

# L'Internet des Objets

1. Définition
2. Concepts
3. Web 3.0
4. Problématiques spécifiques
5. Domaines d'application
6. Communications avec l'IdO

# Définition

L'IdO est une notion complexe à cerner :

- pluralité des technologies qui la forment ;
- besoins multiples → plusieurs domaines d'application ;
- technologie nouvelle ;
- pas de définition partagée de l'IdO ;
- considéré comme le Web 3.0.

Sémantique possible en partant de la racine étymologique

- Internet :

réseau mondial composé de plusieurs réseaux identifiables (adresses IP publiques) et joignables grâce à un protocole de communication standard (TCP/IP)

- Objet :

Chose qui ne peut être précisément identifiée

Définition probable pour l'Internet des Objets :

« réseau mondial de choses communiquant grâce à un protocole standard »

# Concepts

## Machine 2 Machine (M2M)

Désigne les moyens de communication permettant l'échange de données entre machines (sans intervention humaine)

Rôle important dans la gestion:

- des entrepôts (stock)
- chaîne logistique (traçabilité des colis)
- robotique
- ...

C'est la base-même du concept d'IdO

Un système M2M comporte :

- un ensemble de capteurs (température, présence, RFID, ...)
- reliés au réseau (fil ou sans-fil)
- par une unité de traitement (microcontrôleurs)

Il n'existe pas de plateforme d'appareils connectés standard pour la communication M2M (chacun son protocole / système)

Interconnexion souvent lourde → recours à des API !



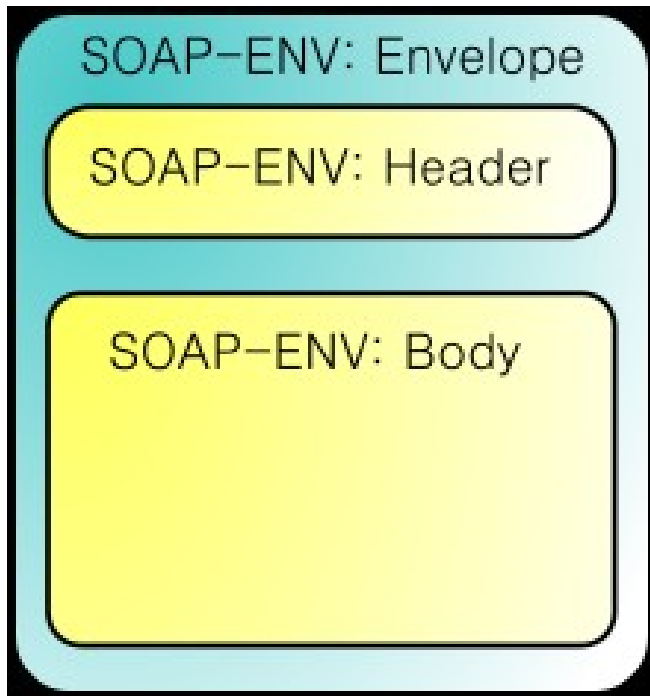
Une API ou Application Programming Interface, est :

- un ensemble de code (classes, méthodes ou fonctions) servant de façade ;
- cette façade permet à des programmes *consommateurs* de se servir des fonctionnalités du programme *fournisseur* ;
- elle est généralement offerte par une bibliothèque logicielle ou un service web ;
- l'utilisation de la façade est détaillée dans un document qui incarne la dénomination d'API ;
- l'utilisation d'API permet de construire de nouveaux logiciels en utilisant des briques de fonctionnalités fournies par d'autres logiciels ;
- l'avantage est de masquer la réalité du code au programme consommateur.

Un service web (Web Service) est :

- une interface informatique de la famille des technologies web ;
- il permet la communication et l'échange de données entre applications hétérogènes ;
- Il expose les fonctionnalités d'un programme sur le web au travers de deux normes : **SOAP** (Simple Object Access Protocol) et **Rest** (Representational state transfer) ;
- SOAP utilise XML (Extensible Markup Language) pour faire transiter ces données alors que Rest préfère JSON (JavaScript Object Notation).

### SOAP redéfinit un format de message (proche de HTTP)



```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:tem="http://tempuri.org/"
  <soapenv:Header/>
  <soapenv:Body>
    <tem:RechercherCandidat>
      <!--Optional:-->
      <tem:nomUser>test</tem:nomUser>
      <!--Optional:-->
      <tem:mdpUser>test</tem:mdpUser>
      <!--Optional:-->
      <tem:NomNaissance>test</tem:NomNaissance>
      <!--Optional:-->
      <tem:Prenom>test</tem:Prenom>
      <!--Optional:-->
      <tem:DateNaissance>01/01/2000</tem:DateNaissance>
      <!--Optional:-->
      <tem:DepartementNaissance>1</tem:DepartementNaissance>
    </tem:RechercherCandidat>
  </soapenv:Body>
</soapenv:Envelope>
```

Rest utilise la sémantique de HTTP pour ses messages :

- Create → POST
- Read → GET
- Update → PUT
- Delete → DELETE

Les codes sont utilisés pour renseigner l'état de la réponse :

- 200 → ok, 201 → objet créé, 204 → pas de contenu, ...
- 400 → mauvaise requête (paramètre manquant), 403 → non autorisé, ...
- ...

Rest utilise JSON pour faire transiter les données :



```
▼ 0:  
  name:      "/dev/sda1"  
  label:     "MUSIC"  
  size:      "7.2 GiB"  
  fstype:    "vfat"  
  UUID:      "84B9-A853"  
  isEnabled: false  
  
▼ 1:  
  name:      "/dev/sdb1"  
  label:     "MUSIC 2019"  
  size:      "7.2 GiB"  
  fstype:    "vfat"  
  UUID:      "74D1-A16F"  
  isEnabled: false
```

```
[  
  {"name":"/dev/sda1","label":"MUSIC","size":"7.2 GiB","fstype":"vfat","UUID":"84B9-A853","isEnabled":false}  
  ,  
  {"name":"/dev/sdb1","label":"MUSIC 2019","size":"7.2 GiB","fstype":"vfat","UUID":"74D1-A16F","isEnabled":false}  
]
```

## L'utilisation du service web est détaillée dans une API :

### Device [\[modifier\]](#)

L'objet périphérique est composé comme suit : {"name":"VdevVsd1","label":"MUSIC","size":"7.2 GiB","UUID":"84B9-A853","fstype":"vfat"}

URL	Corps	Verbe	Action	Code retour	Implémentée
/api/device	• vide	GET	Retourne un tableau avec tous les périphériques	• 200	
/api/device/list/\$\$UUID\$\$ • \$\$UUID\$\$ : un UUID de périphérique • paramètre(s): • path : un chemin sur le périphérique • extension : un filtre par extension de fichier (optionnel)	• vide	GET	Retourne un tableau avec la liste récursive des fichiers	• 200 : si l'UUID existe • 400 : si l'UUID n'existe pas	
/api/device/explore/\$\$UUID\$\$ • \$\$UUID\$\$ : un UUID de périphérique • paramètre(s): • path : un chemin sur le périphérique • extension : un filtre par extension de fichier (optionnel)	• vide	GET	Retourne un tableau avec la liste des fichiers et dossiers: { "directories": ["Pop", "Rock"], "files": ["Gala - Freed From Desire.mp3", "13)Jacques Your Body (Make Me Sweat)(Jcf Pitch Edit).mp3"] }	• 200 : si l'UUID existe • 400 : si l'UUID n'existe pas	
/api/device/content/\$\$UUID\$\$ • \$\$UUID\$\$ : un UUID de périphérique • paramètre(s): • path : un chemin sur le périphérique • download : si présent, le navigateur téléchargera le fichier ('Content-disposition: attachment;' au lieu de 'Content-disposition: inline;')	• vide	GET	Retourne le contenu du fichier avec le type mime en entête	• 200 : si le fichier existe • 400 : si le fichier n'existe pas	

# Web 3.0

Buzzword, le terme désigne le web qui suit le Web 2.0 et chacun à sa vision propre du futur d'Internet !

Les différentes phases du Web :

- Web 1.0 → liaison de pages web entre elles par des hyperliens (1990) ;
- Web 2.0 → web participatif ou social généralisé par les blogs, réseaux sociaux et la technologie wiki (2000) ;
- Web<sup>2</sup> (square) → Web exponentiel, les systèmes génèrent des données qui sont traitées et rendues disponibles aux internautes (implied metadata) (2006) ;
- Web 3.0 → l'Internet des objets (2008-10).

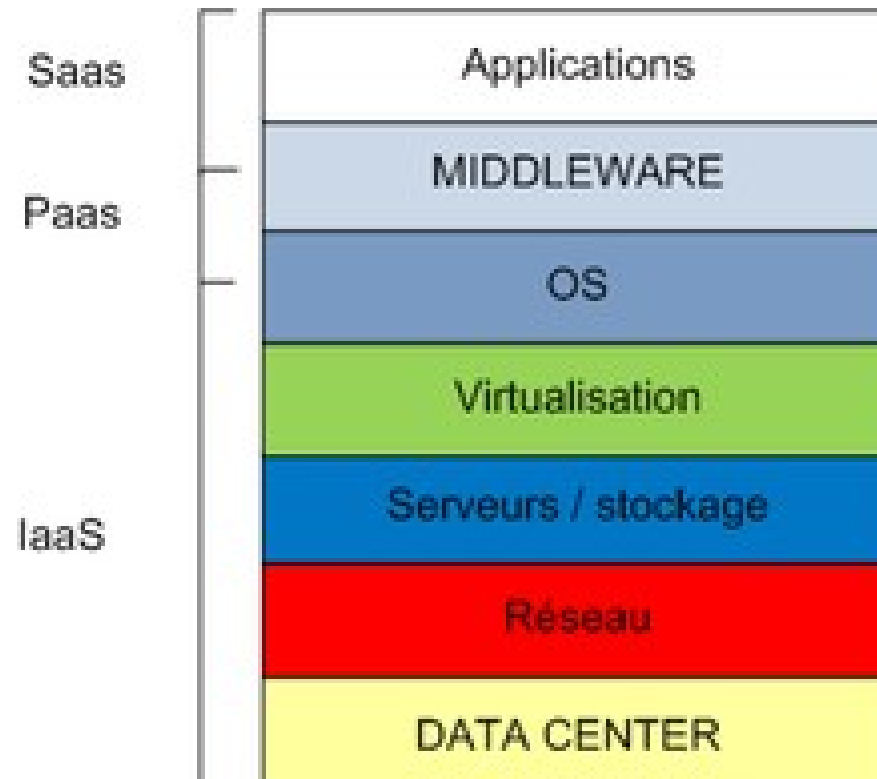


Les mœurs changent, les logiciels ne sont plus achetés mais loués en fonction du besoin (**Software as a Service**) :

- disponibles sur Internet ;
- ne nécessitent qu'un navigateur pour les utiliser (pas de licence) ;
- besoin matériel faible car toute la puissance est déportée côté serveur ;
- délocalisation des serveurs : accès nomade simplifié ;
- Total Cost of Ownership (TCO) grandement réduit (coût d'acquisition faible) ;
- passage des dépenses en OPEX (Operational Expenditure) plutôt qu'en CAPEX (Capital Expenditure = amortissement sur le moyen / long terme).

Le web 3.0 est rendu possible grâce au :

- Software as a Service (SaaS)
- Service Oriented Architecture (SOA) → web service, Json, etc...



Attention quand même :

- les données de l'entreprise ne sont plus en local (hébergées chez le prestataire du SaaS) ;
- le niveau de confidentialité des données dépend de la législation du pays de l'hébergeur ;
- interopérabilité inter-application complexe : émergence des PaaS (Platform as a Service) fournissant des API pour faciliter le dialogue inter-application ;
- délocalisation des serveurs : souci de sécurité des informations lors du départ / licenciement d'un employé ;

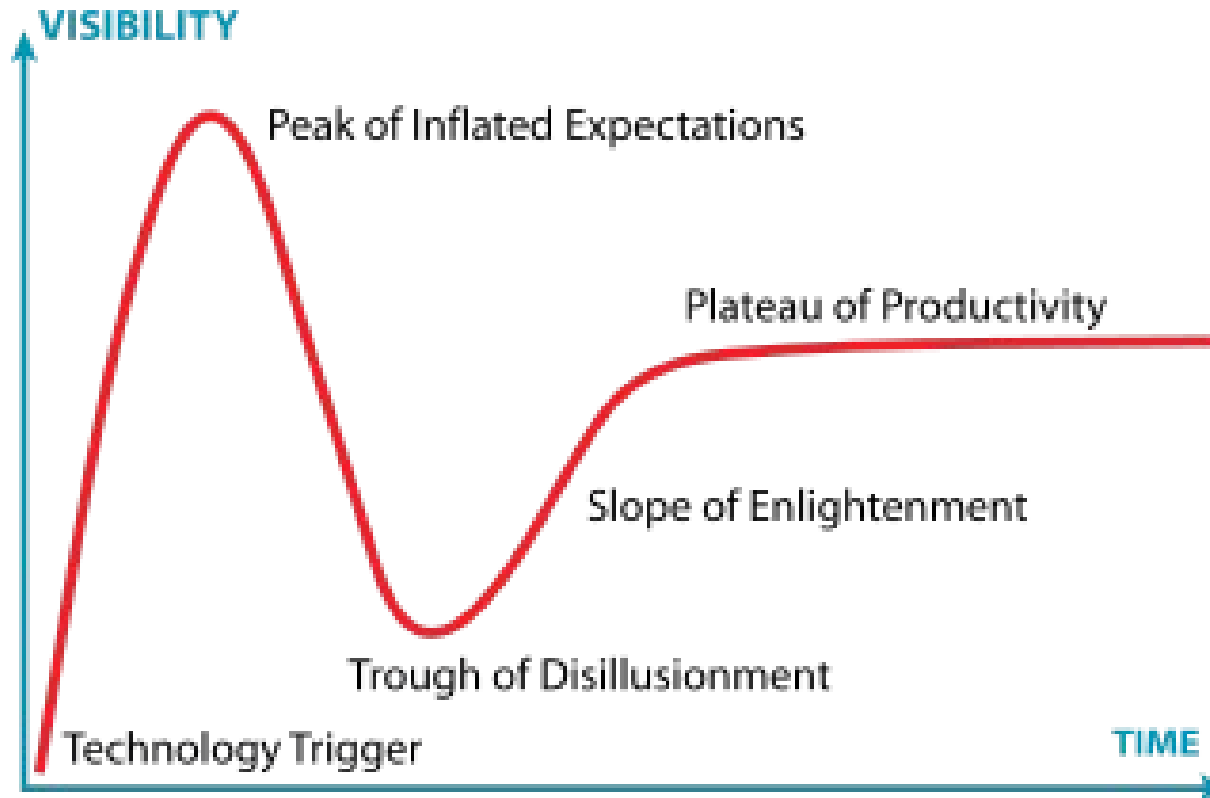
Le Web 3.0, c'est « l'ordinateur qui génère de la nouvelle information et non plus les humains »

Conrad Wolfram

Une application Web 3.0 doit:

- ne plus être uniquement un site web *xHTML* mais une solution **SaaS** ;
- être connectée à une base de données relationnelle (MySQL, Oracle...) ou fichier (XML, JSON...);
- être mobile, indépendante de tout support (taille d'écran) ;
- universelle, indépendante du système d'exploitation / matériel ;
- accessible, en conformité avec le W3C.

Comme toute nouvelle technologie, le web 3.0 suit le cycle de « Hype » défini par Gartner :



Le cycle possède 5 étapes :

- le lancement : arrivés sur le marché de la technologie, les produits ressemblent plus à des prototypes et ne sont pas utilisables simplement ;
- attentes fortes (exagérées) : emballement médiatique autour du produit qui conduit à la création de startups ;
- gouffre de désillusion : les produits ne répondent pas aux attentes exagérées des consommateurs, ce qui conduit à un krach boursier (anti-hype) ;
- pente de l'illumination : les entreprises les plus solides persistent et les consommateurs comprennent les avantages de la technologie ;
- plateau de productivité : la technologie est rodée, l'étendue des applications devient plus vaste.

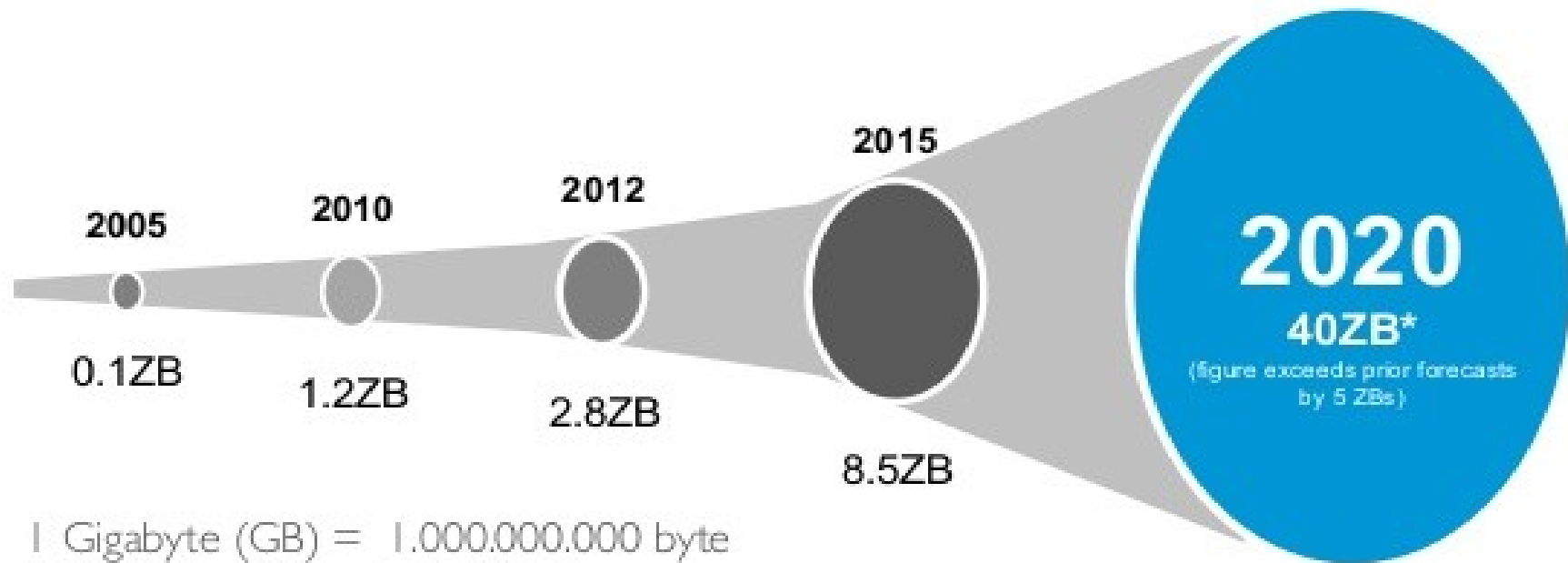
Le cycle possède des limitations :

- ce n'est pas un cycle car suivi une seule fois ;
- cette représentation est factuelle et ne permet pas d'être acteur de la réussite ;
- les termes utilisés pour les phases sont plus marketing qu'objectifs.

# Problématiques spécifiques



### Le déluge de données !

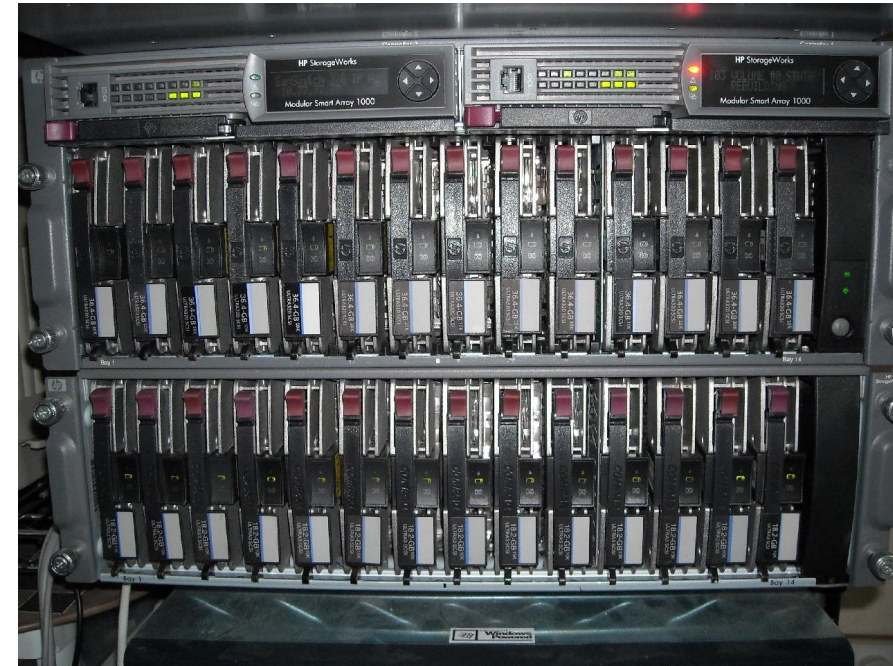


- | Gigabyte (GB) = 1.000.000.000 byte
- | Terabyte (TB) = 1.000 (GB)
- | Petabyte (PB) = 1.000.000 (GB)
- | Exabyte (EB) = 1.000.000.000 (GB)
- | Zettabyte (ZB) = 1.000.000.000.000 (GB)

Source: <http://sys.orexhosting.com/wp-content/uploads/lor-data-info-graphic.jpg>

L'IdO est envisageable grâce au big data et aux bases de données NoSQL / temps réel :

- données trop volumineuses ;
- difficile à travailler (algorithmes) ;
- données redondantes ;
- dé-duplication complexe ;
- sans relation entre elles (déstructuration).



Les données doivent être traitées au plus proche de la source

=> **Edge Computing**

- réseau maillé de micro-datacenters qui traitent / stockent les données localement ;
- déporter la puissance de calcul du datacenter vers la périphérie du réseau ;
- un calcul au plus proche de la production (transit plus faible) ;
- agrégation des données et traitements, si besoin, envoi vers le Datacenter ;

Va décoller quand la 5G arrivera → modification des antennes pour y intégrer ces micro-datacenters ;

Attention quand même :

- IdO plus vulnérables que les machines dans les datacenters ;
- il faut chiffrer les données, contrôler les accès voire utiliser des canaux de communications chiffrés (SSL / IPSec)

Avec l'IdO naît un besoin en énergie.

Pour les datacenters :

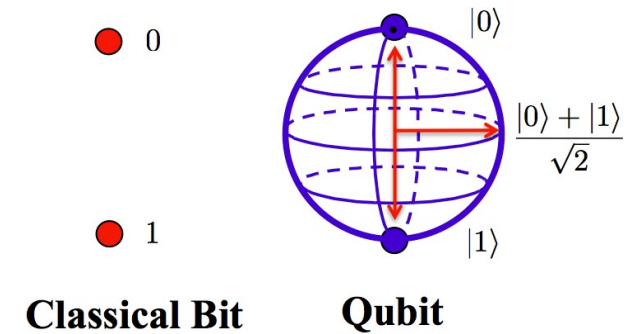
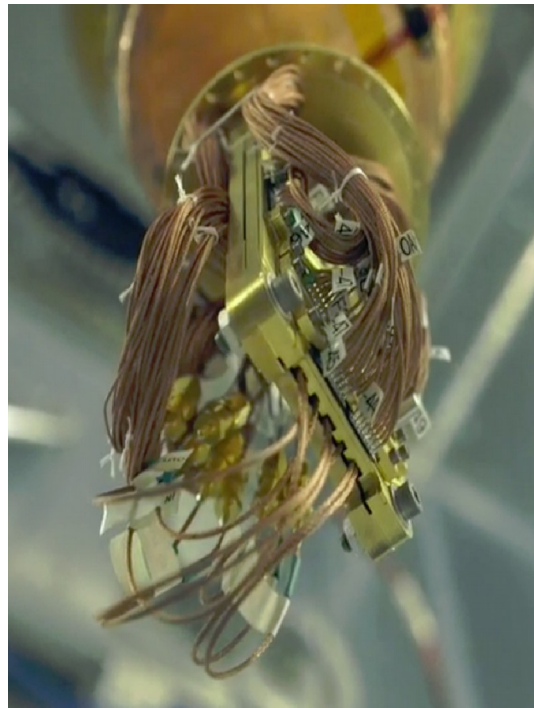
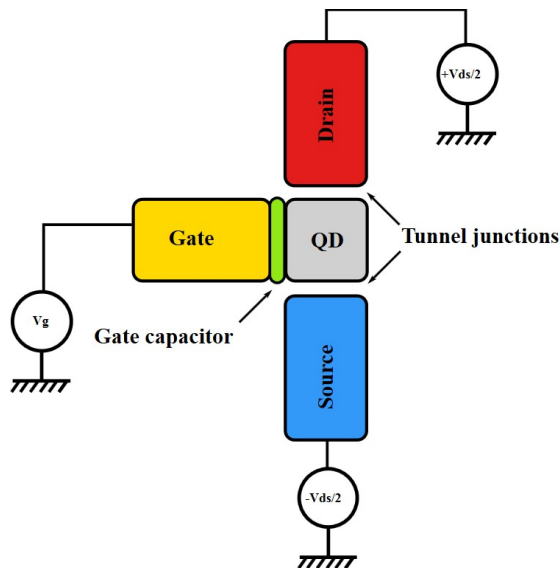
- la consommation maximale est quasiment atteinte ;
- difficile d'ajouter des machines aux infrastructures existantes ;
- remplacement des anciennes machines (temps).

Pour les objets connectés :

- source transportable (LiPo, Imprimée en Polytéréphtalate d'éthylène)
- autonome (générateur et convertisseur d'énergie) ;
- composants moins énergivores. (finesse de gravure)

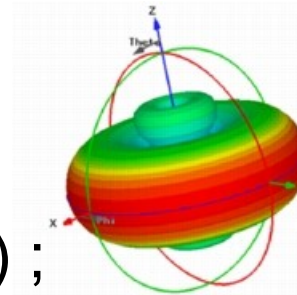
Côté datacenters, on a encore besoin de puissance pour donner du sens aux informations :

- MOSFET → SET ;
- ordinateur quantique ;

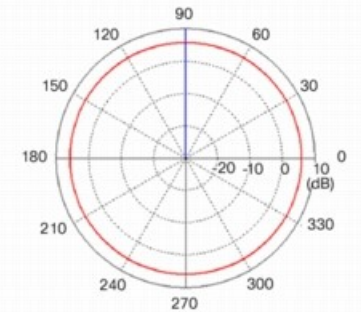


Ces objets ont besoin d'être interconnectés entre eux :

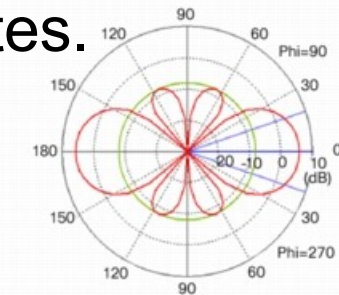
- antenne multi-fréquence ;
- taille optimisée (gravure sur puce) ;
- Phase Local Loop (PLL) plus performantes.



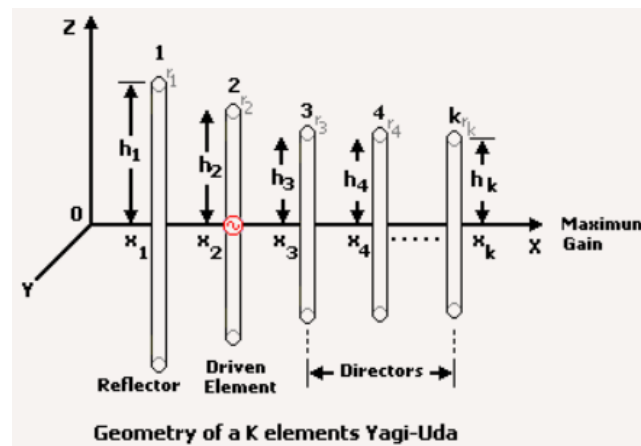
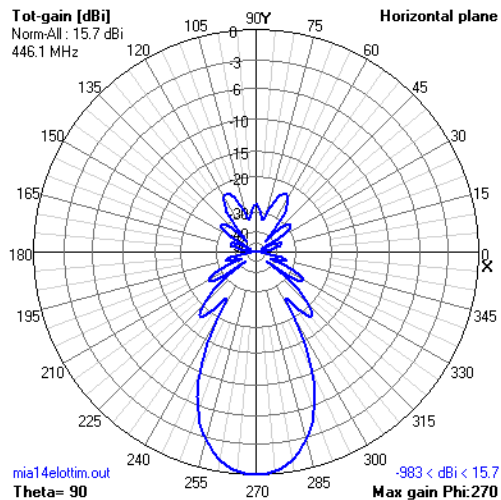
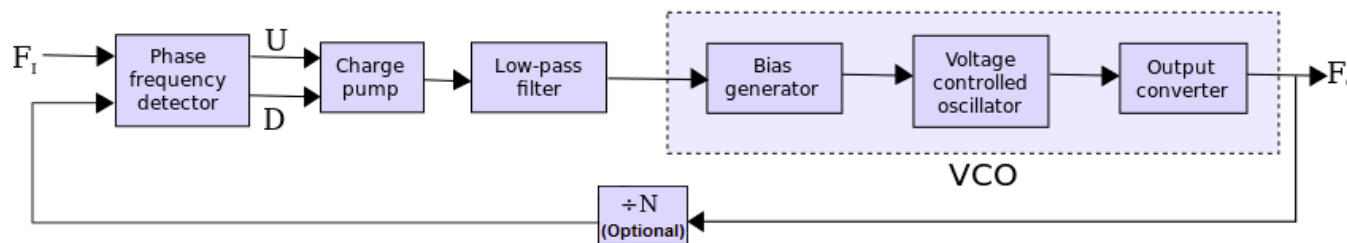
(a) 5.8 dBi Omni 3D Pattern



(b) 5.8 dBi Omni Azimuth Plane Pattern



(c) 5.8 dBi Omni Elevation Plane Pattern



### L'acceptation par le grand public :

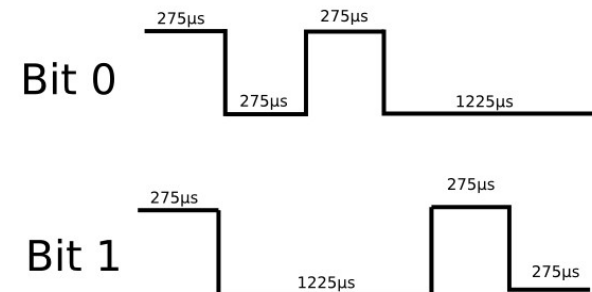
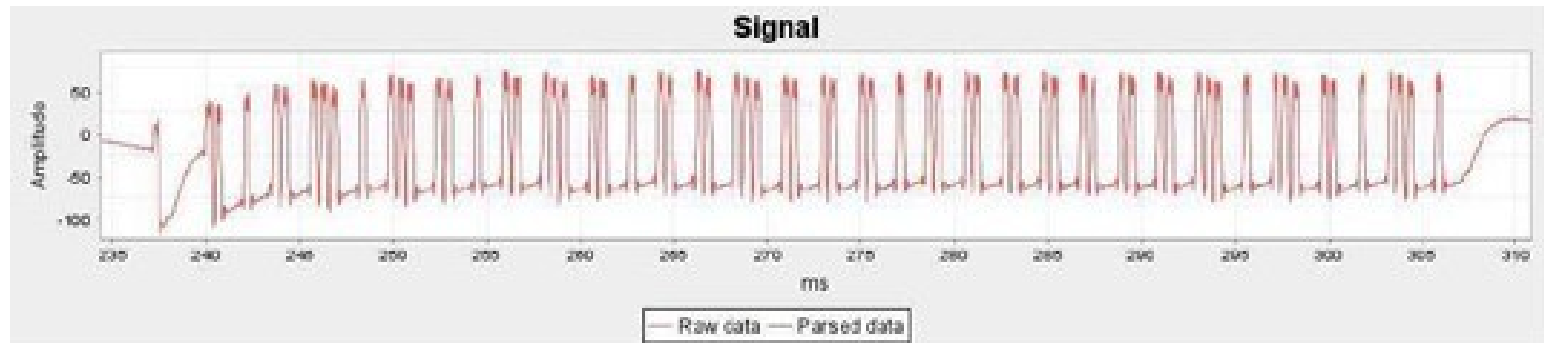
- intégrer les objets connectés dans le packaging voire dans les produits eux-mêmes → baisser les coûts ;
- utilisation de puces ultra-fines pour intégration textile ;
- problématique sécuritaire (piratage des caméras)
- entrer dans les mœurs





### Interopérabilité à améliorer :

- Les objets connectés actuels ne sont pas interopérables (HomeEasy : Dio Chacon, Oregon Scientific, ...)
- problème de standardisation / gouvernance → besoin d'un standard ouvert (eg. TCP/IP) et d'une agence (eg. IEEE) ;



Peu de protocoles utilisent un chiffrement :

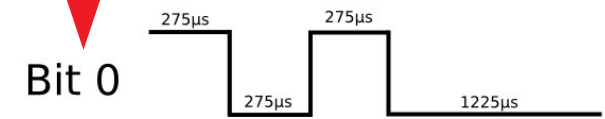
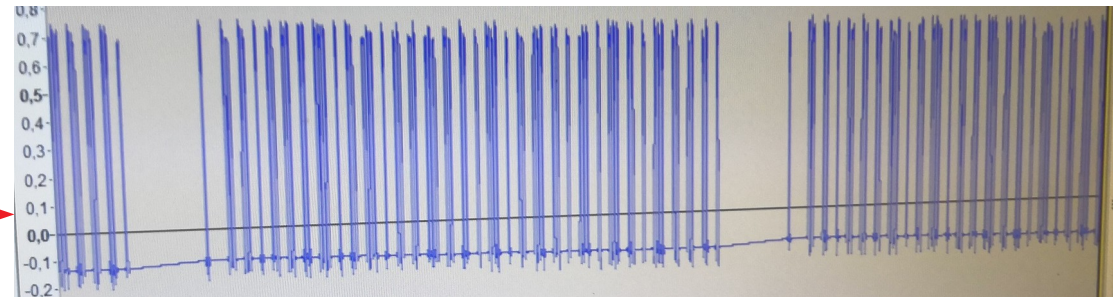
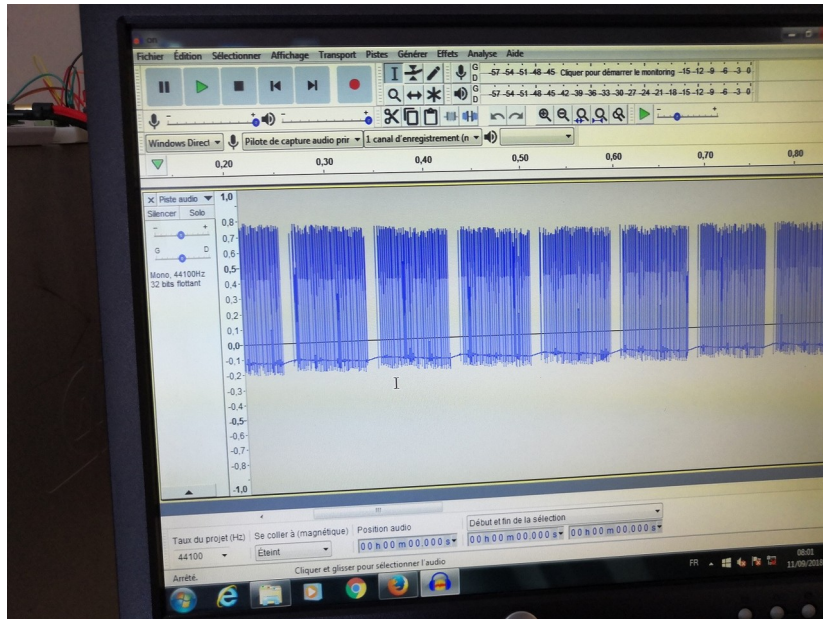


module radio  
(1€)

amplificateur audio *maison* (0.50€)

entrée micro du PC

Enregistrement grâce à un logiciel *professionnel* (**Audacity**) :




Trame :

01101001100101101001011010100101010101010101010100110010110010

Données :

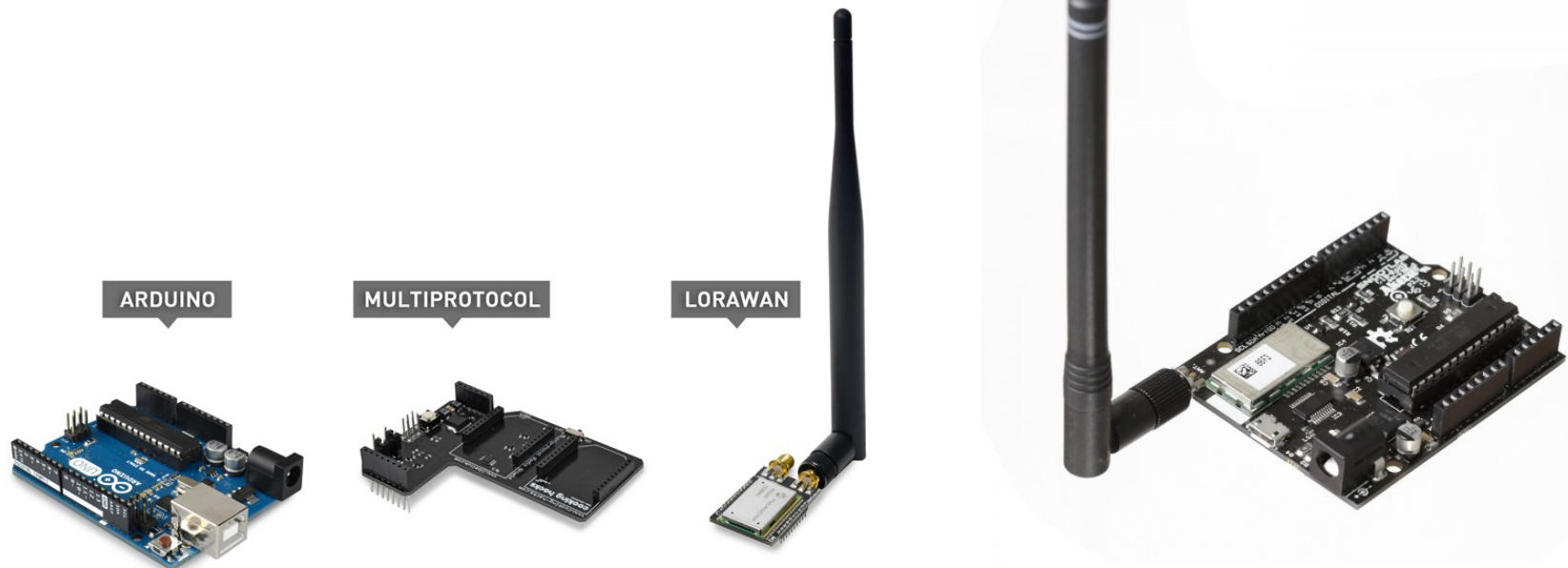
011010011001110000111111110100100

Infrastructure en phase de développement :

 • LoRaWAN (Long Range Wide-area network) ;  
Alliance comportant Cisco, IBM, Orange, STMicro, etc...



• SigFox : Ultra Narrow Band (UNB) réseau cellulaire ultra-bas débit ;



# Domaines d'application

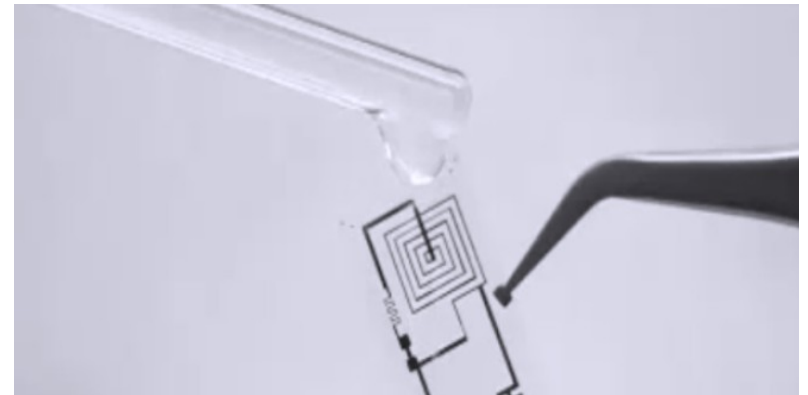
### En logistique :

- utilisation des puces RFID ;
- permet de rendre la marchandise « intelligente » / traçable ;
- entrepôts entiers entièrement automatisés (Amazon).



Dans le domaine pharmaceutique :

- puces bio-dégradables évitant les contre-façons ;
- automatisation de la préparation des ordonnances ;



Dans le domaine de la santé ("quantified self") :

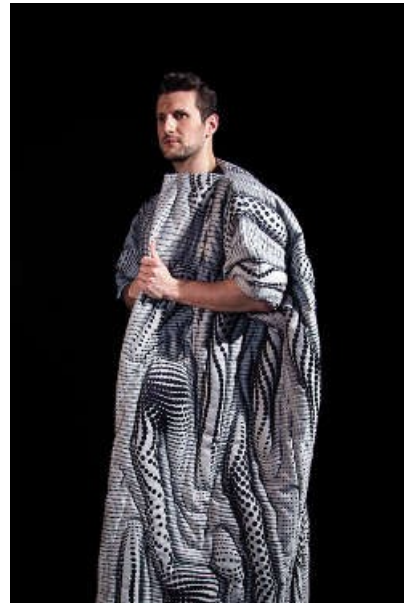
- Mieux se connaître (sport, balance connectée) ;
- Dépistage alerte (tension, pouls, ...) ;





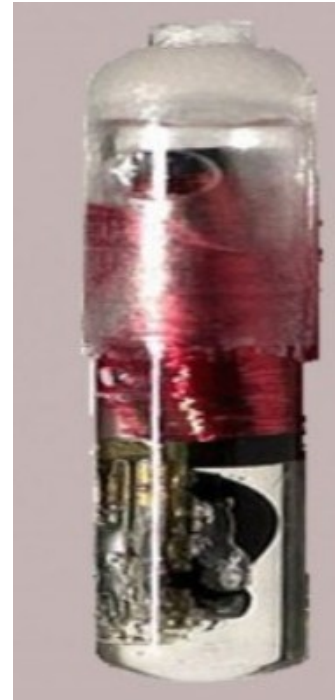
Dans le domaine de la mode ("wearable technologies") :

- combinaison ultra connectée (Wi-Fi, GPS, bluetooth et NFC);
- charger son portable en marchant (smart shoes) ;
- cape d'invisibilité.



Dans le domaine alimentaire :

- puces assurant la traçabilité du bétail ;
- bien utile pour éviter Creutzfeldt-Jakob ;



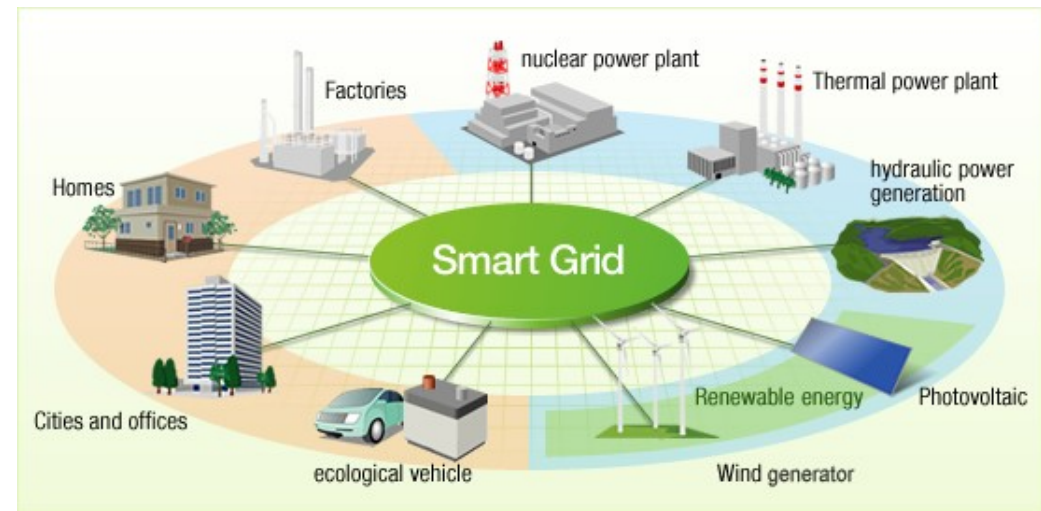
Dans le domaine de la domotique :

- piloter ses ouvrants ;
- contrôler le chauffage ;
- ...



Dans le domaine de l'énergie (smart grid) :

- optimiser la production ;
- optimiser l'acheminement ;
- contrôler le réseau
- ...



# Communication avec l'IdO

### Communications filaires :

- Ethernet : Ethernet, TCP/IP ;
- Bus : 1-Wire, I2C, ModBus, ProfiBus ;
- CPL : X10, In One Legrand, KNX, X2D Delta ;



PhpModbus



### Communications sans-fil :

- 3.1 / 10 Ghz : WUSB
- 2.4 Ghz : Bluetooth, ZigBee ;
- 868 : X2D, ZigBee (Europe), KNX, Z-Wave ;
- 433 : HomeEasy ;
- Infrarouge : IrDa, RC5 Philips

