETI DI MICRODISPOSITIVI INTELLIGENTI

Le *Wireless Sensor Network* ed i *tag RFID* sono tecnologie recenti con grandi potenzialità applicative che causeranno un aumento del grado di penetrazione dei dispositivi elettronici e dei sistemi informatici nella vita di ogni giorno. Al di là dei timori suscitati da alcuni recenti interventi sui mass media, questi nuovi strumenti hanno tutte le carte in regola per incidere profondamente non solo nel mondo dell'ICT, ma anche in molti dei tradizionali settori industriali. Questo articolo presenta una panoramica di queste tecnologie.

Mirko Cesarini  
Carlo Ghezzi  
Fabio Schreiber  
Letizia Tanca



### 1. INTRODUZIONE

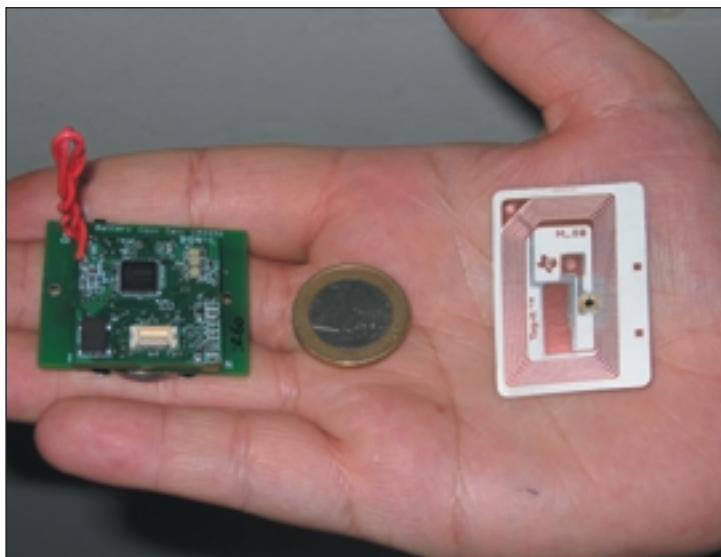
**N**egli ultimi anni il progresso compiuto nel campo della miniaturizzazione degli apparati di elaborazione, memorizzazione e comunicazione ha permesso la creazione di nuove famiglie di dispositivi che per le loro caratteristiche possono essere distribuiti in grandi quantità nell'ambiente per svolgere funzioni di rilevazione dati, di controllo della movimentazione delle merci e in generale per realizzare nuove e avanzate forme di interazione con il mondo circostante. Esempi d'uso si trovano nella logistica, nei sistemi di telecontrollo ed in generale nelle applicazioni di "ambient intelligence".

Tra gli esempi più noti e interessanti di questi sistemi periferici intelligenti e pervasivi troviamo le reti wireless di sensori (*Wireless Sensor Network*, WSN) e le etichette (*tag*) RFID (*Radio Frequency Identification*). Su di essi si concentra lo studio riportato in questo articolo. L'impiego delle WSN e dei *tag* RFID permette di far uscire le informazioni digitali dai confini dei server aziendali per distribuirle sugli oggetti e negli ambienti fisici. Grazie

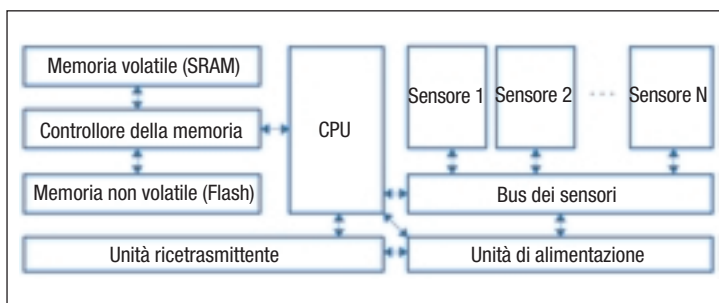
a ciò si aprono nuove possibilità per quanto riguarda la progettazione di software e di sistemi informativi aziendali. Nel corso dell'articolo forniremo una panoramica sugli aspetti più salienti delle due tecnologie e ci focalizzeremo su come questi possono influenzare il processo di sviluppo del software ed i sistemi per la gestione dei dati.

Nella loro forma più nota, le WSN sono reti formate da tanti piccoli elementi chiamati *mote*, dal nome utilizzato per i primi prototipi di WSN sviluppati dall'Università della California Berkeley (USA). Un *mote* è un dispositivo di elaborazione in miniatura dotato di memoria, di un apparato ricetrasmittente, di sensori per la rilevazione di dati ambientali e di una batteria per l'alimentazione. I *mote* (Figura 1) vengono progettati in modo che, una volta accesi e inseriti nel luogo di funzionamento, siano in grado di operare a lungo senza intervento umano.

I *mote* sfruttano le trasmissioni radio per comunicare tra loro e per trasmettere i dati raccolti. L'assenza di cavi di alimentazione e di trasmissione dati facilita il dispiegamento del-



**FIGURA 1**  
Da sinistra verso destra sono visibili rispettivamente un mote, una moneta da 1 euro ed un RFID



**FIGURA 2**  
Schema hardware di un mote

le WSN, consentendone l'utilizzo in tutte quelle situazioni dove i tradizionali sensori cablati non possono essere impiegati efficacemente. Un tag RFID è una memoria, accoppiata ad una antenna di trasmissione, che può essere interrogata a distanza da un apposito lettore. Un tag per funzionare sfrutta l'energia irradiata dai dispositivi con i quali interagisce e per questo motivo normalmente non ha bisogno di una batteria<sup>1</sup>; ciò rende possibile produrre dei tag con le dimensioni ed il volume di un francobollo (Figura 1) e con costi molto bassi. Una volta posizionati su oggetti fisici, gli RFID memorizzano informazioni in formato digitale che possono essere lette a distanza. Gli RFID potenzialmente possono rivoluzionare molti set-

<sup>1</sup> Per essere precisi, all'interno della famiglia dei tag RFID, solo i tag passivi non usano batterie. Per una classificazione dettagliata si rimanda al punto 3.1.

tori applicativi: si tratta di una tecnologia che offre funzionalità di identificazione e di memorizzazione equivalenti a quelle fornite dai codici a barre e che supera alcune delle limitazioni che affliggono le tecnologie ottiche (i primi sistemi RFID sono stati sviluppati proprio con l'obiettivo di sostituire i codici a barre).

Il resto di questo articolo è organizzato nel modo seguente: nel paragrafo 2 analizzeremo le reti wireless di sensori, nel paragrafo 3 analizzeremo in dettaglio la tecnologia RFID. In entrambi i paragrafi ci focalizzeremo sulle problematiche connesse alla gestione dei dati negli scenari d'impiego delle nuove tecnologie. Infine nel paragrafo 4 verranno tratte alcune conclusioni e descritti i possibili sviluppi futuri.

## 2. RETI WIRELESS DI SENSORI

### 2.1. Architettura hardware

Un mote di una rete di sensori è descritto dallo schema riportato nella figura 2. I componenti principali sono: l'unità di elaborazione, le memorie, l'unità ricetrasmittente e l'insieme di sensori di cui il mote è corredato.

Nel seguito di questo paragrafo analizzeremo con qualche dettaglio i singoli componenti dell'architettura hardware delle WSN. Successivamente al punto 2.2 descriveremo l'architettura software e infine al punto 2.3 illustreremo un caso applicativo.

#### 2.1.1. UNITÀ DI ELABORAZIONE

Rispetto ai sensori tradizionali, i componenti di una WSN hanno la capacità di elaborare le informazioni: un mote può rimuovere il rumore di fondo dai segnali raccolti, trasformare i dati raccolti in informazioni di più alto livello e più sintetiche (esempio, il calcolo della media delle temperature rilevate in un periodo) o aggregarle con informazioni ricevute dagli altri mote prima di trasmetterle al destinatario. Queste operazioni sono svolte dall'unità di elaborazione, la quale oltre a elaborare i dati raccolti dai sensori si occupa anche di gestire le risorse di un mote. I microprocessori impiegati devono essere ottimizzati per consumare poca energia, che costituisce una risorsa scarsa nelle normali condizioni di utilizzo, infatti i mote difficilmente possono essere recuperati una volta dispiegati nell'ambiente di lavoro; per

questo motivo prolungare la durata della batteria è importante. Le unità di elaborazione attualmente disponibili sono a 8, 16 e 32 bit. Come criterio per il confronto si possono considerare sia le prestazioni di punta del processore sia l'indice MIPS/Watt che fornisce un'indicazione di quale sia il costo della potenza di calcolo erogata in termini di energia consumata. In linea generale si può affermare che i microprocessori a 32 bit offrono elevate prestazioni di calcolo ma consumano più energia; al contrario, i processori a 8 bit sono parchi nel consumo di energia, ma a scapito delle prestazioni.

### 2.1.2. MEMORIE

I *mote* usano memorie SRAM come memoria di lavoro e memorie flash come memoria di massa. Le memorie SRAM generalmente sono integrate nel processore; le memorie flash hanno capacità di poche centinaia di kbyte, arrivano ad alcuni Mbyte solo nei *mote* più dotati. Le memorie flash conservano il loro contenuto anche quando non vengono alimentate e sono utilizzate per ospitare sia i programmi sviluppati dagli utenti per i *mote* sia i dati rilevati dai sensori in attesa di essere elaborati o trasmessi.

### 2.1.3. UNITÀ RICETRASMITTENTE

Sono disponibili diverse soluzioni per realizzare il sistema di comunicazione di un *mote*. Alcuni *mote*, per esempio i *mote* Crossbow [40] nella serie Mica2, utilizzano dei protocolli di comunicazione proprietari, altri utilizzano protocolli standard come Bluetooth [37], IEEE 802.15.4/ZigBee [63] e IEEE 802.11 (WiFi) [66]. IEEE 802.11 (WiFi) è un protocollo per wireless LAN che risulta carente sotto il profilo del risparmio energetico e per questo motivo è scarsamente utilizzato nelle WSN. Bluetooth e IEEE 802.15.4/ZigBee sono protocolli per WPAN (*Wireless Personal Area Network*). Le WPAN sono nate per sostituire, con collegamenti radio, i cavi di connessione dei vari dispositivi "indossati" da una persona e implementano strategie di risparmio energetico. Nel riquadro è riportato un confronto tra Bluetooth e ZigBee.

Ogni *mote* poi mette a disposizione dei programmatori servizi per crittografare i dati trasmessi.

#### Confronto tra Bluetooth e ZigBee

Concepita inizialmente per sostituire i cavi di collegamento tra PC, telefoni e altri dispositivi elettronici, la tecnologia Bluetooth si è affermata anche per realizzare piccole reti locali in grado di riconfigurarsi automaticamente man mano che si presentano nuovi dispositivi nell'area di copertura. La tecnologia Bluetooth, regolata dalle norme IEEE 802.15.1, utilizza la frequenza radio ISM (*Industrial Scientific Medical*) a 2,45 GHz, di libero uso in tutto il mondo. I dispositivi hanno un raggio utile di comunicazione di circa 10 m in linea d'aria (100 m con Bluetooth 2) che si riduce in presenza di ostacoli. Bluetooth è stato concepito per connettere tra loro un numero limitato di dispositivi (se sono presenti più di 8 dispositivi, la rete si segmenta in gruppi di massimo 8 elementi) garantendo una banda di 1 Mbyte/s, sufficiente per trasmettere voce e video. I dispositivi sono ottimizzati per limitare il consumo energetico delle batterie: le potenze impiegate per trasmettere sono basse e i dispositivi di comunicazione rimangono per la maggior parte del tempo in modalità di funzionamento a basso consumo energetico.

IEEE 802.15.4/ZigBee è una specifica per reti di dispositivi a basso consumo energetico che offre una banda di 250 kbit/s, dà la possibilità di gestire connessioni che coinvolgono fino a 255 altri dispositivi, utilizza la frequenza radio ISM (*Industrial Scientific Medical*) a 2,45 GHz. La massima distanza di comunicazione da un dispositivo all'altro, come per Bluetooth è di circa 10 m in linea d'aria, un po' meno con ostacoli in mezzo. A differenza di Bluetooth, ZigBee è ottimizzato per connettere un elevato numero di dispositivi che comunicano sporadicamente tra di loro e che per le loro comunicazioni non necessitano di banda elevata.

### 2.1.4. SENSORI

I *mote* hanno sia sensori built-in sia un bus di espansione per mezzo del quale è possibile aggiungere altri sensori necessari per svolgere funzioni specifiche. Sono disponibili molti tipi di sensori, per esempio rilevatori di temperatura, rumore, ricevitori GPS ecc.. È tuttavia possibile costruire un sensore ad hoc specifico per le proprie esigenze.

Un sensore è composto generalmente da un trasduttore e da un convertitore di segnali analogici in digitali. I trasduttori sono costruiti sfruttando le caratteristiche di certi materiali che variano le loro caratteristiche elettriche al variare delle condizioni ambientali. Un ADC (*Analog to Digital Converter*, convertitore di segnali analogici in segnali digitali) converte il valore di tensione su un trasduttore in un valore binario che verrà poi utilizzato per le successive elaborazioni.

Molti dei trasduttori utilizzati sui *mote* sono dei MEMS (*MicroElectroMechanical Systems*): si tratta di dispositivi in grado di rilevare una vasta gamma di fenomeni fisici in maniera efficiente ed economica dal punto di vista del consumo energetico; confrontati con i sensori piezoelettrici ad alta precisione i MEMS forniscono una discreta precisione pur avendo un costo di produzione notevolmente inferiore.

Un MEMS viene costruito utilizzando processi di erosione del silicio (in maniera analoga al procedimento utilizzato per stampare circuiti integrati) per costruire delle piccolissime strutture meccaniche con dimensioni dell'ordine 5 – 10  $\mu\text{m}$ . Su queste microstrutture, man mano che vengono costruite, vengono anche stampati dei circuiti a semiconduttore. La forza di gravità o le accelerazioni possono flettere le strutture di silicio, causando delle variazioni alle caratteristiche elettriche dei circuiti semiconduttori. Il primo esempio di impiego su larga scala di un sensore MEMS è l'accelerometro che causa l'apertura degli airbag nelle automobili [9].

### 2.1.5. CONSIDERAZIONI RIASSUNTIVE

Nella tabella 1 riportiamo un confronto delle principali caratteristiche hardware di alcuni sensori disponibili in commercio. Nella tabel-

la 2 abbiamo inserito un elenco di progetti di hardware per WSN con relativi riferimenti per l'approfondimento. Questi progetti descrivono prototipi sviluppati da università e centri di ricerca, alcuni dei quali sono diventati dei prodotti commerciali; per alcuni sono disponibili gli schemi circuitali di realizzazione.

### 2.2. Architettura software

Un *mote* è un elaboratore dotato di risorse hardware molto limitate, che deve gestire un apparato ricetrasmittente ed il routing dei pacchetti dati. Il sistema operativo e le applicazioni sviluppate devono essere in grado di esercitare un controllo fine delle risorse hardware per poter implementare politiche di risparmio energetico efficaci che consentano di far durare il più possibile la batteria che alimenta il *mote*. I sistemi operativi tradizionali (come Linux, OSE, QNX, ...) impiega-

Nome piattaforma	IMote	Mica2	MicaZ	Telos B	Stargate
CPU (produttore) Modello Velocità Bit	ARM ARM7TDMI 12 MHz 32 bit	Atmel ATMEGA128 8 MHz 8 bit	Atmel ATMEGA128 8 MHz 8 bit	Texas Instruments MSP430 8 MHz 16 bit	Intel PXA255 400 MHz 32 bit
SRAM (KB)	64	4	4	10	64.000
Flash (KB)	512 512 dati	128 programmi 512 dati	128 programmi 1024 dati	48 programmi	32.000
Radio	Bluetooth	Proprietario 315/433/915MHz	ZigBee	ZigBee	Espandibile con periferiche di comunicazione via porta PCMCIA, CF, RS 232, USB, Ethernet
Banda (kbit/s)	720	15	250	250	

**TABELLA 1**

Confronto tra diversi hardware per WSN

Crossbow	<a href="http://www.xbow.com/">http://www.xbow.com/</a>
Rockwell	<a href="http://wins.rsc.rockwell.com/">http://wins.rsc.rockwell.com/</a>
BTnode	sviluppati al Politecnico di Zurigo, commercializzati da Vitronics <a href="http://www.vitronics.com/">http://www.vitronics.com/</a>
$\mu\text{amps}$	<a href="http://www_mtl.mit.edu/researchgroups/icsystems/uamps/">http://www_mtl.mit.edu/researchgroups/icsystems/uamps/</a>
Medusa	<a href="http://nesl.ee.ucla.edu/projects/ahlos/hardware.htm">http://nesl.ee.ucla.edu/projects/ahlos/hardware.htm</a>
Intel IMote	<a href="http://www.ieee.or.com/Archive/Intel_mote.pdf">http://www.ieee.or.com/Archive/Intel_mote.pdf</a>
EyesIFX Platform	<a href="http://doc.utwente.nl/41407">http://doc.utwente.nl/41407</a>
Sensoria WINS	<a href="http://www.sensoria.com/">http://www.sensoria.com/</a>
Telos	<a href="http://www.moteiv.com/">http://www.moteiv.com/</a>
EmberNet	<a href="http://www.ember.com/company/index.html">http://www.ember.com/company/index.html</a>

**TABELLA 2**

Elenco di progetti  
su hardware  
per WSN

ti nei sistemi embedded sono carenti per quanto riguarda i requisiti appena introdotti; per questo sono stati sviluppati dei sistemi operativi ad hoc come TinyOS [60], EYES OS [10] e Contiki [11].

L'implementazione delle politiche di risparmio energetico e di routing delle informazioni costituiscono un fattore cruciale di una WSN. Il risparmio di energia può essere ottenuto spegnendo i sensori di rilevazione quando non vengono utilizzati, "addormentando" l'unità di elaborazione quando non deve svolgere calcoli e soprattutto usando oculatamente l'unità ricetrasmittente, il componente che in assoluto consuma più energia. Un *mote* può decidere di spegnere l'unità ricevente per brevi intervalli di tempo, può sfruttare la comunicazione *multihop*<sup>2</sup> e soprattutto può cercare di elaborare localmente le informazioni prima di trasmetterle. Un sensore che deve calcolare la temperatura media di un ambiente in un giorno, può memorizzare i valori man mano che vengono raccolti e a fine giornata calcolare e trasmettere la media. In questo caso il consumo energetico per memorizzare i valori, elaborarli e trasmetterli solo a fine giornata è notevolmente inferiore rispetto a quanto consumerebbe l'apparato di trasmissione per trasmettere singolarmente ognuno dei valori rilevati.

### 2.2.1. TINYOS E NESC

Analizziamo ora NesC e TinyOS, rispettivamente il primo linguaggio utilizzato per scrivere le applicazioni per *mote* ed il primo sistema operativo per *mote* (scritto in NesC), entrambi sviluppati presso l'Università di Berkeley. TinyOS è diventato uno standard industriale, sia per essere stato il primo sistema operativo per WSN sia perché i suoi sorgenti sono pubblici e liberamente utilizzabili.

<sup>2</sup> Si ha comunicazione *multihop* quando un messaggio non viene trasmesso direttamente dal mittente al destinatario, ma per mezzo di un insieme di nodi, ognuno dei quali riceve un messaggio e lo ritrasmette al vicino, formando un percorso che congiunge il mittente con il destinatario. La trasmissione *multihop* permette di coprire lunghe distanze pur avendo a disposizione apparati di comunicazione a basso consumo energetico in grado di coprire solo piccole distanze.

I *mote* sono dei dispositivi di elaborazione particolari, perché "dormono" per la maggior parte del tempo. L'elaborazione svolta è guidata dagli eventi: solo quando il sensore acquisisce nuovi dati oppure quando vengono ricevuti dei messaggi si avvia un processo di calcolo.

L'unità base di programmazione e di compilazione nel linguaggio NesC è il "componente", ogni programma è ottenuto assemblando uno o più componenti tra loro. Ogni componente è specificato da una o più interfacce, le quali, in modo simile a quanto fatto da altri linguaggi, dichiarano un insieme di funzioni e procedure da implementare chiamate "comandi". A differenza di linguaggi come C e JAVA, le interfacce NesC offrono dei costrutti sintattici espliciti sia per specificare il codice di gestione degli eventi (la porzione di codice che deve essere avviata al verificarsi di un particolare tipo di evento) sia per gestire l'accoppiamento tra i componenti che producono eventi e i componenti interessati alla notifica.

I componenti NesC a loro volta si dividono in due categorie: i moduli e le configurazioni. I moduli contengono il codice che implementa una o più interfacce; un componente di configurazione (più brevemente, configurazione) viene utilizzato invece per specificare come assemblare più moduli tra loro. L'obiettivo principale di quest'approccio è poter sviluppare separatamente componenti che possono essere poi facilmente composti tra loro.

TinyOS è formato da un insieme di routine di sistema (implementate per mezzo di componenti NesC) e da un componente che si occupa di pianificare l'esecuzione dei "thread". In un *mote* ci sono solamente due thread in esecuzione: l'esecutore dei "task" e l'esecutore dei "gestori degli eventi hardware". I task sono blocchi di codice applicativo, i gestori degli eventi hardware si occupano di gestire le interruzioni generate dall'hardware del *mote*, in particolare le interruzioni generate dai sensori e dall'unità ricetrasmittente.

Le chiamate di sistema TinyOS che richiedono elaborazioni lunghe in termini di tempo non sono bloccanti, ma vengono eseguite in modo asincrono: la loro invocazione restituisce subito il controllo al chiamante ed il completamento del comando è segnalato per mezzo di un evento. Per esempio, l'interfaccia che offre il comando per trasmettere dati via radio ri-



chiede al chiamante di gestire l'evento "send-Done", generato dal sistema operativo, che informa che il pacchetto è stato inviato.

### 2.2.2. MACCHINE VIRTUALI

I programmi per WSN vengono sviluppati su desktop, provati con dei simulatori [61] e poi trasferiti sui *mote*. Per quest'ultima operazione si utilizza la "programming board", una scheda che si collega ad una porta del PC, sulla quale si può inserire un *mote* e che permette di scrivere direttamente sulla memoria flash del *mote* stesso. Con questo metodo è necessario riprogrammare i *mote* uno ad uno ed inoltre non è possibile riprogrammare quelli già disposti sul campo. Sarebbe auspicabile utilizzare le capacità di comunicazione dei *mote* per distribuire un nuovo programma, tuttavia la maggior parte dell'hardware non permette di riscrivere via radio la memoria delle applicazioni.

Questo inconveniente può essere superato eseguendo su ogni *mote* una "macchina virtuale": un interprete di codice macchina che gira come un normale programma. La macchina virtuale esegue del codice che può essere trasmesso come se fosse una normale sequenza dati, quindi sfruttando le potenzialità

di comunicazione di una rete di *mote*. Le istruzioni interpretate dalle macchine virtuali sono generalmente istruzioni di più alto livello di quelle offerte dall'hardware e da TinyOS, per questo motivo il codice prodotto ha dimensioni inferiori ed è quindi più facile da trasmettere. A differenza di un programma NesC, il quale viene eseguito direttamente dall'hardware del *mote* senza alcun controllo, la macchina virtuale può controllare che il programma interpretato non esegua operazioni pericolose che possano portare a bloccare il *mote*, come scrivere in zone di memoria al di fuori di quelle assegnate. Esempi di applicazioni o sistemi operativi che implementano al loro interno delle macchine virtuali sono: Mate [19], Bombilla [38], Sensorware [57], MagnetOS [51].

### 2.2.3. LA GESTIONE DEI DATI: TINYDB

I *mote* di una WSN hanno la capacità di acquisire informazioni autonomamente dall'ambiente esterno e hanno capacità di elaborazione che possono essere utilizzate per filtrare ed in generale per pre-elaborare localmente i dati raccolti. Queste capacità sono state sfruttate in modo interessante da TinyDB [20], un software che permette di creare una rappresentazione astratta di un insieme di *mote* basata sul paradigma del database. TinyDB permette di recuperare informazioni eseguendo interrogazioni espresse in linguaggio SQL come se ci si trovasse di fronte ad un database vero e proprio. L'architettura software sottostante a TinyDB si occupa di inviare la query ai *mote* interessati, di farla eseguire in locale, di raccogliere le informazioni e di presentarle all'interrogatore, eventualmente in forma aggregata. Essa si occupa anche di minimizzare il traffico di rete, per esempio cercando di inviare le query solamente ai nodi che dispongono delle informazioni cercate e aggregando i dati raccolti dai nodi vicini prima di ritrasmetterli. Il linguaggio di TinyDB include anche un paradigma attivo, che prevede la possibilità di scrivere query che vengono scatenate periodicamente o in presenza di determinati eventi.

### 2.3. Un caso applicativo

Le reti di sensori possono essere impiegate nei più svariati ambiti e contesti; a titolo di esempio nel riquadro si riportano i riferimenti ad alcune applicazioni, suddivise per cate-

#### Esempi di applicazioni basate su WSN

##### Controllo di spazi

Controllo di ambienti ed habitat naturali [44][16].  
Agricoltura di precisione: si rilevano dati ambientali per determinare interventi da compiere, periodo di raccolta o di vendemmia [8].  
Climatizzazione di interni [2][47][46].  
Sorveglianza [54].

##### Controllo di oggetti

Controllo strutture [39][17][32][45].  
Ecophysiology: si utilizzano i sensori per controllare che i parametri di un ambiente rientri nei normali valori di funzionamento [6].  
Gestione e controllo motori ed equipaggiamenti [3].  
Diagnostica medica [26].  
Mappatura di territori [22].

##### Controllo dell'interazione tra oggetti, persone ed ambiente

Ambienti domestici [59][15].  
Interazione tra persone [28].  
Controllo traffico veicoli [62].  
Controllo habitat selvaggi [29].  
Gestione di catastrofi [42].  
Interventi di emergenza [33].  
Ubiquitous computing environments [18].  
Asset tracking: localizzazione di container, beni di valori [34].  
Sanità [21][27].  
Flussi di processi produttivi [12].

gorie, che utilizzano le WSN. In questo paragrafo descriveremo come le reti di sensori sono state utilizzate sul Golden Gate di San Francisco (USA).

Poiché il ponte vibra per effetto del traffico e del vento, arrivando ad oscillare di alcune decine di centimetri quando sottoposto a forti raffiche di vento o a terremoti, sulla struttura sono stati disposti dei *mote* dotati di accelerometri in grado di rilevare le sollecitazioni subite dal ponte. I dati raccolti vengono inviati ad un'apposita stazione per essere consultati dal personale che sovrintende alla manutenzione del ponte [45].

Per misurare le oscillazioni del ponte è stata preferita una WSN ad altri strumenti di misura basati su cavi poiché i *mote* possono essere posizionati con maggior facilità nei punti più difficilmente accessibili.

L'elevata mole di dati originata dal processo di rilevamento e la ridotta banda di trasmissione disponibile sono stati gli aspetti più critici da gestire nello sviluppo del software di controllo della WSN impiegata sul Golden Gate. In 5 min di rilevazioni, 100 *mote* disposti sul ponte raccolgono 24 Mbyte di dati, i quali devono essere tutti trasmessi ad un *mote* che funge da "gateway" tra la rete di sensori e il centro di rilevazione dati. Il *mote gateway* costituisce un collo di bottiglia, in quanto il canale radio attraverso cui riceve i dati dagli altri *mote* della rete ha una banda effettiva<sup>3</sup> di 1,2 kbyte/s e per trasmettere i dati raccolti con tale banda a disposizione sarebbero necessarie più di 5 h [50]. Anche aumentando il numero di gateway tra rete wireless e stazione di raccolta dati non si riuscirebbero ad ottenere prestazioni accettabili. In altri contesti applicativi sarebbe possibile sfruttare le capacità di elaborazione locali di ogni *mote* per ridurre le dimensioni dei dati da trasmettere, tuttavia nel caso del Golden Gate i dati raccolti sui movimenti della struttura devono essere trasmessi per intero.

<sup>3</sup> Sono stati utilizzati dei *mote* Mica2 che offrono una banda massima teorica di 15 kbit/s (circa 2 kbyte/s) che, per effetto della ritrasmissione dei pacchetti corrotti e dell'inserimento di informazioni di controllo da parte del protocollo di comunicazione impiegato, si riduce ad una banda effettiva di 1,2 kByte/s.

Per superare i problemi esposti ogni *mote* spiegato sul ponte è stato programmato in modo da memorizzare temporaneamente i dati raccolti nella memoria flash e per trasmetterli verso la stazione di rilevamento solo alla fine del periodo di rilevazione (o all'esaurirsi della memoria).

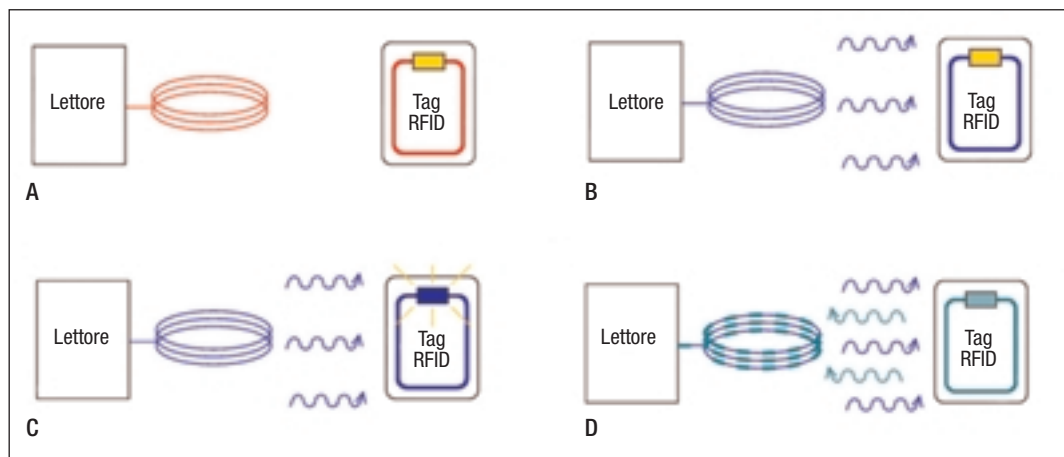
Dato che i dati sono trasferiti alla stazione di rilevamento in differita, è necessario contrassegnare ogni rilevamento con un *time stamp* (orario di rilevamento). Poiché gli orologi interni dei *mote* non sono sincronizzati tra loro, dopo l'invio dei dati è necessario ricondurre i vari *time stamp* ad un unico orario. Un protocollo sviluppato *ad hoc* si occupa di determinare la differenza di orario tra i *mote* e la stazione di rilevamento, in modo da poter correggere i *time stamp*.

Per svolgere le rilevazioni i *mote* sono stati depositati sulle strutture del ponte per poi essere rimossi una volta esaurita la batteria. Sono allo studio soluzioni per dotare il ponte di sensori permanenti. In questo caso il problema da risolvere è come fare in modo che i *mote* possano funzionare per lungo tempo senza essere connessi alla rete elettrica (soluzione che richiederebbe la messa in posa di cavi).

I *mote* fino ad ora sono stati disposti in modo tale da essere in contatto diretto con la stazione di rilevamento. Sono allo studio forme di comunicazione *multihop*: dato che i *mote* vengono spesso disposti "in fila" (per esempio sui tiranti del ponte), questa topologia permette di implementare un semplice algoritmo di routing: è sufficiente che un *mote* inoltri i pacchetti ricevuti al *mote* successivo della fila per fare in modo che le informazioni giungano alla stazione di rilevazione. È in corso di valutazione la possibilità di impiegare dei *super-mote* che raccolgano i dati raccolti dai *mote* normali circostanti e li inviino alla stazione di raccolta per mezzo di un canale radio a banda maggiore.

### 3. RFID

Un sistema RFID è composto da due elementi principali: il lettore (o *reader*) ed il *transponder*. I *transponder* spesso sono montati su etichette e per questo motivo vengono comunemente chiamati etichette o con il corrispondente termine inglese "tag". Il lettore è



**FIGURA 3**

*Esempio di funzionamento di un transponder RFID: A Situazione a lettore spento. B Il lettore genera un campo elettromagnetico che alimenta la bobina del transponder. C Il circuito del transponder esegue delle elaborazioni. D Il transponder restituisce i risultati delle elaborazioni*

un dispositivo ricetrasmittente che svolge una duplice funzione: fornisce energia, con la quale un *transponder* può funzionare, e dialoga con il *transponder* trasmettendo comandi e raccogliendo risposte. Un *transponder* è un piccolo chip accoppiato ad una antenna; quando il *transponder* entra nel campo elettromagnetico generato dal lettore, sull'antenna del *transponder* (che in molti casi assomiglia ad una bobina) viene indotta una corrente che alimenta il chip. Il chip elabora il comando ricevuto e trasmette indietro il risultato delle elaborazioni. Il processo è illustrato nella figura 3.

Ogni *transponder* (*tag* d'ora in poi) è dotato di un identificatore univoco (ID), che lo distingue dagli altri. Ad ogni operazione di interrogazione, un *tag* risponde fornendo l'ID che lo contraddistingue; in questo modo diventa possibile identificare oggetti e merci usando i *tag* RFID come se fossero dei marcatori, in maniera analoga ad un codice a barre. Un codice a barre per essere letto deve essere illuminato con una certa precisione da un lettore laser, in caso contrario la lettura non riesce<sup>4</sup>; un RFID non soffre di questo problema perché utilizza le onde radio le quali non richie-

dono che trasmittente e ricevente siano tra loro visibili. L'impiego di onde radio permette addirittura di rilevare tutti i *tag* RFID contenuti all'interno di uno scatolone senza doverlo aprire. Un'altra differenza fondamentale tra codici a barre ed RFID è il tipo di codifica dei prodotti adottata. Un codice a barre permette di identificare i prodotti solamente a livello di categoria (esempio, le marmellate alla fragola prodotte dall'azienda del signor Rossi hanno tutte lo stesso codice a barre), mentre l'ID di un RFID permette anche di distinguere tra loro prodotti della stessa categoria.

Questa caratteristica, unita con la possibilità di leggere i dati di un *tag* RFID a distanza, apre innumerevoli possibilità, prima fra tutte la possibilità di eseguire gli inventari in modo quasi istantaneo. È sufficiente far passare gli oggetti da inventariare vicino ad un lettore RFID per identificare tutti gli oggetti corredati di *tag*. Con i codici a barre sarebbe necessario leggere ad uno ad uno i codici degli oggetti; inoltre ci sarebbe sempre il rischio di contare un oggetto più volte, se l'operatore passasse per sbaglio il lettore su un oggetto "già letto".

### 3.1. Categorie di tag

Esistono vari tipi e varie categorie di *tag*: si distinguono in base alla presenza o meno di una fonte di alimentazione a bordo, alla capacità ed alla riscrivibilità della memoria, al tipo di codice identificativo adottato, alla frequenza di lavoro e allo standard utilizzato per

<sup>4</sup> È esperienza comune vedere il cassiere di un supermercato costretto a passare più volte un prodotto sopra il lettore ottico della cassa prima che la lettura riesca.



la trasmissione dati. I *tag passivi* non hanno batterie o fonti di alimentazione a bordo, i *tag attivi* hanno una fonte di alimentazione e trasmettono continuamente informazioni, i *tag semipassivi* hanno una fonte di alimentazione e trasmettono informazioni solo quando sono interrogati. Per esempio, il sistema telepass utilizzato nelle autostrade italiane utilizza dei *tag semipassivi*. Invece il dispositivo arva [35] usato dagli escursionisti alpini per rintracciare le persone sepolte dalle slavine costituisce un esempio di *tag attivo*. Nel seguito di questo articolo ci occuperemo solo dei *tag passivi*, per i quali ci si attende una notevole diffusione nel mercato nei prossimi anni, soprattutto a causa delle dimensioni e dei costi di produzione mediamente inferiori rispetto agli altri tipi di *tag*. Le dimensioni ridotte facilitano l'inserimento dei *tag passivi* all'interno degli oggetti ed il basso costo di produzione è un fattore chiave per la diffusione di massa della tecnologia.

I *tag* con memoria a sola lettura (*tag Read Only*, RO in breve) memorizzano solamente il codice identificativo univoco impostato in fabbrica. I *tag* scrivibili (*Read Write*, RW in breve), che pure sono dotati di un codice identificativo univoco, dispongono anche di memorie flash sulle quali un utente può leggere e scrivere informazioni. La scrittura delle informazioni, in maniera analoga a quello che accade per la lettura, è comandata a distanza dal lettore che con il suo segnale fornisce sia i comandi sia l'energia necessaria per far svolgere al *tag* l'operazione.

Esistono due categorie di standard per il codice identificativo di un *tag*, una portata avanti dall'ISO e un'altra portata avanti da EPC Global<sup>5</sup>. Gli approcci seguiti dai due organismi sono molto diversi tra loro, come riassunto nel riquadro di p. 26. Stando alle più recenti informazioni rese disponibili dalla stessa EPC Global, è ormai probabile che lo standard EPC confluisca in un apposito

<sup>5</sup> EPC Global Inc. è un'organizzazione no-profit nata con lo scopo di standardizzare i codici di prodotto impiegati all'interno degli RFID. EPC Global è nata da un'iniziativa del MIT (Massachusetts Institute of Technology di Boston) e attualmente è un network di aziende ed organizzazioni. Il rappresentante italiano di EPC Global è Indicod-Ecr [49].

standard ISO, attualmente in corso di negoziazione tra le due parti.

Per quanto riguarda la frequenza delle onde elettromagnetiche utilizzate, i *tag* si possono classificare in 4 categorie: *tag* a bassa frequenza o LF (da 125 a 134 kHz), *tag* ad alta frequenza o HF (13.56 MHz), *tag* UHF (ultra high frequency 868 – 956 MHz) e *tag* a microonde (2.45 GHz). I *tag* a bassa ed alta frequenza possono essere usati in tutto il mondo liberamente e senza licenze. Per i *tag* UHF invece ci sono delle regole discordanti da Paese a Paese. In Europa i *tag* maggiormente utilizzati sono quelli HF. L'impiego di *tag* UHF e a microonde in Europa è reso difficoltoso dalle norme che disciplinano la telefonia cellulare, l'assegnazione di frequenze televisive e la sicurezza del lavoro. Nel mercato nord americano, una differente disposizione delle frequenze della telefonia cellulare e della tv e leggi più permissive nel campo della sicurezza del lavoro facilitano l'utilizzo di *tag* UHF.

All'aumentare della frequenza aumenta la distanza massima di lettura e scrittura di un *tag*. I *tag* LF, pur avendo delle distanze di lettura inferiori alle altre categorie di *tag*, presentano una maggior insensibilità alla presenza di metalli e liquidi. Per questo motivo vengono utilizzati per la marcatura di animali (il cui corpo presenta una elevata quantità di acqua).

### 3.2. Casi d'uso

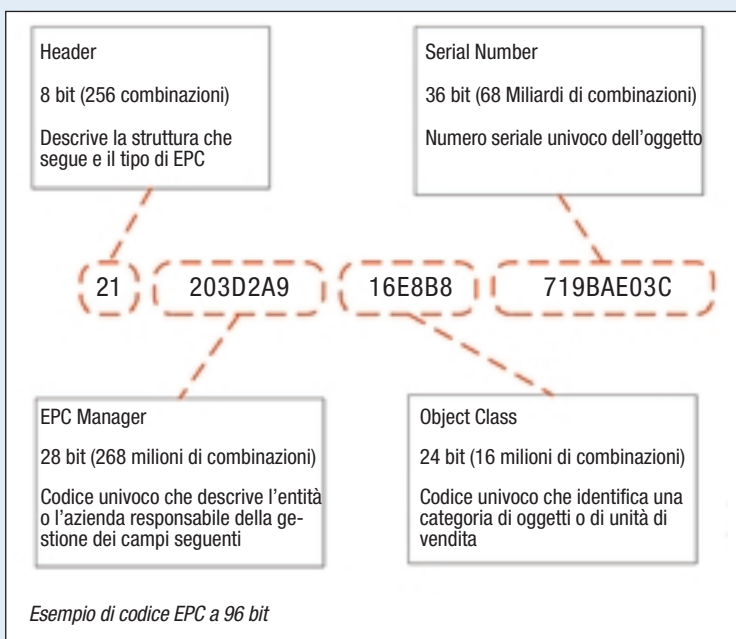
Numerose applicazioni e prototipi innovativi che sfruttano la tecnologia RFID sono stati sviluppate da industrie e centri di ricerca; questi lavori possono essere classificati in base a diversi criteri: il settore industriale di riferimento, il grado di maturità dell'applicazione della tecnologia, il tipo o i tipi di processi aziendali che l'applicazione va ad influenzare.

Nelle tabelle 3 sono riportati dei casi d'uso raggruppati a seconda dell'attività per la quale è stata impiegata la tecnologia RFID. La classificazione è stata ispirata da [1]. Le attività sono state divise principalmente in tre categorie: nella prima i *tag* RFID sono utilizzati per identificare e tracciare il flusso o lo stato di oggetti (o di animali), nella seconda per tracciare e determinare le azioni di individui, nella terza per migliorare le funzionalità

### Confronto tra ISO ed EPC

L'obiettivo principale di EPC Global è la creazione dell'EPC (*Electronic Product Code* "Codice prodotto elettronico"), cioè un codice che permetta di identificare facilmente ogni prodotto, fornendo informazioni sul produttore, sul Paese di provenienza ecc., in maniera analoga ad un codice a barre. Lo standard EPC definisce tre schemi di codici: a 96, a 128 e a 256 bit. Lo schema del codice a 96 bit, riportato nella figura fornisce un identificatore univoco diviso in tre parti più un header: azienda, classe di oggetto e prodotto. La prima parte del codice identifica l'azienda produttrice del bene, e permette di identificare 268 milioni di aziende. Ogni azienda può gestire 16 milioni di classi di oggetti, ognuna delle quali può avere 68 miliardi di numeri seriali diversi. Secondo lo schema EPC, ogni prodotto è contrassegnato da un tag il quale ha come identificatore univoco l'EPC del prodotto stesso. Un tag pur potendo ospitare delle informazioni aggiuntive, secondo lo schema EPC deve memorizzare solamente il codice EPC, le altre informazioni che contraddistinguono un prodotto devono essere fornite da una rete e da un sistema informativo on-line che permettono di recuperare le informazioni associate al tag. In merito a quest'ultimo punto EPC definisce in modo rigido un'infrastruttura per interconnettere i sistemi informativi di più soggetti al fine di condividere le informazioni associate ai tag.

Lo standard ISO è concepito in modo completamente diverso: le informazioni aggiuntive rispetto ad un ID che contraddistinguono un oggetto risiedono per la maggior parte sul tag stesso, o al più su un database e su delle applicazioni locali all'azienda. In questa architettura non vi è una standardizzazione del sistema che si occupa dello scambio di informazioni tra diversi sistemi informativi. Infine per quanto riguarda i criteri per specificare l'ID dei tag, ISO definisce uno standard aperto, delegando al singolo produttore di tag la definizione delle caratteristiche dell'ID utilizzato dal tag stesso. Nella tabella è riportato l'elenco degli standard ISO di interesse per i tag RFID. Sia ISO sia EPC Global si sono occupati della standardizzazione delle frequenze radio utilizzate oltre alla standardizzazione dei codici identificativi dei tag. EPC Global, a differenza di ISO, ha svolto in questo campo un lavoro marginale il cui principale obiettivo è stato la creazione di un hardware standard che possa ospitare l'EPC. Gli sforzi di EPC Global nel campo delle frequenze si sono concentrati principalmente sulla banda UHF, per la quale nel mercato nord americano esistono meno vincoli per l'utilizzo rispetto al mercato europeo.



### ELENCO DEGLI STANDARD ISO DI INTERESSE PER RFID

ISO/IEC 18001	Information Technology, AIDC Techniques, RFID for Item Management, Application Requirement Profiles	Published Standard 2004
ISO/IEC 18000_1	Generic Parameters for Air Interface Communication for Globally Accepted Frequencies	Published Standard 2004
ISO/IEC 18000_2	Parameters for Air Interface Communications below 135kHz	Published Standard 2004
ISO/IEC 18000_3	Parameters for Air Interface Communications at 13.56 MHz	Published Standard 2004
ISO/IEC 18000_4	Parameters for Air Interface Communications at 2.45 GHz	Final Draft Intern. Standard
ISO/IEC 18000_6	Parameters for Air Interface Communications at 860_930 MHz	Published Standard 2004
ISO/IEC 15961	RFID for Item Management _ Data protocol: Application interface	Published Standard 2004
ISO/IEC 15962	RFID for Item Management, Protocol: Data encoding rules and logical memory functions	Published Standard 2004
ISO/IEC 15963	RFID for Item Management, Unique Identification of RF Tag	Final Draft Intern. Standard

Supporto all'attività produttiva principale	<p>Settore agricolo e dell'allevamento</p> <ul style="list-style-type: none"> <li>• Identificazione animali: contrassegnando gli animali con dei tag si snelliscono le procedure burocratiche connesse con la loro compravendita [1].</li> <li>• Controllo animali: contrassegnando ogni animale con un tag si possono riconoscere e fornire automaticamente alimentazione, cure veterinarie personalizzate all'animale [1].</li> <li>• Riconoscimento lotti prodotti agricoli: all'ingresso di serre sono applicati tag RFID su cui vengono registrati i dati del lotto di sementi utilizzate e l'insieme delle attività svolte sulle colture [1].</li> </ul>
	<p>Settore manifatturiero</p> <ul style="list-style-type: none"> <li>• gestione dell'avanzamento lavori lungo le linee di assemblaggio: i lotti di semilavorati vengono contrassegnati con tag sui quali si registrano le lavorazioni effettuate e le lavorazioni da effettuare [1] [30] [65].</li> </ul>
	<p>Settore dei servizi</p> <ul style="list-style-type: none"> <li>• Automazione operazione di ingresso/uscita/rifornimento mezzi di trasporto: l'applicazione di un tag agli automezzi facilita il riconoscimento dei mezzi ai varchi di accesso al deposito e ad aree del centro storico, oppure serve per selezionare automaticamente il carburante più adatto nelle stazioni di rifornimento [1].</li> <li>• Life cycle management di attrezzature: macchinari sottoposti a manutenzione periodica vengono contrassegnati con tag sui quali si registrano le operazioni di manutenzione effettuate [1].</li> <li>• Gestione medicinali in ospedale: tag sono applicati sulle confezioni dei farmaci, in questo modo può essere automatizzata la contabilità dei prelievi e restituzione dei farmaci nei singoli armadietti da parte dei medici [1].</li> <li>• Gestione delle supply chain (catene di approvvigionamento) [64 - 53 - 36].</li> <li>• Utilizzo dei tag RFID per aumentare l'affidabilità e la velocità di elaborazione dei centri di smistamento pacchi di corrieri e spedizionieri [56].</li> <li>• Gestione bagagli negli aeroporti [25 - 7 - 14].</li> <li>• Gestione biblioteche: Inserendo un tag in ogni libro, è possibile velocizzare le operazioni di prestito e restituzione [5].</li> </ul>
Logistica di magazzino	<ul style="list-style-type: none"> <li>• Identificazione univoca delle unità di movimentazione nei magazzini: applicando un tag ai contenitori di magazzino è possibile snellire le procedure di inventario, contabilità degli ingressi e delle uscite [1].</li> <li>• Controllo ricevimento merci in ingresso: applicando un tag ad ogni prodotto è possibile conoscere il contenuto di un contenitore di prodotti (o di una qualsiasi altra unità di movimentazione) senza doverlo aprire [1].</li> <li>• Impiego in magazzini con condizioni climatiche difficili: l'utilizzo dei tag consente di velocizzare le operazioni di riconoscimento prodotti, di avere meno manodopera umana all'interno del magazzino ed infine i tag RFID (non dovendo stare all'esterno delle confezioni come i codici a barre) subiscono meno il degrado causato dalle condizioni ambientali [1].</li> </ul>
Tracciabilità dei prodotti lungo una filiera	Su un tag che contrassegna un prodotto, ogni componente di una filiera produttiva può memorizzare le sue generalità e le lavorazioni svolte, in tal modo il tag diventa un registro della filiera produttiva [1].
Gestione di asset	I tag vengono utilizzati per contrassegnare e tracciare oggetti di valore, per esempio contenitori (damigiane, container, ...) che transitano tra più operatori [1].
Gestione dei punti vendita	Contrassegnando con un tag i beni venduti da un negozio, è possibile automatizzare le operazioni di pagamento in cassa e di inventariazione della merce. Attualmente in Italia sono allo studio soluzioni per ottenere questi obiettivi, però non esiste nessuna realizzazione pratica di larga scala [1], negli Stati Uniti sono stati compiuti degli esperimenti: [23 - 43].
Sensori di stato	Tag RFID che cambiano irreversibilmente stato se sottoposti a condizioni ambientali particolari. Sono adatti per verificare l'interruzione della catena del freddo di un prodotto [31 - 24].

**TABELLA 3/1***RFID associati ad oggetti ed animali*

Bigliettazione elettronica	Si vuole sostituire il tradizionale biglietto o abbonamento cartaceo per i mezzi pubblici con un badge avente un tag RFID che memorizza il contratto di abbonamento e la validazione delle singole corse. Il tag permette di automatizzare le operazioni di pagamento dell'abbonamento, diventa inoltre possibile creare biglietti ricaricabili con credito a scalare. Le amministrazioni pubbliche possono infine ottenere informazioni aggregate sui tragitti giornalieri compiuti dalle persone e utilizzare queste informazioni per migliorare il servizio [1 - 55].
Car sharing nel trasporto pubblico locale	Sono allo studio l'utilizzo di tag RFID per velocizzare le operazioni di ritiro dei mezzi prenotati [1].
Ticketing	Si utilizzano dei badge con tag RFID per gestire l'ingresso a particolari eventi, manifestazioni, centri sportivi e per la fruizione dei servizi accessori [58 - 1].
Customer Relationship Management ed edutainment	Si tratta di un'evoluzione delle operazioni di ticketing: il riconoscimento della persona e il tracciamento del suo comportamento consentono di fornire all'utente servizi personalizzati o migliorare il livello di servizio fornito [1].
Controllo accessi nei luoghi di lavoro	Si utilizzano dei badge con tag RFID per il controllo accessi, i quali possono essere utilizzati anche per accedere ad altri servizi interni all'azienda (es. mensa). In alcune applicazioni, i badge memorizzano dati biometrici cifrati da utilizzare per la validazione dell'identità della persona. Dato che le informazioni non sono memorizzate nei server aziendali, per l'azienda non sussistono problemi per la gestione della privacy [1 - 48].
Identificazione dei pazienti in ambito ospedaliero	L'applicazione di tag sui pazienti (per mezzo di braccialetti) congiuntamente all'applicazione su materiali (esempio, sacche per trasfusione) possono fornire al personale utili indicazioni per la corretta gestione dei processi ospedalieri e prevenire somministrazioni scorrette di farmaci o applicazione di trattamenti errati [1].

**TABELLA 3/2**

*RFID associati a individui*

Misura pressione pneumatici	Un tag RFID munito di sensore di pressione è inserito all'interno di un pneumatico, un lettore RFID è inserito all'interno del parafrangente, in questo modo è possibile determinare la pressione di una gomma in tempo reale [67].
Utilizzo di lettori RFID all'interno di lavatrici e frigoriferi	Una lavatrice è in grado di leggere le istruzioni di lavaggio contenute nei tag attaccati ai vestiti. Il frigorifero può controllare la presenza di alimenti scaduti ed eventualmente ordinarne di nuovi [52].
Utilizzo di tag	Nel processo di firma digitale di documenti per garantire l'univocità (oltre che l'autenticità) di un documento [41].

**TABELLA 3/3**

*RFID per migliorare tecnologie esistenti*

di tecnologie esistenti. All'interno di ognuna di queste categorie sono state individuate delle sottocategorie, per ognuna delle quali sono state riportate informazioni sui singoli casi applicativi e dei riferimenti per l'approfondimento.

Ora analizzeremo in dettaglio un caso d'uso della tecnologia RFID.

**Sistema di messaggistica.** Al Politecnico di Milano, è stato svolto un progetto sperimentale [13] nel quale si sono utilizzati i tag RFID per sostituire la procedura di autenticazione

basata su username e password di un sistema di messaggistica. Il sistema utilizza dei pannelli a messaggi variabili per fornire informazioni ad utenti che si muovono all'interno di un certo ambiente, quale un campus universitario, una stazione o un aeroporto. Esso consente di veicolare messaggi diretti ad un vasto gruppo di persone (esempio, "I passeggeri del volo XYZ 123 sono pregati di presentarsi al gate 2" oppure "La lezione di Informatica A è stata spostata nell'aula So1").

Ogni volta che una persona si avvicina ad un

pannello a messaggi variabili, il pannello rileva il *tag* RFID contenuto nel badge di riconoscimento della persona e visualizza i messaggi destinati all'utente. Il *tag* RFID permette di rilevare la presenza dell'utente e di identificarlo senza un intervento esplicito dell'utente stesso, come invece digitare lo username o strisciare un badge magnetico su un lettore. Se più persone sono vicine ad un pannello, lo schermo del pannello si divide in parti uguali, ognuna delle quali mostra le informazioni dirette ad un singolo utente. Qualora vicino al pannello siano presenti un numero di utenti tale da portare ad una eccessiva frammentazione dello schermo, vengono visualizzati solo i messaggi comuni ai vari utenti. Il sistema gestisce la suddivisione di utenti in gruppi e permette di inviare messaggi di gruppo, che verranno poi notificati a tutti i membri, oppure messaggi con destinatario non specificato, che verranno recapitati a tutte le persone che condividono certi interessi.

Il *tag* contenuto nel badge ospita informazioni sugli interessi dell'utente, le quali sono utilizzate dal sistema per decidere quali messaggi proporre all'utente stesso. Le informazioni memorizzate sui *tag* sono mantenute ed aggiornate personalmente dall'utente (avvalendosi di appositi chioschi che permettono di leggere e scrivere il contenuto dei *tag*) e sono ospitate solamente sul badge. In questo modo un utente può ricevere messaggi tarati sui suoi gusti senza che il sistema debba gestire centralmente le sue informazioni di preferenza. Il sistema è particolarmente indicato per veicolare informazioni generali che interessano un largo pubblico, può essere usato per inviare informazioni anche a singole persone, tuttavia non è adatto per trasmettere informazioni di carattere strettamente personale dato che i messaggi vengono visualizzati da schermi posti in luoghi pubblici.

### 3.3. Gestione distribuita dei dati

I *tag* RFID permettono di realizzare in modo semplice ed economico un sistema informativo "distribuito" e pervasivo cui partecipano, oltre a componenti fissi, diversi dispositivi mobili come palmari, telefoni cellulari e smartcard ed eventualmente componenti per la raccolta di informazioni del territorio come i *mote*.

I *tag* RFID consentono di memorizzare le infor-

mazioni in formato elettronico direttamente sugli oggetti, dando modo agli operatori di recuperarle e (nel caso di *tag* RW) aggiornarle agevolmente. Per esempio un *tag* RFID attaccato ad un prodotto permette di memorizzare il percorso compiuto, man mano che il *tag* avanza, lungo la filiera produttiva.

Il fatto che dei "dati elettronici" viaggino assieme alla merce che caratterizzano, diventa un comodo strumento per scambiare informazioni tra i sistemi informativi delle società e degli enti coinvolti nella vita di un prodotto. Salvare le informazioni alla "periferia del sistema informativo" significa metterle "nelle mani" degli attori che principalmente le utilizzano. Ciò offre delle semplificazioni dal punto di vista dell'accesso alle informazioni, ma solleva delle problematiche riguardanti la sicurezza e la riservatezza delle stesse. Non entriamo qui nel dettaglio di questa discussione; vogliamo però segnalare che le nuove tecnologie offrono possibilità per affrontare questi problemi. La sede di New York dell'American Express utilizza un sistema composto da *tag* RFID e lettore di impronte digitali per gestire il controllo degli accessi dei dipendenti [48]. L'aspetto innovativo sta nel fatto che le impronte digitali di un dipendente sono memorizzate su un *tag* RFID presente all'interno del badge assegnato al dipendente stesso<sup>6</sup>. Quando un utente deve passare per un punto di controllo, preme il suo dito su un lettore di impronte, l'impronta viene rilevata e confrontata con le informazioni memorizzate sul *tag* RFID contenuto nel badge dell'utente. Il sistema di accessi della società non memorizza l'impronta dell'utente ma solo un codice di controllo dei dati del badge per poter riconoscere eventuali manomissioni. Questo approccio evita alla società di dover gestire direttamente le informazioni di natura biometrica dell'utente, i quali rappresentano "dati sensibili" che richiederebbero un trattamento particolare per garantire la riservatezza.

<sup>6</sup> Per essere precisi sul badge non è salvata una descrizione completa dell'impronta di una persona ma solo uno schema (un sottinsieme di punti dell'impronta) che può essere usato per la verifica dell'impronta. Lo schema non permette di risalire all'impronta dal quale è stato generato, ma permette di identificarla.



0

1

0

Salvare i dati su dispositivi mobili, con connessioni transitorie e di capacità variabile, ha come rovescio della medaglia problemi di sincronizzazione e di consistenza dei dati. Per esempio il ricevimento di un pacco da parte di un fornitore richiede lo svolgimento di alcune procedure amministrative e per far ciò è necessario sincronizzare le informazioni gestite dal sistema informativo dell'amministrazione con le informazioni contenute nel *tag* RFID che contrassegna il pacco. Se gli operatori del magazzino non hanno lettori di *tag* collegati con il sistema informativo, l'afflusso delle informazioni a quest'ultimo può subire dei ritardi.

Se più copie della stessa informazione sono memorizzate in dispositivi diversi, c'è il rischio che diversi operatori modifichino le informazioni in modi differenti generando problemi di consistenza delle informazioni. Anche per questo motivo, nel caso in cui i *tag* debbano essere letti ed aggiornati da diversi sistemi informativi, i dati devono essere espressi in un formato comune intelligibile ai vari sistemi coinvolti.

Infine le informazioni contenute nei *tag* possono essere manomesse oppure i *tag* possono subire dei danni accidentali. Nel primo caso le informazioni devono essere protette con opportuni strumenti (crittografia, codici di controllo, ...). Nel secondo caso una copia delle informazioni può essere sufficiente. La copia può essere memorizzata su un altro *tag* attaccato all'oggetto, oppure su un server del sistema informativo.

### 3.4. RFID: una metodologia di progettazione

Per avvantaggiarsi delle possibilità offerte dalle nuove tecnologie è necessario ripensare lo sviluppo dei sistemi informativi. L'approccio tradizionale (gestire centralmente tutte le informazioni), dovrà coesistere con la possibilità di memorizzare dati sugli oggetti fisici messa a disposizione dalle nuove tecnologie. Come conseguenza nasce il problema di decidere, data un'informazione, dove memorizzarla e in che maniera gestire le eventuali repliche. Occorre scegliere le informazioni che devono essere disponibili localmente (memorizzate su *tag* RFID oppure su *mote*) in modo tale da minimizzare l'accesso a fonti di informazioni

non locali, tenendo tuttavia conto dei limiti di memorizzazione dei dispositivi.

Illustriamo ora un esempio in cui applicheremo una metodologia [4] utile per risolvere il problema accennato.

L'esempio riguarda la gestione delle informazioni mediche e cliniche di un gruppo di pazienti. Ad ogni paziente viene fornito un *tag* RFID che contiene le sue informazioni mediche personali. Le informazioni che il sistema informativo nel suo complesso deve gestire riguardano la storia medica, le patologie ed i ricoveri dei pazienti. Una porzione di queste informazioni deve essere replicata nel *tag* in mano alla singola persona: tale porzione deve essere scelta in modo tale che gli operatori che assistono la persona (per esempio i medici in corsia oppure il medico di base oppure il farmacista) trovino nel *tag* del paziente tutte le informazioni di cui hanno bisogno per svolgere le loro operazioni. Purtroppo la memoria di un *tag* RFID (e dei dispositivi di memorizzazione in generale) è limitata e non è possibile ospitare al suo interno tutte le informazioni inerenti un paziente. Occorre quindi modificare il tipo di informazioni contenute nel *tag* al variare del contesto in cui il paziente si trova (esempio, ricoverato, malato cronico, ...).

Vediamo ora come determinare le possibili combinazioni di informazioni che devono essere disponibili localmente. Per prima cosa introduciamo il concetto di ambiente locale (o più brevemente ambiente): l'insieme delle caratteristiche personali ed ambientali che determina la porzione di dati che deve essere disponibile localmente. L'ambiente guiderà la scelta di quale sottinsieme delle informazioni globali sarà reso disponibile anche localmente. La metodologia di progetto prevede i seguenti passi:

**1.** Si individuano tutte le informazioni più rilevanti del dominio applicativo e, sfruttando le tecniche e metodologie tradizionali, si disegna lo schema concettuale dell'intero database. Questo schema dovrà contenere tutte le informazioni di interesse, senza distinguere tra quelle che devono essere disponibili localmente e quelle che sono disponibili solo presso il server del sistema informativo.

**2.** Si identificano le "dimensioni di analisi" che permetteranno di costruire l'ambiente del dispositivo, e quindi individuare i dati che devono essere disponibili localmente. Una

dimensione di analisi è un aspetto comune a più problemi e situazioni, utilizzato con scopi di classificazione ed analisi. Elenchiamo ora alcune dimensioni, che possono essere usate e integrate con altre a seconda del dominio applicativo considerato:

- L'*utilizzatore* del dispositivo fisico al quale sono attaccate le informazioni. A seconda del tipo di utilizzatore cambiano le informazioni che un soggetto desidera avere memorizzate. Per esempio i medici desiderano accedere a informazioni su tutti i loro pazienti, mentre un paziente desidera accedere solo alle sue informazioni personali, magari con un maggior livello di dettaglio.
- L'*argomento di interesse* dell'utilizzatore in un particolare istante. Nell'esempio precedente per un medico gli argomenti possono essere: "ricette" oppure "malattie croniche".
- La *situazione* fa riferimento al fatto che durante la vita dell'utilizzatore, per un determinato argomento, le informazioni desiderate possono cambiare in base alla sua situazione. Nel nostro esempio individuiamo la situazione "paziente ospedalizzato" che si riferisce a persone ricoverate e "paziente regolare" che si riferisce allo stato ordinario di una persona.

Il risultato di questo passo è un elenco di dimensioni di analisi, che fungerà da guida per la scelta delle informazioni da ospitare localmente. Nel nostro esempio l'elenco è: <*utilizzatore, argomento di interesse, situazione*>

**3.** In questa fase, si sceglie la principale dimensione di analisi (per esempio, rispetto all'utilizzatore) e per ogni valore che essa può assumere viene creata una vista concettuale dei dati di interesse. Nel nostro caso viene creata una vista per i medici, una per i pazienti e una per ogni altro tipo di utente (esempio, amministratore dell'ospedale, infermiere ecc.). Le viste devono essere congruenti con lo schema concettuale globale del database precedentemente creato. Per soddisfare questo requisito, in questa fase può essere necessaria una modifica dello schema concettuale globale.

**4.** In questa fase si combinano i valori delle dimensioni di analisi, eliminando le combinazioni di valori senza senso. Queste combinazioni vengono chiamate *chunk* (pezzo).

Esempi di *chunk* per un database di informazioni sanitarie sono:

<*paziente, malattia cronica, ospedale*>

questo *chunk* rappresenta tutte le informazioni di cui un paziente ha bisogno relativamente alle sue malattie croniche quando ricoverato,

<*paziente, ricetta, regolare*>

questo *chunk* contiene tutte le informazioni di cui un paziente ha bisogno relativamente alle sue ricette in una situazione normale,

<*dottore, ricetta, regolare*>

questo *chunk* contiene tutte le informazioni sulle ricette che un dottore ha effettuato su pazienti regolari.

Si noti che il *chunk*

<*dottore, informazioni amministrative, ospedale*>

costituisce invece un esempio di combinazione senza senso nel dominio applicativo considerato.

**5.** A questo punto, tenendo conto dei *chunk* prodotti nelle situazioni precedenti, si possono disegnare lo schema logico (per esempio, relazionale) del database centrale e gli schemi logici dei *chunk*, che dovranno essere ospitati nei singoli dispositivi locali (esempio, tag RFID, smartcard o computer palmari).

Per ogni *chunk* si genera una vista dei dati associati al *chunk* a partire dalle viste associate ai singoli valori della dimensione principale.

I *chunk* generati in questa fase sono raggruppati in modo da definire quali informazioni devono essere immagazzinate in un singolo dispositivo locale. Per esempio, l'RFID di un paziente dovrà contenere tutti i *chunk* relativi alla situazione di un paziente regolare più quelli collegati alle sue malattie croniche e alle sue ricette. Ovviamente, il *chunk* dello specifico paziente è una istanza di quello definito per il paziente generico, e contiene solo i suoi dati. Quando un paziente è ricoverato, i *chunk* relativi alla situazione regolare sono rimossi per lasciare spazio a quelli relativi alla situazione "ospedalizzato".

**6.** Poiché i *chunk* verranno implementati come viste sullo schema logico globale, occorre tener conto anche di alcune caratteristiche fisiche dei dispositivi locali, come le dimensioni di memoria, e alcune caratteristiche di contesto, come la disponibilità di collegamento con il database centrale, il tempo ed il luogo in cui

queste informazioni sono necessarie e i diritti d'accesso degli utenti. Queste informazioni di contesto possono aiutare a ridurre ulteriormente le informazioni che devono essere disponibili localmente.

#### 4. CONCLUSIONI

Sia la tecnologia RFID sia le WSN aumentano la capacità di interazione dei dispositivi di elaborazione con il mondo reale, riducendo in questo modo il divario tra "mondo digitale" e "mondo fisico".

Le WSN permettono di raccogliere informazioni su un ambiente senza richiedere la messa in posa di cavi. Per questo motivo le WSN possono essere impiegate in tutte quelle aree che, per la difficoltà, per il costo, per i tempi richiesti dalle operazioni di cablaggio, difficilmente possono essere dotate dei tradizionali sensori a filo.

Gli RFID sono dei contenitori di informazioni che possono essere disseminati all'interno di un ambiente, possono essere facilmente

identificati, la loro memoria dati (quando presente) può essere letta ed aggiornata da dispositivi elettronici.

Vorremmo far notare che entrambe le tecnologie sono utilizzate per gestire dati: le WSN raccolgono informazioni sul mondo reale (temperatura, suoni, umidità, ...), i tag RFID permettono invece di associare oggetti reali con dei dati che possono essere facilmente letti, elaborati ed aggiornati da dispositivi elettronici.

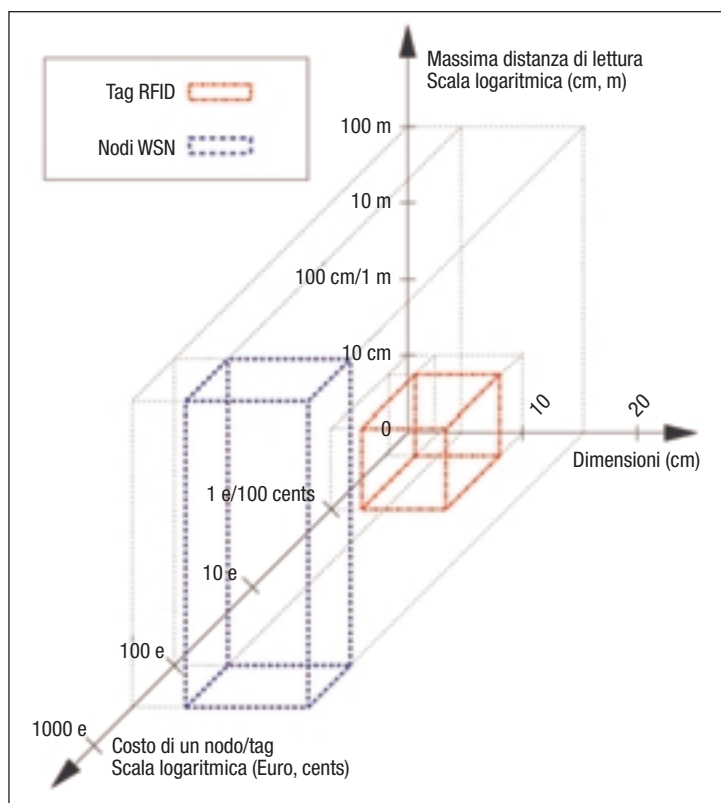
Alcune operazioni possono essere svolte da entrambe le tecnologie: un *mote* di una WSN può essere agganciato ad un oggetto per conservare informazioni, in modo simile ad un tag RFID. Ciò è reso possibile dall'alta versatilità dei *mote* che possono essere programmati in modo da comportarsi come tag RFID attivi. Un tag RFID passivo ha un costo e delle dimensioni inferiori rispetto ad un *mote* di una WSN, per questo nelle situazioni dove le maggiori performance dei tag attivi non servono, si preferisce l'utilizzo dei tag passivi.

Proponiamo ora un confronto simbolico tra le due tecnologie basandoci su 4 dimensioni di analisi:

- costo unitario
- dimensioni<sup>7</sup>
- massima distanza di comunicazione
- versatilità e prestazioni offerte

I valori delle prime tre dimensioni sono rappresentati negli assi della figura 4, all'interno della quale delle aree di diverso colore evidenziano il posizionamento delle due tecnologie all'interno dello schema.

Come si può osservare dallo schema, il costo ridotto di un tag RFID permette di sviluppare applicazioni che ne utilizzano grandi quantità. Le ridotte dimensioni facilitano l'inserimento dei tag nell'ambiente o negli oggetti che devono essere contrassegnati. Le maggiori distanze di comunicazione (fino a 100 m) favoriscono invece l'impiego dei *mote*. La maggior capacità di elaborazione (non riportata nello schema) e le maggiori distanze di comunica-



**FIGURA 4**  
Riepilogo delle dimensioni, dei costi e delle distanze di lettura di mote di WSN e di RFID

7 Come dimensioni abbiamo considerato il raggio della più piccola (ipotetica) sfera, capace di contenere l'oggetto in considerazione. Sebbene il volume sarebbe stata una unità di misura più significativa, non è utilizzabile come strumento di confronto poiché gli RFID hanno tutti una altezza quasi nulla e quindi un volume quasi nullo.

zione dei mote tuttavia incidono sul costo di un singolo *mote*, che risulta notevolmente superiore a quello di un singolo *tag*.

Le tecnologie impiegate dagli RFID e dalle WSN sono in costante evoluzione ed è lecito attendersi per i prossimi anni delle novità significative. Ci si aspetta che le dimensioni dei dispositivi, soprattutto dei *mote* delle WSN, diminuiranno sensibilmente fino ad ottenere dei dispositivi dalle dimensioni di pochi mm. Dispositivi di dimensioni minori di quelle attuali molto probabilmente si diffonderanno negli ambienti di lavoro e domestici.

Ambienti attrezzati con *tag* RFID e WSN permetteranno di raccogliere una gran quantità di informazioni, su processi logistici, processi di business e comportamenti abituali delle persone. Ciò richiederà ai sistemi informativi e ai database di gestire un numero molto maggiore di informazioni rispetto a quelle rese disponibili attualmente da tecnologie simili (per esempio, i lettori di codici a barre). La pervasività di queste tecnologie, la quantità di informazioni che una WSN, dispiegata anche all'interno di un ambiente domestico, può raccogliere solleva complesse problematiche connesse anche con la tutela dei dati personali. In questo settore si stanno muovendo i primi passi e ci si attendono sviluppi futuri interessanti.

## Bibliografia

- [1] AA.VV.: *RFID tra presente e futuro*. Collana Quaderni AIP, Politecnico di Milano \_ Dipartimento di Ingegneria Gestionale, 2005. <http://www.osservatori.net>.
- [2] Borodulkin L., Ivanov B., Zhelondz O., Ruser H.: *Distributed smart sensor system for indoor climate monitoring*. In KONNEX Scientific Conference, ott. 2002.
- [3] Bai H., Atiquzzaman M., Lilja David: *Wireless sensor network for aircraft health monitoring*. Proceedings of the First International Conference on Broadband Networks, IEEE Computer Society, 2004, p. 748-750.
- [4] Bolchini C., Schreiber F. A., Tanca L.: A context-aware methodology for very small data base design. *SIGMOD Rec.*, Vol. 33, n. 1, 2004, p. 71-76.
- [5] Boss R.: *Library RFID technology*. Library Technology Reports, nov.-dic. 2003.
- [6] Brennan S. M., Mielke A. M., Torney D. C.: Radiation detection with distributed sensor networks. *IEEE Computer*, Vol. 37, n. 8, ago. 2004, p. 57-59.
- [7] Brewin B.: *Delta has success in RFID baggage tag test*. Computerworld, 18 dic. 2003.
- [8] Burrell J., Brooke T., Beckwith R.: *Vineyard computing: Sensor networks in agricultural production*. IEEE Pervasive Computing, Vol. 3, n. 1, gen.-mar. 2004, p. 38-45.
- [9] Culler D., Estrin D., Srivastava M.: Guest editors' introduction: Overview of sensor networks. *Computer*, Vol. 37, n. 8, 2004, p. 41-49.
- [10] Dulman S., Havinga P.: *Operating system fundamentals for the eyes distributed sensor network*. In: Proceedings of the 3-rd PROGRESS Workshop on Embedded Systems, Utrecht, The Netherlands, ott. 2002. [http://www.home.cs.utwente.nl/~dulman/docs/paper\\_progress.pdf](http://www.home.cs.utwente.nl/~dulman/docs/paper_progress.pdf)
- [11] Dunkels A., Gronvall B., Voigt T.: *Contiki - a lightweight and flexible operating system for tiny networked sensors*. In LCN '04: Proceedings of the 29-th Annual IEEE International Conference on Local Computer Networks, Washington, DC, USA, 2004, p. 455-462. IEEE Computer Society.
- [12] Manges W. W., e altri: Intelligent wireless sensors for industrial manufacturing. *Sensors Magazine*, Vol. 17, n. 4, apr. 2000.
- [13] Mocchi A., Montalto M.: *Sharp<sup>2</sup>anel: un pannello a messaggi variabili basato su tecnologia RFID*. Tesi di laurea, Politecnico di Milano, 2004.
- [14] Feder B. J.: *Delta to invest in radio tags for luggage at airports*. The New York Times, 1 lug. 2004.
- [15] Kalawsky R.S., Neild I.: *Sensing activities of daily living on a limited power budget*. In 2-nd International Workshop on Body Sensor Network, Imperial College, London, apr. 2005.
- [16] Kaiser W.J., Pottie G.J., Srivastava M., Sukhatme G.S., Villasenor J., Estrin D.: *Networked infomechanical systems (nims) for ambient intelligence*. Relazione Tecnica n. 31, UCLA Center for Embedded Networked Sensing, dic. 2003.
- [17] Kurata N., Spencer B. F. Jr., Ruiz-Sandoval M., Miyamoto Y., Sako Y.: *A study on building risk monitoring using wireless sensor network mica-mote*. Proceedings of the First International Conference on Structural Health Monitoring and Intelligent Infrastructure, nov. 2003.
- [18] LaMarca A., Brunette W., Koizumi D., Lease M., Sigurdsson S. B., Sikorski K., Fox D., Borriello G.: *Making sensor networks practical with robots*. Proceedings of the First International Conference on Pervasive Computing, London, UK, Springer-Verlag, 2002, p. 152-166.



- [19] Levis P., Culler D.: *Mate: a tiny virtual machine for sensor networks*. Proceedings of the 10-th international conference on Architectural support for programming languages and operating systems, New York, NY, USA, 2002, p. 85-95. ACM Press.
- [20] Madden S.R., Franklin M.J., Hellerstein Joseph M., Hong Wei: TinyDB: an acquisitional query processing system for sensor networks. *ACM Transactions on Database System*, Vol. 30, n. 1, 2005, p. 122-173.
- [21] Malan D., Fulford-Jones T.R.F., Welsh M., Moulton S.: Codeblue: *An ad hoc sensor network infrastructure for emergency medical care*. Proceedings of the MobiSys 2004 Workshop on Applications of Mobile Embedded Systems (WAMES 2004), Boston, MA, giu. 2004. Articolo presentato anche a International Workshop on Wearable and Implantable Body Sensor Networks, London, April 2004.
- [22] Maroti M., Simon G., Ledeczi A., Sztipanovits J.: Shooter localization in urban terrain. *IEEE Computer*, Vol. 37, n. 8, 2004, p. 60-61.
- [23] McGinity M.: RFID: is this game of tag fair play? *Communications of the ACM*, Vol. 47, n. 1, 2004, p. 15-18.
- [24] Nambi S., Nyalamadugu S., Wentworth S.M.: *Radio frequency identification sensors*. Proceedings of the 7-th World Multiconference on Systemics, Cybernetics and Informatics, lug. 2003.
- [25] Rosencrance L.: *Las Vegas airport to implement RFID baggage-tag system*. Computerworld, nov. 2003.
- [26] Shih E., Bychkovsky V., Curtis D., Gutttag J.: *Continuous medical monitoring using wireless micro-sensors*. In: SenSys '04: Proceedings of the 2-nd international conference on Embedded networked sensor systems, New York, NY, USA. ACM Press, 2004, p. 310-310.
- [27] Shnyder V., Chen B., Lorincz K., Fulford-Jones T.R.F., Welsh M.: *Sensor networks for medical care*. Relazione Tecnica n. TR-08-05, Harvard University, apr. 2005.
- [28] Srivastava M.B., Muntz R.R., Potkonjak M.: *Smart kindergarten: sensor based wireless networks for smart developmental problem-solving environments*. Proceedings of the 7-th international conference on Mobile Computing and Networking, 2001, p. 132 - 138.
- [29] Szewczyk R., Osterweil E., Polastre J., Hamilton M., Mainwaring A., Estrin D.: Habitat monitoring with sensor networks. *Communications of the ACM*, Vpl. 47, n. 6, 2004, p. 34-40.
- [30] Todorovich P.: *RFID, come rivoluzionare l'identificazione*. Approfondimenti di Computer World Online, 9 lug. 2003.
- [31] Want R.: Enabling ubiquitous sensing with RFID. *IEEE Computer*, Vol. 37, n. 4, 2004, p. 84-86.
- [32] Xu N., Rangwala S., Chintalapudi K., Ganesan D., Broad A., Govindan R., Estrin D.: *A wireless sensor network for structural monitoring*. Proceedings of the ACM Conference on Embedded Networked Sensor Systems(Sensys04), nov. 2004.
- [33] Ying Z., Ackerson L., Duff D., Eldershaw C., Yim M.: Tsam: a system of tracking and mapping in real environments. *Wireless Communications, IEEE*, Vol. 11, n. 6, dic. 2004.
- [34] Zhao F., Guibas L.: *Wireless Sensor Networks*. Elsevier, Morgan Kauffman, 2004. ISBN: 1-55860-914-8.

### Riferimenti a siti WEB

I siti web di seguito elencati sono stati visitati in data 29 luglio 2005.

- [35] <http://www.caifirenze.it/arva/>
- [36] <http://www.rfidjournal.com/article/article-view/344/1/1/>
- [37] <http://www.bluetooth.com/>
- [38] <http://www.ece.osu.edu/~greenc/Tiny%20OS%20Info/>
- [39] <http://www.citris.berkeley.edu/applications/disaster-response/smartbuildings.html>
- [40] <http://www.xbow.com/>
- [41] <http://www.autentica.it/>
- [42] <http://firebug.sourceforge.net/>
- [43] <http://www.ti.com/tiris/docs/solutions/epc/retail.shtml>
- [44] <http://envisense.org/>
- [45] <http://www.geospatial-online.com/geospatial-solutions/article/articleDetail.jsp?id=93546>.
- [46] <http://www.cbe.berkeley.edu/research/xyz/FederspielCEBO2002.pdf>
- [47] <http://www.calitz.net/technology/features/sensorNetClass/AirVentCalibration.pdf>. [48] <http://www.rfidjournal.com/article/view/309/1/1/>
- [49] <http://www.indicod-ecr.it/>
- [50] <http://www.eecs.berkeley.edu/~binetude/course/cs294-1/paper.pdf>
- [51] <http://www.cs.cornell.edu/People/egs/magnetos/>
- [52] <http://www.rfidjournal.com/article/article-view/369/1/1/>
- [53] <http://www.metrogroup.de/servlet/PB/menu/1014631/index.html>
- [54] <http://wins.rockwellscientific.com/>
- [55] <http://www.semiconductors.philips.com/markets/identification/products/mifare/>
- [56] <http://www.semiconductors.philips.com/markets/identification/articles/success/s46/> [57] <http://www.ee.ucla.edu/~boulis/phd/SensorWare.html>





- [58] <http://www.oracle.com/corporate/press/2005-mar/dolomiti.html>
- [59] <http://www.intel.com/research/prohealth/cs-aging-in-place.htm>
- [60] <http://www.tinyos.net/>
- [61] <http://www.tinyos.net/nest/doc/tutorial/tos-sim-lesson.html>
- [62] <http://www.mobilitytechnologies.com/ntdc/>
- [63] <http://www.zigbee.org/>
- [64] <http://informationweek.com/shared/printableArticle.jhtml?articleID=20600021>
- [65] <http://www.rfidjournal.com/article/article-view/527/1/26/>
- [66] <http://wi-fi.org/>
- [67] <http://www.bartecautoid.com/applications-bartec.html>

MIRKO CESARINI ha conseguito il dottorato di ricerca in Ingegneria dell'Informazione presso il Politecnico di Milano nel 2005. I suoi interessi di ricerca riguardano: context aware programming, pervasive computing, sistemi di localizzazione e tecnologia RFID.  
E-mail: cesarini@elet.polimi.it

CARLO GHEZZI è Professore Ordinario di Ingegneria del Software, membro del Senato Accademico del Politecnico di Milano e responsabile scientifico dell'area di ricerca sull'Ingegneria del Software presso il CEFRIEL. È Fellow dell'ACM (Association for Computing Machinery) e Editor in Chief della rivista ACM Transactions on Software Engineering and Methodology. È autore di numerosi articoli scientifici e libri. Ha interessi di ricerca nell'ambito dei linguaggi di programmazione e dell'ingegneria del software, con particolare riguardo ai fondamenti teorici, metodologici e tecnologici dello sviluppo delle applicazioni distribuite su rete.  
E-mail: carlo.ghezzi@polimi.it

FABIO A. SCHREIBER è Professore Ordinario di Tecnologie dei Sistemi Informativi presso la Facoltà di Ingegneria del Politecnico di Milano. I suoi interessi di ricerca riguardano i Sistemi di Elaborazione Distribuiti e i Sistemi Informativi. Attualmente si occupa in particolare di problematiche relative ai Sistemi Informativi per applicazioni mobili, dipendenti dal contesto e comprendenti dispositivi utente molto piccoli (PDA, telefoni cellulari ecc.). È autore di numerosi articoli scientifici ed è associate editor di Data & Knowledge Engineering e di Decision Support Systems.  
E-mail: schreibe@elet.polimi.it

LETIZIA TANCA è Professore Ordinario di Basi di Dati presso il Politecnico di Milano, e presidente del Consiglio di Corso di Studi di Ingegneria Informatica del Politecnico di Milano, campus Milano Leonardo. È autrice di diverse pubblicazioni internazionali sulle basi di dati e sulla teoria delle basi di dati, e del libro "Logic Programming and Databases", scritto con S. Ceri e G. Gottlob. Ha partecipato a vari progetti nazionali e internazionali. I suoi interessi di ricerca riguardano tutta la teoria delle basi di dati, in particolare le basi di dati deduttive, attive e orientate agli oggetti, i linguaggi a grafi per basi di dati, la rappresentazione e l'interrogazione di informazione semistrutturata, le basi di dati per piccoli dispositivi.  
E-mail: tanca@elet.polimi.it