



# Introduction à PHP

Caractéristiques principales

## CONTENU

PHP - Historique .....	1
Caractéristiques du langage PHP .....	1
License .....	1
Portabilité .....	1
Exécution .....	1
Apprentissage de PHP .....	2
Richesse du langage PHP .....	2
PHP, un langage fiable et performant .....	2
PHP, un langage conçu pour les applications web dynamiques .....	3
PHP pour d'autres domaines d'applications .....	4
Les limites de PHP .....	5
PHP, Un langage faiblement typé .....	5
PHP : documentation officielle .....	5

## PHP - HISTORIQUE

Le langage PHP a été inventé par Rasmus LERDORF en 1995 pour son usage personnel (mise en ligne de son CV en l'occurrence). Autrefois abréviation de Personal HomePage devenue aujourd'hui Hypertext Preprocessor, PHP s'impose comme un standard dans le monde de la programmation web par ses performances, sa fiabilité, sa souplesse et sa rapidité.

PHP a été inventé à l'origine pour le développement d'applications web dynamiques qui constituent encore le cas d'utilisation le plus courant et son point fort. Cependant, les évolutions qui lui ont été apportées jusqu'à aujourd'hui assurent à PHP une polyvalence non négligeable. PHP est par exemple capable d'interagir avec Java, de générer des fichiers PDF, d'exécuter des commandes Shell, de gérer des objets (au sens programmation orientée objet), de créer des images ou bien de fournir des interfaces graphiques au moyen de PHP GTK.

## CARACTÉRISTIQUES DU LANGAGE PHP

### LICENSE

PHP est tout d'abord un langage de script interprété (en réalité précompilé en Opcode), gratuit, Open Source et distribué sous une License autorisant la modification et la redistribution.

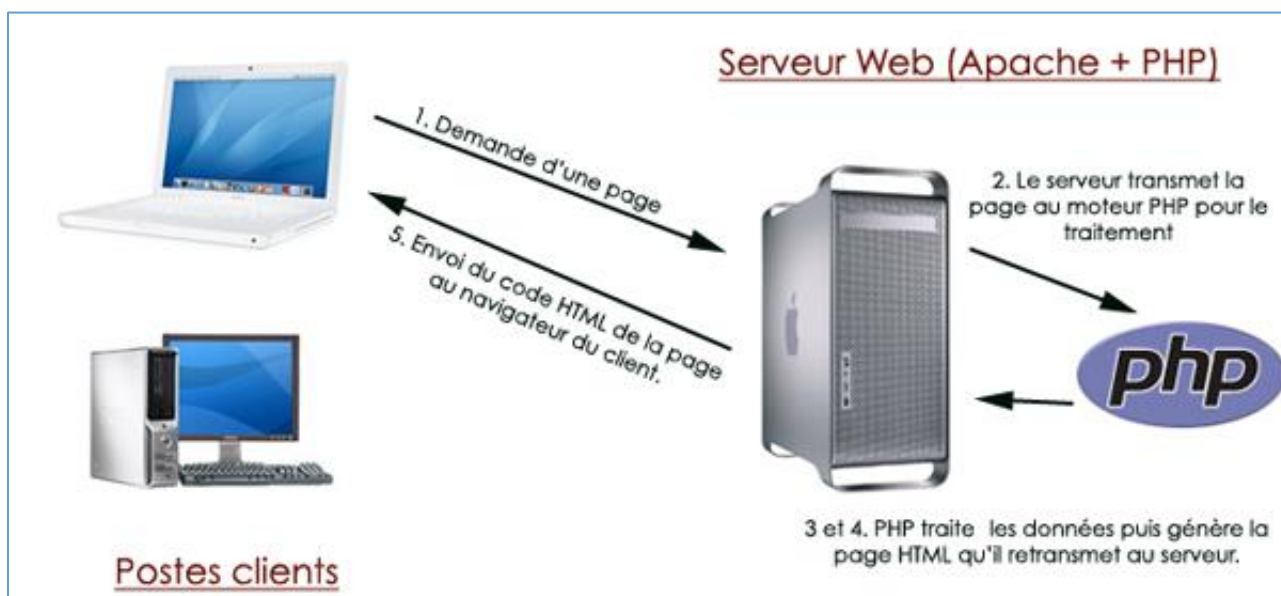
### PORTABILITÉ

PHP est supporté sur plusieurs systèmes d'exploitation. C'est par exemple le cas des versions Microsoft Windows™, mais aussi des systèmes reposant sur une base UNIX (Apple MAC OS X™, distributions Linux ou encore Sun Solaris).

Il sera alors très facile de déplacer une application écrite en PHP d'un serveur Windows d'origine vers un serveur Linux sans avoir à la modifier (ou très peu).

### EXÉCUTION

D'un point de vue exécution, PHP a besoin d'un serveur Web pour fonctionner. Toutes les pages demandées par un client seront construites par le serveur Web, en fonction des paramètres transmis, avant d'être retournées au client. Le schéma ci-dessous illustre le principe de fonctionnement de PHP.



## APPRENTISSAGE DE PHP

PHP est un langage dit de « haut niveau » dont la syntaxe est très proche du langage C. Il propose une syntaxe proche du langage naturel ce qui assure un apprentissage rapide et peu de contraintes d'écriture de code. Néanmoins, la maîtrise rapide de sa syntaxe ne signifie pas la maîtrise de ses fonctionnalités et de ses concepts. Une bonne connaissance et une utilisation avancée de la programmation PHP nécessite un temps d'apprentissage relativement long.

## RICHESSSE DU LANGAGE PHP

Une des forces du langage PHP est sa richesse en termes de fonctionnalités. En effet, il dispose à l'origine de plus de 4000 fonctions natives prêtes à l'emploi garantissant aux développeurs de s'affranchir de temps de développement supplémentaires et parfois fastidieux. Ces fonctions permettent entre autres de traiter les chaînes de caractères, d'opérer mathématiquement sur des nombres, de convertir des dates, de se connecter à un système de base de données, de manipuler des fichiers présents sur le serveur...

PHP puise aussi sa richesse dans le dynamisme de sa communauté de développeurs. Les profils de développeurs de la communauté sont très divers. Il y'a ceux qui apportent de nouvelles fonctionnalités et bibliothèques de version en version, ceux qui traduisent la documentation en plusieurs langues ou encore les programmeurs ayant des compétences plus modestes qui réalisent des applications Open Source prêtes à l'emploi. Parmi les plus connues, nous pouvons citer les CMS (Joomla, Dotclear, Wordpress...), les systèmes de ventes en ligne (OSCommerce, Prestashop...), les forums (PHPBB, Vbulletin...), les frameworks (Zend Framework, Symfony, CakePHP, Laravel...).

## PHP, UN LANGAGE FIABLE ET PERFORMANT

Le langage est maintenant devenu un langage fiable, performant et viable. Il est capable de supporter des sites qui sollicitent des millions de requêtes journalières. De nombreuses entreprises de renommée nationale et internationale lui font confiance pour le développement de leur site Internet. Nous pouvons parmi elles citer TF1, IBM, Le Monde, Le Figaro, Facebook, Orange...



## PHP, UN LANGAGE CONÇU POUR LES APPLICATIONS WEB DYNAMIQUES

Le langage PHP a pour principale vocation la réalisation d'applications web dynamiques. Par définition, une « application (ou page) dynamique » est un programme capable de générer une page unique en fonction de paramètres qui lui sont transmis.

Un script PHP peut donc être intégré directement à l'intérieur d'un code html:

```
<html>
  <head>
    <title>Hello World en PHP</title>
  </head>
  <body>
    <p>
      <?php echo 'Hello World !'; ?>
    </p>
  </body>
</html>
```

Le script PHP, clairement identifié par les deux balises **<?php** (ouverture) et **?>** (fermeture), provoquera l'écriture de la chaîne de caractères Hello World entre les balises HTML après son exécution sur le serveur.

Nous aurions également pu obtenir le même résultat en utilisant le script php suivant :

```
<html>
  <head>
    <title>Hello World en PHP</title>
  </head>
  <body>
    <?php echo '<p>Hello World !</p>'; ?>
  </body>
</html>
```

On peut déduire du code précédent que PHP est capable de générer du code HTML (mais pas que !), ce qui fait tout son intérêt. La présentation du document HTML généré est alors complètement dépendante des conditions passées et des paramètres initiaux fournis.



## PHP POUR D'AUTRES DOMAINES D'APPLICATIONS

Fort de sa richesse, PHP ne se limite pas forcément à l'édition de pages web dynamiques. Il peut par exemple être utilisé en ligne de commande via l'utilisation de l'exécutable `php cli`. Ce cas d'utilisation permet alors d'exécuter des scripts directement sur les machines. Il est possible de coupler l'utilisation de PHP avec un gestionnaire de tâches tel qu'un serveur cron sous Linux.

Un autre point fort de PHP est sa capacité à s'interfacer très facilement avec de nombreux systèmes de gestion de bases de données relationnelles et noSQL. Parmi eux, nous pouvons retrouver MySQL, Oracle, SQLite, MS SQLServer, PostgreSQL, MongoDB...

Les possibilités offertes par PHP sont donc très nombreuses et nous n'allons pas les détailler toutes car cela vous donnerait des jours entiers de lecture.

Retenons néanmoins une liste de ses principales capacités :

- ✓ Manipulation d'un système de fichiers (création, édition, suppression, droits d'accès...)
- ✓ Gestion des sessions utilisateurs
- ✓ Génération et parsing de documents XML
- ✓ Génération d'images
- ✓ Génération de fichiers PDF
- ✓ Accès simplifié aux bases de données
- ✓ Exécution de commandes Shell
- ✓ Gestion des e-mails
- ✓ Compression et décompression d'archives ZIP
- ✓ Chiffrement de données
- ✓ Gestion d'annuaires LDAP
- ✓ Manipulation des dates
- ✓ Manipulation des URL
- ✓ Envoi et lecture de cookies
- ✓ Dialogue avec Java
- ✓ Utilisation d'Ajax
- ✓ ...

La version 5 de PHP a apporté un grand vent de fraîcheur au langage et aux professionnels. Le principal manque jusque-là était son modèle objet trop succinct. Les développeurs se sont alors penchés sur cette problématique et ont finalement implémenté un modèle de programmation objet proche du langage Java. Grâce à un tel modèle, les professionnels encore hésitants à utiliser PHP deviennent de plus en plus nombreux. Des applications complètement objet voient également le jour et intègrent un design pattern MVC. Les plus connus aujourd'hui sont les frameworks Laravel et Symfony.

La version 7 de PHP a, quant à elle, apporté un gain de performance de l'ordre de 100% par rapport à la version précédente (5.6).



## LES LIMITES DE PHP

Malgré toutes les qualités que nous pouvons attribuer au langage PHP, subsistent quelques défauts. Par exemple, dans le cas de très grosses applications, il peut présenter quelques faiblesses et devenir inadapté. Cet argument reste toutefois très subjectif dans la mesure où les développeurs de PHP améliorent constamment la qualité et la robustesse du langage.

Le second défaut (mais qui paradoxalement fait son succès et sa qualité) que nous pouvons lui reprocher est sa grande simplicité d'utilisation. Cela a beaucoup terni l'image de PHP parce que tout webmaster (même très peu expérimenté) devient capable de créer du code et des applications facilement. Cependant, la plupart des codes produits par des développeurs amateurs n'est pas forcément bien conçu, souvent peu sécurisé, peu maintenable et même non optimisé. Les autres langages comme C++, .Net, Java, ASP, Perl, Python ou Ruby ne subissent pas cette mauvaise image dans la mesure où leur apprentissage est plus fastidieux.

Enfin, le dernier défaut reprochable à PHP est son manque de rigueur dans la nomenclature des fonctions et de la syntaxe. Par exemple, un manque de standardisation des noms des fonctions :

Utilisation du séparateur underscore :

- `str_replace()`, `preg_match()`, `mysql_real_escape_string()` ...

Fonctions composées de plusieurs mots écrite en un seul :

- `wordwrap()`, `htmlspecialchars()` ...

Traduction du to en littéral ou numéraire :

- `bin2hex()`, `strtotime()`...

## PHP, UN LANGAGE FAIBLEMENT TYPÉ

Contrairement à d'autres langages tels que C# ou Java, PHP est ce qu'on appelle un langage faiblement typé. Il permet au programmeur de ne pas se soucier du type de variables qu'il manipule. Si cela offre certaines facilités, l'approche comporte toutefois certains pièges dont il vaut mieux être conscient.

La version 7 de PHP corrige en partie ce « problème » et apporte le typage fort mais pas obligatoire...

## PHP : DOCUMENTATION OFFICIELLE

PHP héberge sa documentation sur <http://www.php.net> .

Vous y trouverez toutes les informations nécessaires pour bien utiliser PHP.

--- FIN DU DOCUMENT ---

