

architecte
logiciel

Guillaume Ponçon
Ouvrage dirigé par Libero Maesano

Best practices PHP 5

Préface de Cyril Pierre de Geyer

EYROLLES

© Groupe Eyrolles, 2006,
ISBN : 2-212-11676-4

1

PHP est-il adapté à vos besoins ?

La popularité de PHP ne cesse d'augmenter depuis plus de 10 ans. Sa souplesse et sa grande simplicité d'utilisation séduisent un très grand nombre de développeurs. En revanche, exploiter l'étendue de ses possibilités nécessite, au même titre que n'importe quelle plate-forme de développement complète, de bonnes connaissances théoriques.

Aujourd'hui, de nombreux succès voient le jour dans des projets professionnels réalisés avec PHP. La plate-forme s'avère de plus en plus fiable et performante, ce grâce aux contributions d'une communauté de développeurs très active qui grandit de jour en jour.

PHP est principalement utilisé pour réaliser des applications web, mais il permet aussi de développer des applications en ligne de commande ou avec des interfaces graphiques dites clients lourds (GTK). Après la lecture de ce chapitre, vous aurez une bonne connaissance du panorama PHP actuel.

Nous aborderons entre autres les garanties d'un investissement dans PHP, les points forts et les points faibles de la plate-forme ainsi qu'une vision d'ensemble des activités qui existent autour de PHP. Le but est de vous donner les moyens d'apprécier à quel point et à quel niveau PHP peut répondre à vos besoins.

Qu'est-ce que PHP ?

PHP (PHP Hypertext Preprocessor) est une plate-forme composée d'un langage de programmation très complet et de nombreux outils pour le développement. Elle s'adapte très rapidement aux technologies émergentes et se voit de plus en plus utilisée dans des développements web dynamiques professionnels et Open Source.

Voici quelques-unes de ses caractéristiques principales :

- un très bon compromis entre fiabilité et rapidité d'exécution ;
- une plate-forme avant tout spécialisée pour le développement de sites web dynamiques de toute taille ;
- une plate-forme pratique et complète adaptée aux applications en ligne de commande ;
- une syntaxe complète, souple et permissive, qui ne rebute pas les développeurs débutants et ne limite pas les utilisateurs confirmés ;
- un langage procédural et un langage orienté objet ;
- un outil très complet, doté de nombreuses fonctionnalités, extensions et bibliothèques.

PHP 5 et ses nouveautés propulse PHP dans le monde des plates-formes d'entreprises comme .Net ou J2EE.

À quoi sert PHP ?

L'utilisation de PHP est principalement dédiée aux développements de sites web dynamiques pour toutes sortes d'applications : du simple forum au supermarché en ligne.

PHP intègre de très nombreuses extensions. Il est par exemple possible de créer des fichiers PDF (Portable Document Format), de se connecter à des bases de données ou à des serveurs d'annuaires LDAP (Lightweight Directory Access Protocol), de créer des clients et serveurs SOAP (Simple Object Access Protocol) ou d'établir des communications natives avec d'autres applications développées en Java ou en C/C++.

Les possibilités de PHP ne s'arrêtent pas à la création de sites web. PHP permet également le développement d'applications en ligne de commande et un de ses modules lui permet de fournir des interfaces graphiques classiques (client lourd, sans navigateur ou serveur web), via GTK (Gimp ToolKit).

PHP dispose de près de 3.000 fonctions utilisables dans des applications très variées et couvre pratiquement tous les domaines en rapport avec les applications web. Par exemple, presque tous les SGBD (Systèmes de gestion de bases de données) du marché peuvent s'interfacer avec PHP, qu'ils soient commerciaux ou qu'ils proviennent du monde du logiciel libre.

De nombreuses solutions d'interopérabilité existent aussi, notamment autour de services web (SOAP, etc.).

CULTURE En savoir plus sur les possibilités de PHP

La documentation officielle propose une introduction qui explique ce qu'est PHP et ce que l'on peut faire avec cette plate-forme. Vous pouvez lire ces pages si vous souhaitez des informations détaillées sur les possibilités de PHP.

► <http://www.php.net/manual/fr/introduction.php>

À qui s'adresse PHP ?

PHP n'a aujourd'hui rien à envier aux plates-formes .Net et J2EE pour le développement de solutions applicatives dynamiques pour le Web. Il est d'ailleurs possible de rendre complémentaires PHP et l'une de ces technologies.

En terme de productivité, le principal avantage de PHP est sa simplicité, sa rapidité et sa fiabilité, que ce soit au niveau du développement ou de l'exécution :

- Programmer avec PHP ne nécessite aucune compilation ou construction manuelle. Les modifications effectuées sur le code sont immédiatement opérationnelles.
- Les outils de débogage proposés avec PHP assurent une parfaite maîtrise du comportement des développements vis-à-vis de la mémoire et des ressources.
- PHP est plus qu'interprété, il est compilé à la volée, ce qui signifie qu'à chaque modification effectuée sur le code, une compilation partielle en langage machine (opcodes) est effectuée. À l'échelle d'un être humain, cette étape de compilation à la volée est instantanée.

PHP est également souple et facile à manipuler. Ces caractéristiques sont idéales pour travailler dans le cadre d'une méthode agile telle que l'eXtreme Programming.

CULTURE Méthode agile

Une méthode agile est une nouvelle manière d'aborder les projets de développements informatiques. Pour être plus réactif et au plus près des attentes des demandeurs de logiciels, elle consiste à les impliquer fortement avec les réalisateurs dès le départ et tout au long du projet.

Enfin, PHP est adapté aux administrateurs systèmes qui cherchent un outil de script complet et pratique. Dans ce domaine, PHP est une bonne alternative à Perl.

Pour résumer, PHP s'adresse avant tout aux équipes de développement web, aux architectes du système d'information et aux administrateurs système qui souhaitent gagner temps et fiabilité à travers une plate-forme simple d'utilisation et facile à maintenir.

Qui développe PHP ?

La communauté PHP est ouverte, elle accueille de nombreux développeurs, testeurs, traducteurs et rédacteurs. Un noyau constitué de 7 développeurs confirmés prend les décisions essentielles qui touchent au cœur de PHP.

Plusieurs équipes travaillent autour de PHP :

- L'équipe de développement (environ 135 personnes) s'occupe des évolutions de PHP et de ses extensions : création de nouvelles fonctionnalités, corrections de bogues et remaniements. PHP est développé en langage C.
- L'équipe de documentation (environ 260 personnes) maintient constamment à jour la documentation, traduite en 23 langues.
- L'équipe PEAR (environ 190 personnes) s'occupe des développements des bibliothèques PHP (classes utiles au développement d'applications PHP).
- L'équipe PECL (environ 80 personnes) développe les modules PHP. Ces modules sont développés en langages C/C++. On en dénombre environ 400 à l'heure actuelle.
- L'équipe de qualité (environ 22 personnes) est garante de la qualité des développements effectués sur le noyau de PHP.
- Les équipes Smarty et PHP-GTK (environ 40 personnes) maintiennent ces deux projets importants.

En tout, plus de 1 000 personnes sont enregistrées dans le dépôt de données CVS. Environ 470 committers répartis dans plusieurs groupes participent réellement au développement du projet.

Figure 1-1
Répartition des contributeurs
dans les différentes équipes

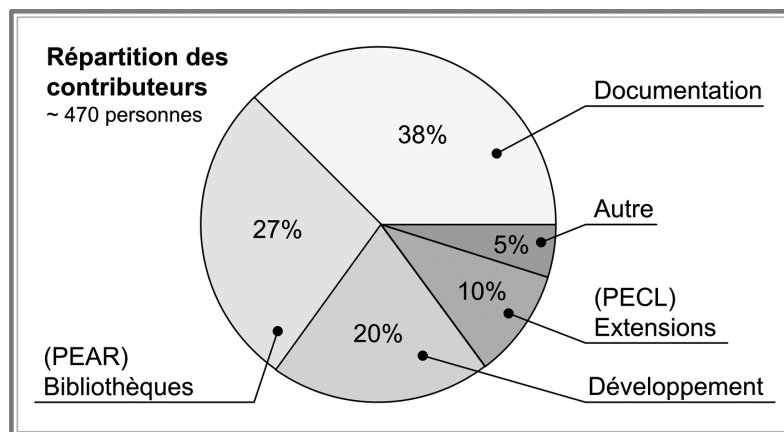
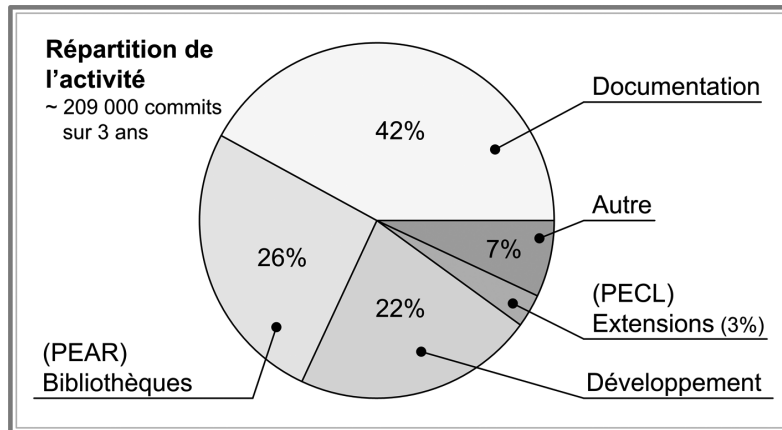


Figure 1-2
Répartition de l'activité
des différentes équipes



RESSOURCES Liens vers les sites officiels des différentes équipes

- Développement : ▶ <http://www.php.net>
- Documentation : ▶ <http://www.php.net/docs.php>
- PEAR : ▶ <http://pear.php.net>
- PECL : ▶ <http://pecl.php.net>
- Qualité : ▶ <http://qa.php.net>
- Smarty : ▶ <http://smarty.php.net>
- PHP-GTK : ▶ <http://gtk.php.net>

Politique d'évolution

La plupart des grands projets Open Source possèdent leur esprit ou leur philosophie. L'évolution de PHP est fortement liée à son esprit, qui peut se résumer en trois mots : simplicité, ouverture et souplesse.

Les évolutions de PHP sont toujours influencées par une écoute massive des besoins de ses utilisateurs, ce malgré l'évolution de cette communauté. Depuis quelques années, nous voyons de plus en plus d'utilisateurs issus de milieux professionnels, et ce phénomène a été déterminant pour les évolutions de la version 5.

Dans ce qui n'évolue pas ou très peu, PHP reste très souple avec le lot d'avantages et d'inconvénients que cela engendre. Il reste également un outil très simple à apprendre et pourtant très complet.

PHP est très à l'écoute des technologies émergentes. La plate-forme est résolument tournée vers l'avenir en proposant des fonctionnalités fortement liées aux nouvelles technologies de l'information et de la communication (NTIC), l'interopérabilité des systèmes et le Web.

Pourquoi PHP est-il libre ?

Retirer PHP du monde du logiciel libre, c'est lui retirer son âme et mettre sa pérennité en danger. PHP repose sur le logiciel libre depuis le commencement. Le développement de PHP est ouvert et vit de ses nombreux contributeurs. Grâce à ce système et à la popularité de la plate-forme qui ne cesse de croître, la réactivité de ses mises à jour est imbattable.

PHP est gratuit, ouvert, simple et populaire. Il est de ce fait largement documenté. La documentation, forte de ses nombreux contributeurs, est constamment à jour depuis des années. Elle est accompagnée de nombreux exemples et retours d'expériences utiles. Plusieurs forums traitent de sujets liés au PHP, accompagnant ce mouvement de diffusion de l'information.

Quels sont les apports de PHP en termes de productivité ?

Sa simplicité

Elle est depuis le début la clé de son succès auprès de nombreux développeurs de tous niveaux. Son installation et son apprentissage est rapide, grâce à la documentation, aux nombreux ouvrages sur le sujet et à une communauté grandissante de développeurs passionnés.

Sa similitude avec d'autres plates-formes populaires (C, Java)

PHP est syntaxiquement similaire aux langages C et Java. Les développeurs issus de ces mondes ont la possibilité de se lancer dans PHP plus rapidement que les autres. Nous verrons par la suite qu'ils possèdent également un atout : celui d'avoir acquis une discipline et des connaissances indispensables au développement de projets ambitieux.

Son adaptation aux débutants

Pour ceux qui ne sont pas familiarisés avec la programmation, l'apprentissage des bases de PHP est rapide. Les ressources PHP pour les débutants abondent et permettent à n'importe qui d'être rapidement opérationnel.

En revanche, il faut du temps et de la rigueur pour aller plus loin et progresser dans de bonnes conditions. Un débutant sera toujours limité par son manque de connaissances en informatique fondamentale et en génie logiciel.

Sa souplesse

Les possibilités de remaniement et de mises à jour sont faciles et rapides avec PHP, contrairement à des configurations figées qui demandent un investissement et des manipulations lourdes pour accepter des changements.

PHP est, d'un point de vue utilisateur, interprété. Il possède un système de compilation interne et transparent destiné à optimiser ses performances.

Quelles sont les garanties d'utilisation de PHP ?

PHP n'a pas de support technique commercial. En revanche, il dispose d'une alternative très efficace : une communauté ouverte et réactive de nombreux passionnés répartis à travers le monde.

Cette communauté est à l'écoute de tous les problèmes et de toutes les attentes des utilisateurs. Si PHP 5 a su gagner son pari de combler les principaux reproches que l'on faisait à PHP 4 dans le monde professionnel, c'est grâce à la mobilisation de cette communauté de contributeurs.

À juste titre, les professionnels ont tendance à ne pas faire confiance aux programmes Open Source pour leur manque de garanties contractuelles. Mais il existe quelques applications d'exception dans lesquelles on a naturellement confiance. Le serveur HTTP Apache et PHP en font partie.

À qui dois-je m'adresser en cas de problème ?

Cela dépend du problème et de l'implication que l'on peut avoir dans sa résolution. Il existe plusieurs solutions :

- en parler aux contributeurs et aux développeurs à travers les forums et les listes de diffusion mises à disposition ;
- proposer une contribution auprès du PHPGroup ;
- s'adresser à des spécialistes qui sauront vous répondre rapidement.

RESSOURCE Trouver du support sur PHP

Le lien suivant est un bon point de départ pour trouver des contacts :

► <http://www.php.net/support.php>

PARTICIPER Faire partie d'un réseau d'utilisateurs professionnels de PHP

En France, l'association AFUP (Association française des utilisateurs de PHP) regroupe un grand nombre de professionnels partageant régulièrement leurs expériences et organisant des rencontres liées à l'utilisation de PHP en entreprise. Si vous avez un problème technique ou des questions liées à l'utilisation de PHP, vous pouvez également participer à un forum de discussions ou vous abonner à un journal spécialisé tel que Direction | PHP.

- AFUP : ▶ <http://www.afup.org/>
- Forum Nexen : ▶ <http://www.nexen.net/forum/>
- Direction PHP : ▶ <http://www.directionphp.biz/>

Le parcours de PHP

L'historique complet de PHP, ainsi qu'un musée mettant à disposition les toutes premières versions de la plate-forme est disponible sur le site officiel à l'adresse suivante :

- ▶ <http://www.php.net/manual/fr/history.php>

Le court historique proposé dans cet ouvrage est inspiré de l'historique officiel.

Naissance de PHP

Initialement, PHP était une bibliothèque Perl rédigée par Rasmus Lerdorf en 1995. Il s'en servait pour mettre à disposition son curriculum vitae sur Internet. Au fur et à mesure qu'il ajoutait des fonctionnalités Rasmus a transformé la bibliothèque Perl en une implémentation C. Il décida par la suite de partager son code pour que tout le monde puisse en profiter, ce qui attira les premiers contributeurs.

PHP/FI

PHP/FI est le sigle de Personal Home Page/Form Interpreter. Cette première version possédait déjà une syntaxe similaire à celle que nous connaissons aujourd'hui.

PHP/FI 2

Publiée en novembre 1997, c'est la deuxième refonte en langage C de PHP. À ce moment là, plusieurs milliers de personnes dans le monde utilisent déjà PHP et environ 50 000 noms de domaines indiquaient utiliser PHP. La durée de vie de cette version aura été très courte avec l'arrivée de PHP 3.

PHP 3

PHP 3 fut une refonte complète initiée en 1997 par deux développeurs : Zeev Suraski et Andi Gutmans. La plate-forme telle que nous la connaissons actuellement est issue de cette version. Dans un effort de coopération et de compatibilité avec les anciennes versions de PHP/FI, Zeev, Andi et Rasmus décidèrent d'annoncer PHP 3 comme étant le successeur de PHP/FI.

CULTURE En savoir plus sur les fondateurs de PHP !

Au chapitre 18 de cet ouvrage, vous trouverez des témoignages d'utilisateurs de PHP avec en exclusivité celui de Zeev Suraski.

À partir de ce moment, PHP change de nom. Il devient PHP Hypertext Preprocessor (acronyme récursif). Ce fut le signal de la publication d'une nouvelle plate-forme qui n'est plus le projet d'un seul homme, mais d'une communauté.

En 1998, plusieurs dizaines de milliers d'utilisateurs et plusieurs centaines de milliers de sites font déjà confiance à PHP. PHP 3 a couvert environ 10 % du parc mondial de serveurs web.

PHP 4

PHP 4 a été initié durant l'hiver 1998. Andi Gutmans et Zeev Suraski décidèrent de réécrire le moteur interne de PHP afin d'améliorer les performances et la modularité du code.

Le nouveau moteur, baptisé Zend Engine (Zend est une combinaison de Zeev et Andi), a atteint haut la main son objectif. La première version de PHP 4 a été publiée officiellement en mai 2000. Des performances plus élevées, une compatibilité avec de nombreux serveurs et plusieurs nouvelles fonctionnalités utiles ont contribué au succès de cette version.

PHP 4 a battu des records de popularité. Plusieurs millions de sites web indiquent qu'ils sont installés avec PHP et des centaines de milliers d'utilisateurs à travers le monde lui font confiance.

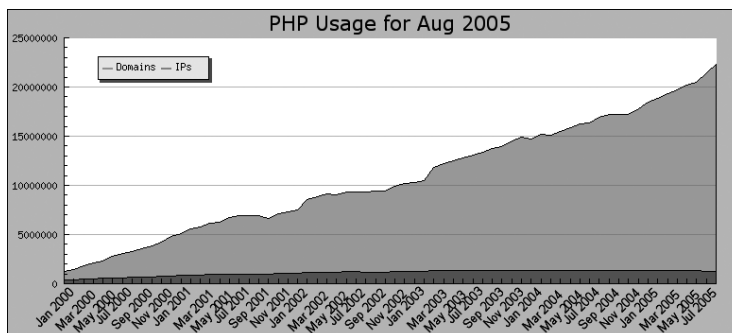
L'équipe s'est également considérablement agrandie avec de nombreux contributeurs pour la documentation et des projets stratégiques comme PEAR.

PHP 5

PHP 5 est sorti en juillet 2004. Il propose un nouveau moteur, Zend Engine II, optimisé pour les nouvelles fonctionnalités que nous lui connaissons, notamment l'approche objet.

Cet ouvrage est basé sur la version 5 de PHP. Nous aurons l'occasion de la découvrir en long, en large et en travers.

Figure 1-3
Les statistiques d'utilisation
de PHP par Netcraft



L'évolution du nombre d'utilisateurs de PHP est en progression constante depuis plusieurs années, comme nous pouvons le voir sur la figure 1-3 (source Netcraft, <http://news.netcraft.com/>). Ces statistiques sont mises à jour régulièrement sur le site officiel de PHP à l'adresse suivante :

► <http://www.php.net/usage.php>

Un outil simple pour résoudre des problèmes complexes

Une plate-forme intuitive, facile à assimiler

Depuis le lancement de PHP 3, la popularité de PHP n'a cessé de progresser auprès des jeunes et des étudiants. PHP est facile à assimiler, donc un choix idéal pour les débutants à la recherche d'un outil simple qui produit rapidement un résultat concret.

Installer PHP, Apache et MySQL en moins de 10 minutes !

De nombreux installeurs tels que Wampserver et EasyPHP permettent d'installer PHP chez soi en moins de 10 minutes :

Wampserver : ► <http://www.afup.org/>

EasyPHP : ► <http://www.nexen.net/forum/>

RESSOURCES Présentation de Wampserver par son auteur

Romain Bourdon, auteur de Wampserver : « WAMP5 est une plate-forme de développement basée sur Apache 1.3.x et PHP 5. Son principal but est de permettre à la communauté PHP de découvrir facilement la nouvelle version PHP. Il dispose également de la base de données MySQL version 4 et de l'application PhpMyAdmin. »

Découvrez également le témoignage de Romain Bourdon sur un projet de migration Lotus vers PHP au chapitre 18.

Développer sa première page PHP en moins de deux minutes !

Développer une première page en PHP est un jeu d'enfant. La procédure est la suivante :

- 1 Après avoir installé PHP avec un installeur, créer un fichier nommé `index.php` dans le répertoire racine du serveur HTTP, contenant le code suivant :

```
<?php
echo "Bonjour, ceci est mon premier fichier PHP !";
?>
```

- 2 Ouvrir un navigateur (Firefox ou Internet Explorer) et taper l'URL de la page (généralement `http://localhost/`). Le message « Bonjour, ceci est mon premier fichier PHP ! » devrait alors s'afficher.

Bien entendu, la plate-forme installée permet d'aller beaucoup plus loin, ce qui explique l'abondance de documentation et d'ouvrages qui existent autour de PHP.

Une syntaxe souple, qui requiert de la rigueur

Si la souplesse syntaxique de PHP est très appréciée des débutants, elle est souvent un obstacle à la rigueur. La communauté des utilisateurs PHP est de ce fait très hétérogène et composée de plusieurs catégories de développeurs :

- Les débutants ou anciens débutants qui ont commencé à développer sérieusement avec PHP.

Ces utilisateurs ont tendance à être limités par un manque de connaissances théoriques de la programmation informatique. Certains, rigoureux et travailleurs, progressent. En revanche, une majorité se contente de l'extrême permissivité de PHP pour développer de petites applications.

- Les développeurs issus d'autres technologies, telles que C ou Java.

Ils ont déjà des réflexes et une expérience de la programmation qui leur permettent d'aller plus loin avec PHP.

AVIS D'EXPERT Cyril Pierre de Geyer

« PHP : une plate-forme intuitive et facile à assimiler.

Avec l'arrivée de PHP 5 nous sommes entrés dans une nouvelle ère pour PHP. S'il ne s'agit pas d'une révolution on peut du moins parler d'une évolution majeure. On ne parle plus de PHP comme d'un langage de script mais on prend de la hauteur et on parle de PHP comme d'une plate-forme complète.

Sa force réside dans ses deux langages (procédural et objet) qui tendent à faciliter la prise en main de l'outil et laissent le développeur libre d'adopter la méthode de travail qu'il souhaite. Ce n'est plus la technologie qui décide mais le développeur.

Énumérer la liste des avantages de PHP serait trop long mais on peut au moins citer :

- Sa souplesse

PHP propose deux syntaxes : l'une procédurale, l'autre orientée objet. Chacune de ces syntaxes permet de mettre en œuvre les mêmes fonctionnalités mais vise des publics différents. La syntaxe procédurale est destinée aux webmasters et aux informaticiens qui travaillent sur l'interface graphique des applications. La seconde syntaxe, orientée objet, est très proche de Java et C# dont elle s'inspire volontairement pour diminuer les coûts de formation des entreprises.

Un développeur Java ou C# pourra ainsi migrer vers PHP 5 avec peu ou sans formation, les concepts et syntaxes clés étant identiques.

- Sa richesse fonctionnelle

Avec PHP vous disposez de plus de 3 000 fonctions permettant de gérer rapidement la majorité des besoins communs.

- Ses performances

Avec des exemples d'applications gérant plus de 150 000 visiteurs par jour (neowiz.com) on trouve difficilement des plates-formes aussi robustes.

- Son interopérabilité

Ce n'est pas pour rien que PHP est de plus en plus souvent choisi comme plate-forme pour interfacier des systèmes d'information. PHP peut manipuler des services web facilement, instancier des objets Java et .NET, dispose de connecteurs techniques vers toutes les bases de données du marché, vers LDAP, Lotus Notes, SAP, etc.

Bref, PHP est la plate-forme incontournable pour tout ce qui touche aux applications web. Elle a commencé à séduire par la base et continue à croître en puissance au fur et à mesure des années. »

Cyril Pierre de Geyer est pionnier de la démocratisation de PHP en France. Il est à l'origine de nombreuses actions et associations telles que l'AFUP (Association française des utilisateurs de PHP), le Forum PHP qui a lieu tous les ans à Paris, PHPTeam, PHPFrance et Anaska Formation. Il est également co-auteur d'un ouvrage pratique sur PHP 5 :

 *PHP 5 Avancé*, 2^e Édition, d'Éric Daspét et Cyril Pierre de Geyer aux éditions Eyrolles.

Alors que d'autres plates-formes ne laissent pas de choix à leurs développeurs sur la ligne de conduite à adopter, PHP est lui un champion de la liberté. De la syntaxe au modèle, en passant par les solutions techniques, le choix est incroyablement vaste.

Une nécessité : un minimum de connaissances en génie logiciel

Savoir afficher du texte et faire des boucles avec PHP est quasiment immédiat. Mais PHP ne se limite pas à cela, il intègre un véritable langage de programmation procédural et objet.

PHP ne sait pas tout faire

PHP ne sait pas construire une architecture logicielle, déterminer l'algorithme optimal ou exploiter le débogueur à la place du développeur.

Les débutants en PHP peuvent avoir tendance à se considérer comme des développeurs après avoir écrit un livre d'or. Faites attention de bien faire la part des choses, les bons développeurs PHP sont encore rares. Ils doivent avoir de bonnes connaissances générales et théoriques en programmation informatique et une certaine expérience de développement dans d'autres langages.

Un atout très appréciable : la connaissance du Web

Parmi les utilisateurs expérimentés, ceux qui connaissent le contexte web du développement informatique ont un avantage très appréciable sur les autres.

Parfaitement maîtriser PHP ne suffit pas à être un développeur fiable de sites web. Beaucoup de programmeurs rencontrent actuellement des problèmes en milieu professionnel pour ces raisons.

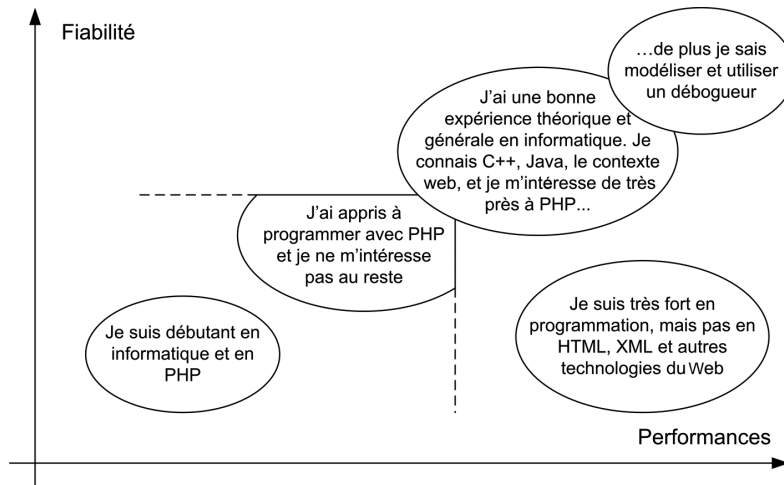
Comment aborder PHP ?

Il n'y a pas de recette miracle pour devenir un bon développeur PHP, les parcours sont très hétérogènes. En revanche, le développeur PHP est vite limité par un manque de connaissance ou d'expérience dans des domaines qui ne sont pas directement liés à PHP :

- la maîtrise des fondements de l'algorithmique et de l'informatique ;
- le respect de règles et de conventions à tous niveaux ;
- la connaissance des technologies nouvelles, notamment celles qui tournent autour du Web et des réseaux ;
- la maîtrise d'autres technologies différentes de PHP, telles que Java, C, C++, Delphi, C# ou toute autre plate-forme dont l'expertise demande un investissement conséquent.

La figure 1-4 tente de mettre en avant les différents types de développeurs PHP que l'on peut rencontrer. Cette généralisation fait suite aux nombreux témoignages et retours d'expérience de professionnels ayant travaillé avec des développeurs.

Figure 1-4
Quelques profils types
de développeurs PHP



Pour en savoir plus, le chapitre 18 propose cinq témoignages. Les personnes choisies sont des professionnels du PHP ayant eu un parcours très différent les unes des autres.

De nombreuses fonctionnalités qui en font une plate-forme communicante

L'un des points appréciables de PHP est sa forte capacité d'interopérabilité. De nombreuses extensions natives mettent à disposition des outils de gestion de flux et de protocoles de communication à tous les niveaux.

À partir des versions 4.3.x et à plus forte raison avec la version 5, PHP est plus que jamais un outil d'interopérabilité. Les outils de manipulation XML tels que DOM, SAX et SimpleXML sont stables et fiables. De nombreux outils de gestion de protocoles (ouverts et propriétaires) sont intégrés à PHP : SOAP, WDDX, SAP, Hyperwave, etc.

Souplesse d'adéquation à la complexité du problème et à la compétence des équipes

Obtenir des résultats concrets avec PHP est facile et rapide. En revanche, c'est à double tranchant. Davantage qu'avec d'autres technologies, les étapes de finalisation, de refactoring et d'optimisation sont essentielles.

À forte complexité, PHP a prouvé qu'il restait très efficace, à condition que la compétence des équipes le permette.

Figure 1-5
Quelques protocoles et outils
d'interopérabilité gérés par PHP

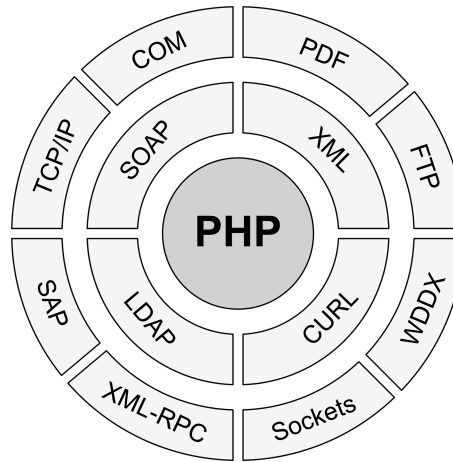
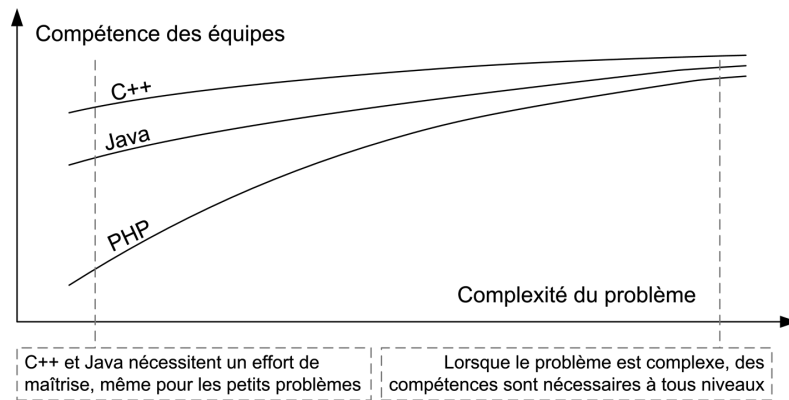


Figure 1-6
Niveaux de compétences
nécessaires pour résoudre
des problèmes



En d'autres termes, une équipe qui arrive rapidement à un résultat concret avec PHP n'est pas forcément capable de résoudre des problèmes complexes qui demandent un investissement conséquent. Être conscient de cela permet d'aborder PHP dans de meilleures conditions.

Une architecture simple, fiable et portable

PHP est un programme. Ce programme prend du code en entrée, exécute des opérations et fournit un flux de données en sortie.

PHP est un interpréteur. Il ne nécessite aucune opération manuelle de compilation ou de construction pour exécuter ce que lui demande l'utilisateur. Il lit et exécute directement ce qu'écrit le programmeur.

PHP est néanmoins fiable. Un script interprété par un programme doté d'un excellent parseur (analyseur syntaxique) accompagné d'un compilateur et d'un moteur efficaces est plus rapide et plus stable qu'un programme mal compilé ou exécuté par dessus des couches logicielles lourdes.

PHP est portable. Son utilisation sur Mac, Microsoft Windows ou Unix est garantie, à l'exception de rares extensions spécifiques à une plate-forme donnée.

Architecture minimale : PHP CLI et GTK

Même si nous le voyons souvent associé à un serveur HTTP pour délivrer des pages web, en utilisation ligne de commandes PHP se suffit à lui-même.

Exécuter du PHP en ligne de commande

```
$ php -r 'print_r(posix_uname());'  
Array  
(  
  [sysname] => FreeBSD  
  [nodename] => guillaume  
  [release] => 5.4-STABLE  
  [version] => FreeBSD 5.4-STABLE #4: Mon Jun 13 19:28:13 CEST 2005  
  [machine] => i386  
)
```

PHP est un interpréteur de script shell performant et plein de possibilités. Il dispose, comme Perl, d'un ensemble vaste et varié de fonctionnalités à mettre à disposition de l'administrateur système.

Cet ouvrage par exemple a été exporté aux formats PDF et HTML à partir de documents OpenOffice.org, pendant toute la durée de son élaboration, par un script PHP en ligne de commande.

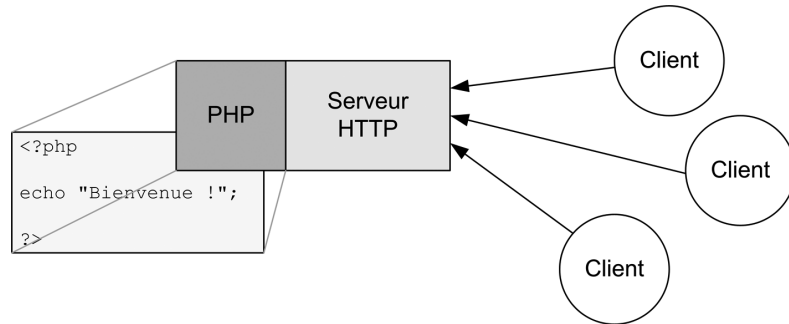
PHP-GTK est la version graphique (client lourd) de PHP. L'avantage de GTK est sa portabilité. Les applications GTK en PHP sont en pleine émergence. Pour en savoir plus sur PHP-GTK, vous pouvez vous rendre sur le site officiel à l'adresse suivante :

▶ <http://gtk.php.net>

Architecture web : PHP + HTTP

C'est l'utilisation la plus courante et également la plus éprouvée de PHP. De nombreux serveurs HTTP, tels que Apache, IIS ou Caudium, sont compatibles avec PHP.

Figure 1-7
PHP associé à un serveur HTTP

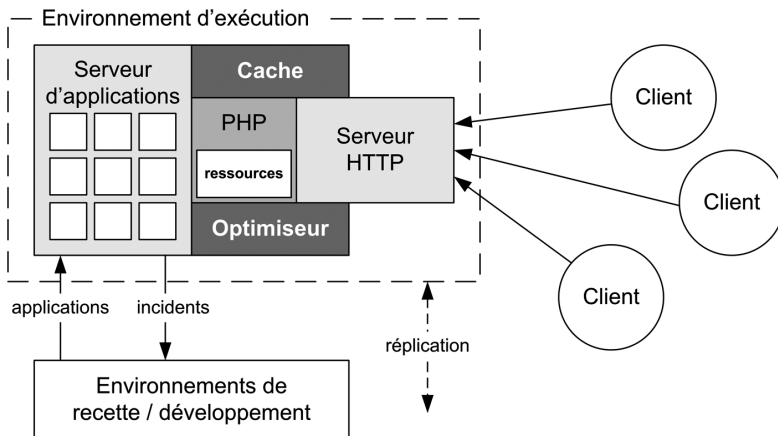


Comme nous l'avons vu précédemment, cette installation de PHP avec un serveur web fait l'objet de plusieurs installateurs pour Windows. En revanche, nous verrons dans cet ouvrage comment installer PHP de manière optimale avec un serveur HTTP en environnements de développement et de production.

Architecture web avancée

Lorsqu'il s'agit de mettre des applications PHP en production et de les maintenir régulièrement, une simple installation de PHP et d'un serveur HTTP ne suffit pas.

Figure 1-8
Une architecture web avancée pour applications PHP



Les concepts d'environnement d'exécution et de serveur d'applications sont nouveaux dans le monde PHP. Les besoins des applications de grande taille évoluent également à grand pas.

La mise en place d'une architecture web avec serveur d'applications et environnement d'exécution nécessite les connaissances et le recul d'un architecte et d'un administrateur. Des sociétés comme Yahoo, Boursorama, Wanadoo et bien d'autres sont les preuves vivantes que PHP est capable de s'intégrer à une architecture web de production conséquente.

Ce livre est en partie consacré à la mise en place d'un environnement propice à l'élaboration d'applicatifs PHP professionnels.

Richesse fonctionnelle et possibilités de PHP

Que peut faire PHP ?

Les possibilités de PHP en tant que plate-forme évolutive sont illimitées. S'il n'existe pas une fonctionnalité dont on a besoin à l'heure actuelle, elle peut exister plus tard.

Mais cela reste rare. Le nombre et la diversité des extensions existantes couvre déjà 98 % des besoins des utilisateurs, professionnels ou non.

Dans la plupart des cas, PHP sert à élaborer des sites dynamiques. De nombreux forums, magasins en ligne, sites de réservation et portails grand public sont réalisés en PHP.

Comme nous l'avons vu précédemment, PHP peut également servir à développer des applications autonomes, généralement en ligne de commande. Plusieurs applications permettent de maintenir un système, générer des configurations serveur ou s'exécutent comme des démons (firewall, etc.).

Une puissance et une simplicité inégalées pour les front-ends

PHP est très souvent utilisé pour la mise en place de front-ends. Vous pouvez parfaitement développer votre application ou votre logique métier en langages Java ou C et la couche présentation en PHP.

Pour cela, de nombreux moteurs de templates, API de bases de données et gestionnaires de données sont disponibles. Selon son fondateur, PHP est lui-même un moteur de templates très efficace.

CULTURE Front-end

Un front-end, ou frontal, est de plus en plus assimilé à un ensemble de pages graphiques constituant l'interface entre l'utilisateur et un autre système qui peut être complètement différent de celui utilisé par le client.

Une puissance et une simplicité inégalées pour les traitements XML

PHP supporte DOM (Document Object Model), SAX (Simple API for XML) et SimpleXML (un outil permettant de manipuler très simplement des documents XML, basé sur le back-end de DOM). Il propose également un support natif pour SOAP et diverses fonctionnalités supplémentaires (transformations XSLT, XML-RPC, etc.).

Traitement simple d'un flux RSS avec SimpleXML

```
<?php
// Ce code affiche les titres de l'actualité Yahoo
$rss_url = 'http://rss.news.yahoo.com/rss/world';
$xml = simplexml_load_file($rss_url);
foreach ($xml->channel->item as $item) {
    echo $item->title."<br />\n";
}
?>
```

Un traitement simple et unifié de trois métastructures : objets/documents/tables

Souplesse et simplicité sont les maîtres mots lorsqu'il s'agit de manipuler des tableaux, des documents XML ou des objets. Ces trois métastructures maintenant parfaitement maîtrisées par PHP sont l'avenir de la manipulation des données informatiques.

Les objets pour l'architecture logicielle, les tableaux pour les données structurées de petite taille, les documents pour l'interopérabilité et l'organisation des données : tout ceci est détaillé dans le chapitre 11.

Performances et montée en charge

Le noyau de PHP : Zend Engine II

Le noyau de PHP est le principal responsable des performances et des fonctionnalités bas niveau, telles que l'approche objet, la gestion de la mémoire, le comportement de l'interpréteur de code et la qualité des opcodes.

CULTURE Que sont les tableaux d'opcodes (opération codes) ?

L'exécution d'un programme PHP se fait en plusieurs étapes : il est d'abord parcouru par un interpréteur et transformé en tableaux d'opérations élémentaires de bas niveau (car très proches du langage machine). Ces opérations élémentaires sont ensuite lues et exécutées par un moteur : Zend Engine. Ce sont ces tableaux d'opérations élémentaires que nous appelons tableaux d'opcodes.

Les premiers objectifs des Zend Engine furent d'assurer des performances optimales. La version 4 de PHP en particulier, a bénéficié de ces améliorations.

Zend Engine II, qui apparaît dans PHP 5, est une refonte de la première version. Tout en maintenant des performances optimales, son objectif est de gérer les mécanismes de programmation orientée objet parmi de nombreuses autres améliorations :

- <http://www.zend.com/php5/zend-engine2.php>

Limite des performances

PHP n'est pas aussi rapide qu'un langage compilé à la main en instructions machines natives, accompagné d'un typage fort et d'une structure rigide, comme le sont les langages C et C++. Si votre application doit gérer des traitements lourds nécessitant un très grand nombre de calculs, vous avez la possibilité de créer facilement une extension en langage C que vous pouvez compiler avec PHP ou en tant que module dynamique.

En revanche, pour un langage interprété, les performances de PHP sont très bonnes. La grande majorité des traitements peuvent être confiés à PHP, même à forte charge.

Peut-on optimiser les performances de PHP ?

La seule chose que PHP ne gère pas par lui-même pour l'instant est la mise en cache du code compilé à la volée (opcodes) qui évite une réinterprétation du code PHP à

chaque requête envoyée par le client. Pour pallier cela, plusieurs optimiseurs d'opcodes sont disponibles : eAccelerator, APC, Zend Optimizer.

Nous verrons dans plusieurs chapitres de cet ouvrage comment optimiser les performances de PHP par des configurations et des réflexes de codage adéquats.

Peut-on déployer des applications PHP sur plusieurs serveurs ?

Plusieurs solutions s'offrent à vous si vous souhaitez déployer vos applications en cluster. En revanche, ce concept est nouveau et pas encore bien abouti avec PHP, bien que de sérieux travaux s'engagent dans ce sens. Voici deux solutions que vous pouvez d'ores et déjà adopter :

- Utiliser un environnement d'exécution qui le permette, tel que la Zend Platform que nous découvrirons au chapitre 17.
- En tenir compte dans votre architecture logicielle. Il est parfaitement possible de gérer une répartition de charge à travers des mécanismes haut niveau assurés par votre code PHP.

CULTURE Cluster

Un cluster, ou grappe d'ordinateurs, est un ensemble de plusieurs machines physiques constituant une seule et unique machine logique. Chaque ordinateur constitue un nœud de la grappe du gros serveur que représente le cluster. Ce type d'architecture se retrouve notamment dans les domaines de la répartition de charge et de la haute disponibilité.

Qualité de développement

Rigueur et élégance avant tout

Le maniement d'outils souples et permissifs nécessite une certaine discipline. Comme nous l'avons vu précédemment, s'imposer des conventions afin d'assurer la pérennité et la fiabilité des développements s'avère primordial.

Savoir développer en PHP est un art que l'on maîtrise peu à peu par la pratique et par une grande diversité de connaissances. C'est un peu comme savoir piloter un avion : c'est très facile au début, il suffit de tirer sur le manche pour décoller, mais cette seule connaissance ne fait pas de vous un bon pilote.

Les méthodes agiles pour le développement d'applications PHP

Elles ont l'avantage d'être adaptées à l'esprit PHP. Elles privilégient l'émergence de la technologie, le changement, l'acquisition rapide de connaissances, la souplesse et la simplicité.

Connaître une méthode agile (eXtreme Programming, Crystal Clear, etc.) en plus de PHP pour du développement en équipe est un atout considérable pour atteindre rapidement le stade des développements fiables.

Le chapitre 2 aborde les pratiques et les valeurs des méthodes agiles. Il est un bon point de départ si vous souhaitez en adopter une.