

Thème 3 : Infrastructures logicielles et communicantes pour l'embarqué : protocoles, OS, middleware, réseaux de capteurs, sécurité, réseaux sur puce

Présentation

Didier Donsez (LIG, Adele)

Marine Minier (CITI)

David Navarro (INL)

Ian O'Connor (INL)

Fabrice Theoleyre (LIG, Drakkar)

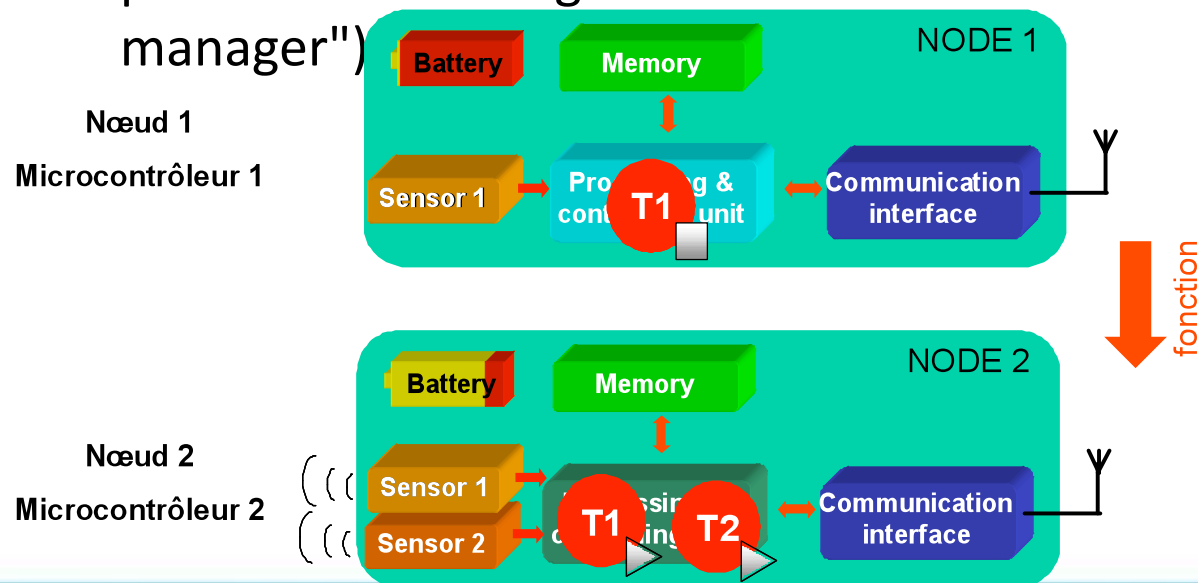
Echange de fonctionnalité et Reconfiguration dynamique dans un WSN hétérogène (nœuds matériellement différents) :

- ✓ **MODULARITE** du réseau (ajout de nœuds ou de fonctions)
- ✓ **INTEROPERABILITE** des nœuds et des réseaux
- ✓ **SURETE DE FONCTIONNEMENT** au niveau du réseau (conservation des tâches effectuées)

Cible: microcontrôleurs "légers" 8 ou 16 bits (< 6 mW)

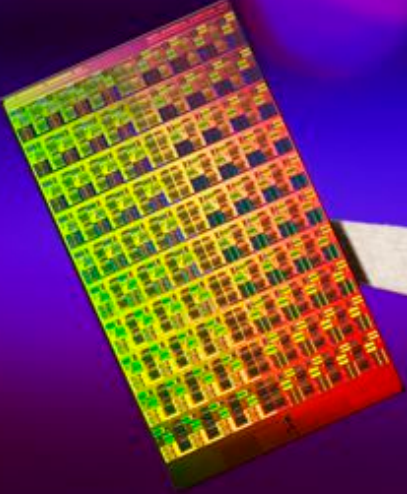
Modélisation systemC et universalité logicielle dans les réseaux de capteurs sans fil (WSN)

- ✓ Modélisation **SystemC** pour définir le langage d'échange de fonctionnalités compatible 802.15.4
- ✓ Développement d'un **interpréteur léger**
- ✓ Développement d'un **OS léger** ("lightweight OS") modulaire pour la gestion des fonctionnalités ("task manager")



Communication in Manycore Computing Architectures

80 cores
Teraflop performance



1.01TFI@3.2GHz/0.95V/62W
1.8TFI@5.7GHz/1.35V/265W

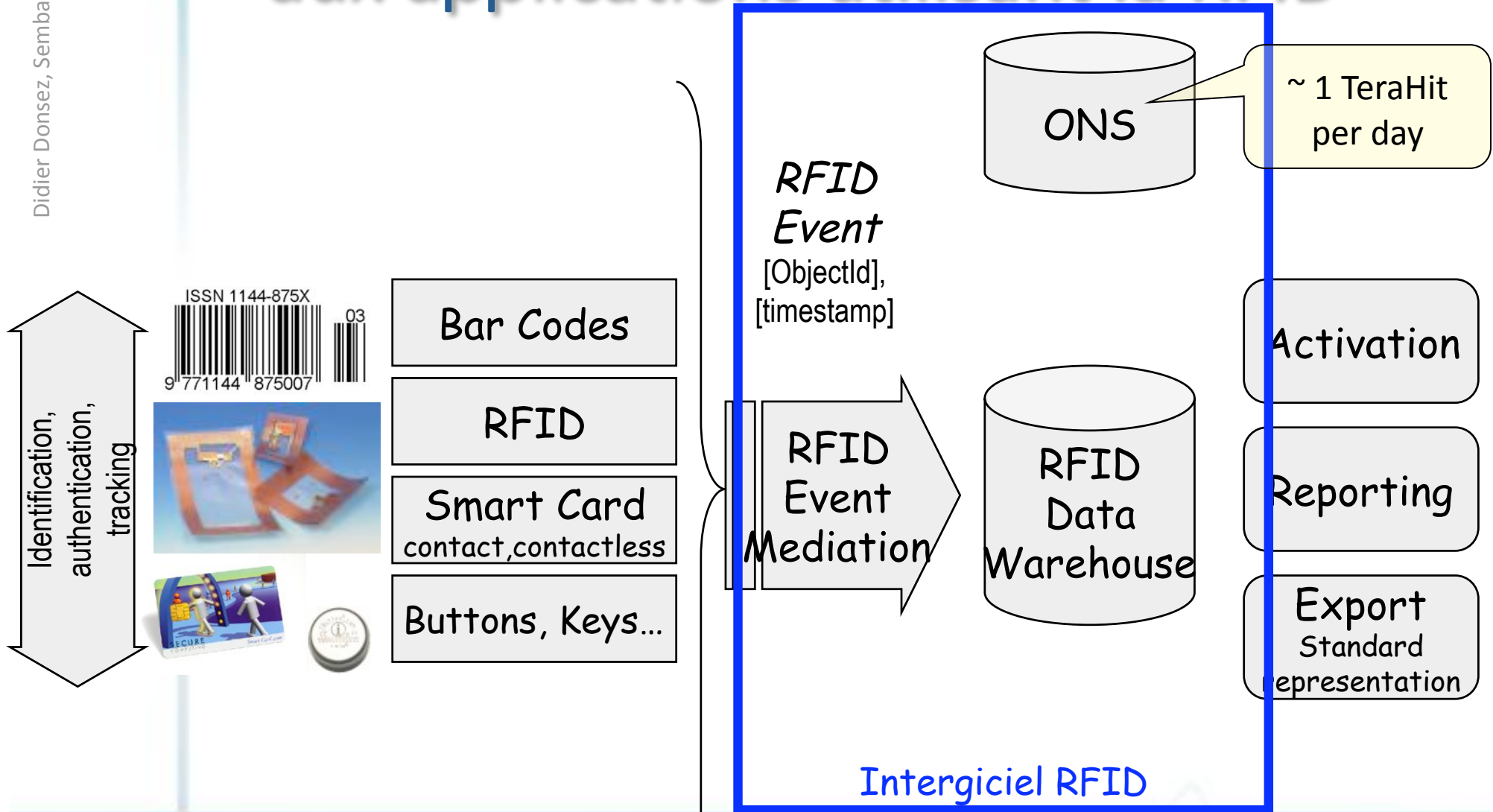
- ❖ manycore processing for a flexible and scalable computing platform
 - ✓ high-performance energy-efficient computing
 - ✓ multiple standards and constrained power
- ❖ requires organized high-speed interconnect architecture
 - ✓ Intel's Teraflops: 2.92Tb/s aggregate bandwidth
 - ✓ expected to be over 100Tb/s before 2015
- ❖ interconnect should also satisfy low power budget, reduce congestion
- ❖ all factors point to widespread use of Network-on-Chip strategies

Ian O'Connor, Semba 2009

Source: Intel

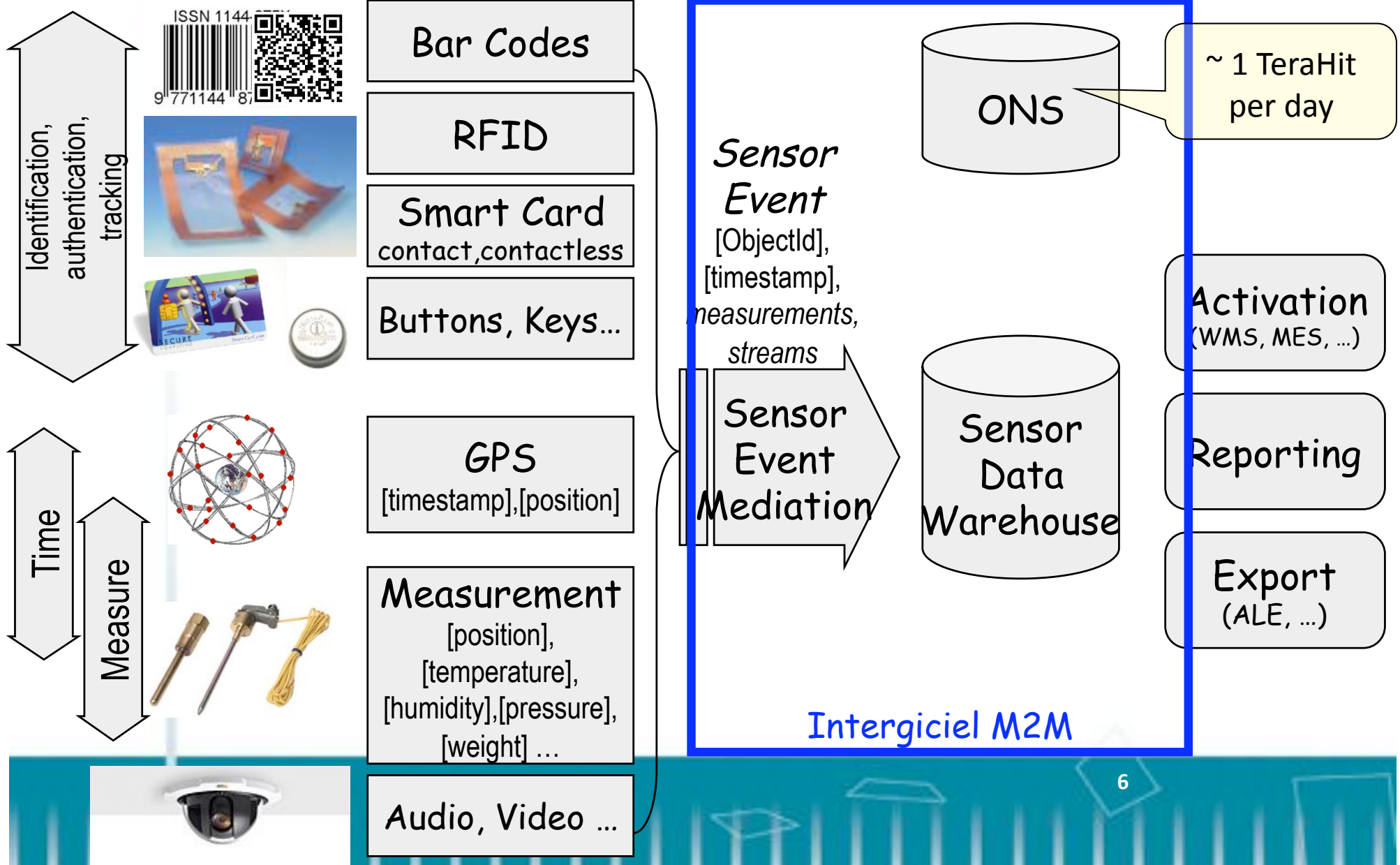
Middleware pour l'embarqué aux applications utilisant la RFID

Didier Donsez, Semba 2009



Evolution des besoins M2M :

RFID + Capteurs environnementaux



Défis

- ❖ Pour nos disciplines
 - ✓ Génie logiciel, Intergiciels, Bases de données, Grilles, Réseaux, Sécurité ...
- ❖ Problème à très grande échelle (volumétrie)
 - ✓ 80 milliards d'objets (versus 6 milliards d'humain dont 3 milliards abonnés cellphone)
- ❖ Multi-organisations
 - ✓ *Boucle ouverte vs boucle semi-ouverte/semi-fermé vs boucle fermé*
- ❖ Multi-modales
- ❖ Multi-plateformes (hétérogénéité, mixité)
- ❖ Multi-modèles logiciels
 - ✓ *Composants vs services, J2EE vs .NET, WS vs ESB/JBI, OSGi, J2ME vs Android vs iPhone ...*
- ❖ QoS hétérogènes (*RT and near-RT, Mission-Critical ...*)
- ❖ Sécurité (*propriété intrinsèque de certains objets ...*)
- ❖ Prise en charge du patrimoine (legacy, COTS, ...)
- ❖ Administration de bout en bout
- ❖ Fonctionnement (*deeply*) autonome
 - ✓ Comportement compatible avec les usages de leurs usagers/porteurs
- ❖ ...

Réseaux de capteurs

❖ Automatisation

- ✓ Air conditionné, ventilation
- ✓ Détecteur de feu et de présence
 - Organisation des secours

❖ Domotique

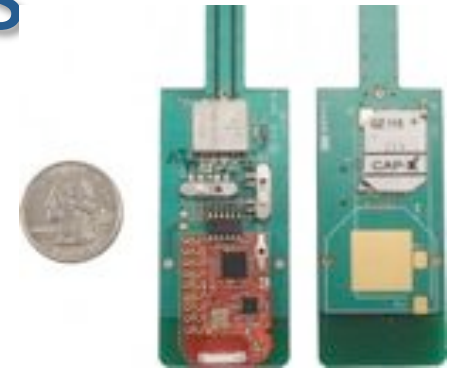
- ✓ Conservation de l'énergie
- ✓ Surveillance à distance & alarmes
- ✓ Santé

❖ Industrie

- ✓ Contrôle (boucles automatiques ou non)
- ✓ Surveillance (maintenance)

❖ Urbains

- ✓ Taux de remplissage des conteneurs
- ✓ Éclairage public
- ✓ Gestion des places de parking



Contraintes



- ❖ Optimisation de la consommation d'énergie
 - ✓ Cross-layer, protocoles, algorithmie, architecture OS
 - ✓ *Endormir* une sous-partie du capteurs (architecture en bloc)
- ❖ Capacités limitées
 - ✓ Empreinte mémoire
 - ✓ algorithmes peu gourmands en CPU
- ❖ Passage à l'échelle
 - ✓ Algorithmique
- ❖ Dynamique
 - ✓ Prendre en compte un monde variable
 - ✓ Instabilité des liens radio
- ❖ Tolérance aux fautes
 - ✓ Auto stabilisation, conception de l'OS
- ❖ Mobilité
 - ✓ Puits mobiles de collecte des données

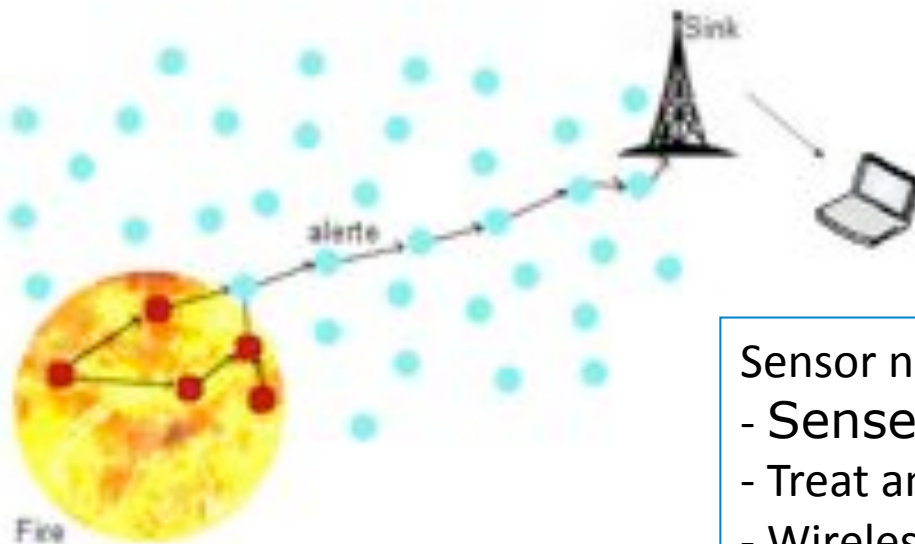


Défis

- ❖ Optimisation de la consommation d'énergie
 - ✓ Cross-layer, protocoles, algorithmie, architecture OS
 - ✓ *Endormir* une sous-partie du capteurs (architecture en bloc)
 - ✓ Limiter les transmissions (routage)
 - ✓ Protocoles MAC économes
- ❖ Sécurité
 - ✓ Environnement hostile : capteur vulnérable
 - ✓ Cryptographie peu coûteuse en distribué
 - ✓ Déni de Service
- ❖ Qualité de Service
 - ✓ Comment assurer une garantie de performances de bout en bout ?
- ❖ Synchronisation
 - ✓ Pour dater les données captées
 - ✓ Pour de nombreux algorithmes
- ❖ Orienté données
 - ✓ Routage
 - ✓ Flux continus de données

Context: Wireless Sensor Networks (WSNs)

Many applications military, temperature, health, home, ...



- No infrastructure
- Hundreds/Thousands of tiny devices
- Difficult/impossible to fix nodes: random topology
- A typical application: fire detection

Sensor nodes:

- Sensed data
- Treat and transmit them to a Sink
- Wireless communication

Embedded Systems:

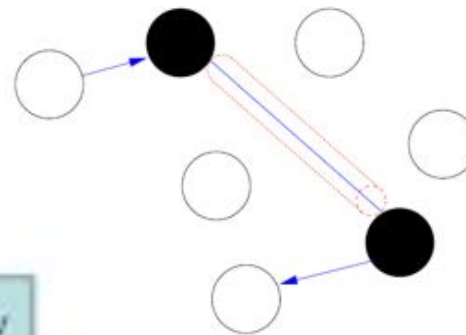
- limited energy, memory and computation resources
- Non tamper-resistant devices (physical compromising)

Security is required !

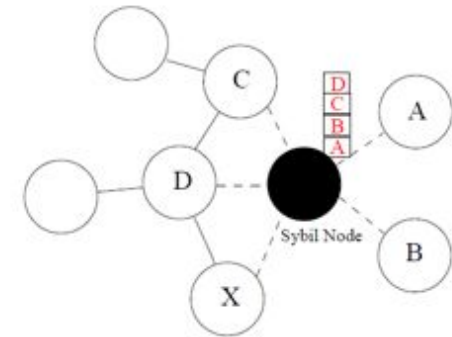
- ⇒ Confidentiality
- ⇒ Authenticity
- ⇒ Integrity
- ⇒ Authentication

Security Challenges in WSNs

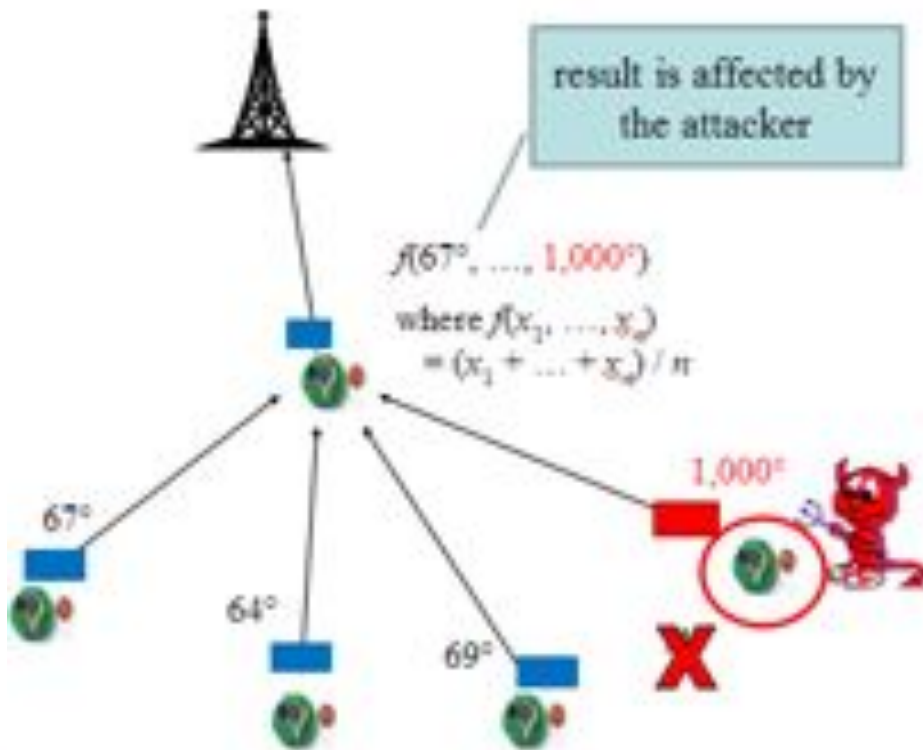
Cryptography could help for outsider attacks but are often insufficient for insider attacks



Wormhole attack



Sybil attack



Challenges:

1. Cryptographic challenges: aggregation, homomorphic functions
2. Secure Neighbors Discovery / Secure positioning
3. Detection of wormhole attacks, Replication attacks, Sybil attacks