

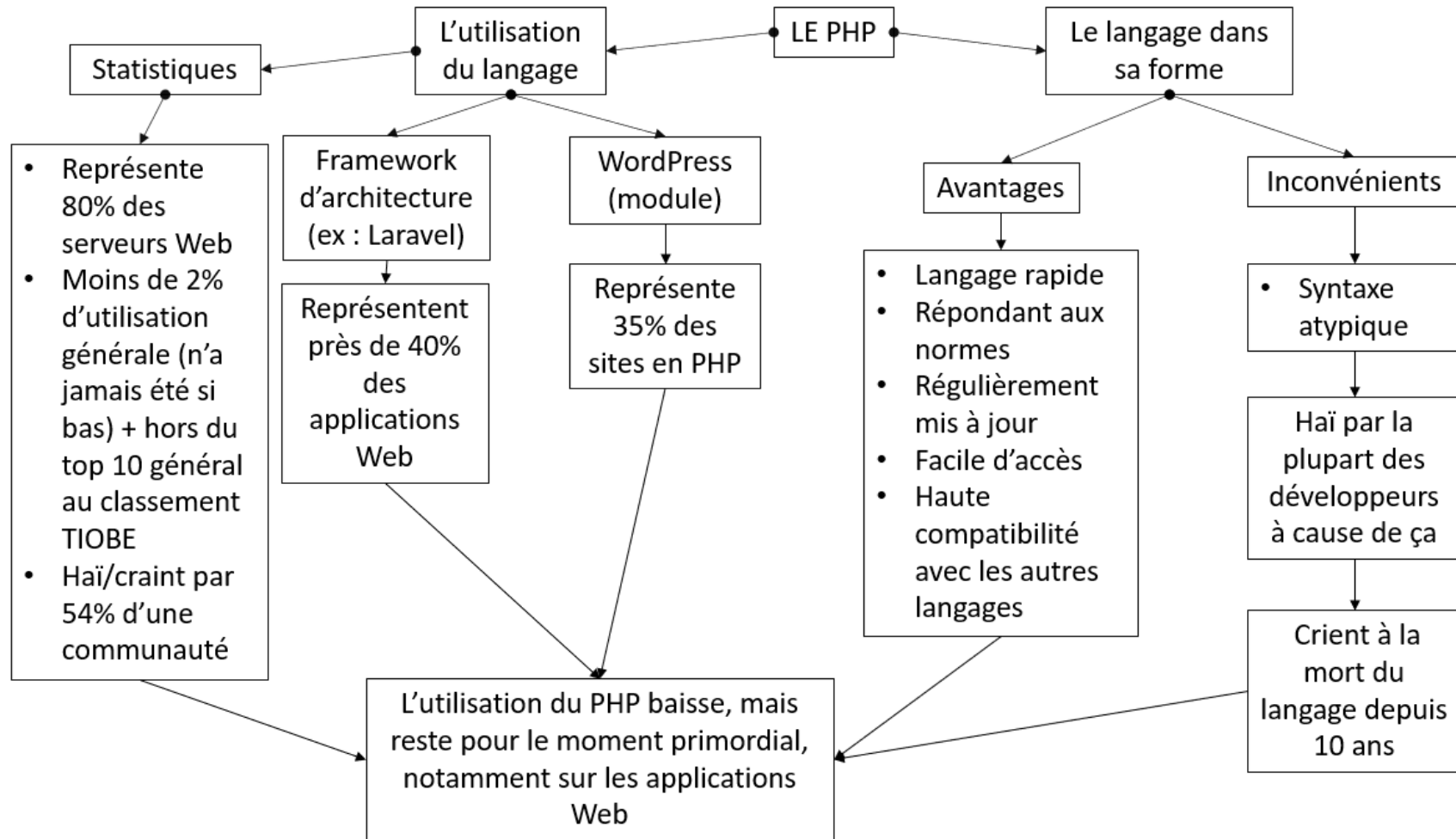
VEILLE INFORMATIONNELLE

L'extinction progressive du PHP

Table des matières

Carte mentale de préparation à la veille	1
Pourquoi cette veille ?	2
Justification économique	2
Justification de production	2
Justification managériale.....	2
Les version de PHP pur, son histoire, sa création	3
Présentation générale	3
Historique et dates importantes	3
L'utilisation du PHP, ses frameworks et modules	5
Le PHP est-il mort ?	7
Sitographie	8

Carte mentale de préparation à la veille



Pourquoi cette veille ?

Si l'on se penche sur l'utilisation du PHP, c'est qu'elle comporte des risques, il est à l'heure actuelle primordial dans le monde du développement, et notamment dans les applications Web. Mais qu'est-ce qui pourrait réellement rationaliser socialement cette veille ?

Justification économique

Tout d'abord, cette question peut se justifier économiquement en cas de disparition du PHP. Si cela vient à arriver, comment réagir ? Quel langage choisir pour le remplacer, les développeurs de l'entreprise sont-ils qualifiés pour ce nouveau langage, cette nouvelle technologie ? Quels seraient les coûts de formation de ces salariés, ou faudrait-il directement embaucher de nouveaux salariés ? De plus, le temps de réagir et de changer de plateforme, quelles seraient les pertes engendrées ? En bref, la situation économique soulève énormément de questions.

Justification de production

Ensuite, l'arrêt d'utilisation du PHP poserait de gros problèmes de production de l'entreprise, pour peu qu'elle ait une activité basée sur un site web. En effet, le temps de réfléchir, de penser à une nouvelle structure applicative dans un autre langage, le reste de l'équipe (commerciale etc...) ne peut plus faire grand-chose tant que le service web n'est pas rétabli.

Justification managériale

Enfin, il faudra savoir adopter la bonne méthodologie, la bonne gestion de l'équipe, en somme le bon management par le bon manager, pour garder une dynamique, et pour pouvoir faire face aux changements de manière efficace.

Les version de PHP pur, son histoire, sa création

Présentation générale

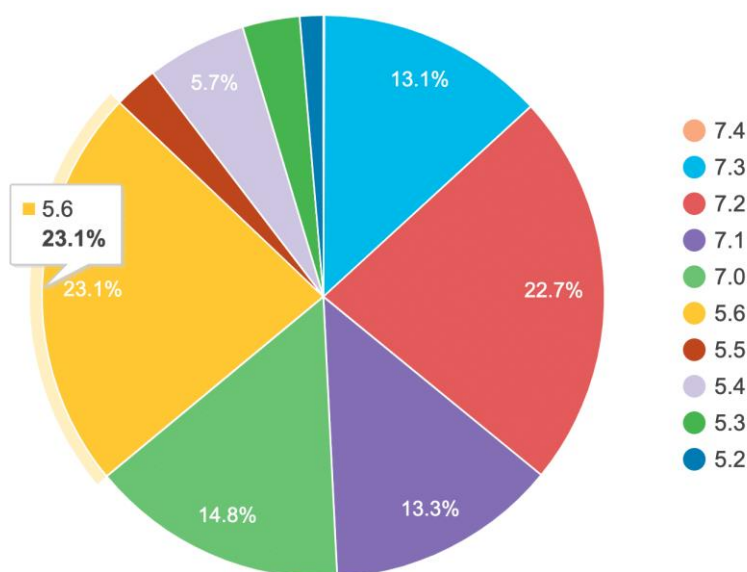
Le PHP, ou « Hypertext Processor » est un langage de programmation. Il est défini comme un langage impératif orienté objet, et principalement utilisé pour produire des pages Web dynamiques via un serveur utilisant un protocole HTTP.

Ce langage est actuellement derrière certains des sites les plus utilisés à l'heure actuelle comme Wikipédia et Facebook. C'est pourquoi le PHP est à juste titre considéré comme la base des sites Web dynamiques, mais surtout des applications Web.

Historique et dates importantes

- Le PHP à été créé en 1994 par Ramus Lerdorf pour son site web. Son but était juste de conserver les traces des utilisateurs qui voyaient son site, à partir d'une librairie en C.
- Son code source a donc été publié à partir de 1995 pour devenir accessible à tous.
- Son cœur a été retravaillé en 1997 par deux étudiants. En 1998, PHP3 était publié
- Le langage connaît depuis des améliorations et des mises à jour régulières, le rendant plus pratique et plus confortable depuis plus de 20 ans.
- Dès 2002, plusieurs millions de sites Web utilisent PHP, et il gagne en part de marché sur l'utilisation côté serveur atteignant aujourd'hui près de 80%.
- Les versions 5 & 7 sont considérées par les mises à jour majeures de PHP et les versions les plus utilisées comme en atteste ce graphique

PHP Versions



Depuis Septembre

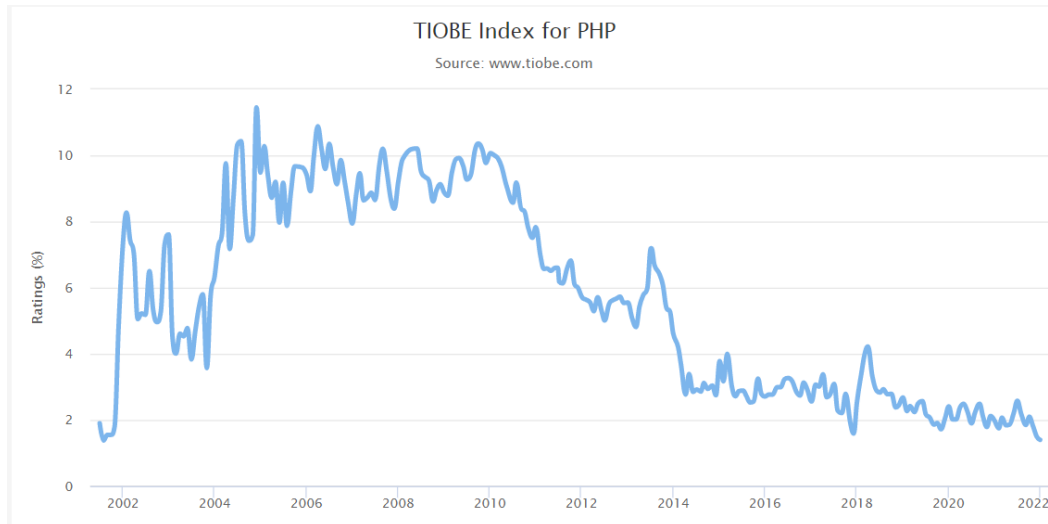
2021, de nombreuses versions ont été rendues publiques pour les versions 7.3, 7.4, 8.0 et une des dernières en date : 8.1

Son fonctionnement est expliqué par le schéma ci-dessous :



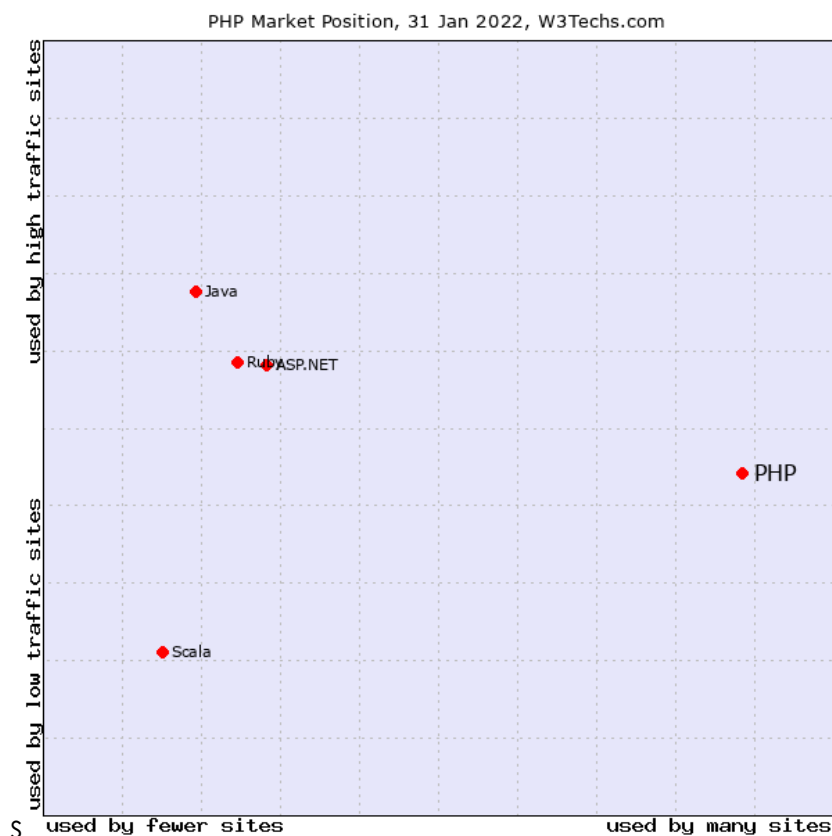
L'utilisation du PHP, ses frameworks et modules

Si l'on s'en réfère aux statistiques, l'utilisation du PHP fait débat. De manière générale, il n'a jamais été si peu utilisé comme en atteste ce graphique du site TIOBE :



Cependant, depuis près de dix ans, une grande partie de la communauté des développeurs parle régulièrement de la mort du PHP. Cela peut paraître pertinent quand l'on voit qu'il représente environ 2% des applications totales.

Par opposition, on peut aussi observer que le PHP est un langage utilisé, côté serveur, par près de 80% des applications Web existantes.



Cela peut s'expliquer par plusieurs points simples et cohérents. Car quand bien même la chute du PHP depuis plus de 10 ans est indéniable, on ne peut se reposer sur une unique statistique, il faut creuser. Et quand on creuse, on trouve que les sites les plus utilisés aujourd'hui utilisent PHP, notamment sur les principaux réseaux sociaux.

Mais ce qui fait également que le PHP est encore aujourd'hui utilisé, c'est grâce à ses différentes fonctionnalités uniques, comme les framework d'architecture que sont Laravel et Symfony par exemple. Ces framework permettent d'implémenter automatiquement un paterne MVC dans l'architecture de l'application web, et permet d'utiliser diverses technologies déjà implémentées.

Également, il se trouve que le module de conception de sites internet WordPress utilise également du PHP.

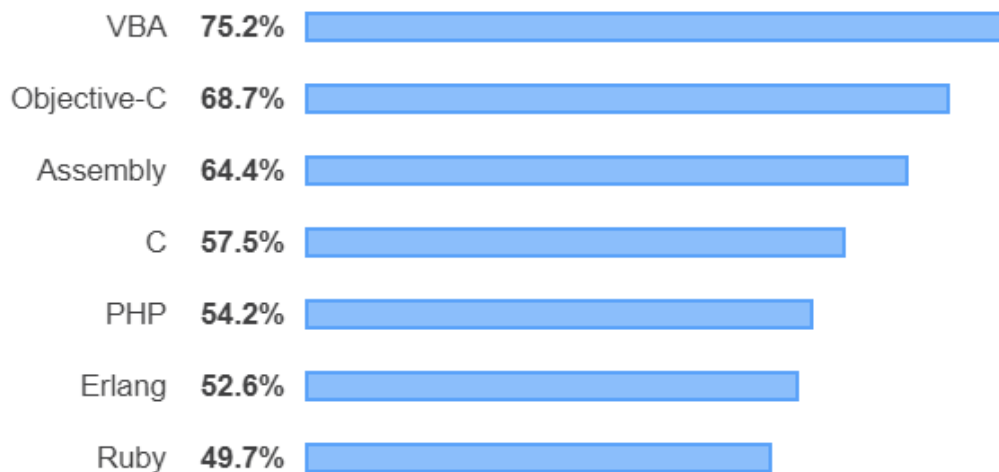
Tous ces avantages font que le PHP existe toujours et est encore bien plus utilisé que l'on ne pense. Mais au-delà de ses atouts, le PHP possède également des qualités intrinsèques indéniables :

- Il est accessible à tous
- Il est facile d'apprentissage, utilisable par des débutants comme des utilisateurs confirmés
- C'est un langage encore beaucoup demandé au sein des entreprises.
- Son exécution est rapide
- Il est compatible avec de nombreux langages comme le balisage HTML et le JavaScript.
- Répond à de nombreuses normes de développement grâce à ses mises à jour régulières.

Toutes ces qualités au PHP en font encore à ce jour un langage extrêmement utilisé, pratique et nécessaire pour beaucoup.

Le PHP est-il mort ?

Mais malgré toutes ces qualités, le PHP fait encore et toujours débat au sein de la communauté des développeurs. Selon un sondage réalisé par la plateforme StackOverflow, le PHP figure dans les 5 langages les plus haïs des développeurs.



Cette haine envers le langage réside notamment dans sa syntaxe atypique, ne ressemblant à aucun autre langage, lui valant de réguliers déferlements de haine, pas toujours justifiée. Cette haine du PHP a fait que nombre de développeurs crient à sa mort et à son extinction depuis près de 10 ans au fur et à mesure que son utilisation décroît jour après jour.

Mais qu'en est-il réellement, le PHP peut-il vraiment s'éteindre d'un jour à l'autre ? Si oui, qu'est-ce qui pourrait le remplacer ?

Nous avons pu constater par l'intermédiaire de statistiques que oui, le PHP était de moins en moins utilisé. Cependant, nous avons pu établir le fait que ce langage, autant utilisé que haï disposait de nombreuses qualités et apports essentiels pour l'heure irremplaçables. Il faudrait alors se pencher sur plusieurs hypothèses que certains framework d'autres langages pourraient remplacer le PHP. Si cela venait à arriver, qu'advierait-il du PHP ? Serait-il peu à peu remplacé, ou, à l'image du Pascal il y'a quelques dizaines d'années, disparaîtrait-il sans prévenir ?

Aussi improbable que cela puisse paraître à l'heure actuelle, un crash est toujours imprévisible mais néanmoins tout autant possible, il faut donc rester méfiant, et être prêt à s'adapter aux nouvelles conditions.

Ainsi, le jour où le PHP s'éteindra complètement, le monde du développement, notamment pour les applications Web changera complètement tant le PHP est encore omniprésent.

Sitographie

<https://oclock.io/blog/5023/est-ce-que-php-est-encore-utile>

<https://kinsta.com/fr/blog/php-est-il-mort/>

<https://www.tiobe.com/tiobe-index/>

<https://w3techs.com/technologies/details/pl-php>

<https://insights.stackoverflow.com/survey/2019#most-loved-dreaded-and-wanted>

<https://www.miroir-mag.fr/actualites-high-tech/php-vs-java-quelle-est-la-meilleure-option-pour-le-developpement-web/>

Lien de mon portfolio : <https://galifire.github.io/> (vous pourrez trouver le résultat de ma veille dans la section veille)