

Réseau personnel sans fil (WPAN)

Wireless personal area networks: Overview

Lecturer: Keun-Woo Lim

COMASIC

22-09-2020





Bienvenue dans le monde de WPAN

■ technologies spécifiques de WPAN

- IEEE 802.15.4
- 6LoWPAN

■ TPs utilisant FIT/Iot-LAB

- Capteurs pour WPAN

■ Exercices de WPAN et FIT/Iot-LAB

- Communication / Réseau / Application



avant d'entrer dans les détails ...

- **Nous devons savoir: Qu'est que le WPAN?**
- **Je vais vous expliquer brièvement l'aperçu**
 - Applications?
 - Pourquoi utiliser le WPAN?
 - Quelles sont les exigences?



INSTITUT
Mines-Télécom

Aperçu général





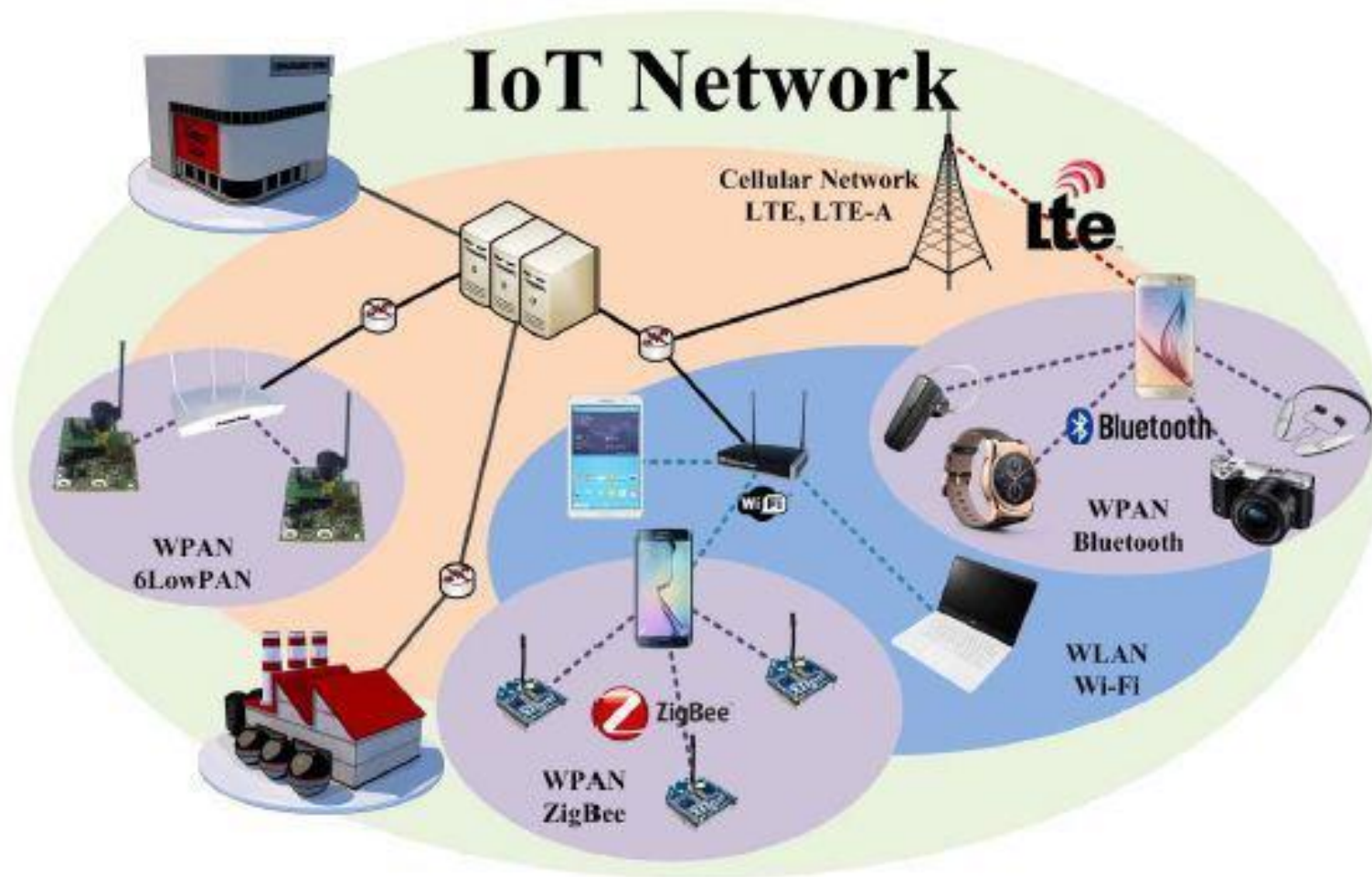
Qu'est ce que le WPAN?

■ C'est un réseau

- réseau autour d'un individu
- réseau sans fil

■ Connexion d'appareils dans votre vie

WPAN



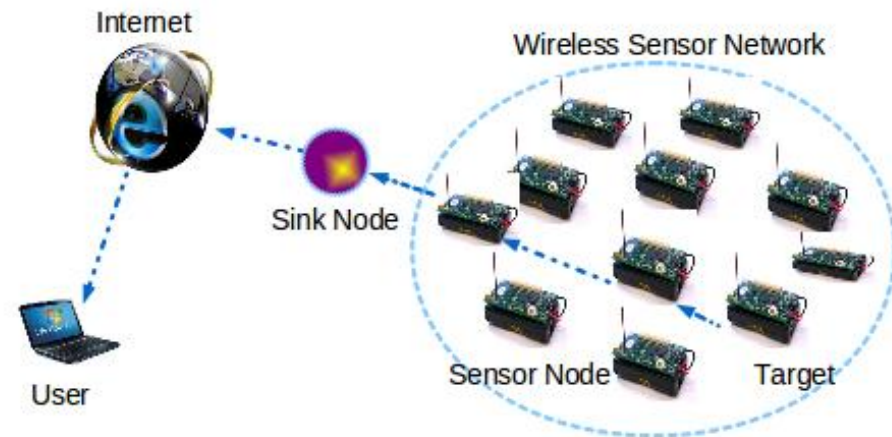
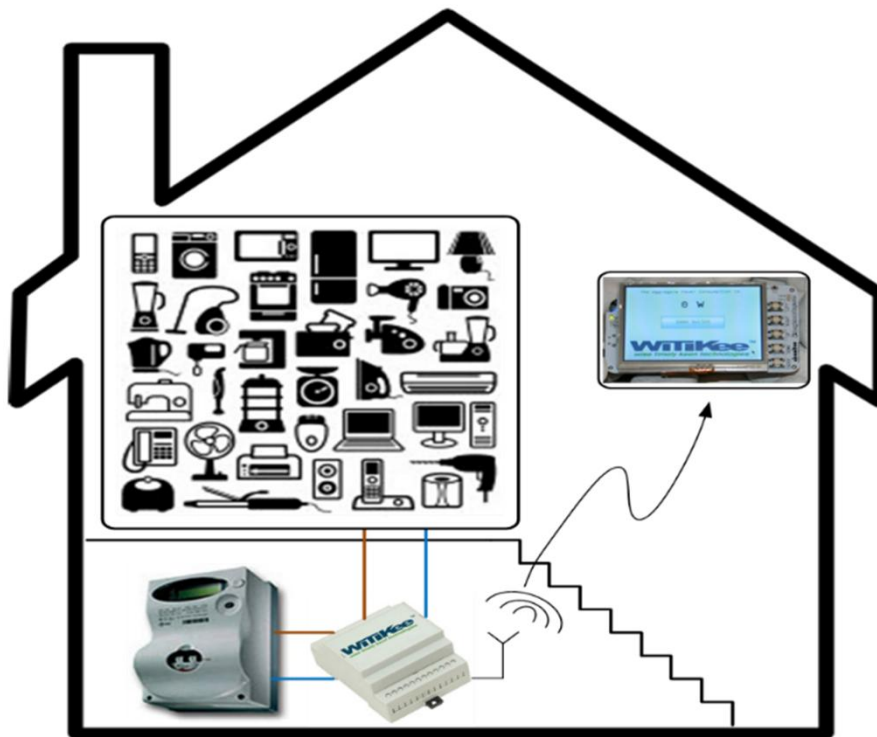
Applications de WPAN

■ Connexion d'appareils (sans fil → “cordless”)



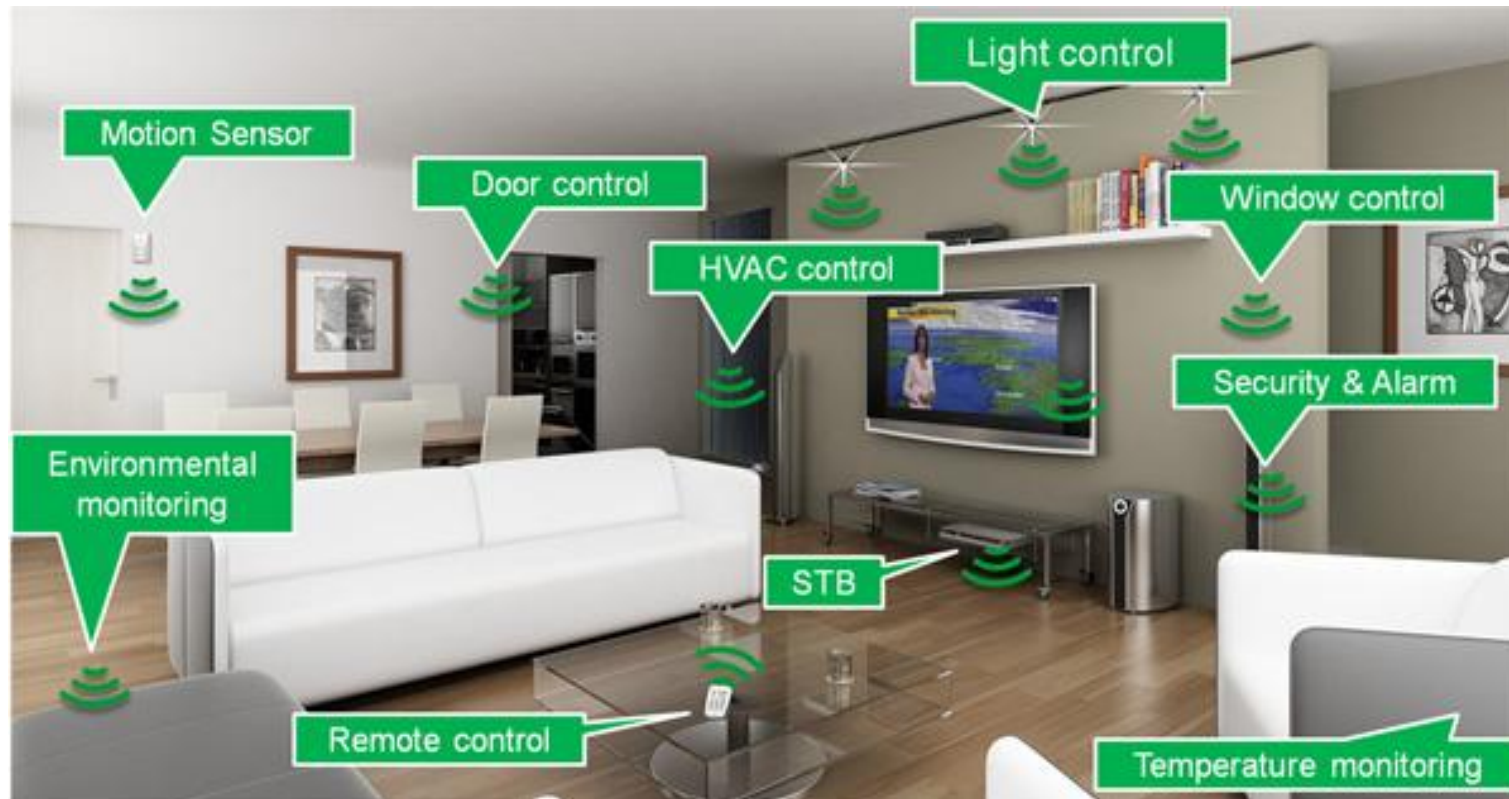
Capteurs sans fil

- Considéré comme une technologie essentielle pour les capteurs



Applications de WPAN

■ automatisatisation de la maison (maison intelligente)



Applications de WPAN

■ localisation à l'intérieur



Quelque chose de plus familier

■ Bluetooth = WPAN





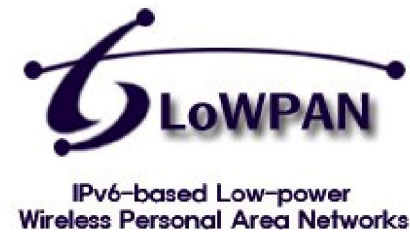
Applications de WPAN

■ Il y a beaucoup d'applications!

- bâtiments intelligents
- voitures intelligentes
- soins de santé sans fil
- surveillance de la nature
- Etc.

■ vous pouvez créer vos services de WPAN!

Quelles sont les technologies WPAN?





Un peu d'histoire...

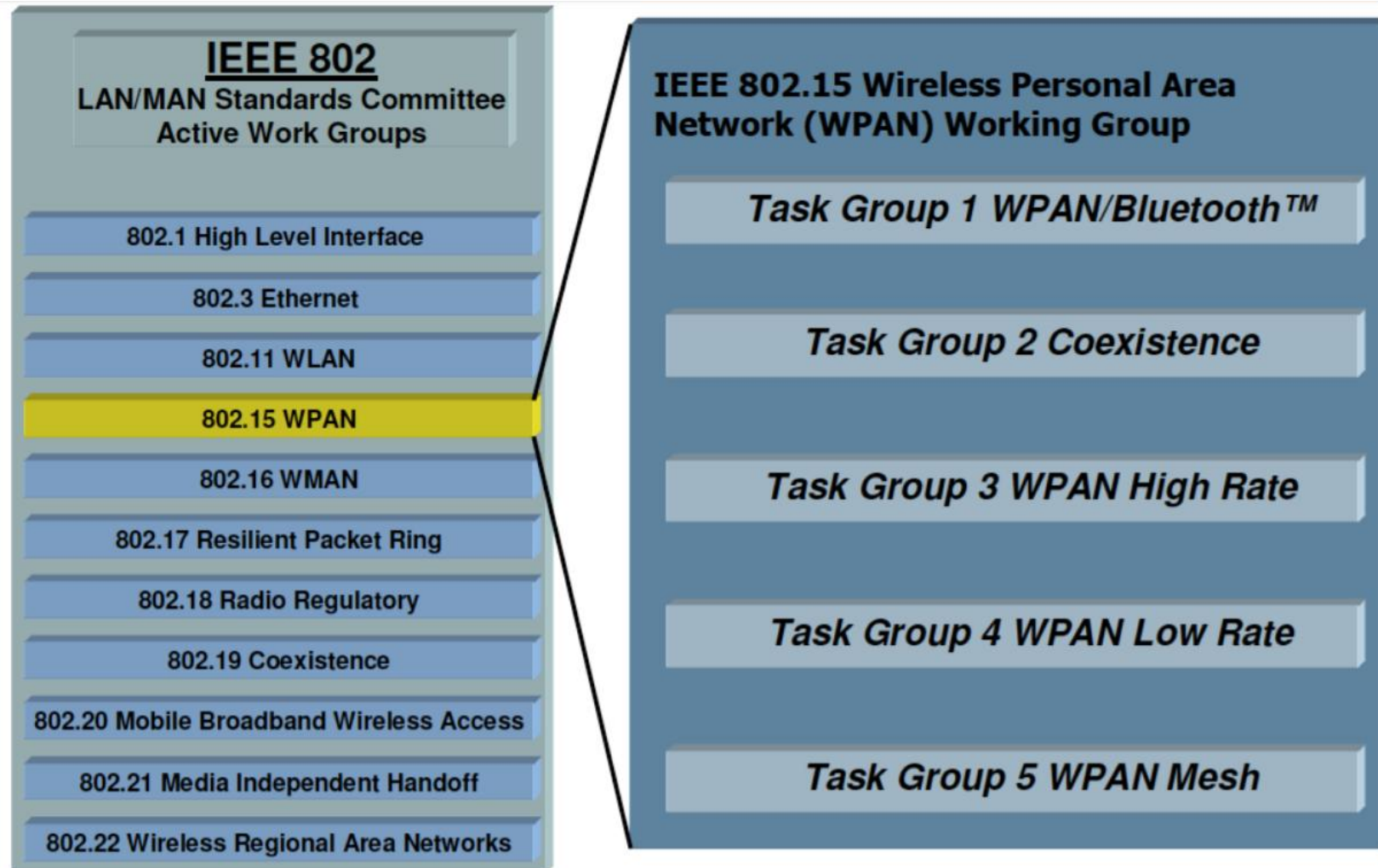
■ terme généralisé par IEEE 802.15

- standardisation pour WPAN
- IEEE 802.15.1 → ancien Bluetooth
- IEEE 802.15.4 → technologie de base pour Zigbee, 6LoWPAN

■ avec la différence d'autres standardisations connexes

- 802.11 (WLAN) : Wi-Fi
- 802.16 (WMAN) : WiMAX
- 802.22 (WRAN) : TV white space

Un peu plus details...





récemment...

- **Il existe de nombreux consortiums pour WPAN**
- **Bluetooth SIG**
 - Bluetooth LE, 5
- **Zigbee Alliance**
 - Zigbee
- **IETF**
 - 6LoWPAN
- **Etc.**



Caractéristiques de WPAN

■ Courte portée

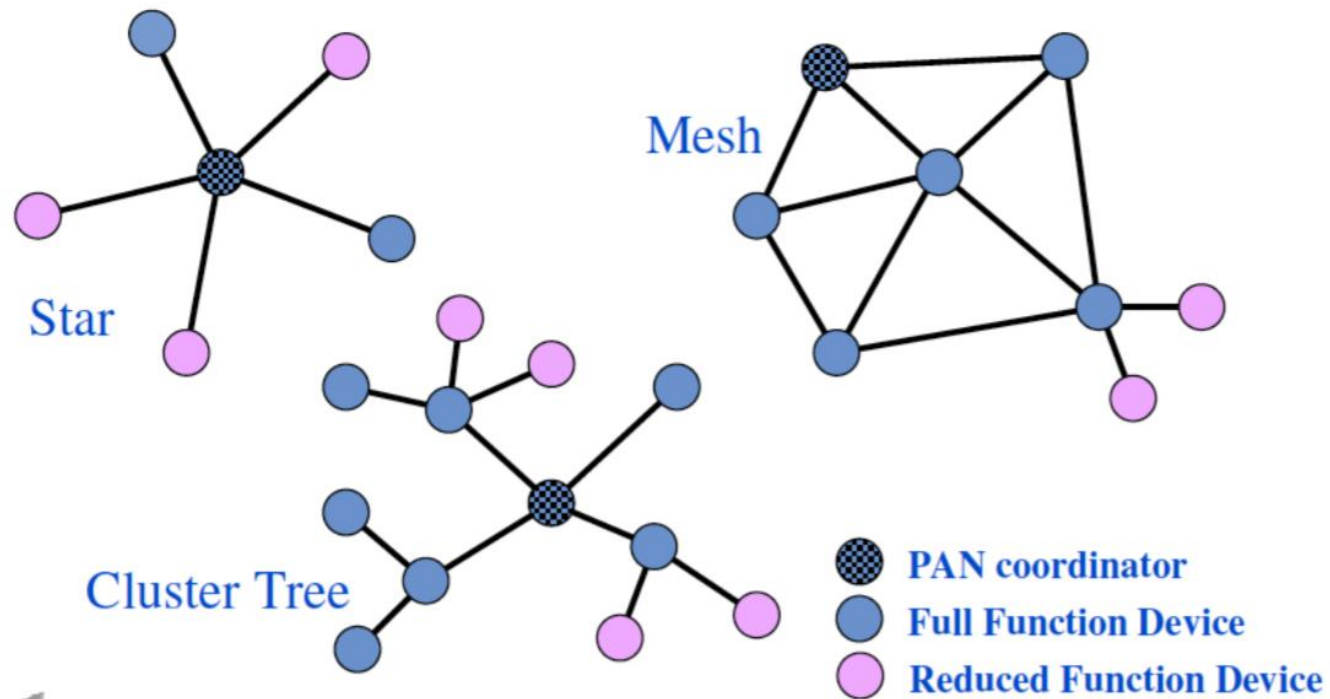
- <100m pour Zigbee
- <10m pour Bluetooth classique

■ Taux bas

- 1~2Mbps pour Bluetooth
- < 10Mbps pour 802.15.4

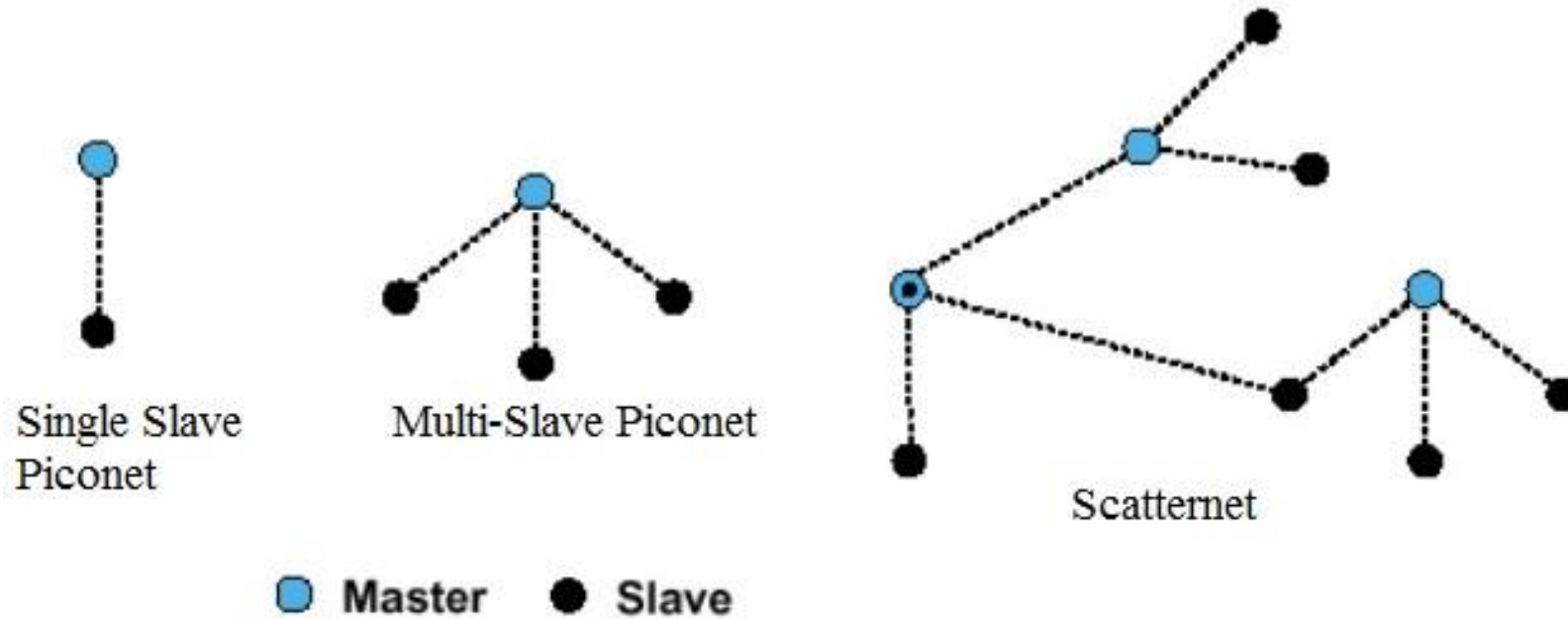
Topologies de réseau

■ selon IEEE 802.15.4



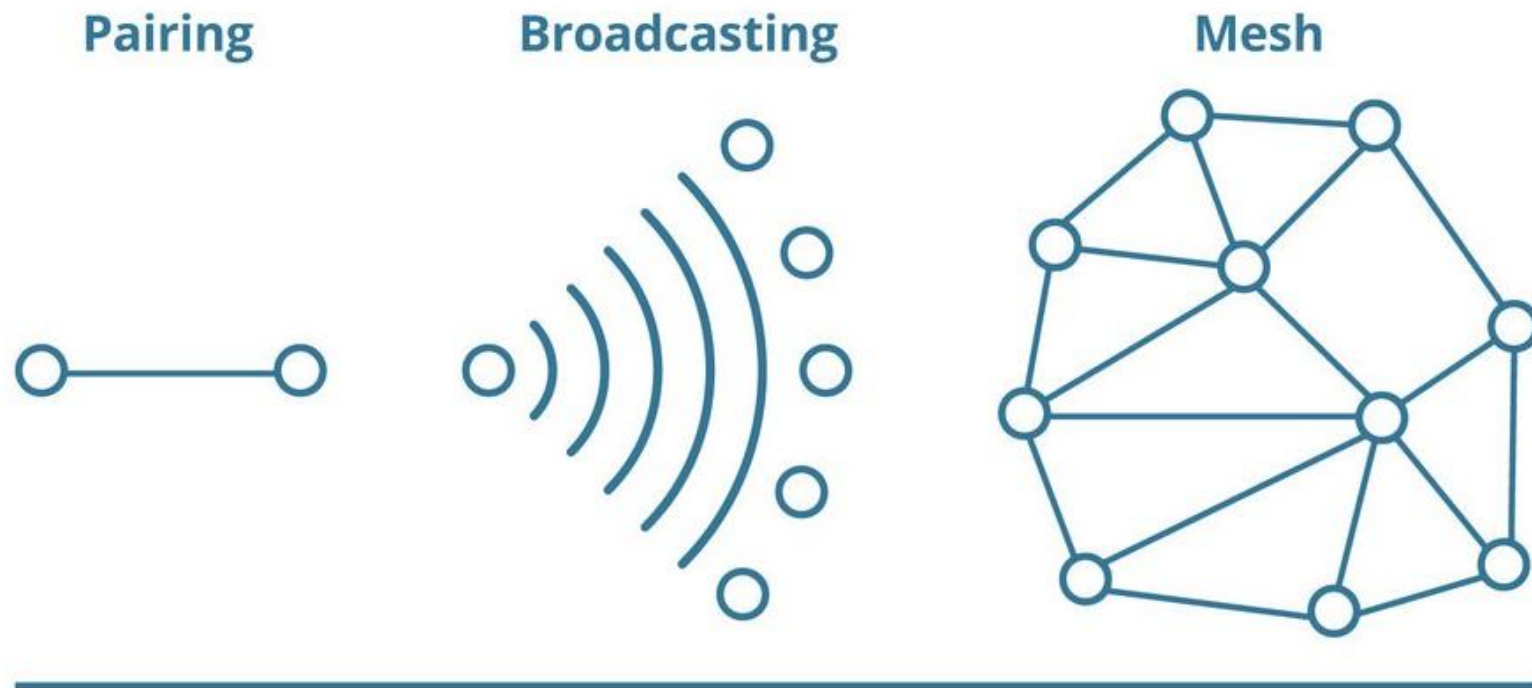
Topologies de réseau

■ selon Bluetooth



Topologies de réseau

■ selon Bluetooth LE et 5





Question!

■ Pourquoi WPAN?

- Courte portée
- c'est lent
- les services sont limités

■ Pourquoi utiliser WPAN?

- WMAN a plus de portée!
- WLAN a vitesse plus rapide!
- Cellulaire a meilleure prestation de services!



la réponse

■ Portée courte?

- C'est bon!
- moins d'interference

■ Lent?

- les services de WPAN n'ont pas besoin de vitesse élevée

■ Efficacité énergétique!

- priorité absolue

Performance de WPAN

	WiFi	Bluetooth	Zigbee	6LoWPAN	UMTS
Peak Data	High	Medium	Low	Low	Medium
Range	Medium	Low	Medium	Medium	High
Power	High	Low	Very Low	Very Low	Very High
Native IPv6	Yes	*No	*No	Yes	Yes
Topology	Star	Star	Mesh	Mesh	Star
Consortium	WiFi-Alliance	Bluetooth SIG	Zigbee Alliance	IETF, Google	3GPP

* Both Bluetooth and Zigbee are working to bring Native IPv6 in future specification

<https://www.linkedin.com/pulse/20141113232539-28901359-iot-connecting-the-dots-part-2>

■ Efficacité énergétique

	Wi-Fi	Zigbee	Bluetooth Low Energy
Sleep	10 μ W	4 μ W	8 μ W
Receive (Rx) Power	90 mW	84 mW	28.5 mW
Transmit (Tx) Power	350 mW	72 mW	26.5 mW
Average Power for 10 Messages Per Day	500 μ W	414 μ W	50 μ W

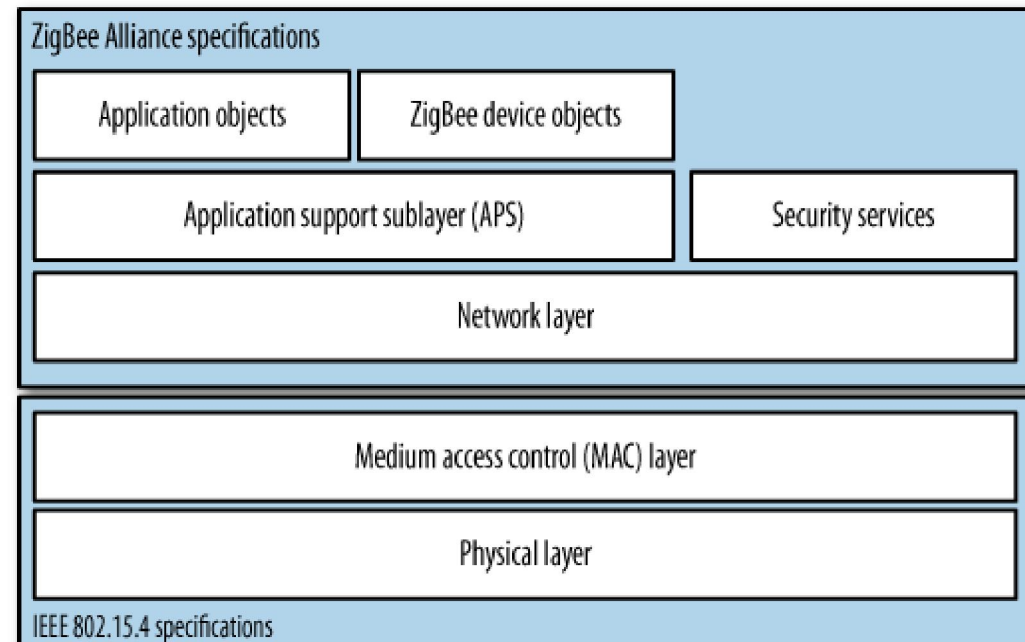
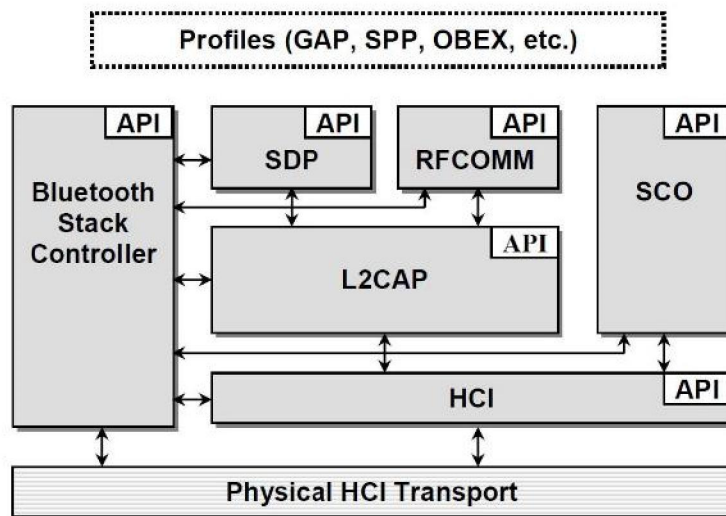
■ <https://www.rfidjournal.com/articles/view?11062>



Architecture de la pile WPAN

- **Il n'y a pas de solution de pile unique pour WPAN**
 - toutes les technologies ont une pile de réseau différente
- **Toutefois, les objectifs et la portée sont similaires**

exemples de la pile WPAN





exigences de la couche PHY

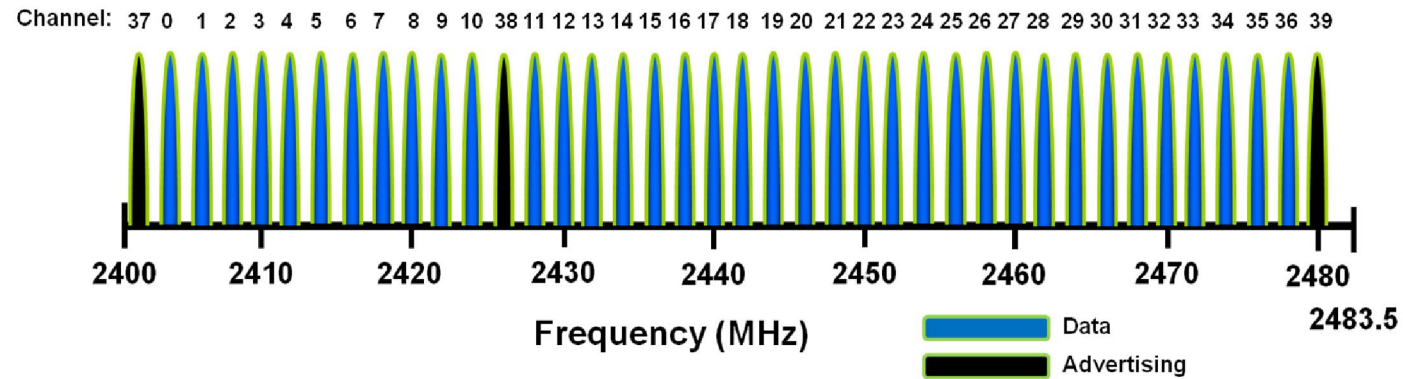
■ Faible consommation d'énergie

- Utilise une modulation plus résistante aux erreurs
 - Bluetooth: GFSK (1Mbps) (802.11ac: 256-QAM, >1Gbps)
- Contrôle de puissance
 - moins d'interférence aussi

■ utilisation de canaux plus petits

- 1~10MHz canaux (Wi-Fi: 20 ~ 160MHz)
- meilleure sensibilité / fiabilité

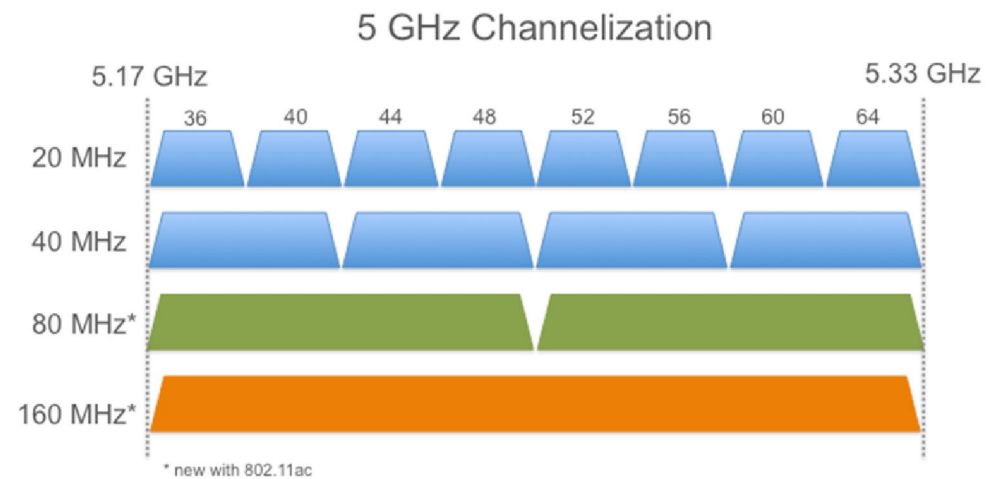
Un exemple



Bluetooth LE

VS

IEEE 802.11ac





exigences de la couche MAC

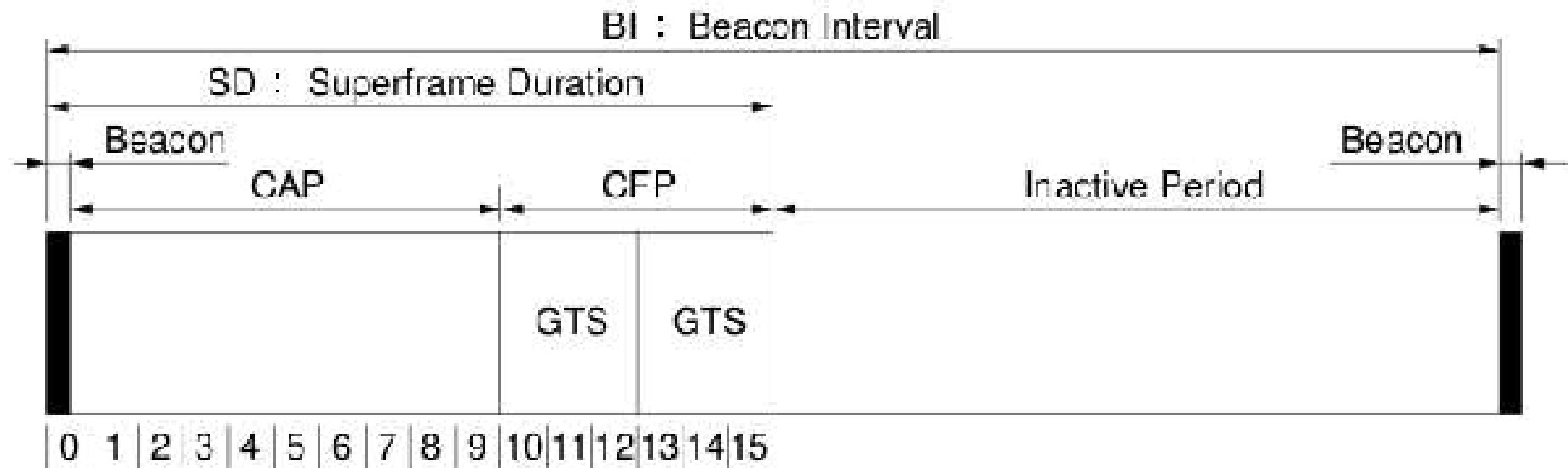
■ Faible consommation d'énergie (encore)

- Rapport cyclique
 - L'appareil s'endort

■ Par exemple,

- appareil de mesure avec intervalle de données de 60 secondes
 - L'appareil peut dormir pour 59 seconds (rapport cyclique 1.6%)

Un exemple



IEEE 802.15.4 Duty cycle



Exigences de la couche réseau

■ Service à grande échelle

- Compatibilité avec autres services
- Facile à utiliser
- Facile à trouver

■ Par exemple,

- 6LoWPAN utilise IPv6 pour utilisation globale
- Et compatibilité aussi



exigences de la couche Routage

■ Faible consommation d'énergie (et encore...)

- Les protocoles de routage multi-hop peuvent entraîner une grande complexité dans un réseau
- doit être efficace!

■ Par exemple,

- RPL dans 6LoWPAN
- Zigbee routage

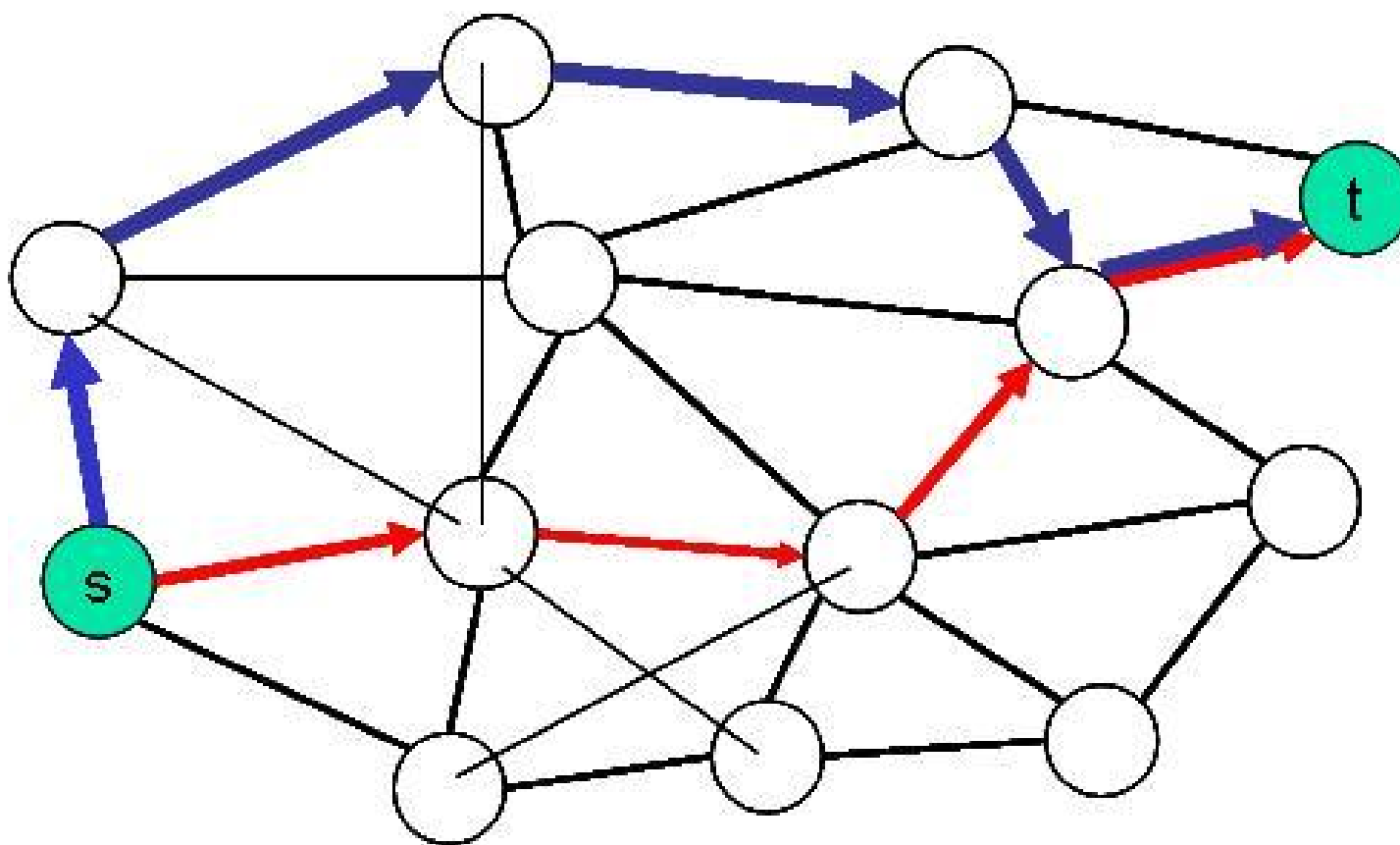


Question

- **Connaissez-vous les differences entre les protocoles routages filaire vs sans fil?**
 - algorithme?
 - efficacité énergétique?
 - comprenez-vous pourquoi il existe des protocoles de routage sans fil?

Une exemple d'efficacité énergétique

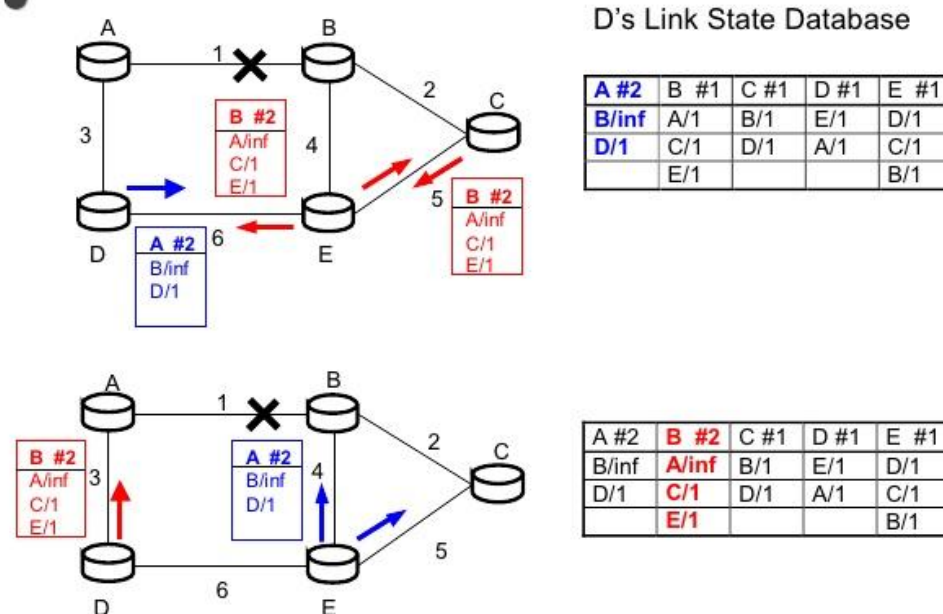
Problem with bad routing with extra hops



Une exemple d'efficacité énergétique

Control message problem

Operation of Link State Routing: The Flooding Protocol at Work (2)



Copyright © 1998

Internet Routing Protocols: Fundamental Operation

25



exigences de la couche de sécurité

- **doit être sécurisé! (bien sûr)**
- **en même temps, doit avoir une faible complexité!**
 - les méthodes d'authentification / cryptage solides ne peuvent pas être utilisées
 - problème énergétique!
 - doit avoir une faible complexité / faible échange de transmission



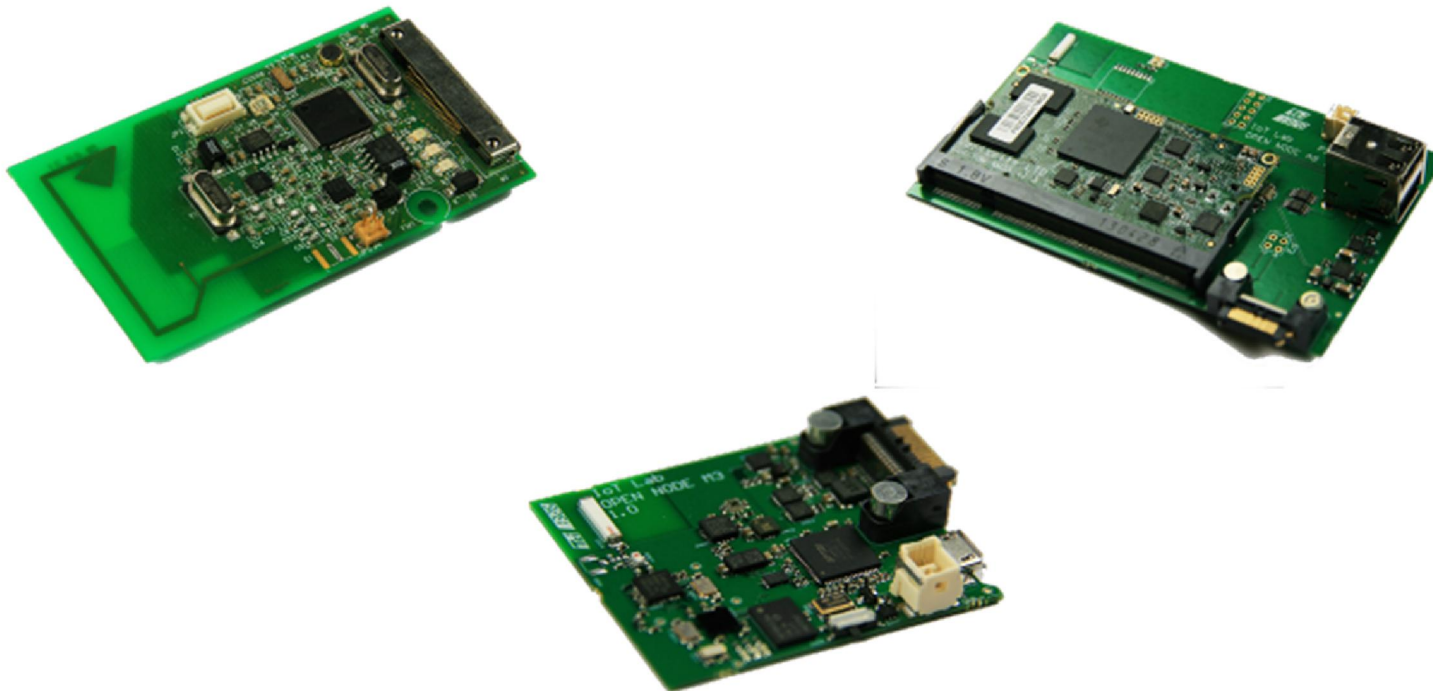
INSTITUT
Mines-Télécom

Plate-forme logicielle FIT/IoT-Lab



Un rapide résumé...

- Quelles sont les plates-formes matérielles dans FIT / Iot-Lab?





Capteurs dans IoT?

■ En fait, qu'est-ce que l'IoT?

- Appareils connectés
- Mise en réseau de tous les appareils

■ Les capteurs sont un sous-ensemble de ces appareils

- Tout appareil qui détecte un phénomène physique à enregistrer en tant que donnée = un capteur

le travail du capteur

■ Détecter

- Après tout, c'est un capteur

■ La communication

- Capable de donner ses données à un serveur / utilisateur

■ Réseau

- Transférer les données d'un autre capteur
- Gérer efficacement le réseau pour la longévité du service

Qu'est-ce qui permet la connexion de capteurs?

- Détection et communication, c'est tout bon, mais quelque chose manque?
- un indice:

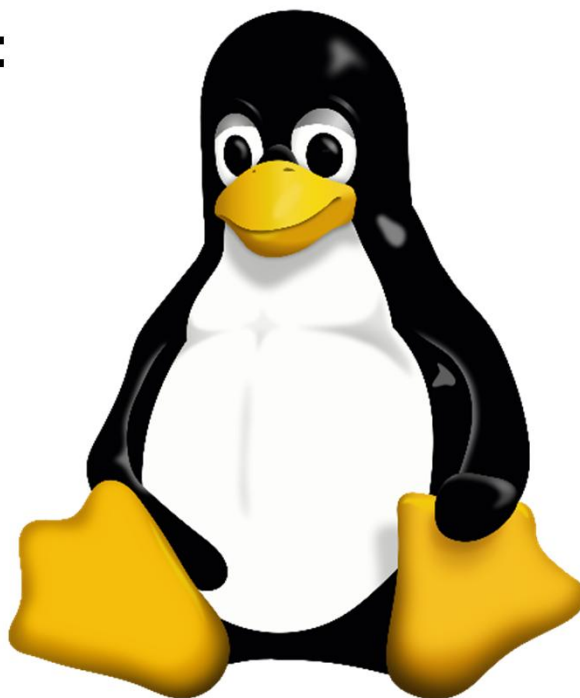




Plate-forme logicielle

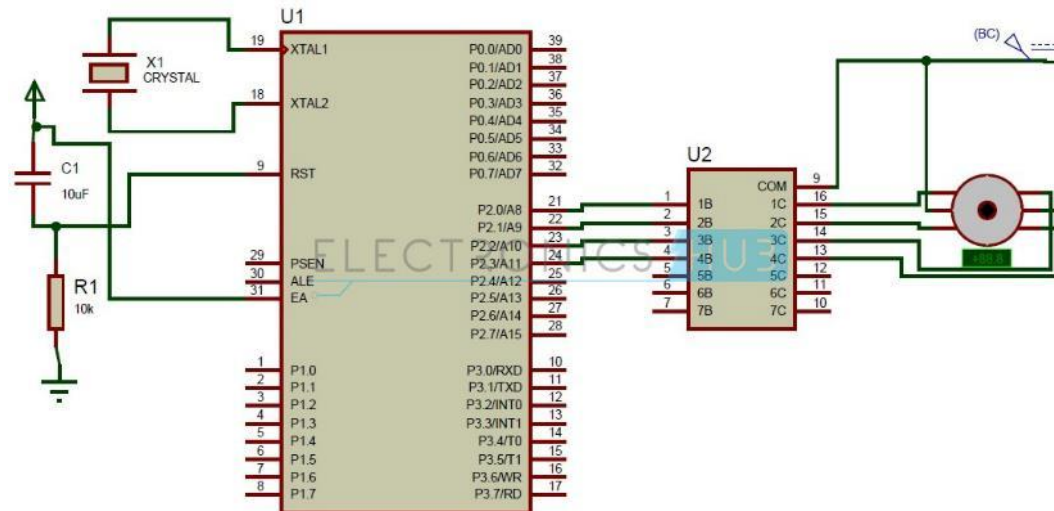
- **Nous utilisons les plates-formes logicielles pour contrôler tous les aspects d'un appareil capteur et son réseau**
 - Activation / désactivation de la détection
 - Activation / désactivation de la communication
 - Mise sous / hors tension de l'appareil
 - Commandes du logiciel

- **Etc.**

Sans plate-forme logicielle..

■ si les capteurs sont de simples firmwares?

- Stupide vs Intelligent
- Après tout, c'est ce que les OS sont pour ... pour rendre l'ordinateur plus intelligent!



Types de plates-formes logicielle

■ Il existe de nombreux types de plates-formes disponibles

- Embedded linux
- TinyOS
- Contiki
- RIOT
- Etc.

■ Mais pourquoi différentes plates-formes ???

- Meilleure performance
- Meilleure compatibilité

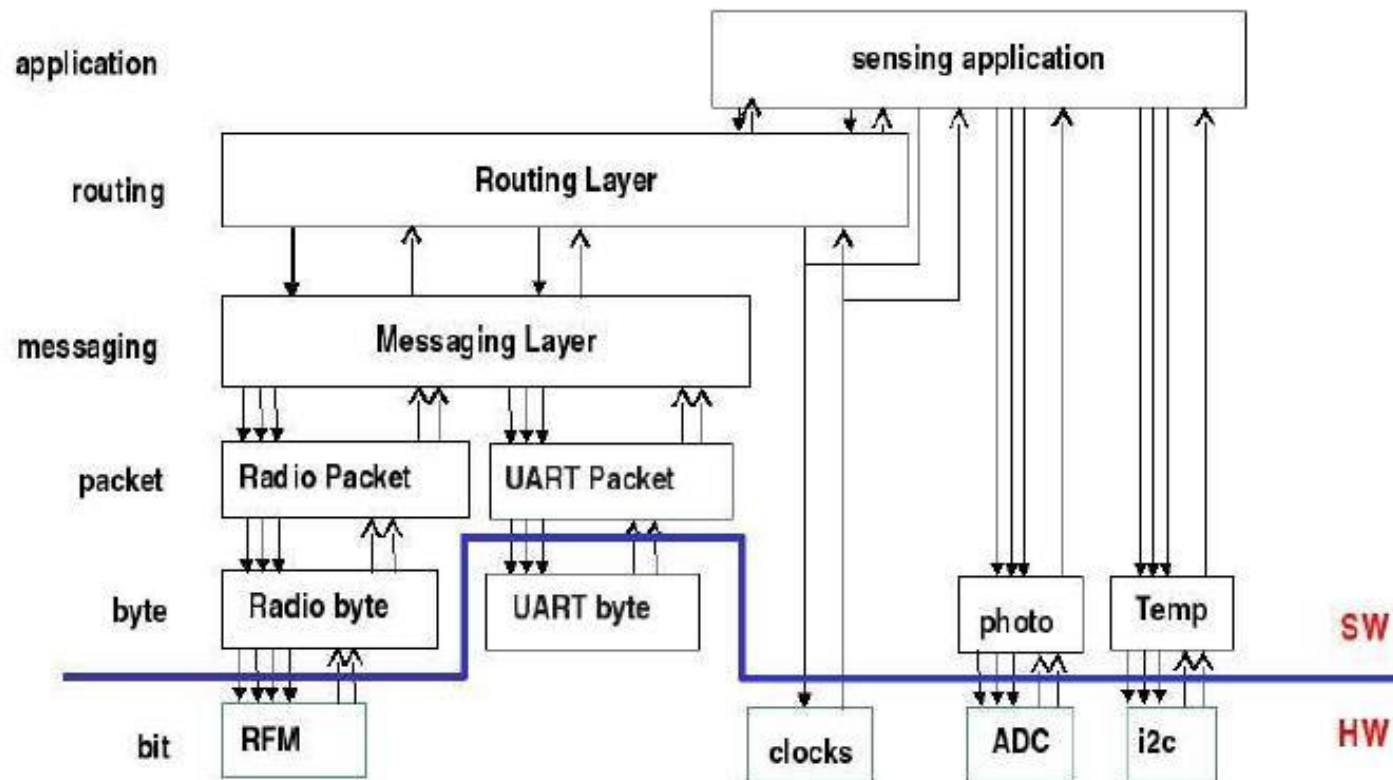
TinyOS

- Réseau de capteurs OS développé par UC Berkeley
- Très petite taille de codage
- Utilisation du langage de programmation nesC
 - Bon ou Mauvais?
- Grande communauté



TinyOS architecture

Typical Application architecture





Contiki

- **Créé par Adam Dunkels – Suède**
- **Multitâche autorisé par multithreading**
 - Prémption
 - GUI forte - Cooja
- **Plus lourd que TinyOS mais aussi plus puissant!**
 - La plus petite pile IPv6 lors du premier développement
- **Grande communauté**

Contiki

The Open Source OS for the Internet of Things



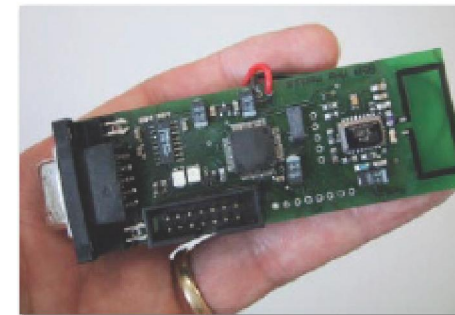
Contiki Contexte

- **Chargement dynamique des programmes (par opposition à statique)**
 - Dynamic loading
- **Exécution gérée simultanée multithread**
 - En plus d'événements entraînés
 - Événement utilisateur, Événement Thread, etc.
- **Disponible sur MSP430, AR, HC12, x86 ...**
- **Supporter le simulateur Cooja**

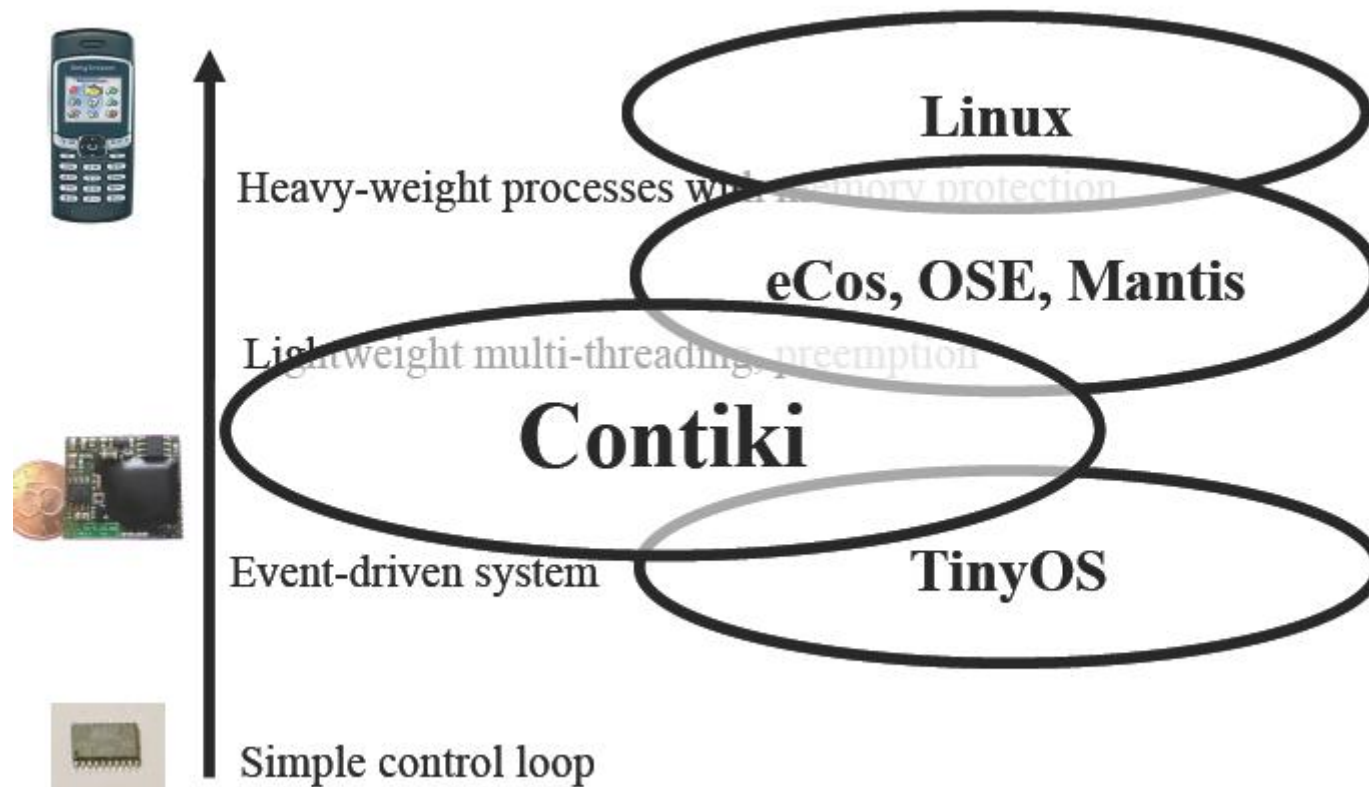
Objectifs de Contiki

■ “mote” – device

- 10-100 Ko de code FROM
- 1-10 Ko de RAM
- la communication



Système d'exploitation embarqué



vous vous souvenez??

■ Contiki dans FIT/Iot-Lab

```
apps          doc          Makefile.include  README.md
CONTRIBUTING.md examples      platform          regression-tests
core          iotlab.makefile README-BUILDING.md tools
cpu          lib          README-EXAMPLES.md
dev          LICENSE     README-IoT-LAB.md
klim@grenoble:~/iot-lab/parts/contiki$ cd examples
klim@grenoble:~/iot-lab/parts/contiki/examples$ ls
antelope      fat           nrf52dk         sky-shell-webserver
avr-rss2       ftp           openmote-cc2538 stm32nucleo-spirit1
cc2530dk      galileo      ping-ipv6        tcp-socket
cc2538-common hello-world  powertrace       telnet-server
cc2538dk      http-socket  ravenusbstick   timers
cc26xx        iotlab      rime             trickle-library
cfs-coffee    ip64-router  rime-tsch        udp-ipv6
collect       ipso-objects rssi-scanner     udp-stream
econotag-ecc-test ipv6         seeyeye          webbrowser
econotag-flash-test irc          sensniff         webbrowser-80col
eeprom-test   irc-80col    servreg-hack     webserver
email         jn516x      settings-example webserver-ipv6
er-rest-example llsec       sky              webserver-ipv6-raven
er-rest-example-raven mbxxx       sky-ip           websockets
example-shell multi-threading sky-shell        wget
extended-rf-api netperf     sky-shell-exec   zolertia
klim@grenoble:~/iot-lab/parts/contiki/examples$
```



familier?

■ Contiki dans FIT/Iot-Lab

```
cc2538-common      hello-world        powertrace          telnet-server
cc2538dk            http-socket        ravenusbstick      timers
cc26xx             iotlab             rime               trickle-library
cfs-coffee         ip64-router        rime-tsch          udp-ipv6
collect            ipso-objects       rssi-scanner       udp-stream
econotag-ecc-test  ipv6               seeyeye            webbrowser
econotag-flash-test irc                 sensniff           webbrowser-80col
eeprom-test        irc-80col          servreg-hack       webserver
email              jn516x            settings-example   webserver-ipv6
er-rest-example    llsec              sky                webserver-ipv6-raven
er-rest-example-raven mbxxx             sky-ip             websockets
example-shell      multi-threading    sky-shell          wget
extended-rf-api    netperf            sky-shell-exec     zolertia

klim@grenoble:~/iot-lab/parts/contiki/examples$ cd ipv6
klim@grenoble:~/iot-lab/parts/contiki/examples/ipv6$ ls
http-server  native-border-router  rpl-tsch          sky-websense
json-ws      rpl-border-router     rpl-udp           slip-radio
multicast    rpl-collect           simple-udp-rpl
klim@grenoble:~/iot-lab/parts/contiki/examples/ipv6$ cd rpl-udp
klim@grenoble:~/iot-lab/parts/contiki/examples/ipv6/rpl-udp$ ls
Makefile      rpl-udp.csc          rpl-udp-scale-wismote.csc
Makefile.target rpl-udp-powertrace.csc udp-client.c
project-conf.h rpl-udp-scale.csc    udp-server.c
klim@grenoble:~/iot-lab/parts/contiki/examples/ipv6/rpl-udp$ make TARGET=iotlab-m3
```

Pour les personnes qui ne sont pas familières

■ C'est bon, il y a beaucoup de tutoriels qui peuvent être facilement suivis

- Sans connaissances sur Contiki

Operating Systems

Contiki

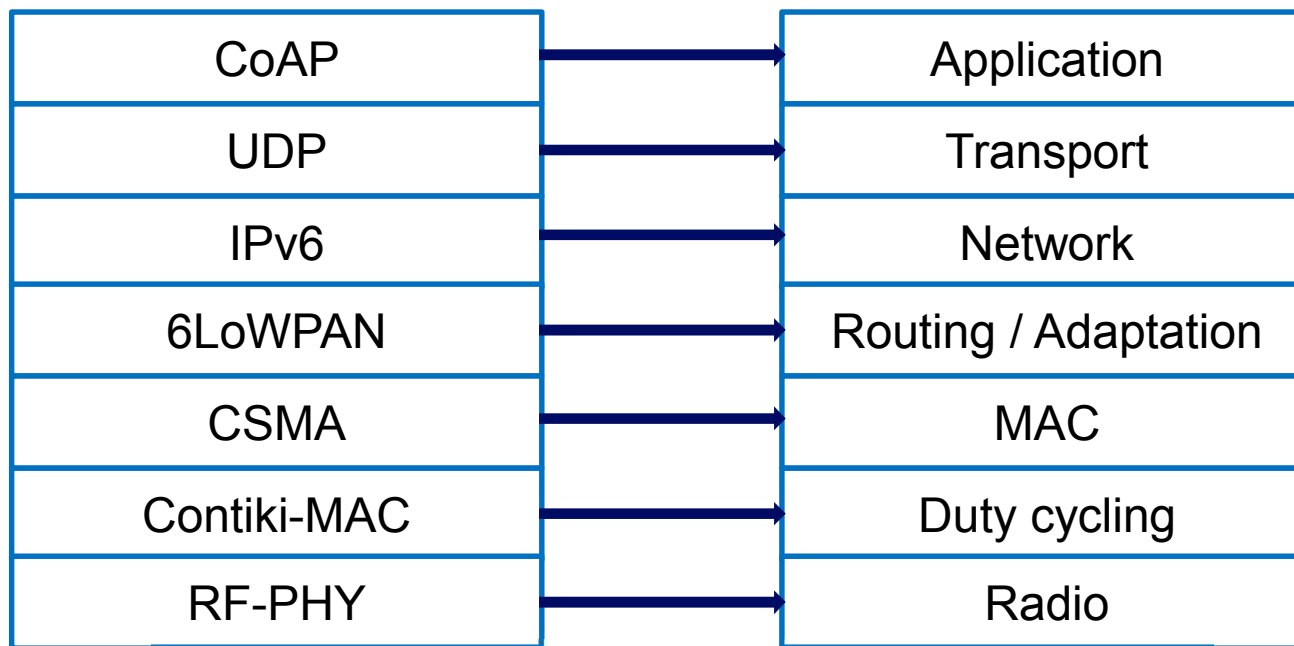


RIOT

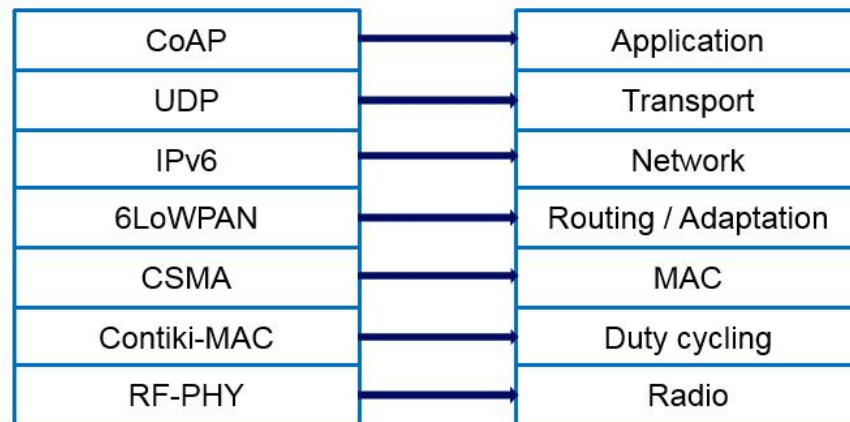


Contiki Réseau pile architecture

Notre architecture de pile



Radio / MAC – IEEE 802.15.4





Radio

■ Tous les appareils basés sur 802.15.4 PHY

- <http://www.atmel.com/images/doc8111.pdf>

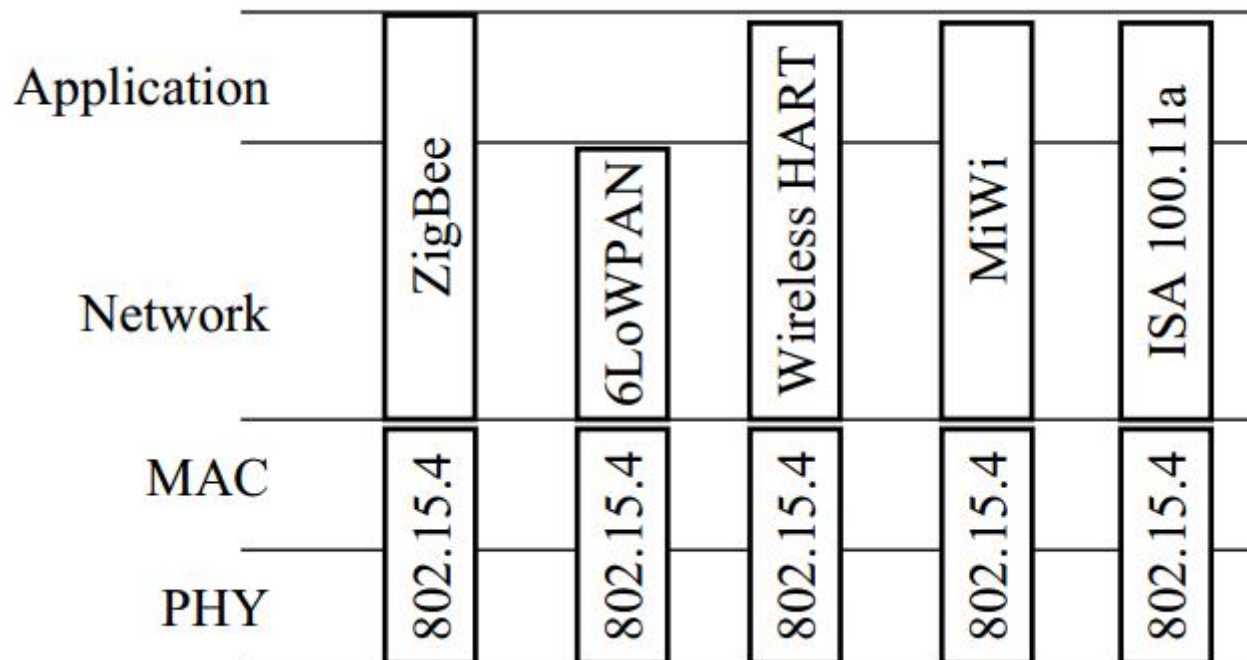
MCU	ARM Cortex M3, 32-bits, 72 Mhz, 64kB RAM – ST2M32F103REY
sensors	<ul style="list-style-type: none">• Ambient sensor light – ISL29020• Atmospheric pressure and temperature – LPS331AP• Tri-axis accelerometer/magnetometer – L3G4200D• Tri-axis gyrometer – LSM303DLHC
radio communication	802.15.4 PHY standard, 2.4 Ghz – AT86RF231

System on Module	High-performance ARM Cortex-A8 microprocessor, 600 Mhz, 256 MB – Variscite VAR-SOM-AM35 CPU
co-microcontroller	
MCU	ARM Cortex M3, 32-bits, 72 Mhz, 64kB RAM – ST2M32F103REY
sensors	<ul style="list-style-type: none">• Tri-axis accelerometer/magnetometer – L3G4200D• Tri-axis gyrometer – LSM303DLHC
radio communication	802.15.4 PHY standard, 2.4 Ghz AT86RF231

IEEE 802.15.4

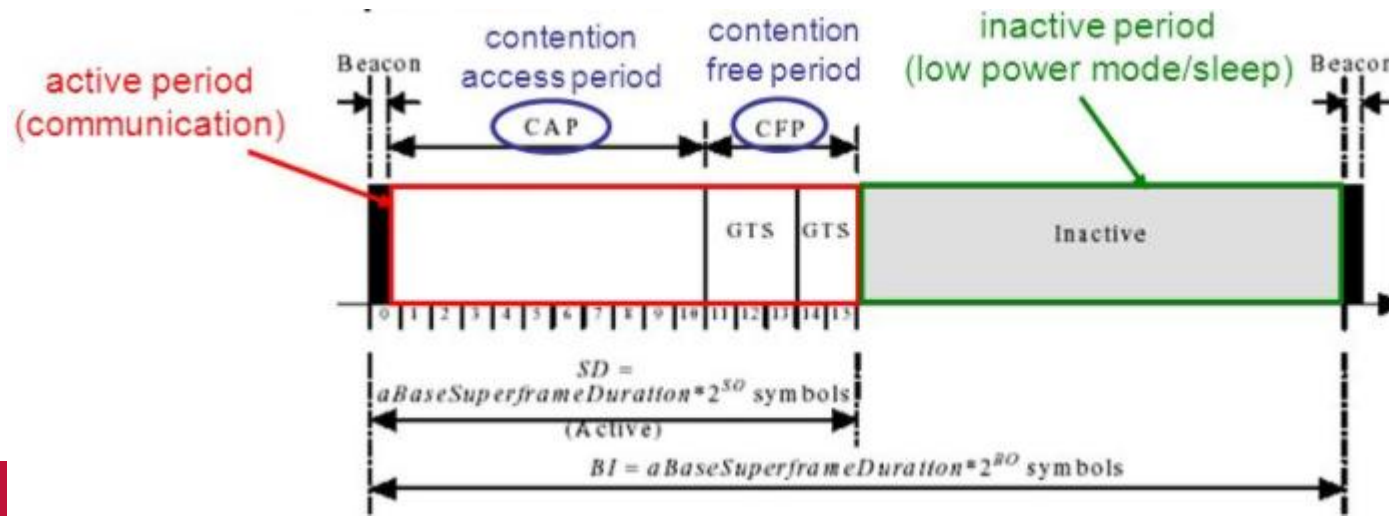
■ Une norme pour WPAN PHY/MAC (Bas débit)

- Beaucoup de solutions WPAN l'utilisent

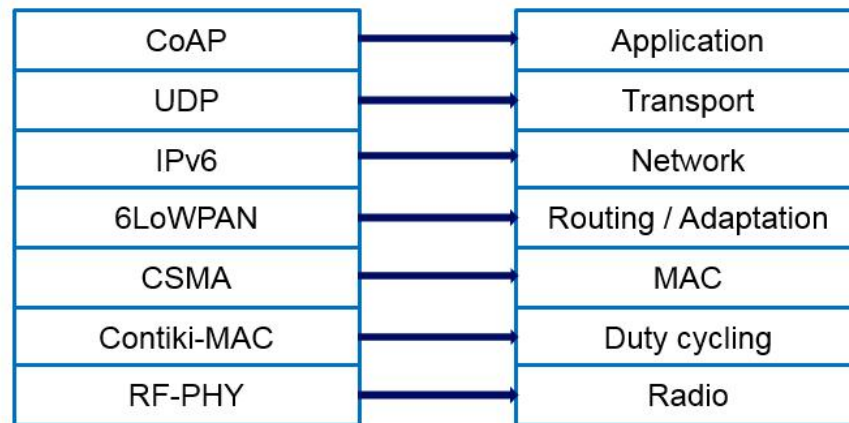


MAC - Beacon-enabled CSMA/CA (Balise)

- Le coordonnateur envoie régulièrement des balises
- Période d'accès à la contestation
 - Contention Access Period (CAP). Slotted CSMA.
- Période sans contestation
 - Contention Free Period (CFP).
 - Services de transmission garantis (GTS): pour les services en temps réel.
- Partie inactive - tous les nœuds dorment



Routing and Adaptation (6LoWPAN)

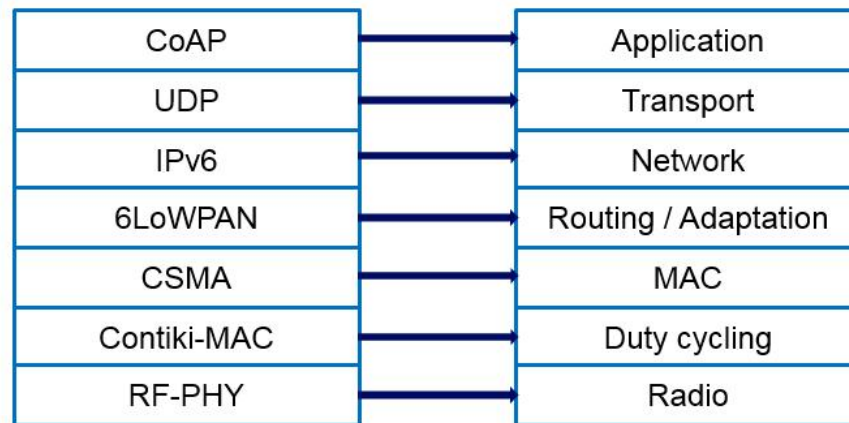




Importance de 6LoWPAN

- **Un réseau de capteurs est supposé avoir:**
 - De nombreux nœuds!
 - Ils ont besoin d'un ensemble d'adresses étendues
 - Aussi besoin de systèmes d'adressage automatique
- **Tous les réseaux doivent être connectés**
 - Connecté à Internet? = Utiliser IPv4 ou IPv6
- **Pourquoi ne pas utiliser IPv4?**
 - Il est déjà assez congestionné
 - IPv6 est énorme!

Application - CoAP





Qu'est-ce qu'une couche d'application?

- **une pile qui permet et gère le transfert de données / application**
 - Par des hôtes
 - Connexion aux serveurs qui ont les données / service



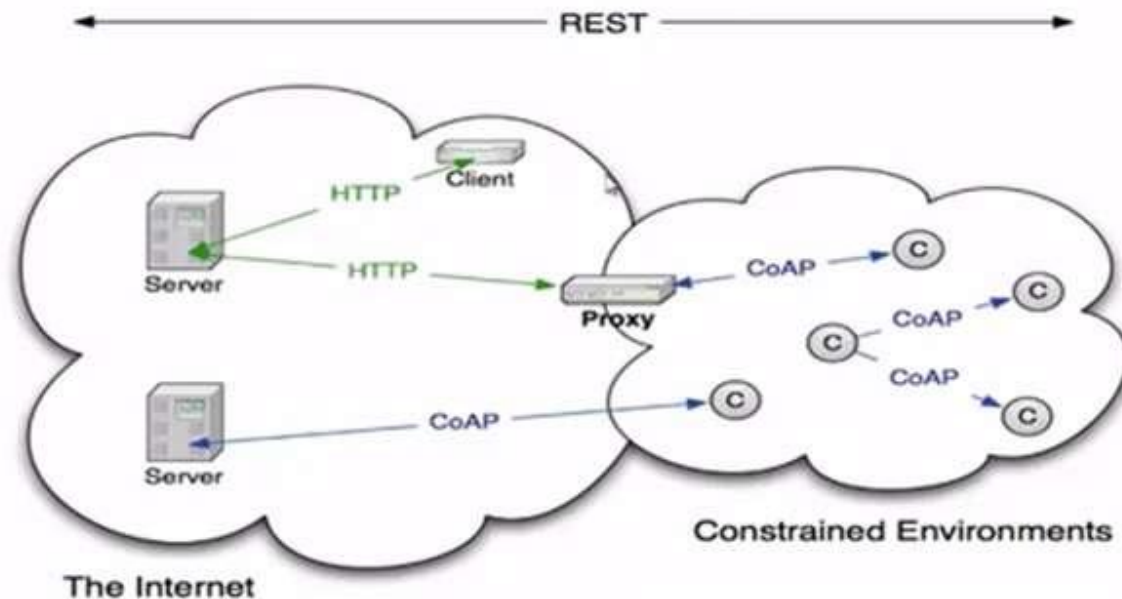
Êtes-vous confus?

- **PHY - signalisation sans fil**
- **MAC - contention pour utiliser la communication sans fil, à simple saut**
- **NET - Communication multi-hop**

- **APP - Communication hôte-serveur**
 - Nous ne nous soucions pas des couches ci-dessous

■ Constrained Application Protocol

- Protocole d'application utilisé pour les appareils contraints, comme les capteurs
- Agit comme HTTP, et aussi facilement adaptable avec HTTP





Pourquoi utiliser CoAP?

■ Protocole un-à-un

- Connexion directe entre le serveur et le client

■ Compatibilité URL

- Facilement adaptable à l'architecture Internet actuelle
- coap: //sensor-network.com/1st-sensor
- Se connecte avec votre navigateur Web

■ Poids léger!

- Simple et adapté aux applications qui ont besoin d'UDP
- Réseaux de capteurs!



CoAP sur le web

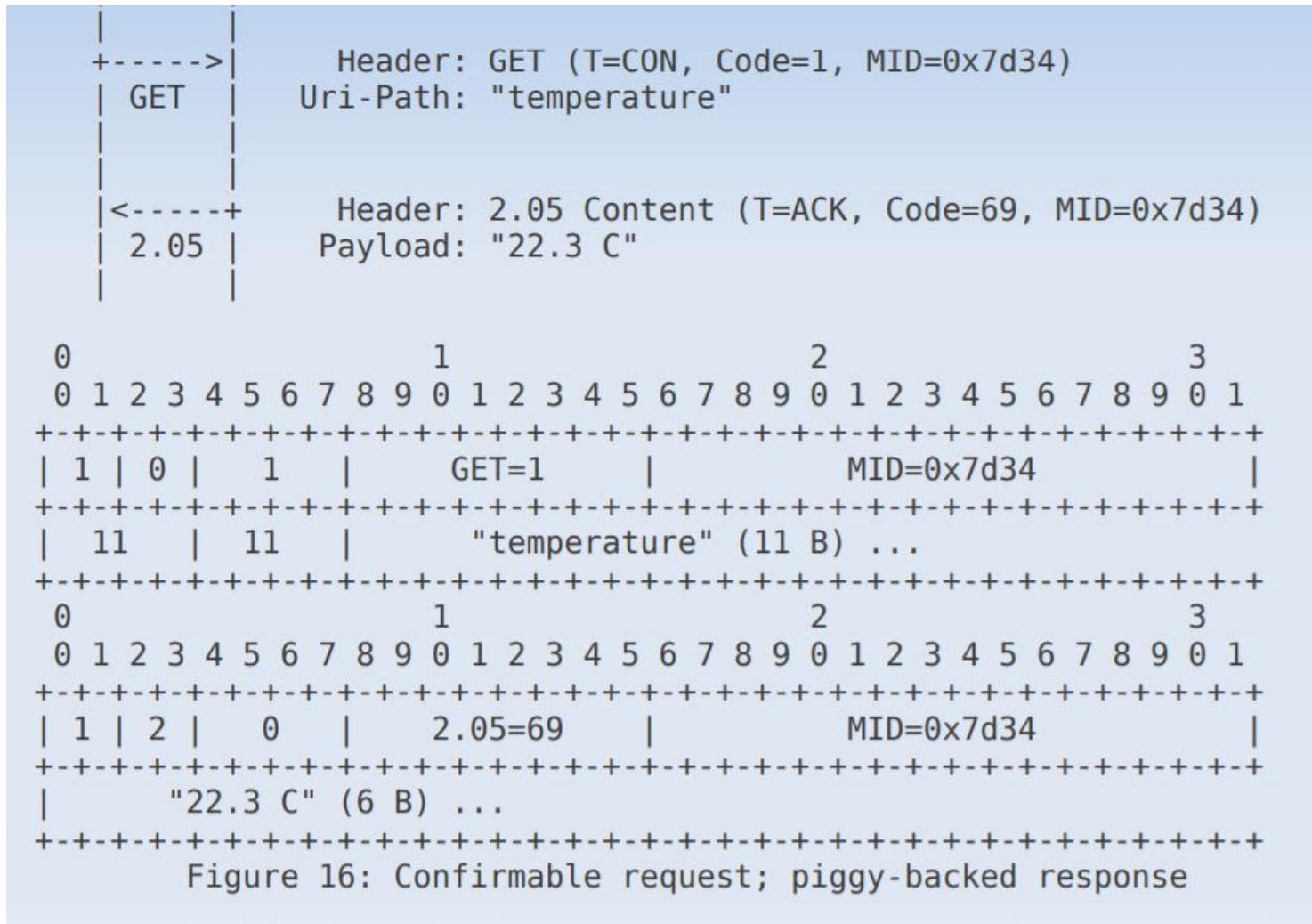
■ Sémantique compatible avec le web actuel

- GET, POST, PUT, DELETE
- Identique à HTTP

■ Fonctions dans le Web possible aussi

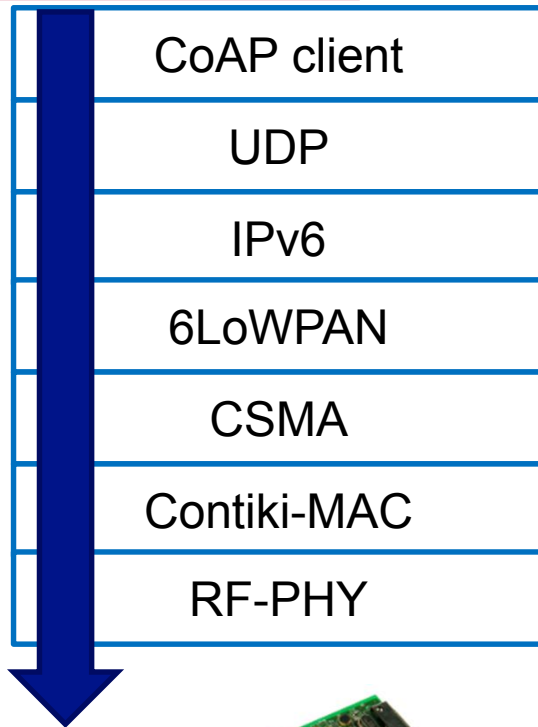
- Mise en cache
- Proxies
- Etc.

Exemple de messagerie dans CoAP

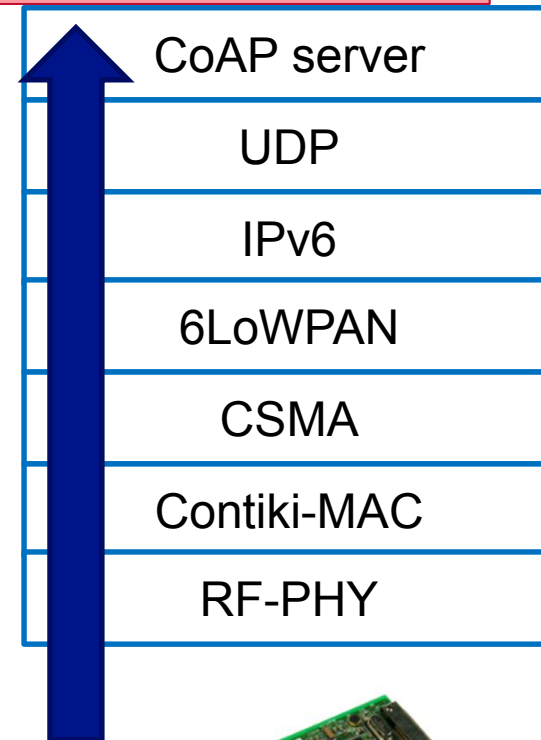


Résumons

GET /temperature



ACK Content « 12.4 »



Temp:
12.4





Notre objectif

■ Il y aura 3 TP

- TP1 et TP2, nous allons nous concentrer sur l'utilisation des tutoriels sur FIT / IoT-Lab pour nous familiariser
- TP3, nous utiliserons les protocoles d'application pour concevoir une application simple et une solution de service!

