

# ÉCOCONCEPTION DE SERVICES NUMÉRIQUES

## Partie 3 - Mise en œuvre de l'écoconception : outils pratiques pour les développeurs

● ECO-TIC  
CONCEPTION ET  
DÉVELOPPEMENT  
LOGICIEL

ÉCOCONCEPTION

TIC ET DÉVELOPPEMENT DURABLE



EN PRATIQUE

DESIGN  
NUMÉRIQUE  
RESPONSABLE

LOGICIEL  
ECO-RESPONSABLE

GREEN IT

# MISE EN ŒUVRE DE L'ÉCOCONCEPTION : OUTILS PRATIQUES POUR LES DÉVELOPPEURS

- Outils d'évaluation de l'impact Conception et environnementale
- Outils de conception et développement logiciel

# L'ÉCOCONCEPTION EN PRATIQUE

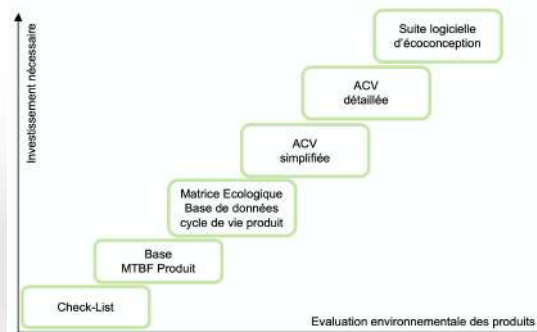


## Quelles outils au service de l'ecoconception ?

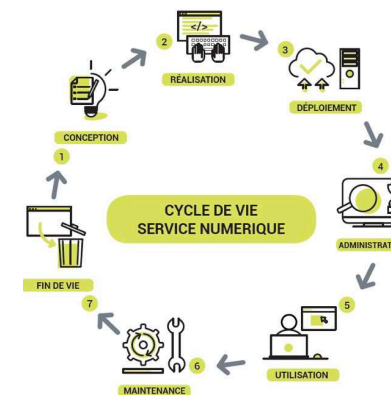


### Deux types d'outils :

#### Evaluation de l'impact environnementale

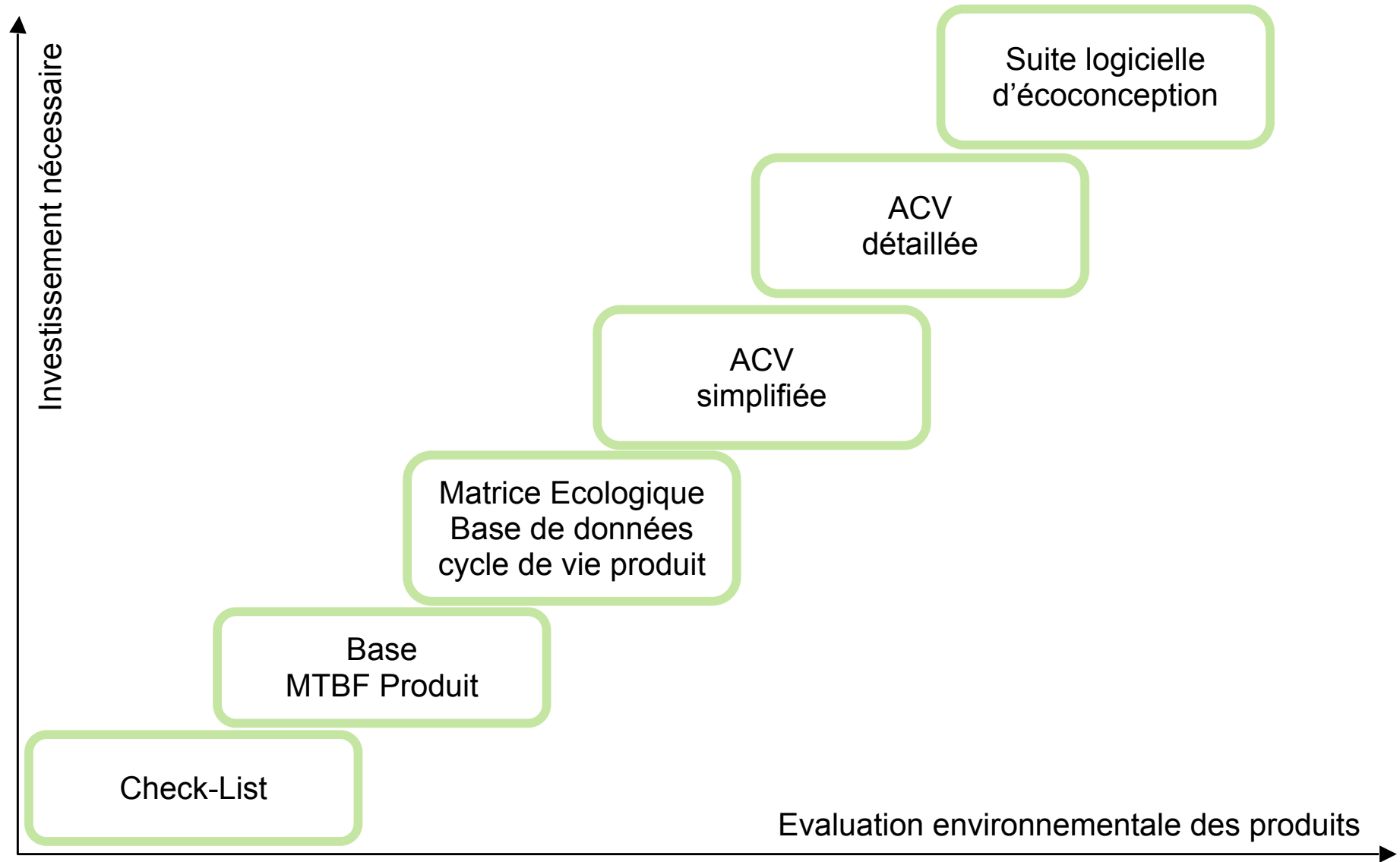


#### Conception et développement logiciel



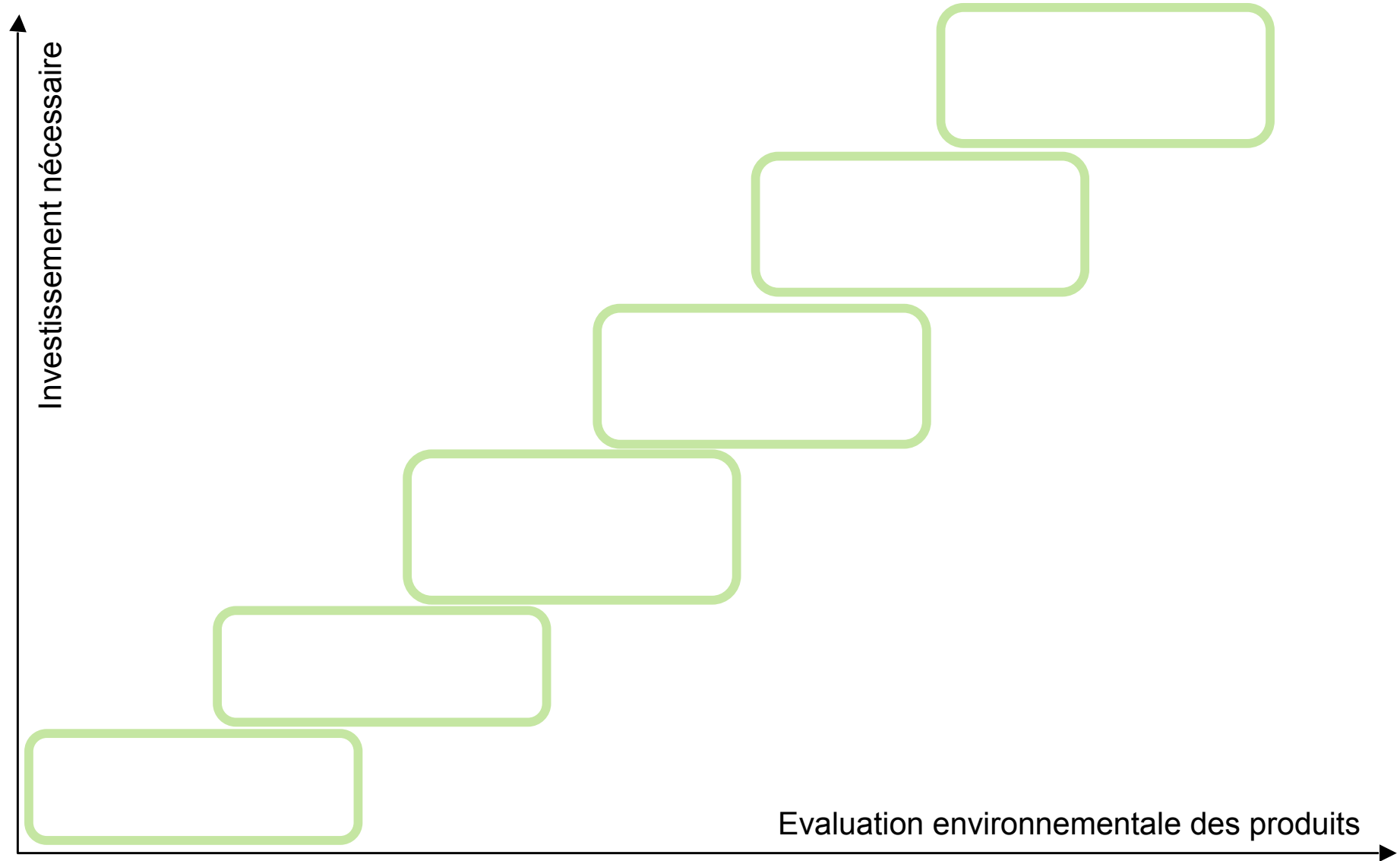
# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS D'ÉVALUATION DE L'IMPACT ENVIRONNEMENTALE



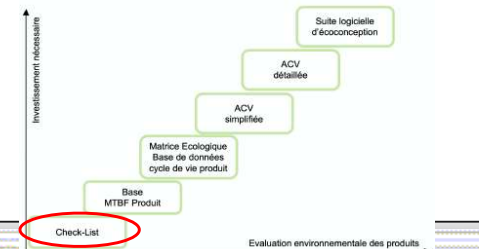
# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS D'ÉVALUATION DE L'IMPACT ENVIRONNEMENTALE



# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS : CHECK-LIST



## Check-list Généraliste : exemple Check-List Green-IT

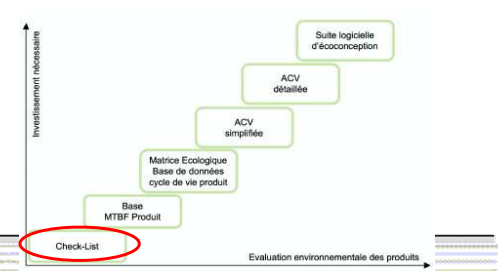


<https://club.greenit.fr/doc/2017-12-ClubGreenIT-RefGIT-checklist.v2.pdf>

ID	Périmètre	Domaine	Bonne pratique	Priorité
625	3. Entreprise	8. Logiciels	Entretien des ordinateurs pour éviter qu'ils ralentissent et deviennent instables	1
626	3. Entreprise	8. Logiciels	Désinstaller régulièrement les logiciels inutilisés	1
627	3. Entreprise	8. Logiciels	Mettre à jour les logiciels uniquement lorsque c'est indispensable	1
415	3. Entreprise	9. Services numériques et applications métier	Mettre en œuvre les bonnes pratiques d'accessibilité numérique	2
622	3. Entreprise	9. Services numériques et applications métier	Optimiser les états et sorties d'impression	3
623	3. Entreprise	9. Services numériques et applications métier	Mettre en œuvre les bonnes pratiques d'écoconception	1
624	3. Entreprise	9. Services numériques et applications métier	Privilégier une architecture applicative modulaire	1
628	3. Entreprise	9. Services numériques et applications métier	Systematiser la revue de code en sortie de développement	2
223	3. Entreprise	10. Centres informatiques	Exiger une efficacité énergétique minimum pour les équipements non IT des salles	3
224	3. Entreprise	10. Centres informatiques	Optimiser l'architecture et l'agencement des salles	1
227	3. Entreprise	10. Centres informatiques	Adapter l'architecture physique des serveurs à leur usage	1
231	3. Entreprise	10. Centres informatiques	Réutiliser l'énergie thermique produite par le data center	3
250	3. Entreprise	10. Centres informatiques	Mettre en place un suivi régulier des indicateurs énergétiques du datacenter	2
312	3. Entreprise	10. Centres informatiques	Privilégier les opérateurs qui ont ratifié le code européen de bonne conduite pour les datacenters	2
517	3. Entreprise	10. Centres informatiques	Confiner les baies des salles serveurs	2
631	3. Entreprise	10. Centres informatiques	Privilégier les équipements IT conformes aux exigences ASHRAE	1

# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS : CHECK-LIST




Check-List

## Check-list spécialisées

### Exemple éco-conception Web

115 bonnes pratiques eco-conception pour le Web

 [https://collectif.greenit.fr/ecoconception-web/115-bonnes-pratiques-eco-conception\\_web.html](https://collectif.greenit.fr/ecoconception-web/115-bonnes-pratiques-eco-conception_web.html)

Ecometer :

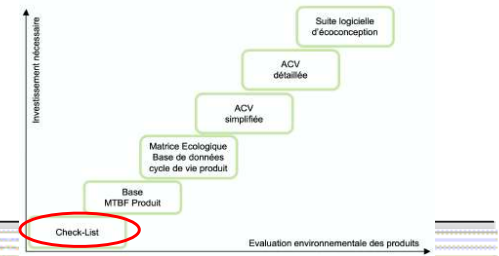
- ✓ [Best Practices for web pages](#)
- ✓ <http://www.ecometer.org/rules/>



N°	Rubrique	Description
1	Conception fonctionnelle	Éliminer les fonctionnalités non essentielles
2	Conception fonctionnelle	Quantifier précisément le besoin
3	Conception fonctionnelle	Fluidifier le processus
4	Ergonomie	Préférer la saisie assistée à l'autocomplétion
5	Ergonomie	Favoriser un design simple, épuré, adapté au web
6	Ergonomie	Créer un site responsive
7	Ergonomie	Respecter le principe de navigation rapide dans l'historique
8	Conception fonctionnelle	Proposer un traitement asynchrone lorsque c'est possible
9	Conception technique	Limiter le nombre de requêtes HTTP
10	Conception technique	Stocker les données statiques localement

# L'ÉCOCONCEPTION EN PRATIQUE

OUTILS : CHECK-LIST -- SITE WEB « RESPONSIVE DESIGN »



## ➔ Adaptation du contenu des pages du site en fonction du contexte de visualisation

ordinateur de bureau, tablette Wi-Fi, smartphone EDGE, 3G,4G,5G...

=> Réduction de la bande passante consommée, tout en améliorant le service rendu au client en lui offrant une expérience utilisateur enrichie quelque soit le terminal utilisé, cette approche.

## ➔ Images optimisées

afin de réduire autant que possible leur taille.

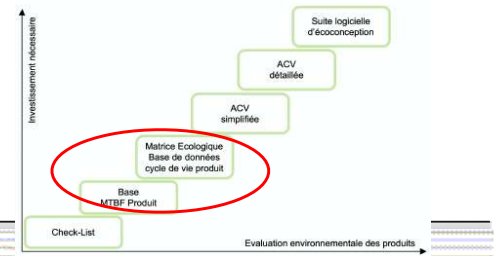
## ➔ Codes CSS de mise en forme et JavaScript non embarqués dans le code HTML de la page

=> réutilisation du code sur l'ensemble des pages, celui-ci n'étant pas re-transféré pour chaque page demandée.

Codes CSS et JavaScript inclus dans leurs propres fichiers permet au navigateur de les stocker dans le système de cache local afin de ne pas les redemander systématiquement, ce qui réduit les requêtes réseau.

# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS : BASES DE DONNÉES PRODUIT



### Vérifier la longévité de la solution

MTBF (Mean Time Between Failures), durée moyenne entre 2 défaillances

et MTTF (Mean Time To Fail), durée moyenne de bon fonctionnement, équivalent du MTBF pour les entités non réparables

*Exemple, disque dur standard : MTBF > à 100 000 heures = 11 sans panne. Le solid-state drives de Intel est donné avec un MTBF de 1,2 million d'heures.*

**Guide et BD :** MIL-HDBK-217 Notice 1 & 2, RDF 93, RDF 2000, FIDES 2004 A, ...



### Vérifier la consommation des ressources

TOP500 GREEN List : <https://www.top500.org/lists/green500>

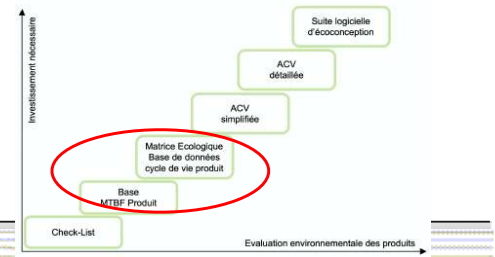
classement des superordinateurs les plus économes en énergie au monde

Green500 Data

Rank	TOP500 Rank	System	Cores	Rmax (TFlop/s)	Power (kW)	Power Efficiency (GFlops/watts)
1	301	<b>MN-3</b> - MN-Core Server, Xeon Platinum 8260M 24C 2.4GHz, Preferred Networks MN-Core, MN-Core DirectConnect, Preferred Networks Preferred Networks Japan	1,664	2,181.2	55	39.379
2	291	<b>SSC-21 Scalable Module</b> - Apollo 6500 Gen10 plus, AMD EPYC 7543 32C 2.8GHz, NVIDIA A100 80GB, Infiniband HDR200, HPE Samsung Electronics South Korea	16,704	2,274.1	103	33.983

# L'ÉCOCONCEPTION EN PRATIQUE

## CAS CONCRET : CHOIX DE SOLUTION UNITÉ CENTRALE



### → Unité centrale Raspberry Pi

- ✓ Faible coût (une trentaine d'euros pour la version 512 Mo)
- ✓ Solutions applicatives “smart” domotique, immotique, robotique,...
- ✓ Alimentation électrique 5V-700mA - Consommation < 4 W (en condition les plus énergivores)

### → Consommation électrique annuelle

- ✓  $[\text{nombre d'heures d'utilisation}] \times [\text{nombre jours d'utilisation}] \times ([\text{puissance appareil en watts}] / 1000) = \text{nombre kWh}$
- ✓ Heures creuses : 22h - 6h soit 8h/jour  $\Rightarrow 8 \times 365 \times 4 / 1000 = 11,68 \text{ kWh}$  Heures pleines : 6h - 22h soit 16h/jour  $\Rightarrow 16 \times 365 \times 4 / 1000 = 23,36 \text{ kWh}$  **Total : 11,68 + 23,36 = 35,04 kWh/an**

*Tarif à ajuster en fonction des conditions contractuelles*

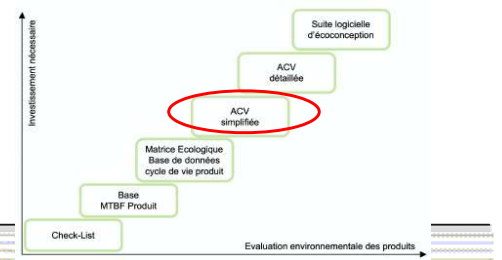
<u>Puissance souscrite (kVa)</u>	<u>Réglage disjoncteur (A)</u>	<u>Heures pleines pour 1kWh (€)</u>	<u>Heures creuses pour 1kWh (€)</u>
6	30	0,1467	0,1002

### → Coût à l'année

- ✓ Le coût énergétique 24h/24h pendant 365 jours de l'unité centrale Raspberry PI est de l'ordre de **4€ à 5€ à l'année**.

# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS ACV



➔ Base de données pour l'analyse du cycle de vie (ACV) des produits.



**Ecoinvent** : c'est la base de référence la plus fournie ;



**Base IMPACTS** : la base française gratuite, régulièrement mise à jour;



**GaBi** : en plus de l'édition logiciel, Gabi dispose de leur propre base avec de plus en plus d'éléments agrégés comme des caméras, micros, etc.



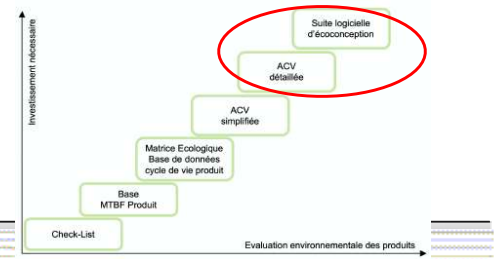
**Agribalyse** : la base européenne gratuite, projet abandonné en 2019

Théoriquement, il est possible d'effectuer une ACV à la main si on dispose de l'accès aux bases de données.

On peut alors utiliser un tableur pour organiser les flux.

# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS : LOGICIELS D'ÉCOCONCEPTION



- ➔ Effectuer une analyse du cycle de vie d'un produit manuellement est extrêmement chronophage.

**Les entreprises qui font de l'écoconception un axe stratégique investissent souvent dans des suites logicielles ACV adaptées.**

- ➔ Exemple de logiciels d'écoconception :



EcoInfo : <https://ecoinfo.cnrs.fr/ecodiag-calcul/>

Outil simple de calcul de consommation électrique d'équipements, impacts liés à la fabrication et transport des équipements achetés.



GaBi LCA

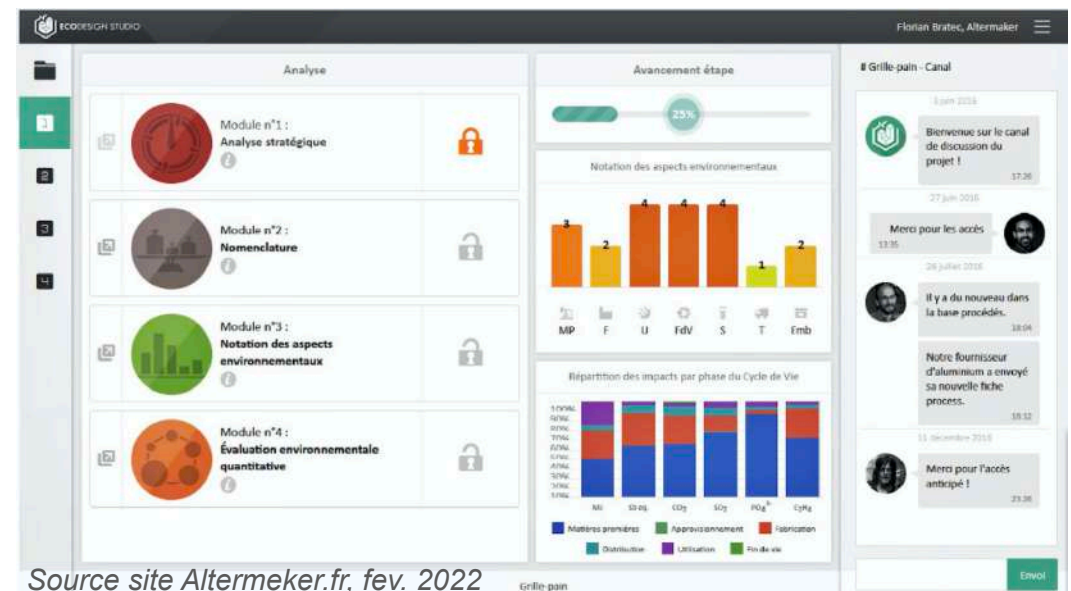


openLCA



Ecodesign Studio

**Outils logiciel  
d'Analyse de  
Cycle de Vie**



Source site Altermeker.fr, fev. 2022

## MISE EN ŒUVRE DE L'ÉCOCONCEPTION : OUTILS PRATIQUES POUR LES DÉVELOPPEURS

- Outils d'évaluation de l'impact Conception et environnementale
- Outils de conception et développement logiciel

# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## PHASE PRÉLIMINAIRE D'ANALYSE

Initier

Sélection d'un service numérique  
Quels sont les enjeux prioritaires associés à l'écoconception ?  
Comment ce projet s'intègre dans la stratégie d'entreprise?  
L'entreprise est-elle prête à consacrer des ressources sur ce projet?

Porter

Identifier qui va porter le projet en interne

Mobiliser

Identifier quels acteurs vont participer au projet en interne et en externe.  
Mobiliser les équipes concernées et impliquer les acteurs de la chaîne de valeur

Evaluer

Décrire le service numérique, sa ou ses fonctionnalités, définition de l'unité fonctionnelle retenue, du périmètre, de l'usage.  
Description du cycle de vie du service numérique  
Quantification des impacts environnementaux

Analyser

Identification des points les plus impactants qui seront à améliorer:

- Quelles étapes du cycle de vie?
- Quels « tiers » physique (terminaux, réseaux télécom, datacenter)
- Quels impacts? (indicateurs)

Imaginer

Imaginer des pistes d'amélioration et étudier la faisabilité:

- à partir des référentiels de bonnes pratiques ;
- à partir des objectifs de l'entreprise ;
- à partir d'une consultation interne & externe auprès des acteurs de la chaîne de valeur (brainstorming) ;
- à partir de conseils externes

# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## OUTILS DE CONCEPTION



### Eviter l'obésiciel

- ✓ Maitriser le nombre de fonctionnalités
- ✓ Limiter la richesse des IHM
- ✓ Concevoir l'ensemble du workflow
- ✓ Alerter sur les fonctionnalités à fort impact sur l'environnement

⇒ **L'utilité de chaque unité fonctionnelle doit être justifiée**

*Exemple d'outils : UML*

*Diagramme de Cas d'utilisations*

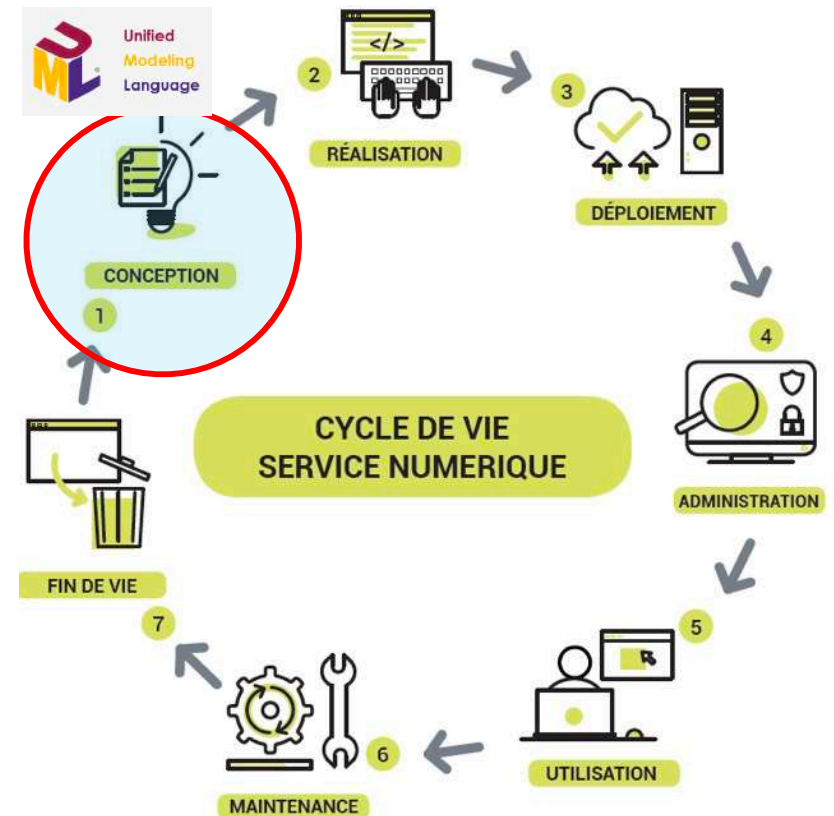


### Eviter les impacts en chaîne

- ✓ Maitriser l'infrastructure logicielle ou matérielle nécessaire
- ⇒ **L'utilité des évolutions sur les infrastructures d'accueil du service logiciel doit être surveillée attentivement**

*Exemple d'outils : UML*

*Diagramme de déploiement*

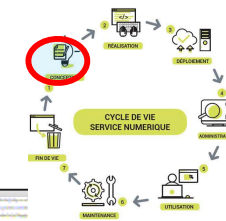


### Pensez à l'efficacité énergétique de l'architecture d'accueil

- ✓ Cibler un OS svelte et standard
- ✓ Penser au partage (cloud,...)

# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## CONCEPTION ET PARTAGE DE RESSOURCES



- **Le cloud initialement épinglé pour son caractère énergivore**
  - ✓ inquiétude de l'énorme hausse de la consommation énergétique associée à au déploiement de solution type Cloud.
  - ✓ gros distributeurs (Amazon, Google et Microsoft) pointés du doigt.
  
- **Le cloud devenu « Champion » de l'énergie renouvelable**
  - ✓ Les Serveurs Cloud sont désormais les plus gros clients d'énergie renouvelable
  - ✓ Google, qui couvre 100 % de sa consommation avec de l'électricité renouvelable, et dit être neutre sur le plan carbone depuis 2017. Microsoft est lui aussi neutre en carbone, mais il ne devrait se fournir uniquement en énergie renouvelable qu'à partir de 2025, comme Amazon.
  
- **Efficacité énergétique supérieure du Cloud**
  - ✓ regroupement de services similaires de plusieurs organisations utilisatrices.
  - ✓ taux d'utilisation supérieure, sans que cela ne provoque pas une hausse proportionnelle de la consommation électrique.
  - ✓ gros datacenters professionnels consomment relativement moins d'énergie que les centres plus petits et les datacenters d'entreprise.

# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## OUTILS DE CODAGE



### Réutiliser des briques logicielles

- ✓ Externe : ne pas réinventer la roue
- ✓ Interne : développement modulaire
- ✓ utiliser des briques logicielles portables optimisées invoquant les couches basses ou routines des noyaux des systèmes.



### Accroître la durée de vie des modules

- ✓ Respect de standards d'interfaçage
- ✓ Utilisation de technologies ouvertes



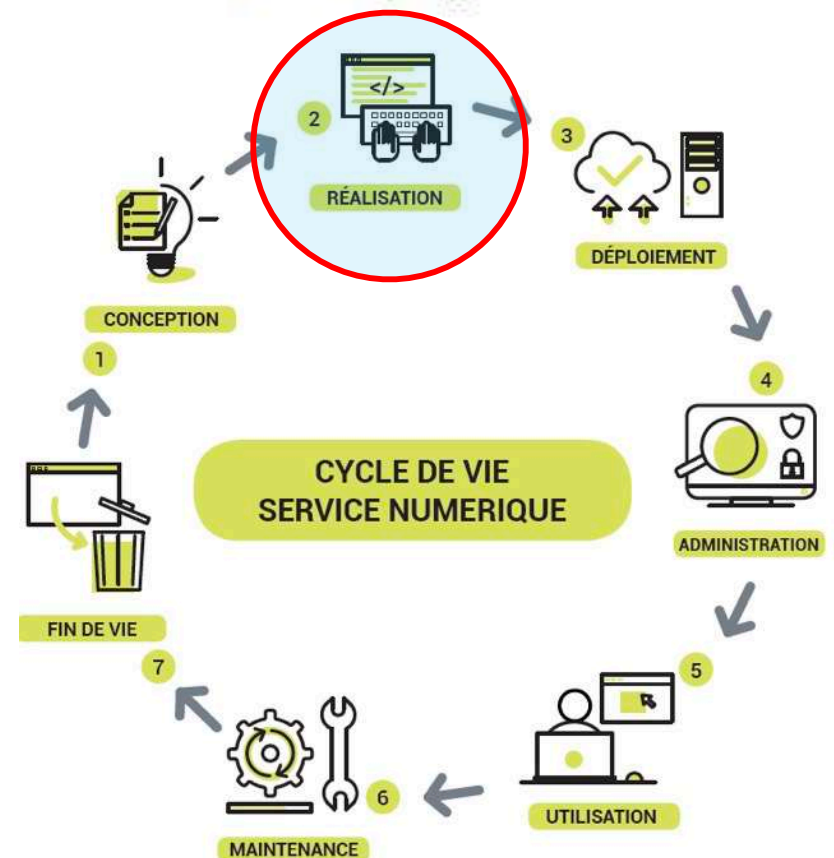
### Contribuer aux bien communs

- ✓ Distribution de modules en licence libre



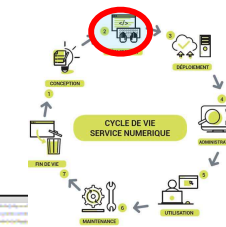
### Bien choisir la « pile logicielle »

- ✓ langage non gourmand en ressource,
- ✓ langage compilé (natif) à privilégier pour haute performance ou temps réel
- ✓ langage facile d'accès (interprété) à privilégier pour les traitements moins contraints, facilite la maintenance, le ré-usage et ainsi la durabilité
- ✓ Multiplicité des langages si nécessaire



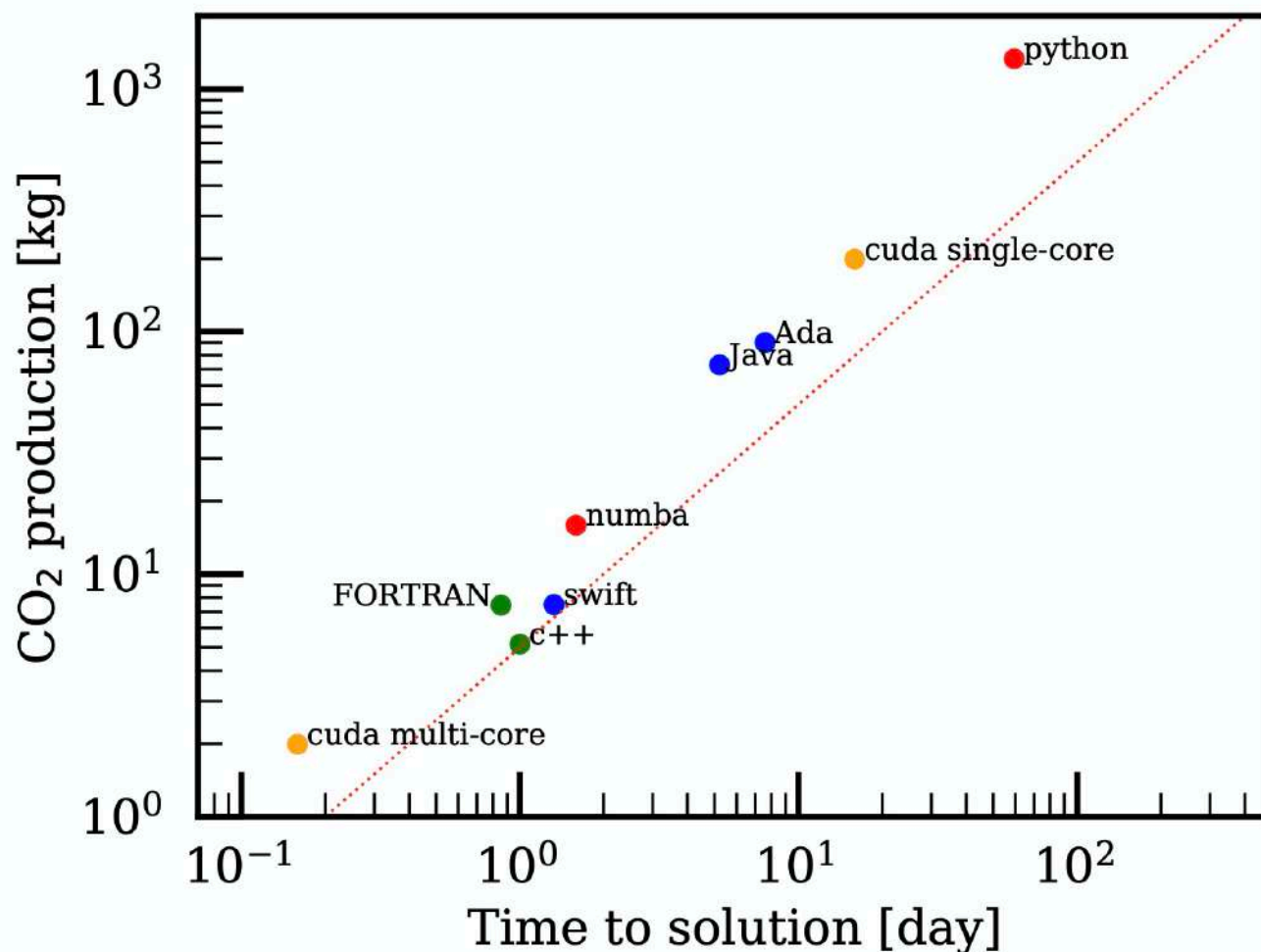
# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## OUTILS DE CODAGE



Efficacité énergétique en fonction du langage

Ref. Simon P. Zwart, The Ecological Impact of High-performance Computing in Astrophysics, Nature Astronomy) avec une simulation N-Body



**Cherches des alternatives aux langages peu efficace :**

**Par exemple il existe d'excellentes alternatives à Python :**



Utiliser des bibliothèques performantes, telles que NumPy et Numba.



⇒ Utiliser des langages fortement typés similaires à Python, comme Alice, Julia, Rust et Swift. Ces langages offrent la flexibilité de Python mais avec les performances du C++ compilé.

# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## OUTILS DE CODAGE

Source : Rui Pereira et al, SLE17 basé sur le « Computer Language Benchmark Game »).

<https://github.com/greensoftwarelab/Energy-Languages>

Time & Memory	Energy & Time	Energy & Memory	Energy & Time & Memory
C • Pascal • Go	C	C • Pascal	C • Pascal • Go
Rust • C++ • Fortran	Rust	Rust • C++ • Fortran • Go	Rust • C++ • Fortran
Ada	C++	Ada	Ada
Java • Chapel • Lisp • Ocaml	Ada	Java • Chapel • Lisp	Java • Chapel • Lisp • Ocaml
Haskell • C#	Java	OCaml • Swift • Haskell	Swift • Haskell • C#
Swift • PHP	Pascal • Chapel	C# • PHP	Dart • F# • Racket • Hack • PHP
F# • Racket • Hack • Python	Lisp • Ocaml • Go	Dart • F# • Racket • Hack • Python	JavaScript • Ruby • Python
JavaScript • Ruby	Fortran • Haskell • C#	JavaScript • Ruby	TypeScript • Erlang
Dart • TypeScript • Erlang	Swift	TypeScript	Lua • JRuby • Perl
JRuby • Perl	Dart • F#	Erlang • Lua • Perl	
Lua	JavaScript	JRuby	
	Racket		
	TypeScript • Hack		
	PHP		
	Erlang		
	Lua • JRuby		
	Ruby		

Considérer les impacts  
en énergie, mémoire,  
temps d'exécution dans  
le choix de son langage




# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES



## OUTILS DE CODAGE

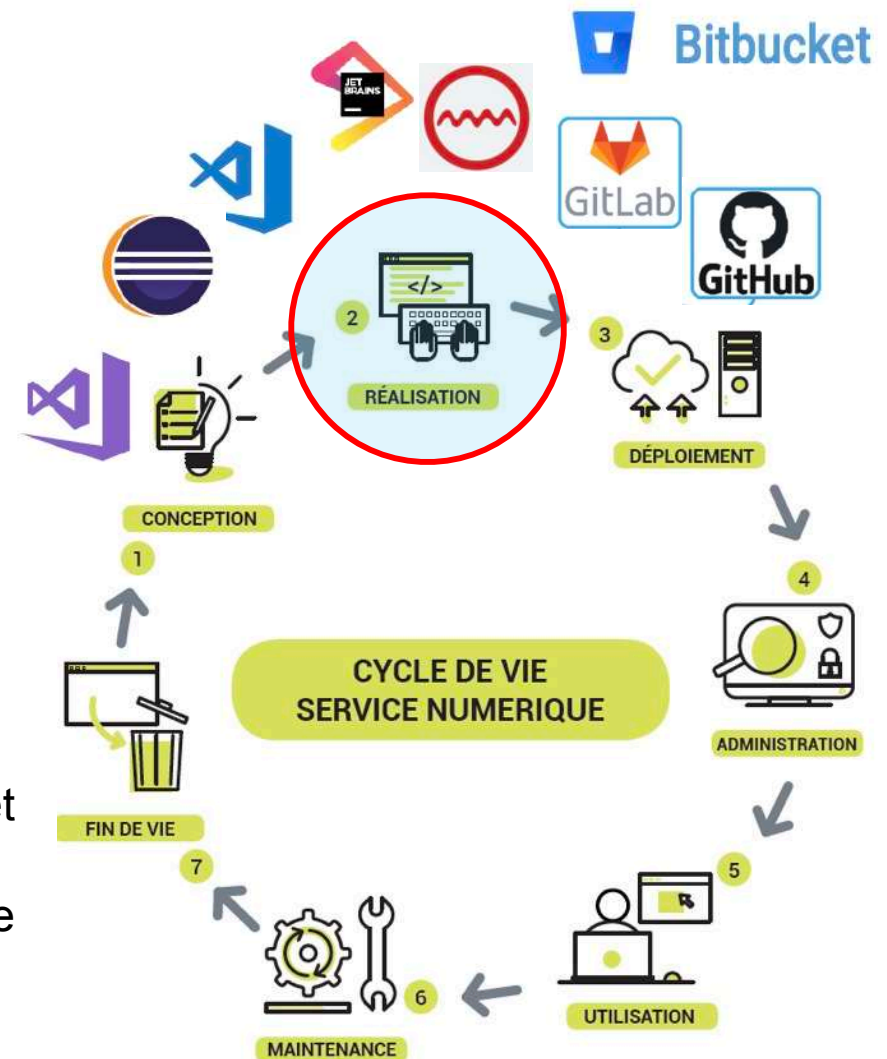
### Bien choisir sa pile « d'outils de développement »

#### → « Clean Code starts in your IDE »

-  Choisir un environnement de développement intégré (IDE) le plus adapté possible à votre langage
  - ✓ Exemple d'IDE : VS Code, Eclipse, Visual Studio, JetBrains, SonarLint.

#### → « Better code starts with pull requests »

- ✓ Gestion de version (oui mais...)
-  Eviter ou limiter le stockage des binaires et les jeux des données non indispensables
-  Ne pas stocker les produit intermédiaire de compilation ni les fichiers de sortie
  - ✓ Exemple GitLab, GitHub, Bitbucket, ...



# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## OUTILS DE TEST

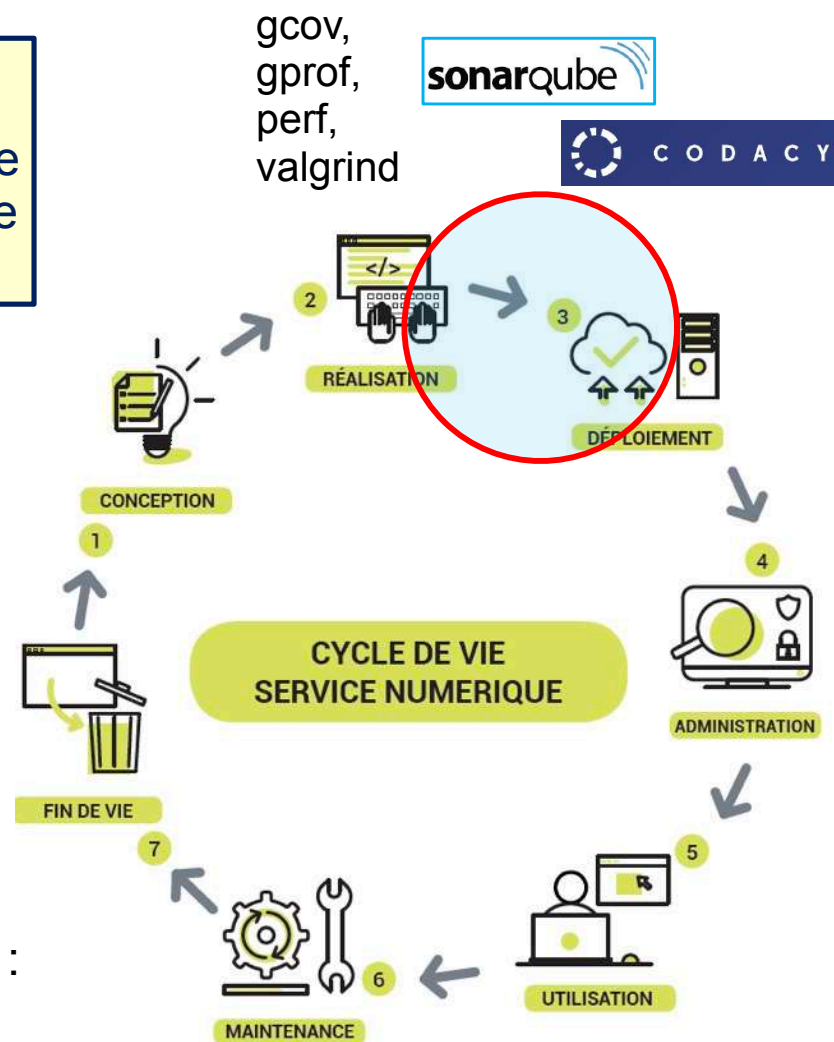
### Logiscope

Outil permettant de vérifier et d'évaluer le code source en calculant mesure de grandeur sur le code d'un logiciel.

#### → Mesures de la qualité du code :

- ✓ Complexité cyclomatique
- ✓ Mesures de la complexité d'Halstead
- ✓ Points de micro-fonction pondérés  
Weighted Micro Function Points (WMFP)

Mesure du nombre de lignes de codes, évaluation du nombre d'opérateurs et d'opérandes. Mesure du nombre cyclomatique : nombre de chemins indépendants dans les graphes de boucles et d'appels de fonctions



# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS DE TEST



## Métriques calculé par la méthode Points de micro-fonction pondérés **Weighted Micro Function Points (WMFP)**

Les éléments mesurés WMFP sont représentés en pourcentage de l'effort de l'unité entière (projet ou fichier) et sont traduits en temps.

- Complexité du flux (FC) - Mesure la complexité du chemin contrôle de flux d'un programme (complexité cyclomatique).
- Vocabulaire objet (OV) - Mesure la quantité d'informations uniques contenues dans le code source.
- Conjuraison d'objets (OC) - Mesure la quantité d'utilisation des informations contenues dans le code.
- Complexité arithmétique (AI) - Mesure la complexité des calculs arithmétiques
- Données transfert (DT) - Mesure la manipulation des structures de données
- Structure de code (CS) - Mesure l'effort consacré à la structure du programme, comme la séparation du code en classes et fonctions
- Données en ligne (ID) - Mesure l'effort consacré à l'incorporation de données codées en dur
- Commentaires (CM) - Mesure l'effort consacré à la rédaction des commentaires

# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS DE TEST



C:\Program ...mindReviewer.ttp - IBM Rational Logiscope Viewer - machine\_update\_scores - Source - machine.c

File Edit Select Navigate View Options Window Help

Workspace1 - Call gra

- consistent
- end\_game
- find\_digit
- format\_output
- get\_code\_mac
- get\_code\_play
- help
- hi\_scores\_disp
- hi\_scores\_writ
- instruction
- machine\_plays
- machine\_print
- machine\_read
- machine\_upde
- make\_code
- master/main
- play
- player\_plays
- player\_score
- print\_help
- print\_instructio
- print\_instructio
- print\_score\_pl
- prompt
- refresh
- rest
- score\_mac
- score\_player
- set\_dummy

```

103     }
104     number [i] = car;
105     car = getc (hi_scores_file);
106     i++;
107 }
108
109     number [i] = '\0';
110     (hi_scores_tab [last_hi_score]).score =
111     free (number);
112
113     car = getc (hi_scores_file);
114     last_hi_score++;
115 }
116

```

machine\_update\_scores - Relative call graph

machine\_update\_scores - Control Graph

machine\_update\_scores - Criteria Kiviat Graph

CRITERION	CLASS
TESTABILITY	FAIR
STABILITY	GOOD
CHANGEABILITY	GOOD
ANALYZABILITY	FAIR
SYNTHESIS	FAIR

Works...

Loading CodeReducer data...  
Loading RuleChecker data...  
+ loaded 55 function(s)  
+ loaded 0 class(es)

Press F1 for Help

RECEPTION

# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS DE TEST



## Réaliser du profilage de code

Le profilage consiste à analyser l'exécution d'un logiciel afin de connaître son comportement à l'exécution.

- ➔ Exemple d'outils **d'analyses statiques** permettant, entre autres, de déterminer la difficulté de maintenance d'un logiciel :

- ✓ Sonarqube
- ✓ Codacy



- ➔ Exemple d'outils de **d'analyses dynamiques** permettant de quantifier l'usage des ressources

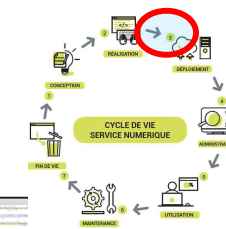
- ✓ gcov, gprof, perf, Valgrind, ...



**Ces analyses de logiciel, qui font partie des bonnes pratiques classiques, sont indispensables afin de réduire l'impact d'un service numérique.**

# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS DE TEST

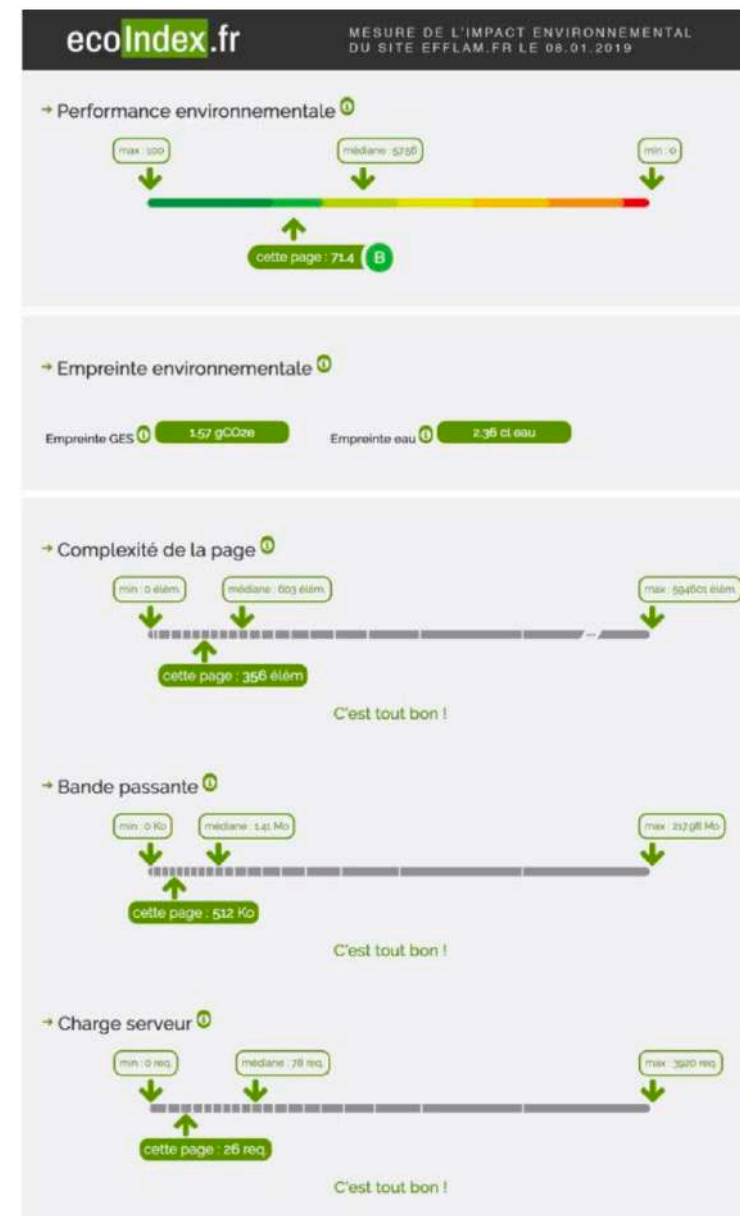


### → Mesure en fonctionnement :

- ✓ temps de chargement
- ✓ temps d'exécution
- ✓ consommation des protocoles réseau
- ✓ taille des données
- ✓ nombre de requêtes
- ✓ performance web
- ✓ consommation électrique

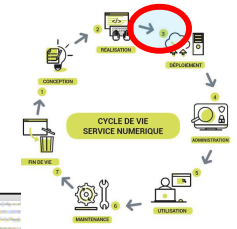
### → Exemples d'outils :

- ✓ Firefox web tools (trafic, requêtes, JavaScript),
- ✓ Apache JMeter (scénarios web),
- ✓ ecoindex (conception web),
- ✓ Cardonalysier (navigation web),
- ✓ Profiler, intégré et adapté au langage
- ✓ Watts'up Pro - PDU Eaton (Wattmètre pour mesurer la conso électrique)



# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS DE TEST



### ➔ Limitation d'impact :



Optimiser un logiciel peut induire à lancer davantage d'opérations ou traiter davantage de données, donc l'empreinte écologique du service ne sera pas réduite (Paradoxe de Jevons).



L'optimisation devrait servir simplement à réduire la consommation énergétique et des ressources, et si possible d'arriver plus vite au résultat. Chaque exécution a un impact !



Il est primordial de n'optimiser que ce qui a le plus d'impact (Loi de Pareto).

Il ne faut pour autant pas négliger la qualité des tests dans le processus d'optimisation (régressions).



Pour le codage web, utiliser des outils d'analyse automatique de vérification du codage pour les site web :

Ecometer : <http://www.ecometer.org>, [GreenIT-Analysis](#)

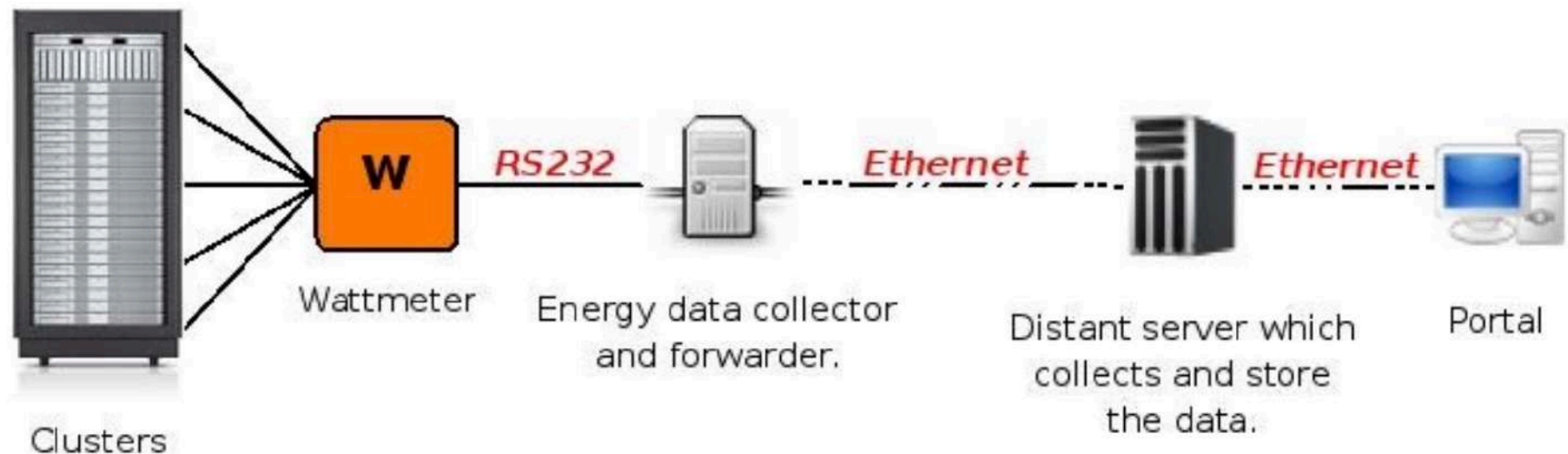
# L'ÉCOCONCEPTION EN PRATIQUE

## OUTILS DE TEST



### Test de l'usage de ressources dans mon logiciel : Chacun de ces usages provoque une consommation électrique.

- Mesurer l'impact énergétique des logiciels en phase d'usage : besoin d'équipements matériels (PDU, wattmètres) ou de sondes logicielles.
- Ressources : processeur, mémoire, espace de stockage, requête réseau, ...



# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES DÉPLOIEMENT ET ADMINISTRATION



Les pratiques actuelles impliquent :

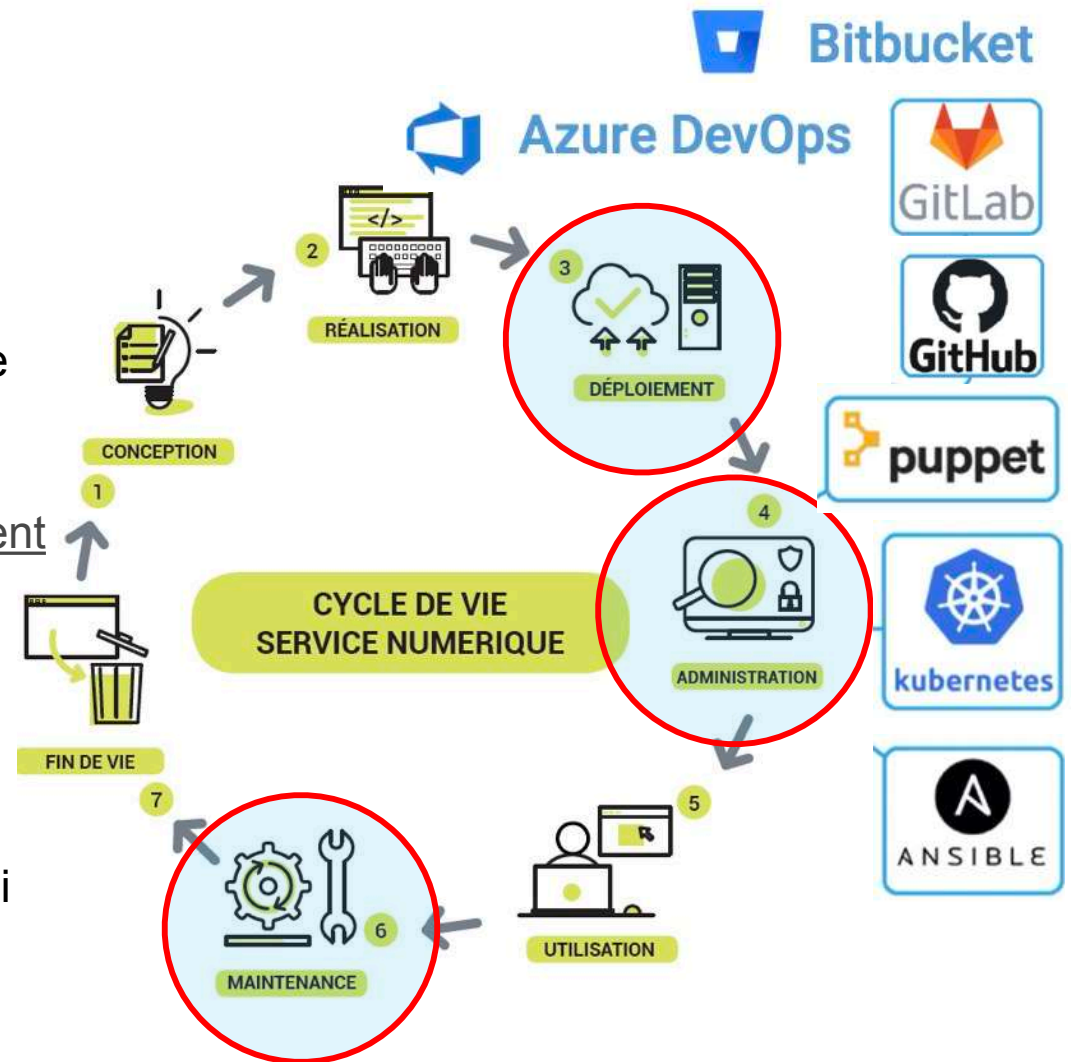
- ✓ développement continu,
- ✓ le test continu,
- ✓ l'intégration continue,
- ✓ le déploiement continu
- ✓ la surveillance continue

des applications logicielles tout au long de leur cycle de vie.

Cette pratique de l'Intégration/déploiement continu  
(CI/CD) est au cœur du **DevOps**

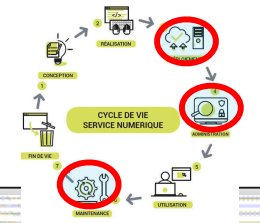
CD/CI : pratique positive vu de l'écoconception mais il faut contrôler aussi son impact.

!










# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## DÉPLOIEMENT ET ADMINISTRATION



### ➔ Limiter l'impact de l'Intégration Continue (CI/CD)

-  Choisir un docker de taille minimum,
-  Activer une CI uniquement sur certaines branches
-  S'assurer que tous les tests n'ont pas besoin d'être réalisés à chaque CI
-  Déployer les codes dans différents environnements prod/non-prod.
-  Créer des environnements de test parallèles (CI/CD).
-  S'assurer que l'on ne produit pas tous les fichiers à chaque modification
-  surveiller la durée des jobs, leur nombre, la taille des artefacts, le trafic réseau

# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## UTILISATION ET MAINTENANCE

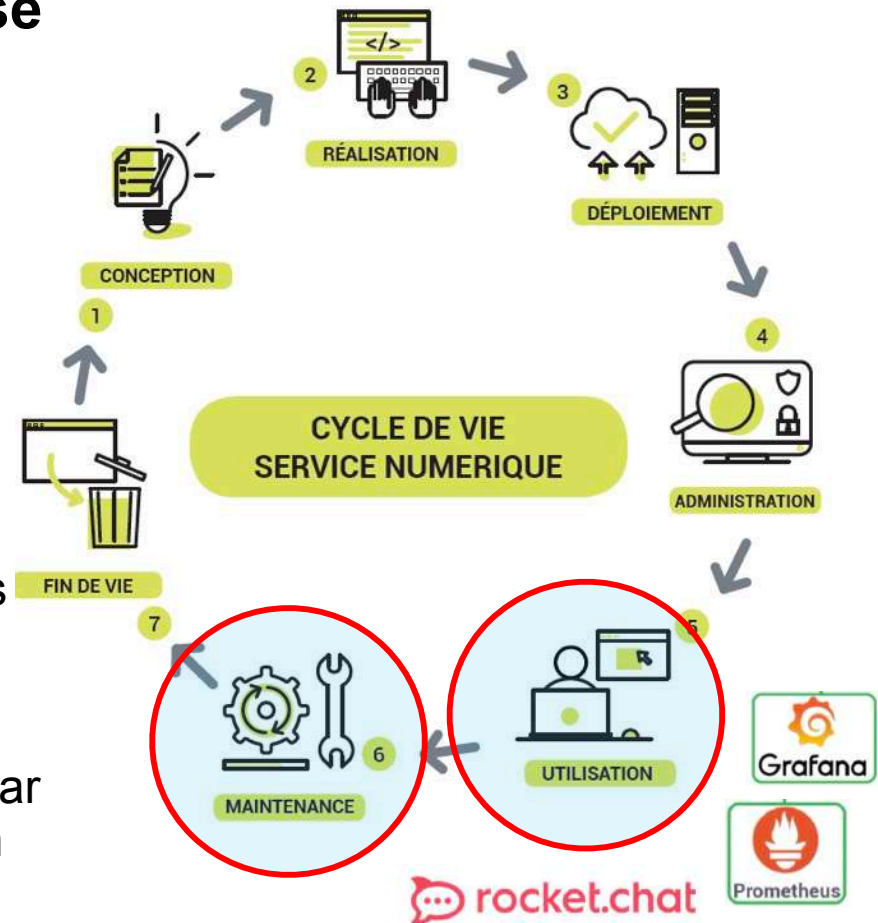
### Mise en place de systèmes de mise en veille et de communication

Créer des tableaux de bord d'indicateurs d'impact :

- ✓ Traquer les ressources inutiles
- ✓ Mesurer les flux réseaux en site
- ✓ Mesurer Les consommations énergétiques réelles
- ✓ Traquer les usages pour détecter les unités fonctionnelles inutiles

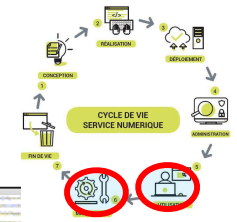
#### ➔ Converser avec les utilisateurs

- ✓ Converser de façon personnalisée par de multiples canaux de conversation tels que le chat en direct sur le site web, l'e-mail, la page Facebook, Twitter, WhatsApp, Instagram, etc.



# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

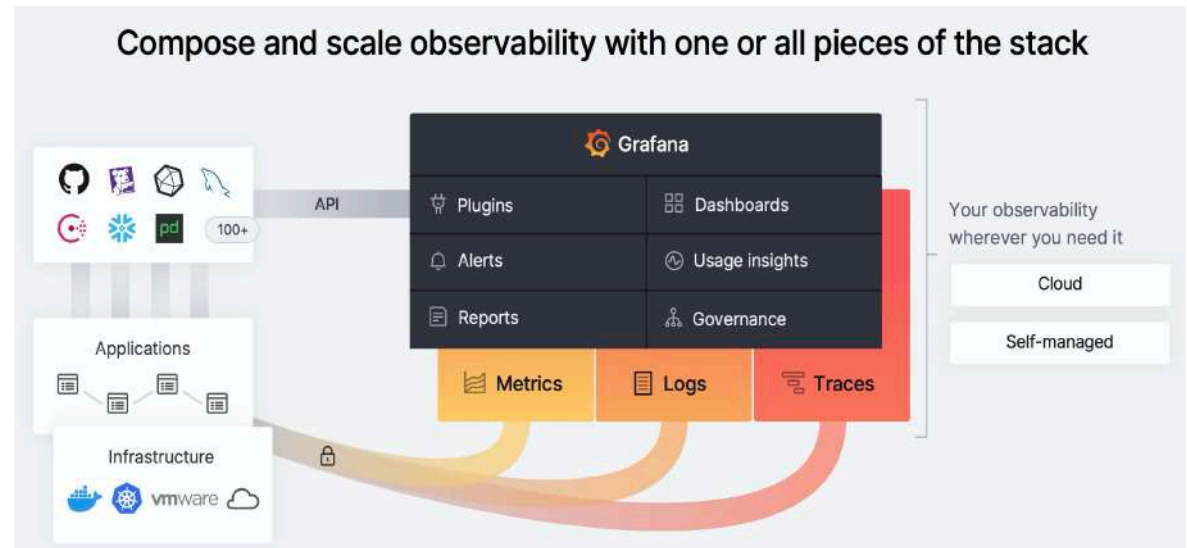
## MAINTENANCE ET EXPLOITATION



### Tableaux de bord de suivi d'un service numérique :

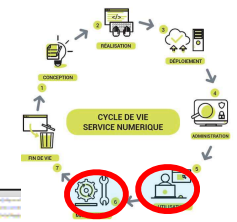
Visualisation, interrogation, analyse en temps réel de métriques d'exécution d'applications.

Exemple : Grafana



# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## MAINTENANCE ET EXPLOITATION

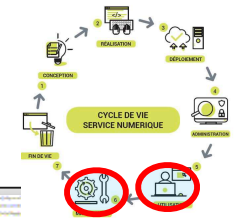


### Limitation d'impact de la phase d'exploitation maintenance :

- évaluer seulement les métriques pertinents
- réaliser le paramétrage des calculs de métriques par un expert
- exploiter la supervision (et les alertes) pour observer les pics CPU, ressources utilisées (disque, réseau), consommation électrique
- modifier le service numérique pour l'adapter en fonction de l'usage observé (amélioration continue)
- Réduire les fréquences et volumes des sauvegardes Mise en place de systèmes de mise en veille
- Eteindre les machines virtuelles avec empreinte mémoire et CPU
- limiter le nombre de services déployés
- favorise les extinctions d'hôtes et de ressources inutilisés

# L'ÉCOCONCEPTION DES SERVICES NUMÉRIQUES

## MAINTENANCE ET EXPLOITATION



### Et enfin...

Sensibiliser les utilisateurs de mon service numérique sur les impacts environnementaux de leurs usages



Participer à des actions de sensibilisation

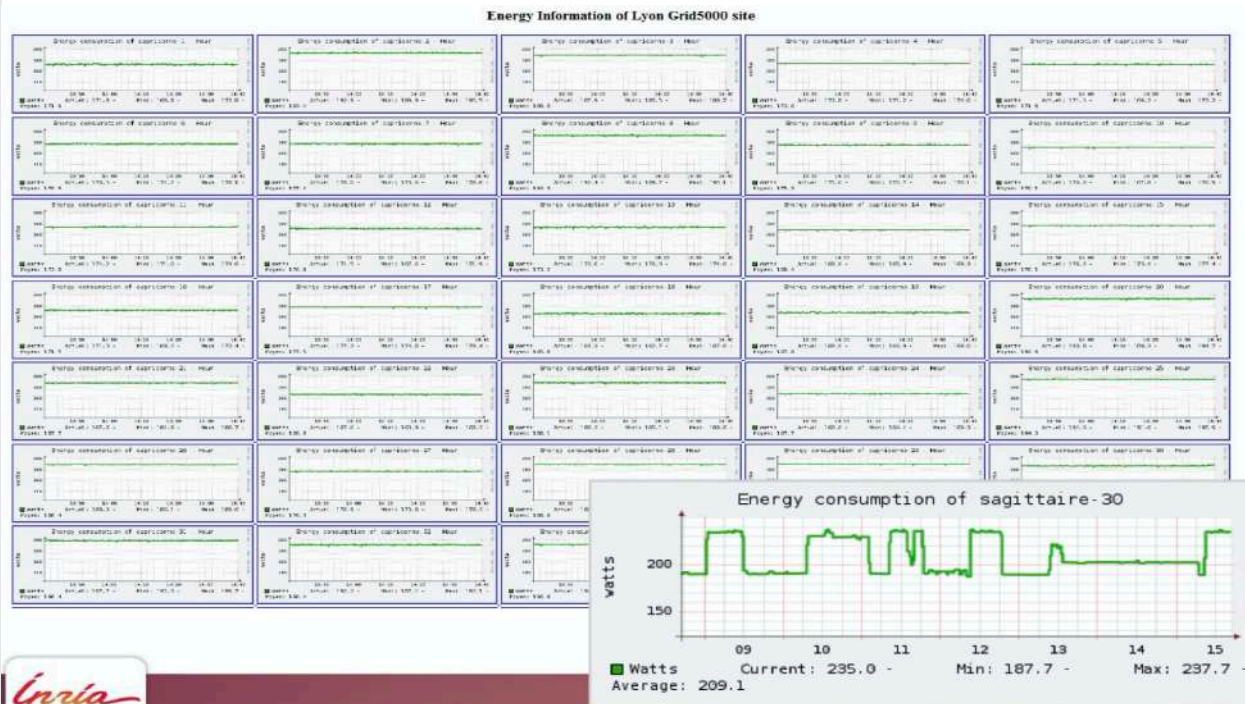


Afficher des informations de suivi de consommation

Participation de votre organisation au « **Cyber World CleanUp Day** »

- Journée de sensibilisation à l'empreinte environnementale du numérique par l'action,
- 19 mars 2022, et la semaine précédant cette date.
- L'objectif est de créer les conditions d'une prise de conscience globale de l'impact environnemental du numérique

<http://www.nexa.re/index.php?id=25&art=2333>



inria

# L'ÉCOCONCEPTION OUTILS

L'écoconception logiciel met en synergie de très nombreux outils participant à toutes les phases du cycle de vie d'un service numérique.

