

# Internet of Things (IoT)

Les objets connectés

Mickaël Bettinelli  
([mickael.bettinelli@univ-smb.fr](mailto:mickael.bettinelli@univ-smb.fr))



# Objectif

- Définir l'IoT
- Comprendre son cadre applicatif
  - Liens avec l'environnement physique
  - Communications réseau
- Comprendre le fonctionnement d'un IoT et sa conception
- Comprendre ses enjeux
  - Les possibilités qu'il offre
  - Ses problématiques

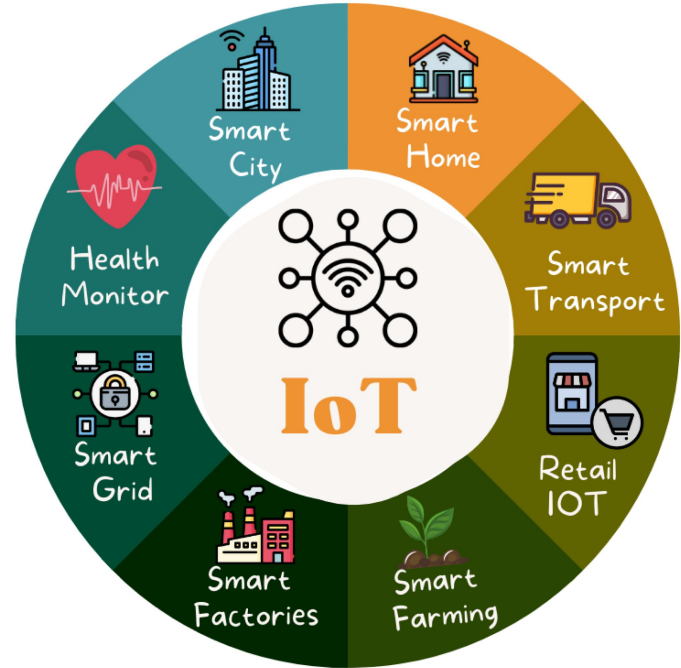


# Qu'est ce qu'un IoT ?

Désigne la connexion d'objets physiques à internet.

*Exemple.*

Ampoules, appareils médicaux,  
téléphone portable, *etc.*





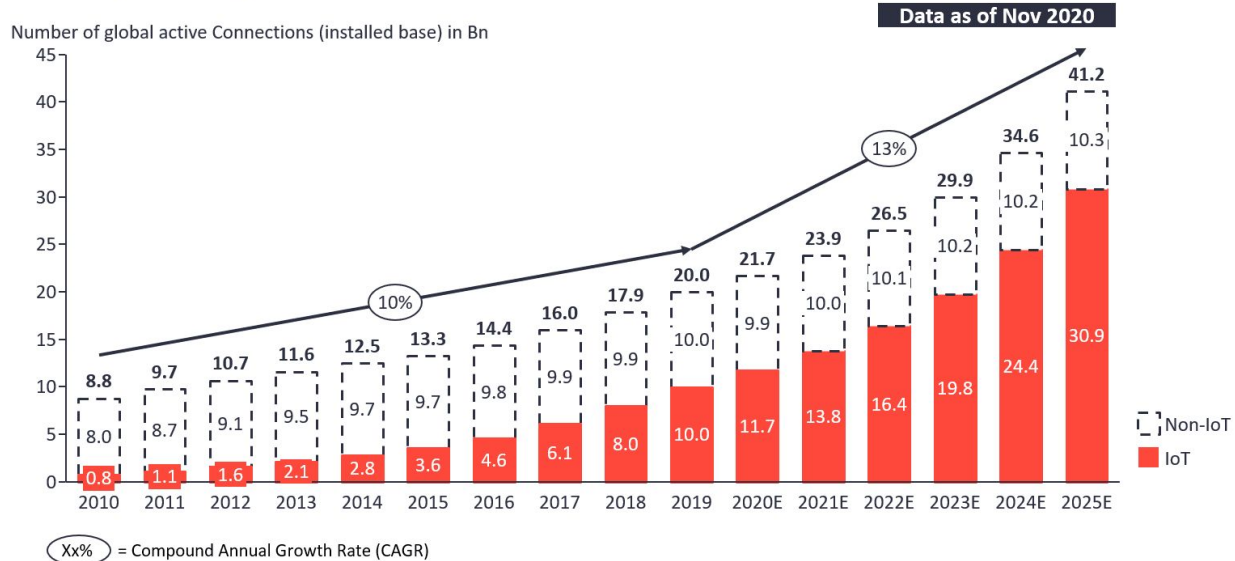
# Expansion des IoT



## Total number of device connections (incl. Non-IoT)

20.0Bn in 2019– expected to grow 13% to 41.2Bn in 2025

Insights that empower you to understand IoT markets



Note: Non-IoT includes all mobile phones, tablets, PCs, laptops, and fixed line phones. IoT includes all consumer and B2B devices connected – see IoT break-down for further details

Source(s): IoT Analytics - Cellular IoT & LPWA Connectivity Market Tracker 2010-25



# Fonctionnement d'un IoT

Un objet communicant est fait comme une interface entre le monde physique et le monde virtuel.





# Architecture matérielle

Choisir le type de matériel :

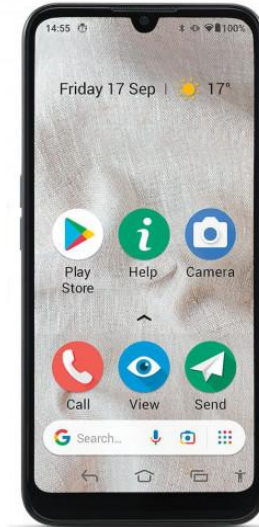
- Mini-ordinateur
- Une carte de développement existante (raspberry pi, arduino, *etc.*)
- Fabriquer le sien à partir de zéro



# Architecture matérielle

Choisir le type de matériel :

- Mini-ordinateur
- Une carte de développement existante (raspberry pi, arduino, etc.)
- Fabriquer le sien à partir de zéro





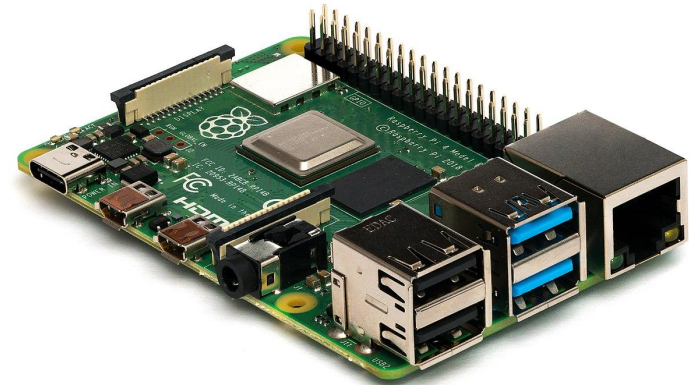
# Architecture matérielle

Choisir le type de matériel :

- Mini-ordinateur
- Une carte de développement existante (raspberry pi, arduino, etc.)
- Fabriquer le sien à partir de zéro



Carte de développement STM32



Raspberry Pi

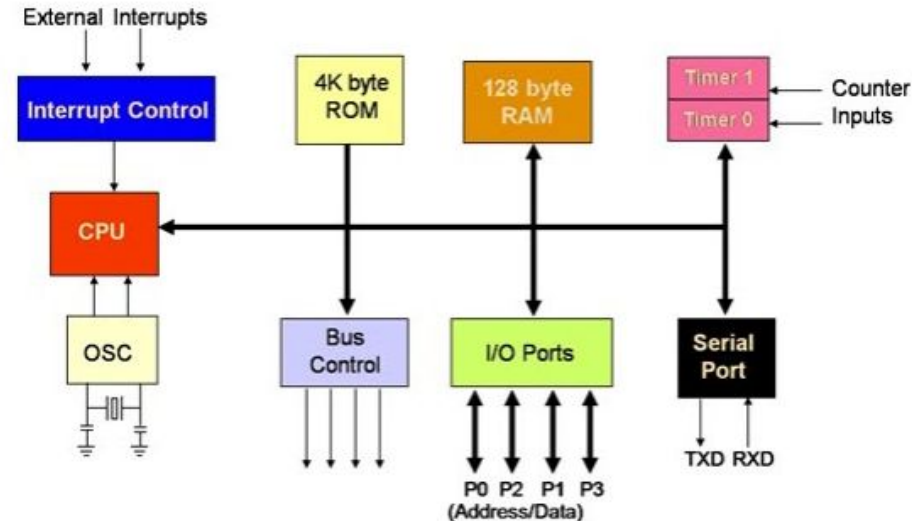




# Architecture matérielle

Choisir le type de matériel :

- Une carte de développement existante (raspberry pi, arduino, etc.)
- Mini-ordinateur
- Fabriquer le sien à partir de zéro

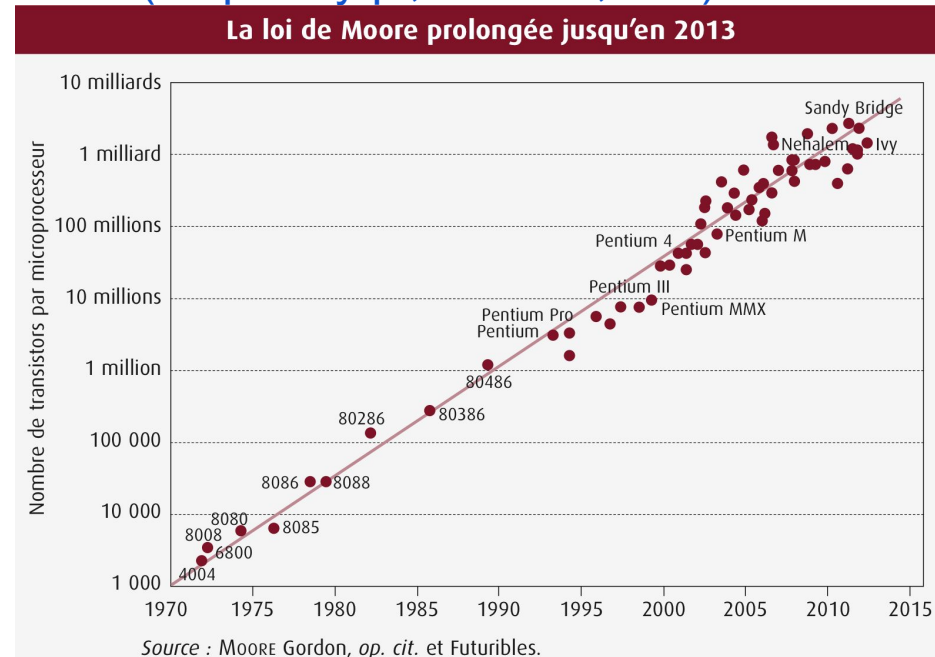




# Architecture matérielle

Choisir le type de matériel :

- Une carte de développement existante (raspberry pi, arduino, etc.)
- Mini-ordinateur
- Fabriquer le sien à partir de zéro





# Infrastructure

## INDUSTRIAL IoT DATA PROCESSING LAYER STACK

### CLOUD LAYER

Big Data Processing  
Business Logic  
Data Warehousing

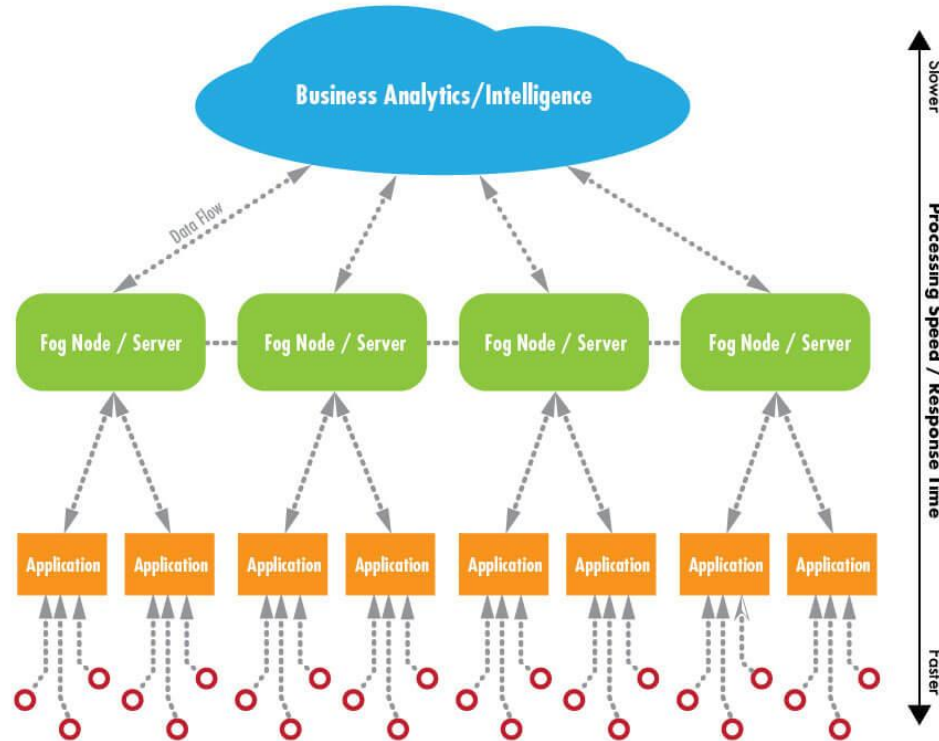
### FOG LAYER

Local Network  
Data Analysis & Reduction  
Control Response  
Virtualization/Standardization

### EDGE LAYER

Large Volume Real-time Data Processing  
At Source/On Premises Data Visualization  
Industrial PCs  
Embedded Systems  
Gateways  
Micro Data Storage

Sensors & Controllers (data origination)





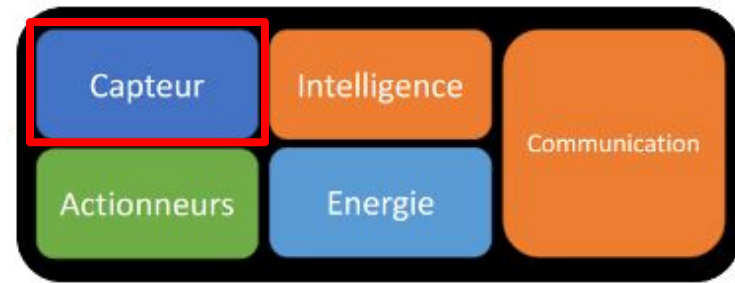
# Capteurs

De nombreux capteurs existants.

A titre d'exemple, nous allons utiliser :

- Capteur d'humidité
- Capteur de CO<sub>2</sub>
- Détecteur de mouvement

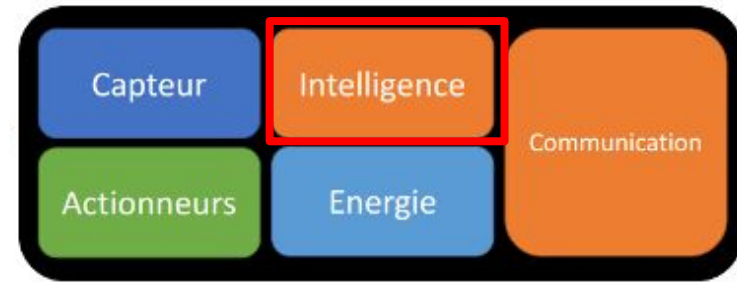
(durant la SAé !)



Détecteur de GPL, i-butane, propane, méthane, alcool, hydrogène et fumée



# Intelligence



Comportement embarqué :

- L'loT est autonome dans sa prise de décision

Avantages :

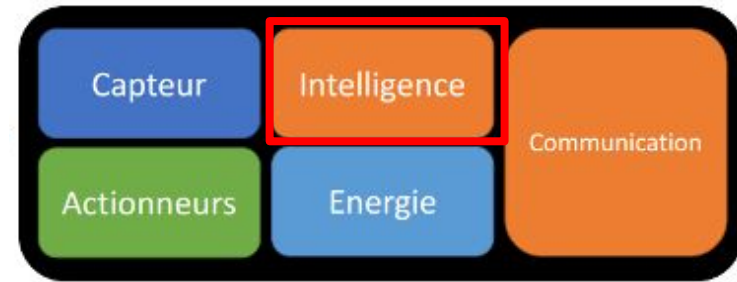
- Prise de décision rapide
- Economie d'énergie (moins de transmission de données)

Inconvénients :

- L'loT a une faible puissance de calcul



# Intelligence



Comportement déporté :

- Un serveur avec plus de capacité de calcul prend les décisions pour l'IoT

Avantages :

- Plus de puissance de calcul sur les serveurs distants

Inconvénients :

- Prise de décision plus lente (latence réseau)



# Communication

*Courte portée*



	Zigbee	Z-wave	Bluetooth	Wi-Fi
Vitesse de transfert (max)	250Kb/s	100Kb/s	1 Mb/s	100Mb-10Gb/s
Portée	10-100m	Max 100m Plus en réseau	10-100m	300m
Autonomie en énergie	Années	Années	Jours	Heures



# Communication

*Longue portée*



	5G	LoRa
Vitesse de transfert (max)	20Gb/s	27Kb/s
Portée	5km	16km
Autonomie en énergie	Heures	Années





# Concevoir le bon IoT

Définir le besoin :

- Traitement du signal
- Communiquer des données
- Contrôle de systèmes
- *etc.*

Et en déterminer :

- La puissance de calcul
- La capacité de stockage
- La capacité à communiquer
- Les capteurs
- Les actionneurs
- L'OS à utiliser
- Les outils logiciels



# Tester le système

Avant de construire votre IoT :

1. Définissez des indicateurs que vous voulez suivre (*aka* **KPI** → key performance indicator)
2. Simulez votre système virtuellement (ex. SimGrid)
3. Vérifiez vos indicateurs
4. Produisez et faites interagir un IoT avec le reste de votre système virtuel (ex. MASH)
5. Vérifiez vos indicateurs
6. Si tout est bon, vous pouvez produire vos IoT !



# Problématiques actuelles

Du point de vue données et intelligence :

- La confidentialité des données produites
- Migrer l'intelligence du cloud au edge
- La sécurité des IoT



## A retenir

- Ce que sont le cloud, le fog et l'edge computing
- Ce qu'est un IoT
- Comment concevoir un IoT (communication, placement du comportement, *etc.*)
- Comment concevoir un système d'IoT