

Transition environnementale de la filière du numérique : réflexions technologiques

S. Pillement

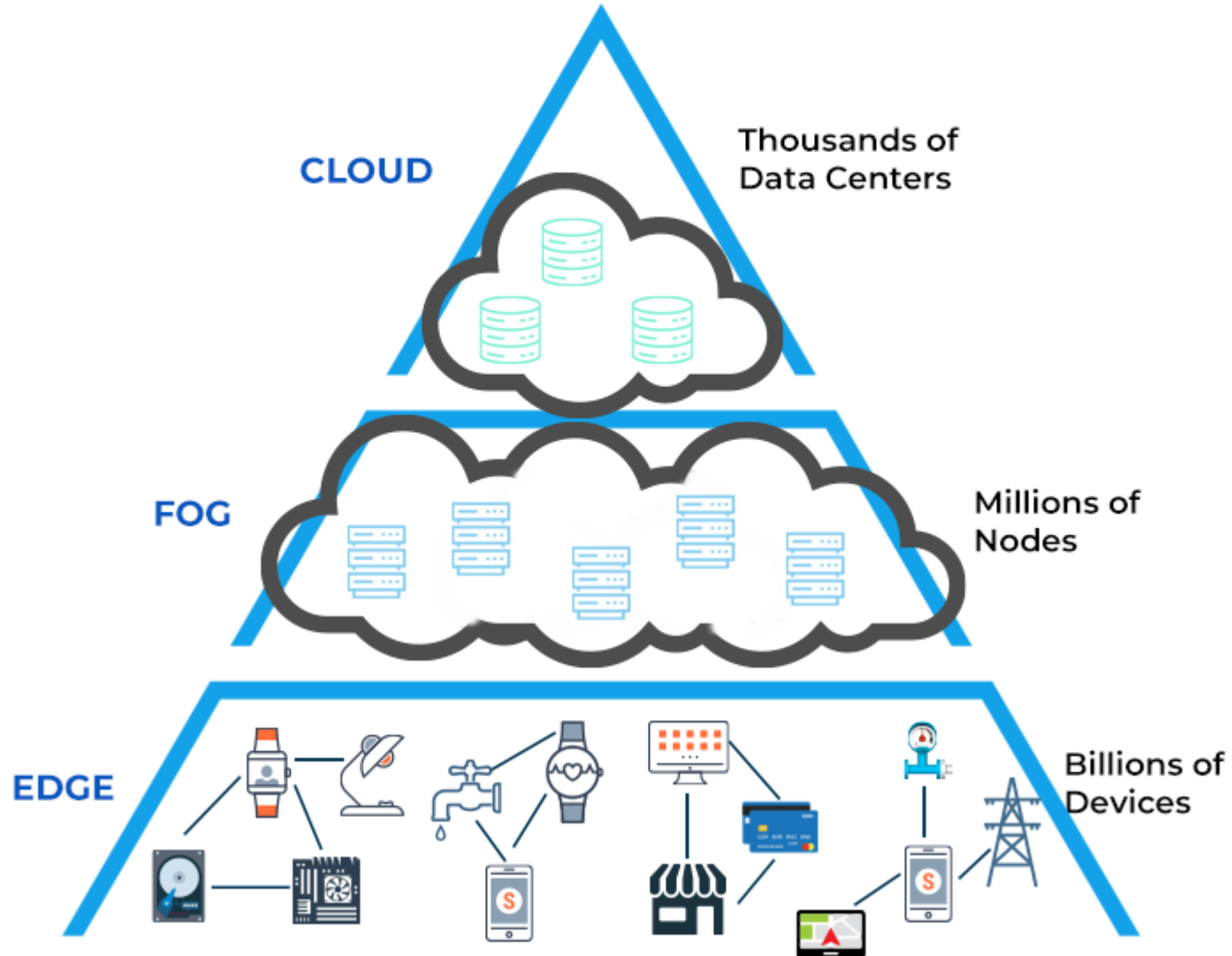
Professeur Nantes Université, IETR



- De l'impact de la numérisation de la société.
- Problèmes et pistes de solution au niveau device
- Problèmes et pistes de solution au niveau edge
- Problèmes et pistes de solution au niveau cloud
- Conclusions







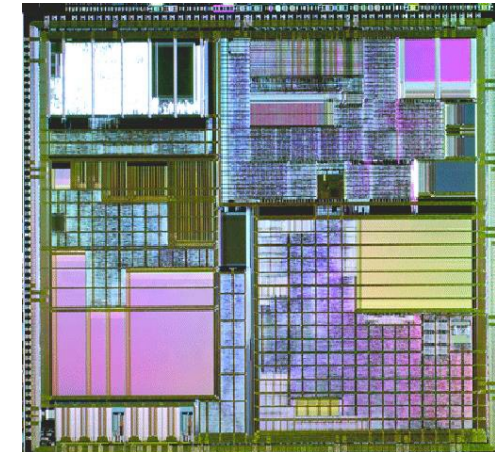
- **Cloud**
 - Traitement des données massives (Big data)
 - Stockage
- **Edge**
 - Traitements et réduction des données
 - Cache
 - Contrôle des réponses RT
 - Virtualisation
- **Device**
 - Capteurs et contrôleurs
 - Communications sans fil

- Fabrication de systèmes complexes
 - Energie
 - Matières premières
 - Produits chimiques
 - Eau
 - Emission de GES
- Utilisation et usages des produits électroniques
 - Consommation d'énergie et de puissance
 - Souvent connecté = opéré sur batterie
- Gaspillage de la production électronique
 - Haut degré de toxicité
- Responsabilité sociétale



IETR Des initiatives au niveau μ -électronique

- Approvisionnement des matériaux
 - WSC (World Semiconductor Council)
 - CDP (Carbon Disclosure Project)
 - RBA (Responsible Business Alliance)
 - RMI (Responsible Mineral Initiative), including conflict mineral
- Recherche
 - Remplacement de matériaux
 - Or par du cuivre ou autre
 - Elimination de produits dangereux (Brome, Chlore, antimoine)
 - Amélioration process de fabrication (taille wafer, couches métal, ...)



Columbite-tantalite



Cassiterite



Wolframite



Gold



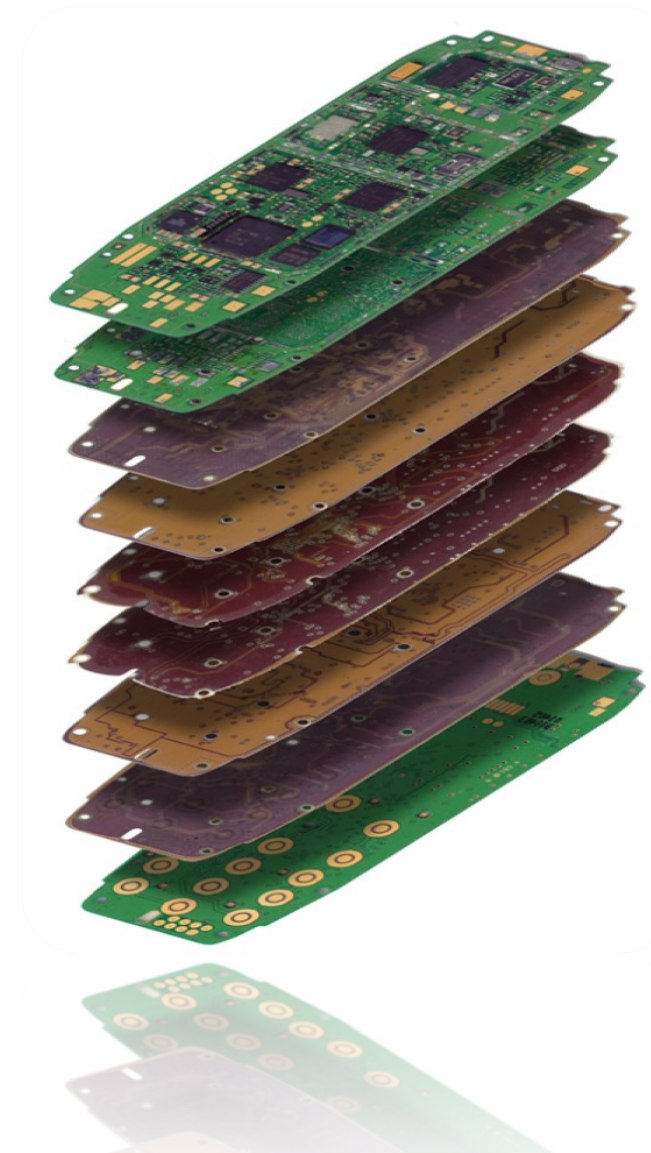
Couche devices

Ou objets connectés

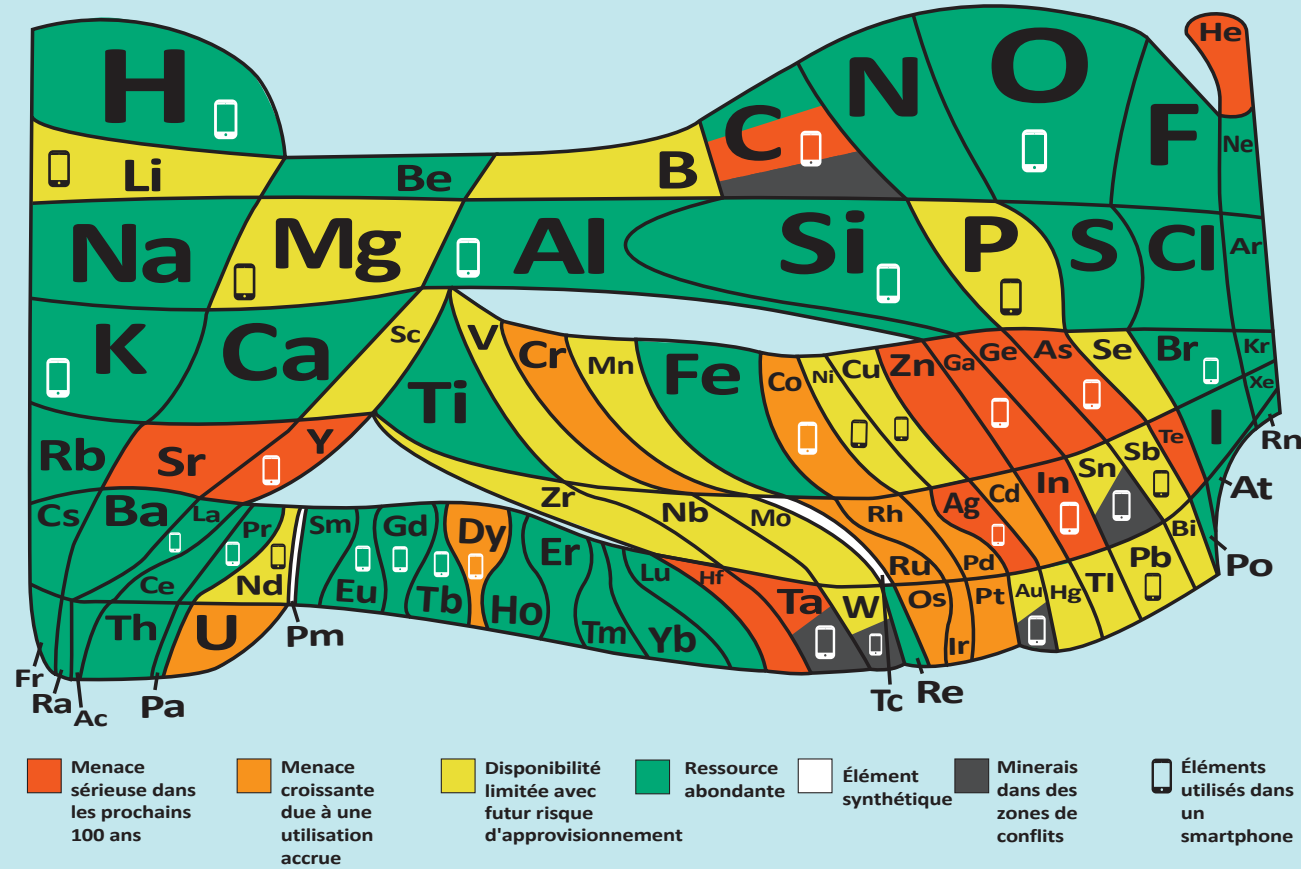
You can find a lot of electronics items

- Integrated circuits
 - Digital signal processor / processor
 - Memory (SRAM, FLASH)
 - RF frontend
 - Antenna
 - Analog-to-digital/digital-to-analog convertors
 - Smartcard (SIM)
 - Near field communication (NFC)
 - ...
- Sensors
 - CMOS image sensor
 - Accelerometer
 - Gyroscope
- Battery
- Touchscreens
- Printed circuit board (PCB)

And up to 54 chemical components



Les 90 éléments qui composent notre monde
Combien en reste-t-il? Y en a-t-il assez? Est-ce durable?



Inspiré par WF Sheehan's 'A Periodic Table with Emphasis' publiée dans Chemistry, 1976, 49, 17-18'

Lisez la suite et jouez au jeu vidéo en ligne : <http://bit.ly/euchems-pt>

2^{ème} édition (2021)



Ce travail est sous la license de Creative Commons Attribution-NoDerivs CC-BY-ND

EuChemS
European Chemical Society

Quelques pistes

Augmenter la réutilisation et la flexibilité des composants
Utilisation d'architectures reconfigurables.

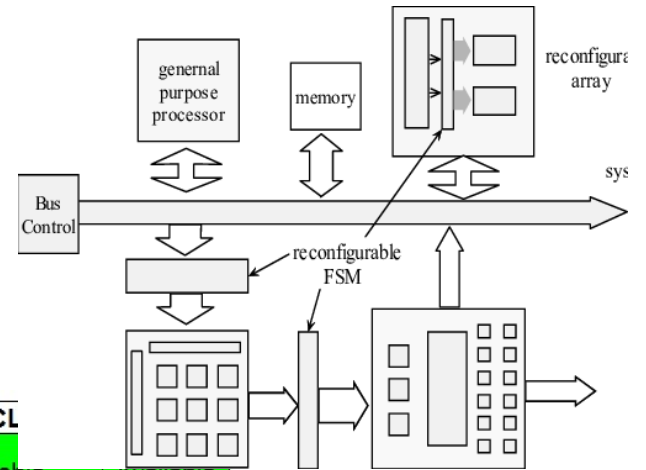
Rallonger le cycle de vie

Maintenabilité, réparabilité

Gestion avancée de la consommation

Diminuer la complexité des assemblages

Nouveau matériaux, nouvelles techniques



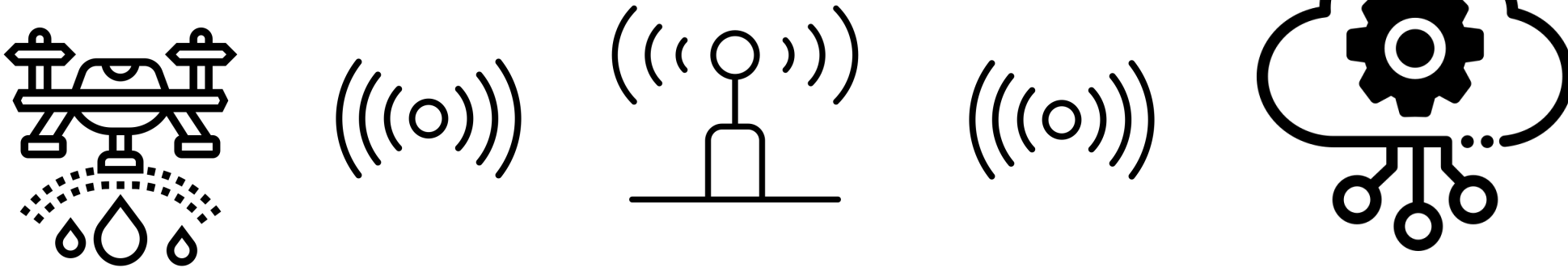
	CPU	MCLK	DCO	SMCLK	
LPM0	InActive	InActive	Available	Available	Available
LPM1	InActive	InActive	InActive	Available	Available
LPM2	InActive	InActive	InActive	InActive	Available
LPM3	InActive	InActive	InActive	InActive	Available
LPM4	InActive	InActive	InActive	InActive	InActive

SMCLK is not available in LPM1 if sourced from DCO

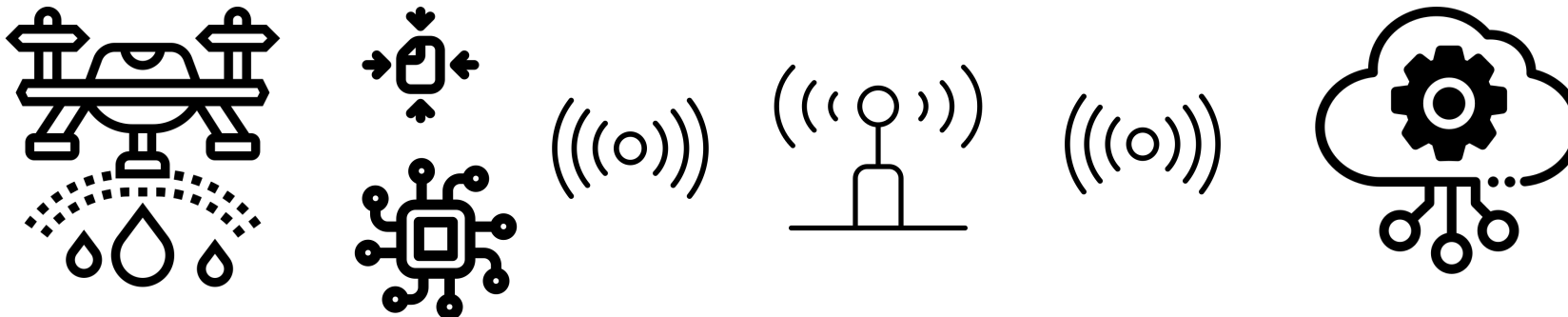


Augmenter la « pertinence » des données

De l'envoi de données brutes



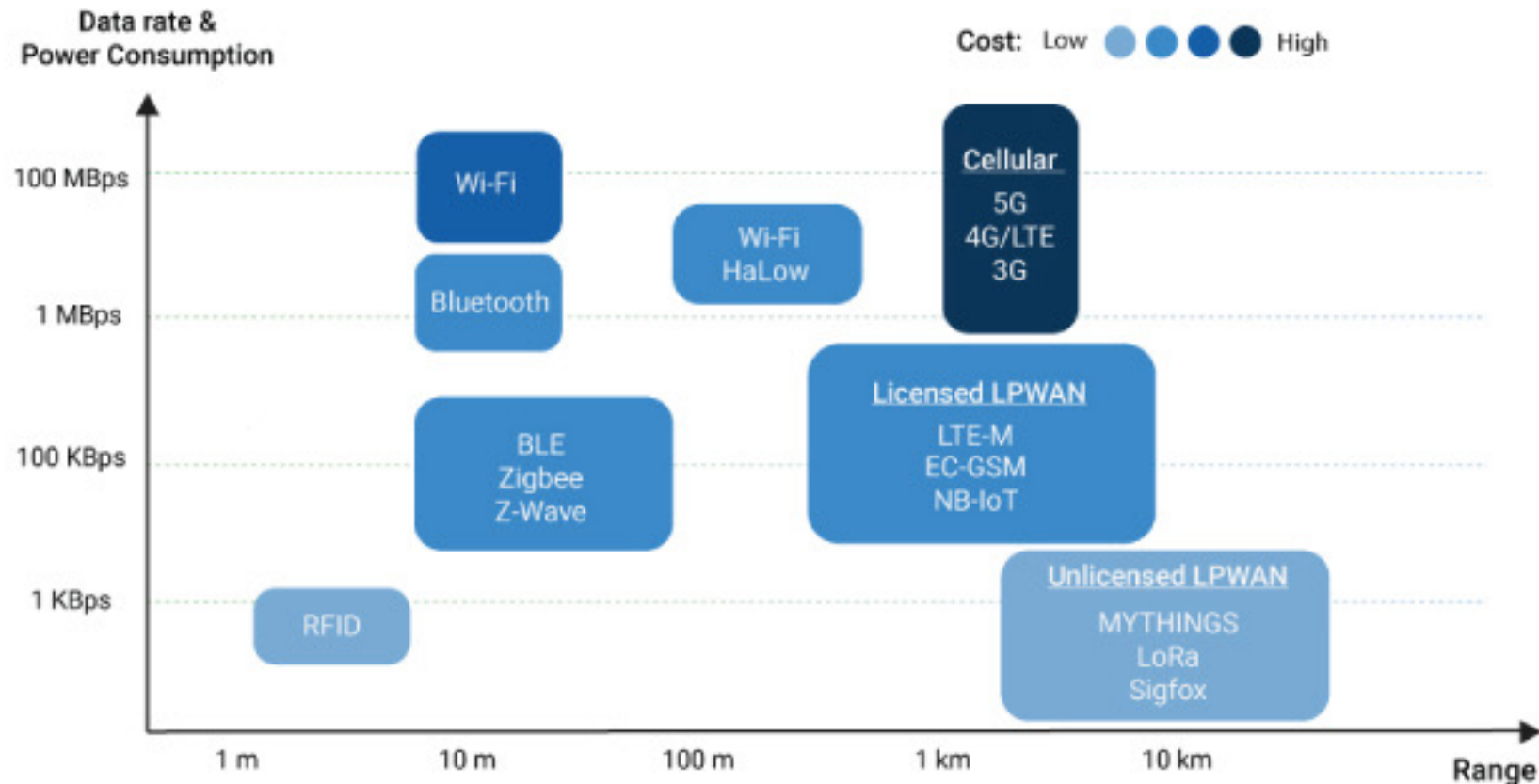
Au traitement local (envois de données traitées et qualifiées)



Le traitement local (compression, classification, identification) est plus efficace (1aJ/b/op) que la transmission de données massives (1pJ/b/mm)

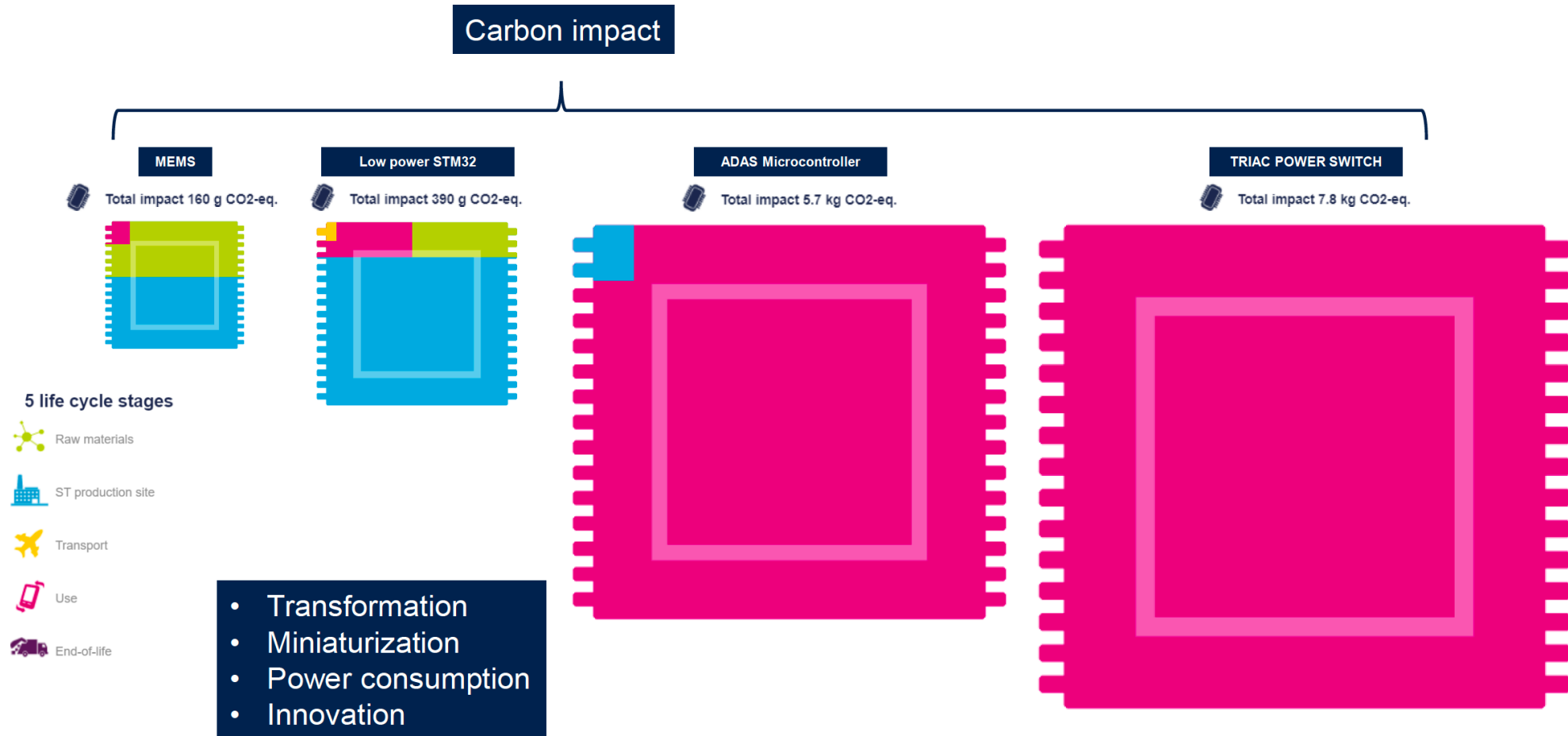
Choisir le bon protocole

Exemple : Le WiFi consomme 3* moins que la 4G (source ADEME).



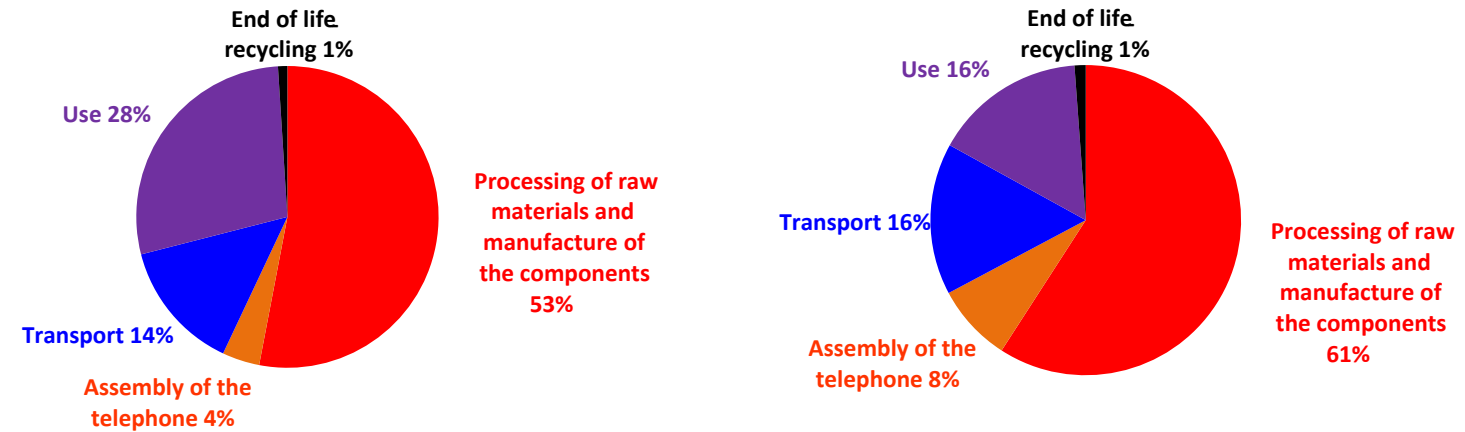
<https://behrtech.com/blog/6-leading-types-of-iot-wireless-tech-and-their-best-use-cases/>

Dépend de l'usage et de la fonction

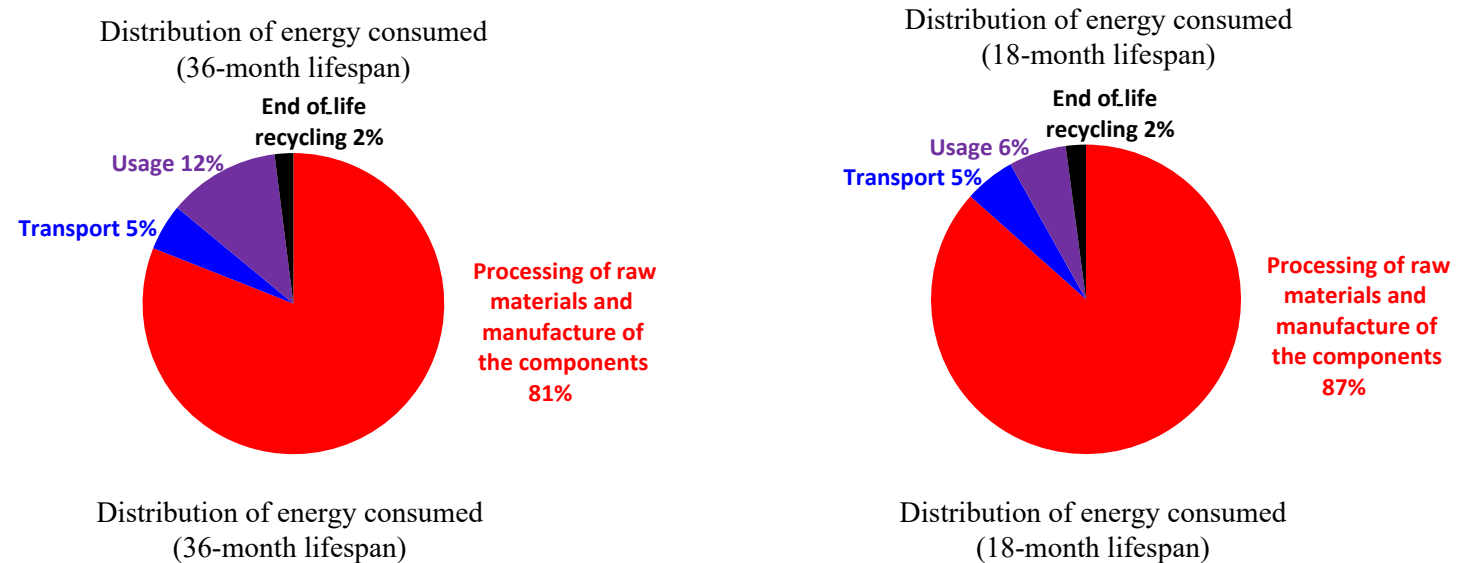


Distribution de la consommation d'énergie sur le cycle de vie

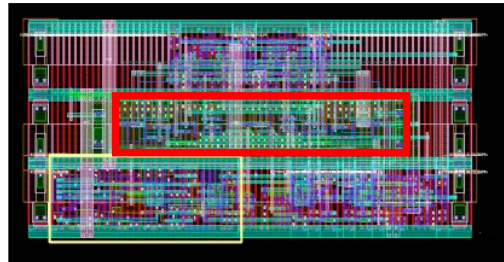
- Nokia [1]



- Apple (iPhone 5S) [2]

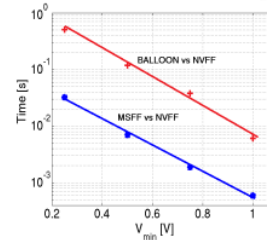


Vers des architectures normalement éteinte



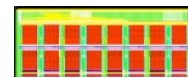
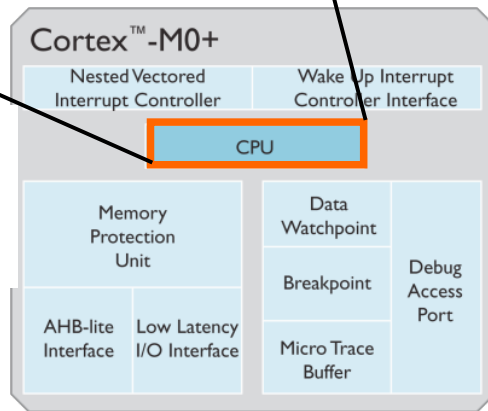
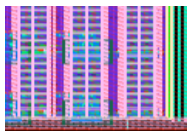
CPU w/ NV-logic

- Hybrid NV-FF
- ReRAM elements



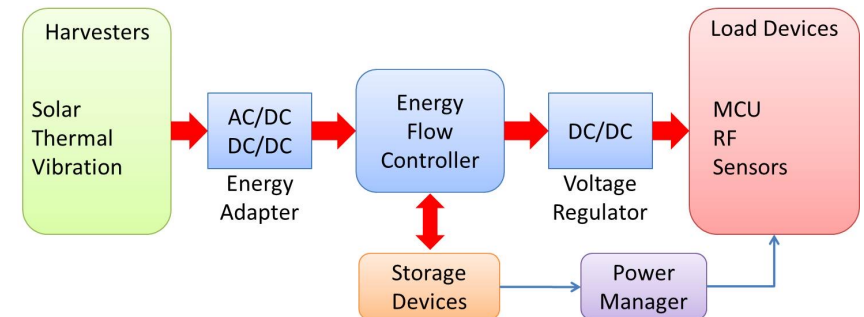
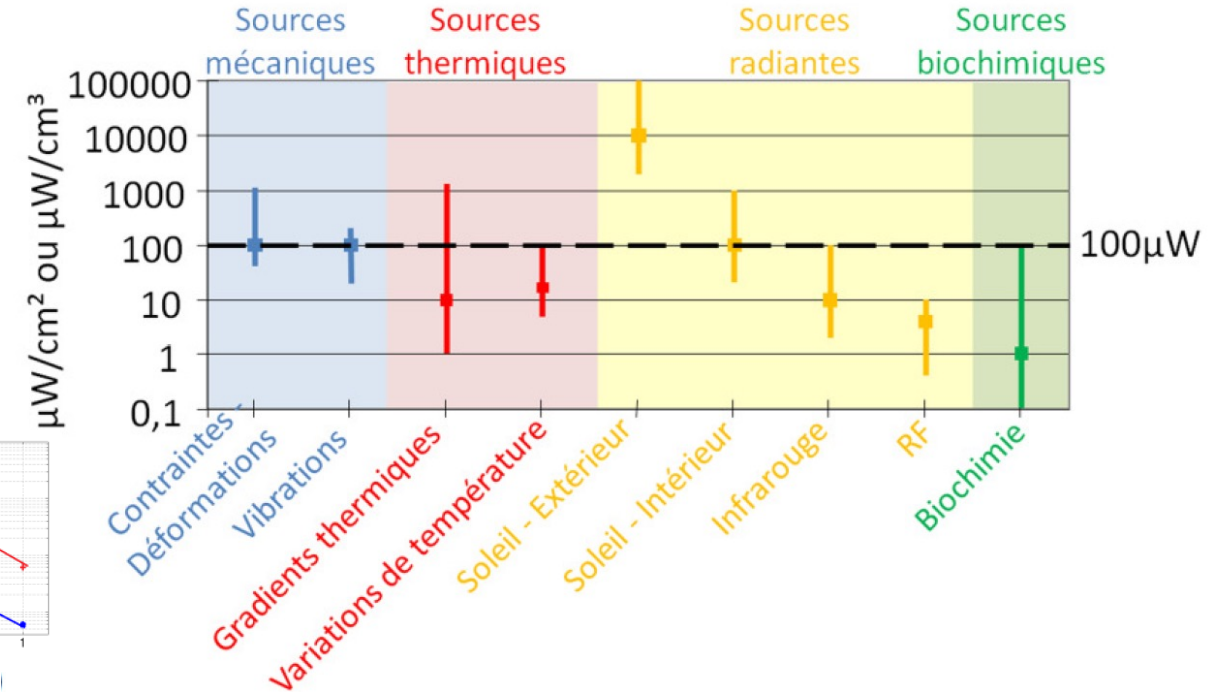
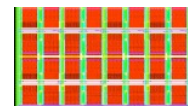
32Kb NV-SRAM

- 32Kb SRAM
- 32Kb ReRAM



128Kb ReRAM Instruction Memory

- HfOx ReRAM
- TiN-Ti electrodes



1. Decrease Transmit Power

- Channel coding, cooperation

2. Optimize radio activity

- MAC protocols, Wake-Up radio

3. Reduce the amount of data

- Compression, feature extraction

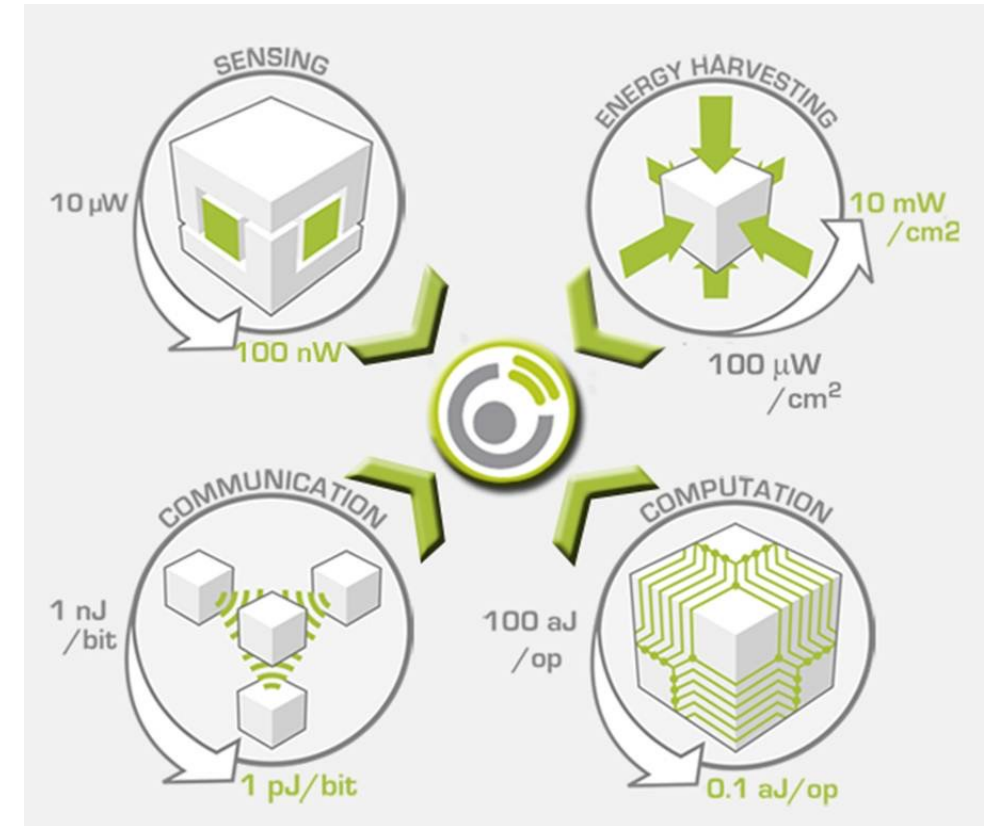
4. Optimize hardware architecture

- Co-processing, DVFS, power-gating, Energy harvesting

5. Optimize technologies

- Matériels
- Assembling

6. Augment life cycle





Couche edge

Electronique à la frontière

Journée Tech4Green - S. Pillement - IETR

[1] clients “care first and foremost about speed of development, and secondly about reasonable quality and performance.”

“It’s more often the hardware rather than the software that we are interested in when we talk about energy consumption.”

[2] These results show that these programmers lacked knowledge of how to accurately measure software energy consumption.



Software (App)

Software (Framework)

Software (JVM)

Software (container)

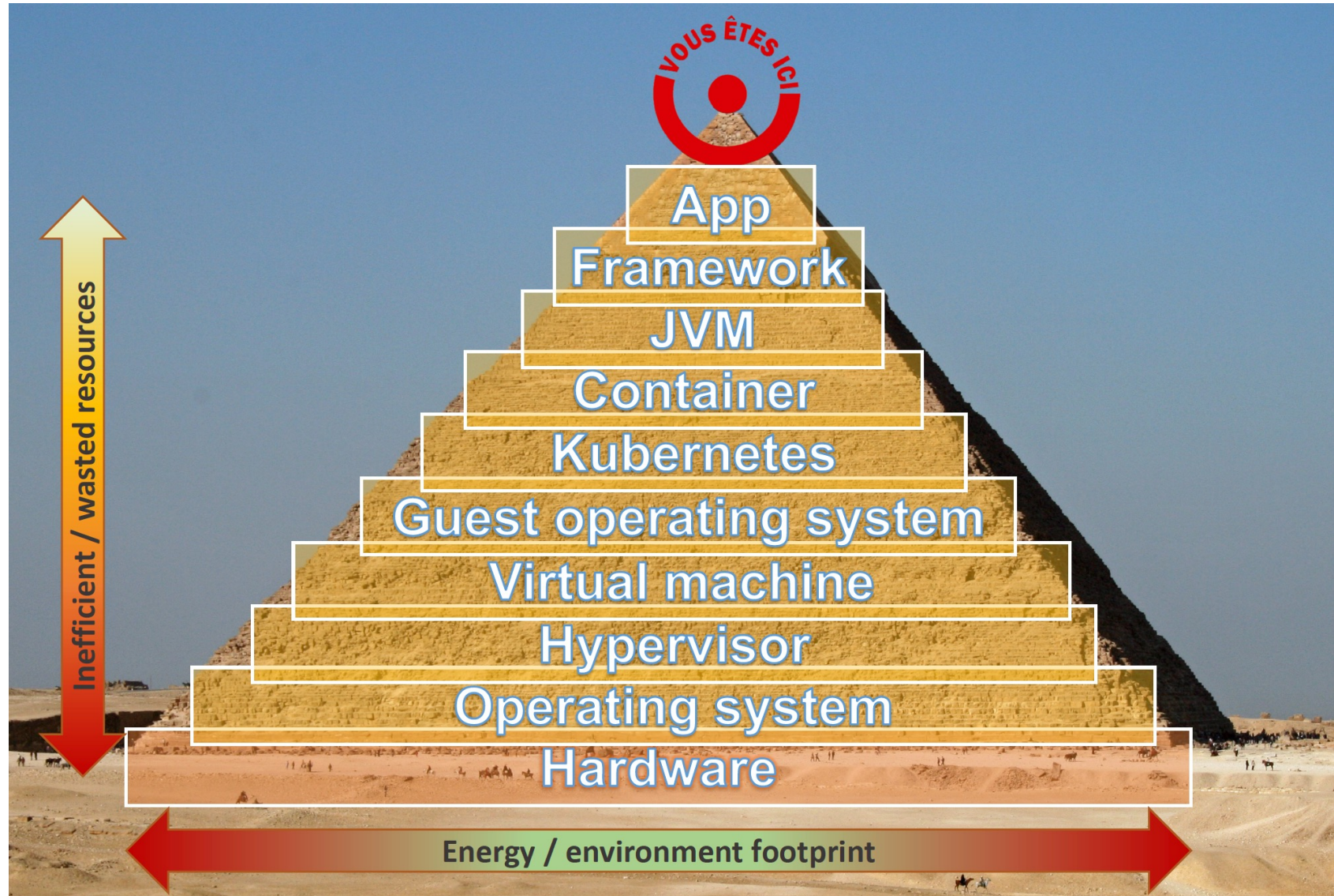
Software (OS)

Software (VM)

Software (Hyperviseur)

Software (OS)

Hardware



1. Etude réalisée sur le Computer Language Benchmarks Game (CLBG)

13 applications

2. Est ce que le langage le plus rapide est toujours le plus EE ?

Si c'est vrai pour les 5 premiers, pas vrai pour les autres

3. Quel est l'impact de la mémoire sur l'E ?

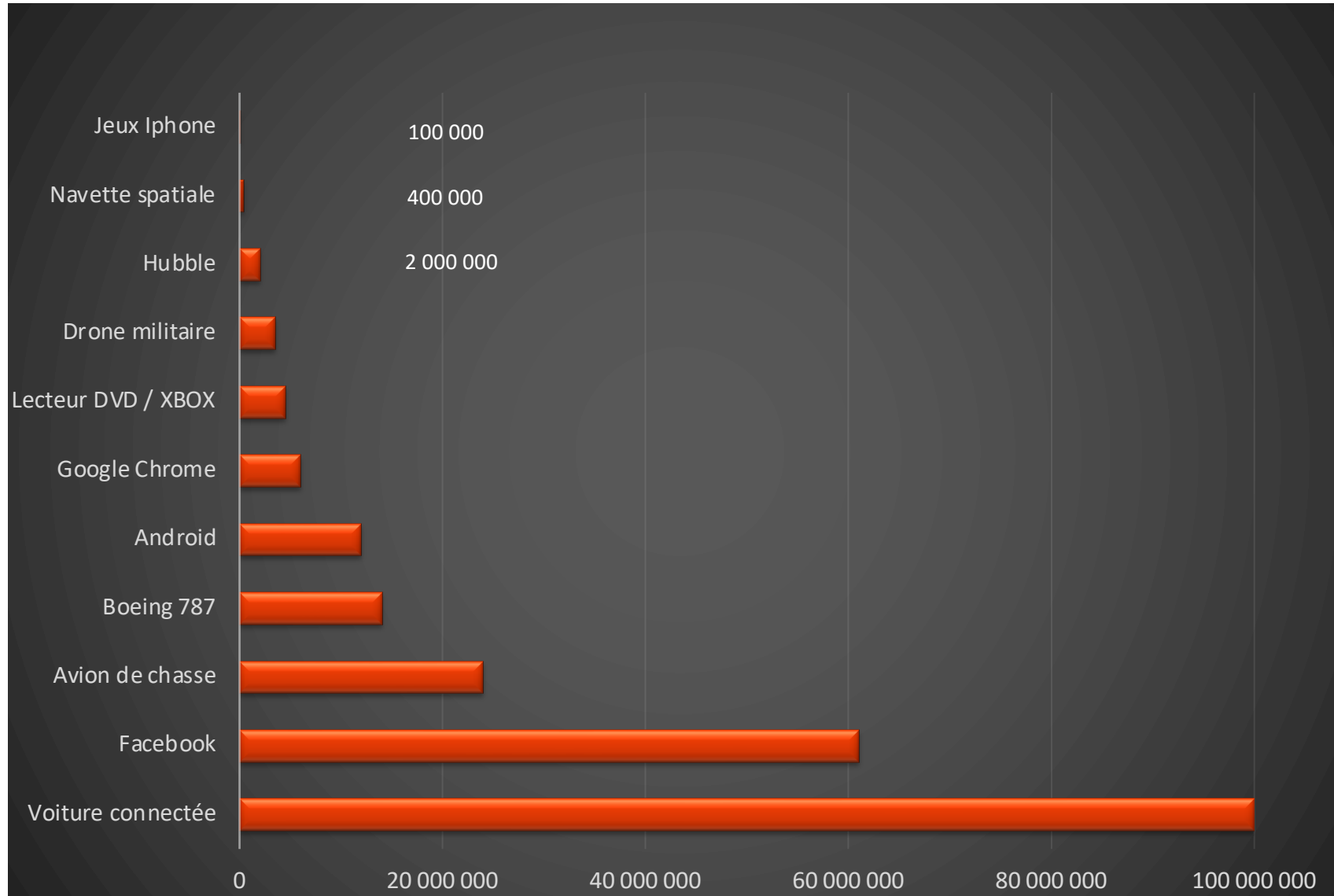
C'est plus comment la mémoire est utilisée qui importe plutôt que sa taille

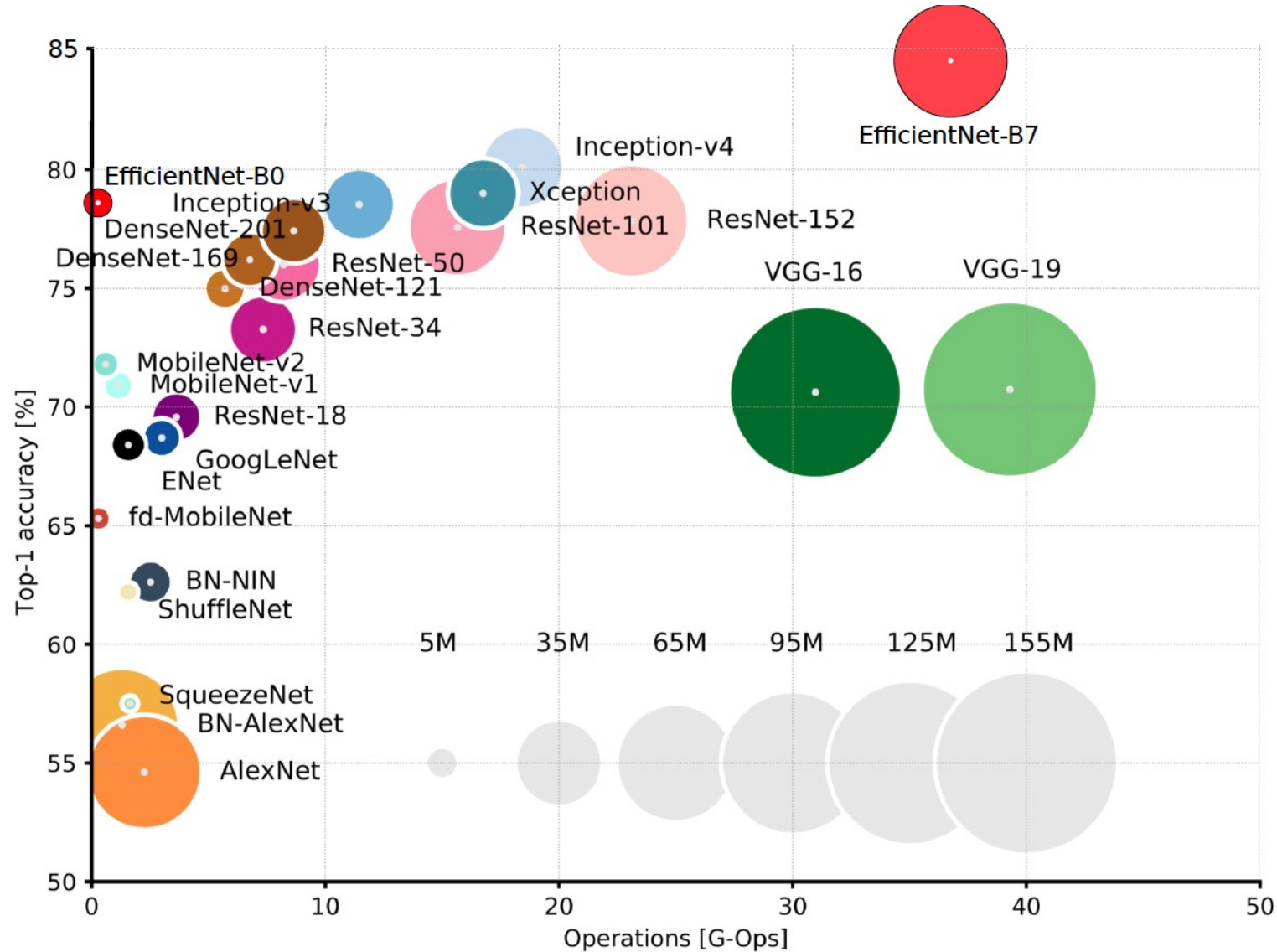
4. Comment choisir le langage ?

Facile si seul le timing et énergie sont considérés. Plus difficile si la taille mémoire est aussi un facteur.

Total					
	Energy		Time		Mb
(c) C	1.00	(c) C	1.00	(c) Pascal	1.00
(c) Rust	1.03	(c) Rust	1.04	(c) Go	1.05
(c) C++	1.34	(c) C++	1.56	(c) C	1.17
(c) Ada	1.70	(c) Ada	1.85	(c) Fortran	1.24
(v) Java	1.98	(v) Java	1.89	(c) C++	1.34
(c) Pascal	2.14	(c) Chapel	2.14	(c) Ada	1.47
(c) Chapel	2.18	(c) Go	2.83	(c) Rust	1.54
(v) Lisp	2.27	(c) Pascal	3.02	(v) Lisp	1.92
(c) Ocaml	2.40	(c) Ocaml	3.09	(c) Haskell	2.45
(c) Fortran	2.52	(v) C#	3.14	(i) PHP	2.57
(c) Swift	2.79	(v) Lisp	3.40	(c) Swift	2.71
(c) Haskell	3.10	(c) Haskell	3.55	(i) Python	2.80
(v) C#	3.14	(c) Swift	4.20	(c) Ocaml	2.82
(c) Go	3.23	(c) Fortran	4.20	(v) C#	2.85
(i) Dart	3.83	(v) F#	6.30	(i) Hack	3.34
(v) F#	4.13	(i) JavaScript	6.52	(v) Racket	3.52
(i) JavaScript	4.45	(i) Dart	6.67	(i) Ruby	3.97
(v) Racket	7.91	(v) Racket	11.27	(c) Chapel	4.00
(i) TypeScript	21.50	(i) Hack	26.99	(v) F#	4.25
(i) Hack	24.02	(i) PHP	27.64	(i) JavaScript	4.59
(i) PHP	29.30	(v) Erlang	36.71	(i) TypeScript	4.69
(v) Erlang	42.23	(i) Jruby	43.44	(v) Java	6.01
(i) Lua	45.98	(i) TypeScript	46.20	(i) Perl	6.62
(i) Jruby	46.54	(i) Ruby	59.34	(i) Lua	6.72
(i) Ruby	69.91	(i) Perl	65.79	(v) Erlang	7.20
(i) Python	75.88	(i) Python	71.90	(i) Dart	8.64
(i) Perl	79.58	(i) Lua	82.91	(i) Jruby	19.84

IETR Nbr de lignes de code





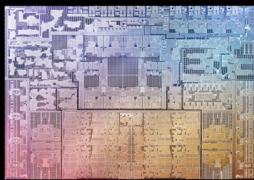
PUCE M1 Ultra de Apple

2 puces M1 Max cintégra,nt 114 Milliards de Transistors. Supporte jusqu'à 128 Go de mémoire hautes performances, avec un CPU de 20 cœurs, un GPU de 64 cœurs et le Neural engine de 32 cœurs.

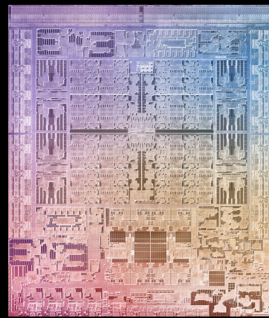
Toujours de l'embarqué ?



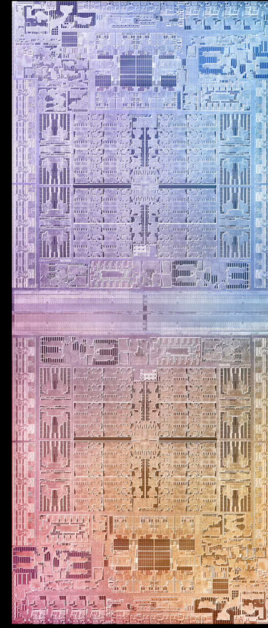
Apple M1



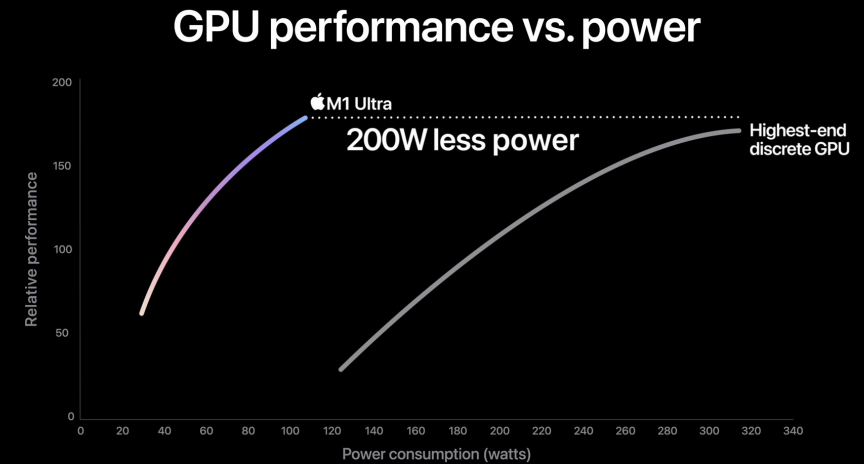
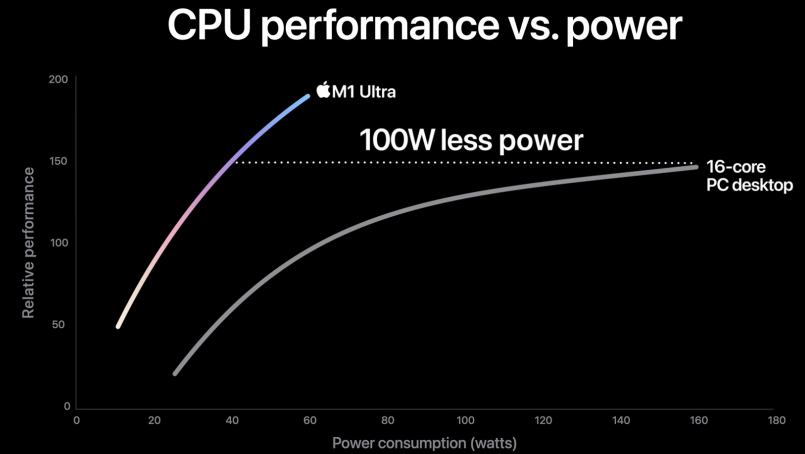
Apple M1 Pro



Apple M1 Max



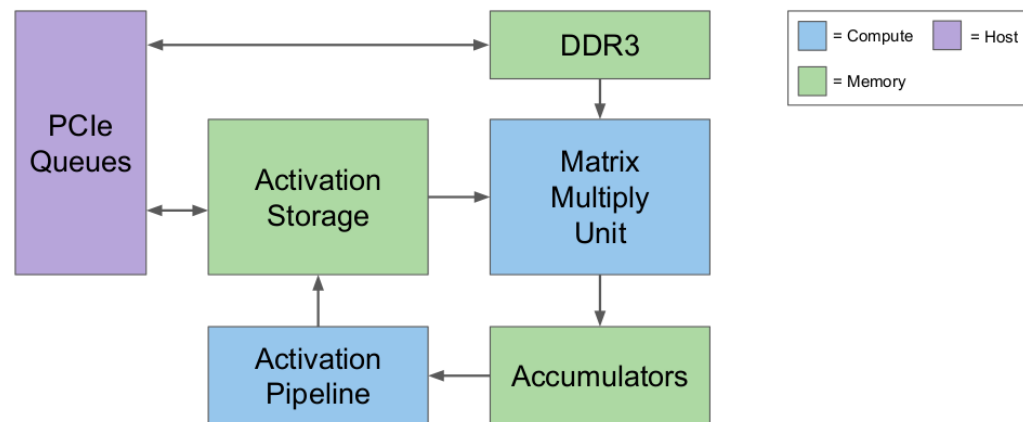
Apple M1 Ultra



Vers la spécialisation, mais ???

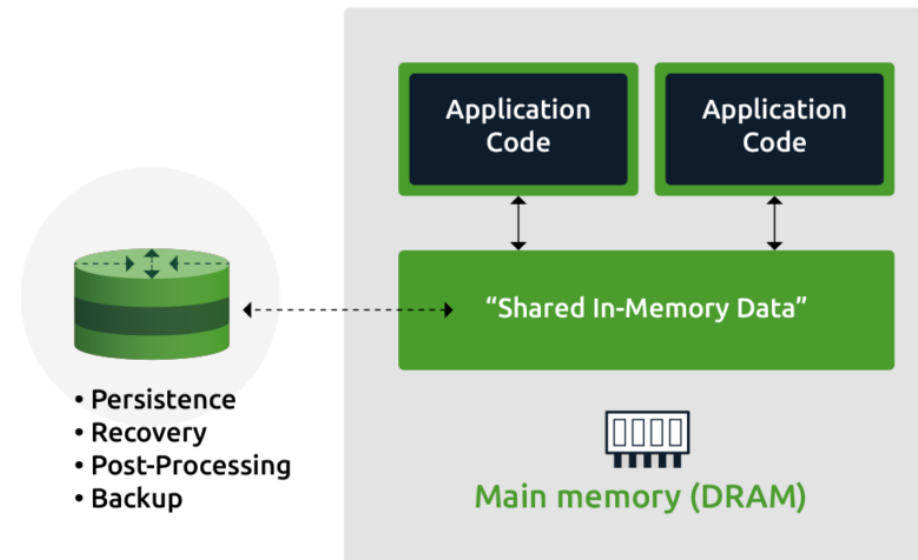
- Acheminer "la donnée" (mémoire CPU) reste le véritable défi aujourd'hui :`
 - cela demande de repenser toute la chaîne (logiciel-compilation-matériel)
 - In-memory computing
- Revenir aux circuits spécialisés (optimisation des ressources) vs CPU, GPU ?
 - Contradictoire avec la réutilisation
 - Avènement de l'Open-HW, pour la réutilisation ultime

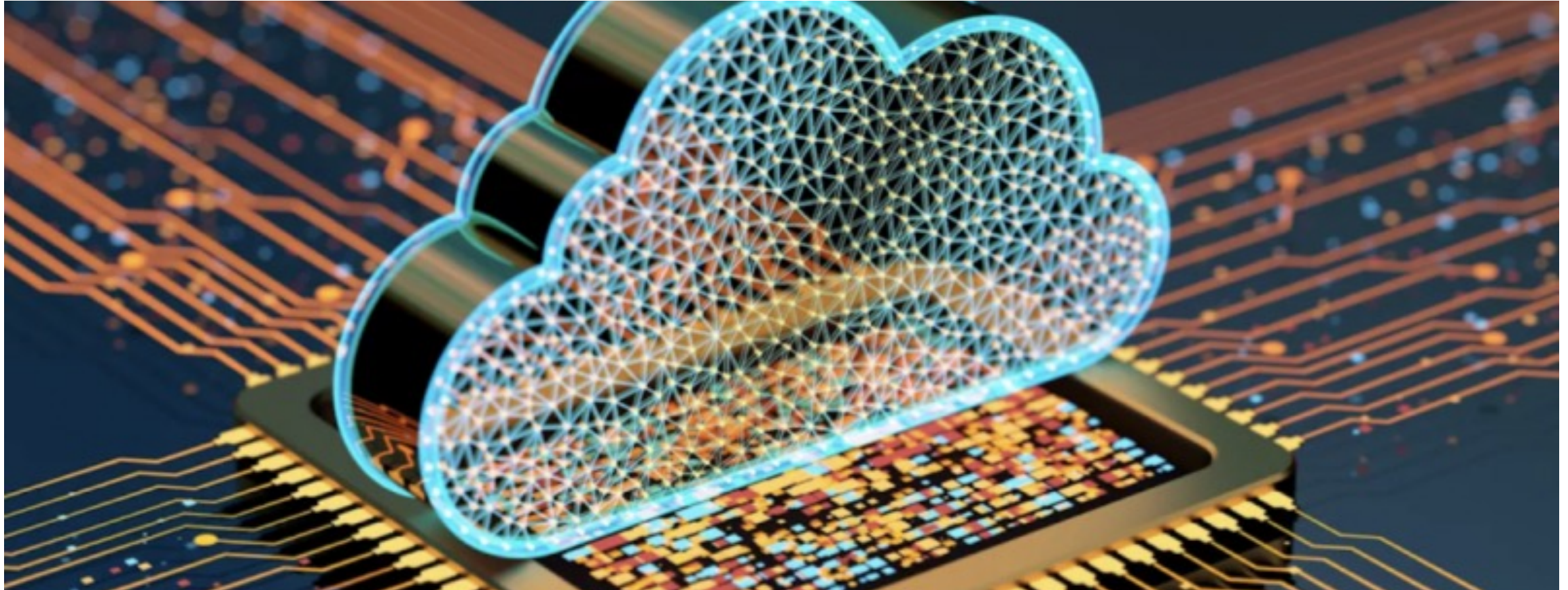
TPUv1 Recap



Journée Tech4Green - S. Pillement - IETR

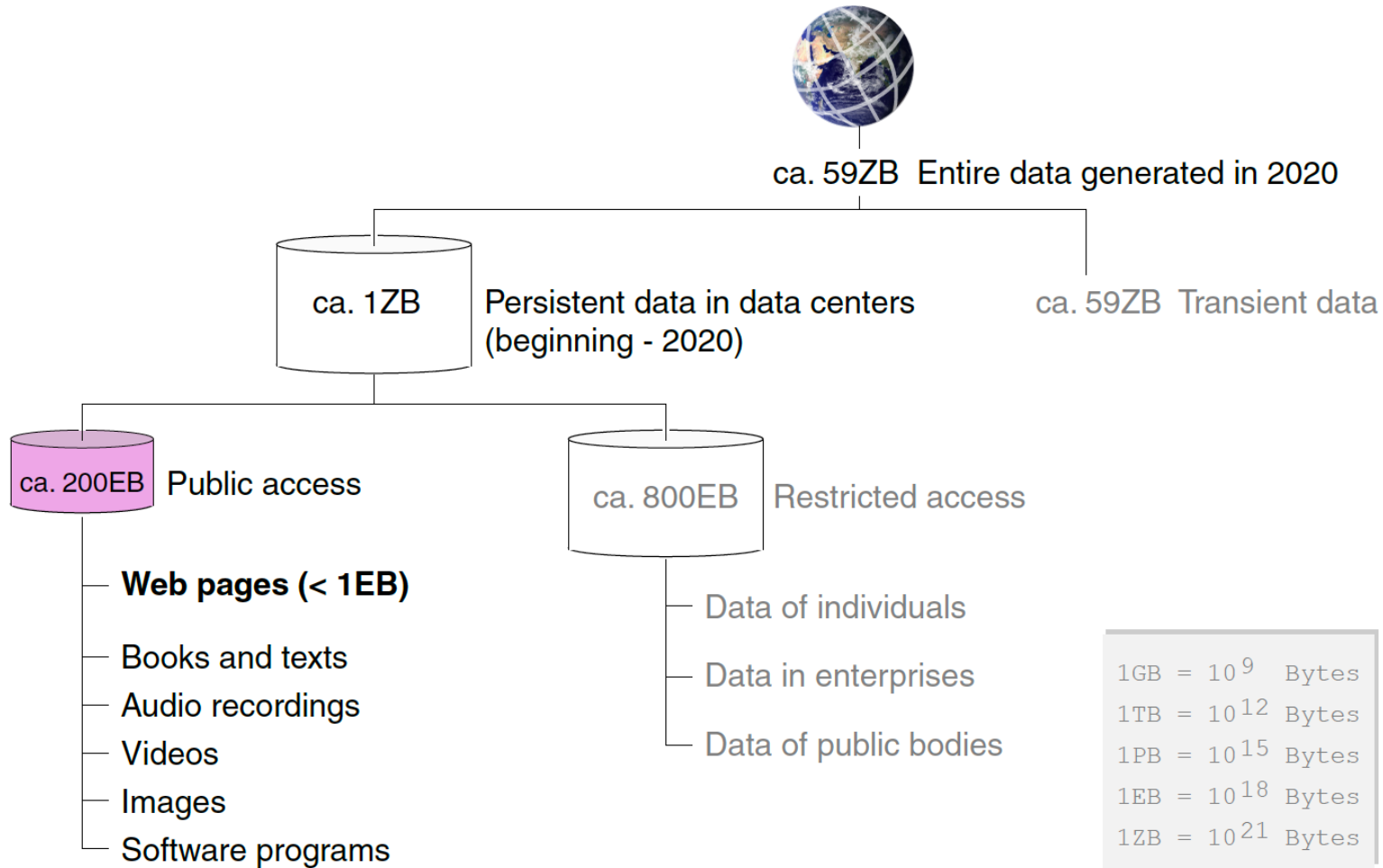
In-Memory Computing





Couche cloud

IETR L'explosion des données



Consumption	CO ₂ e (lbs)
Air travel, 1 passenger, NY↔SF	1984
Human life, avg, 1 year	11,023
American life, avg, 1 year	36,156
Car, avg incl. fuel, 1 lifetime	126,000

Training one model (GPU)	
NLP pipeline (parsing, SRL)	39
w/ tuning & experimentation	78,468
Transformer (big)	192
w/ neural architecture search	626,155



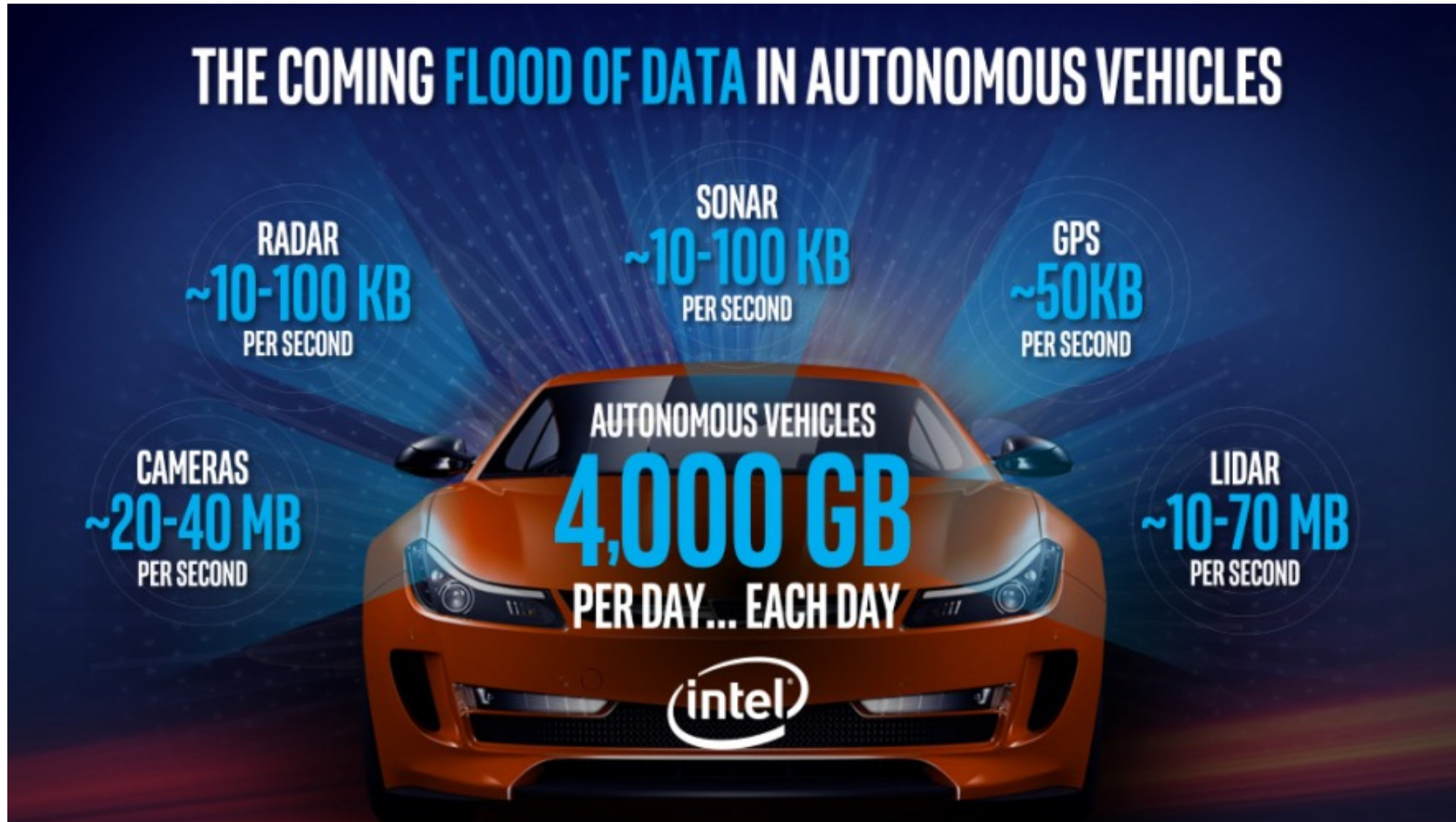
Artificial Intelligence / Machine Learning

Training a single AI model can emit as much carbon as five cars in their lifetimes

Deep learning has a terrible carbon footprint.

Deep learning has a terrible carbon footprint.

lifetimes
as five cars in their



- Limiter le nombre de données :
 - Nouveaux usages : réduire la dépendance aux données (stockage).
 - Limiter le besoin : quelle qualité de données ? TVHD sur écran 7"
- Optimiser les couches matérielles, logicielles et traitements du cloud
 - Pour une voiture l'AI nécessite 2000W (sur WAYMO) (cerveau humain 20W)
 - FPGA in the cloud
 - Réutiliser, recycler
- Traitements distribués (locaux)
 - Compression, moyennage
 - IA fédéré par exemple





Conclusions

- Obsolescence :
 - Conception durable
 - Adaptabilité, flexibilité peut être pas sur tous les éléments (i.e. capteur)
 - Approvisionnement
- Interopérabilité :
 - Lien obsolescence
 - Virtualisation
- Modélisation/Simulation
 - Dimensionnement
 - Compatibilité entre objets
- Traitement des données
 - Pluridisciplinarité
 - Fusions des données (capteurs)
 - Interprétation et présentation
 - Traitement adaptatif



- Autonomie
 - Ultra-basse consommation, power management
 - Récupération d'énergie
- Flexibilité
 - Auto organisation
 - Adaptation
- Sécurité
 - Des systèmes
 - Des utilisateurs
 - Données
 - Santé
 - Mise à jour externe via réseau & co
- Usages (SHS)
 - Acceptation, réduction
 - Protection des données/anonymisation
 - Ethique



- Impact généraux positif et négatif du numérique :
 - Santé, télétravail, accès à l'information, ...
 - Utilisation excessive, réseaux sociaux, ...
- Impact environnementaux positif et négatif
 - 4% des émissions de gaz à effet de serre (en augmentation) + déchets (source: <https://theshiftproject.org>)
 - Potentiel pour réduire l'impact d'autres secteurs
- L'électronique n'est pas (et ne sera certainement pas) une low-tech mais on peut améliorer les choses (<https://longuevieauxobjets.gouv.fr>)
- Nécessité d'une révolution sociologique (usage, effet rebond ou Paradoxe de Jevons)
 - Différentié l'utile du futile

Le numérique est une partie du problème mais aussi une partie de la solution
mais dans une approche beaucoup plus globale

**Merci pour votre
attention**

