

PHP Profissional

**APRENDA A DESENVOLVER SISTEMAS PROFISSIONAIS
ORIENTADOS A OBJETOS COM PADRÕES DE PROJETO**

**Alexandre Altair de Melo
Mauricio G. F. Nascimento**

Novatec



CAPÍTULO 1

Apresentação do PHP

Este capítulo e o seguinte têm como objetivos servir como revisão para aqueles que já conhecem PHP, e como guia inicial para aqueles que estão sendo apresentados à linguagem pela primeira vez. Apesar de este livro abordar conteúdos avançados e não ter o propósito de ensinar princípios da linguagem, uma explanação básica da mesma se faz necessário. Se este for o seu caso, parabéns pela decisão em aprender e aprofundar seus conhecimentos em uma das linguagens web que mais cresce em utilização no mundo. Siga adiante por este capítulo. Caso contrário, se o leitor já tiver conhecimentos prévios, pode se sentir à vontade para avançar nestes dois capítulos iniciais e partir diretamente para o Capítulo 3.

Nas linhas a seguir entenda o porquê do PHP ter se tornado uma das linguagens mais utilizadas na web, e como ele vem sendo usado nos mais diferentes projetos de diversas magnitudes.

1.1 Começando com PHP

Afinal o que é PHP? PHP é a abreviação de Hypertext Preprocessor ou em tradução livre “Pré-processador de Hipertexto”. É uma linguagem de script de código aberto que tem como objetivo primário a geração de conteúdo dinâmico para páginas da internet. Isto quer dizer que ao invés de criar um programa para gerar e imprimir HTML, você pode escrever HTML com o código PHP embutido para gerar conteúdo dinâmico. Como as tags HTML são estáticas, cabe ao PHP ou outra linguagem como ASP ou Java a criação do conteúdo dinâmico que vemos na web. Outra vantagem do PHP é que por ser executado no lado do servidor, seu código-fonte não é exibido ao internauta, que apenas terá acesso ao seu HTML resultante.

1.2 Histórico da linguagem

No final de 1994, Rasmus Lerdorf criou uma série de utilitários para monitorar sua página pessoal e obter informações sobre seus visitantes. Com o passar do tempo, mais funcionalidades foram requeridas. Rasmus escreveu então uma implementação na linguagem C destes utilitários, que deu origem ao núcleo do que ficou conhecido como PHP/FI. Este pacote foi o início do PHP como o conhecemos hoje. Originalmente, o nome PHP era um acrônimo para Personal Home Page Tools / Forms Interpreter (Ferramentas para Página Pessoal / Interpretador de Formulários).

Assim como ocorreu com o sistema operacional Linux, diversos entusiastas do software livre tomaram conhecimento do pacote PHP/FI, os quais passaram a contribuir e acrescentar suas idéias ao mesmo. Entre elas estavam, por exemplo, a conexão a banco de dados e exibição de resultados dinâmicos. De 1994 até o ano de escrita deste livro, mais de dez anos se passaram e o PHP está em sua quinta versão, com previsão do lançamento da versão 6 em breve.

No decorrer desses anos podemos verificar que cada versão do PHP teve o seu papel fundamental para ajudar a propagar e difundir sua linguagem. A seguir, veremos de uma maneira geral o que cada uma delas fez pela evolução do PHP:

- **Versão 2** – Em 1997, o PHP/FI 2.0, segunda versão da implementação C, obteve o apoio e reconhecimento de milhares de usuários ao redor do mundo. Aproximadamente 50.000 domínios reportavam sua instalação e uso, constituindo assim uma base de 1% dos domínios da internet. Enquanto isso, havia milhares de pessoas contribuindo com pequenos trechos de códigos para o projeto. Apesar disto, ele foi apenas lançado oficialmente em Novembro de 1997, após demandar a maior parte de sua vida em versões betas. Foi rapidamente substituído pelas releases alfas do PHP 3.0.
- **Versão 3** – Criada por Andi Gutmans e Zeev Suraski em 1997, o PHP 3.0 foi a primeira versão da linguagem que se assemelha ao PHP como conhecemos hoje. Esta versão foi totalmente reescrita, após seus autores descobrirem que o PHP/FI 2.0 era uma linguagem versátil e que poderia ser utilizada para seus projetos acadêmicos de comércio eletrônico. Num esforço conjunto a partir da base de usuários PHP/FI já existente, Andi, Rasmus e Zeev decidiram se unir e anunciaram o PHP 3.0 como uma versão oficial e sucessora do PHP/FI 2.0, cujo desenvolvimento foi totalmente descontinuado. Nasceu assim o PHP 3.0, que além de oferecer aos usuários finais uma infra-estrutura sólida para diversos bancos de dados, protocolos e APIs, a extensibilidade apresentada pela versão atraiu dezenas de desenvolvedores que se juntaram com o propósito de submeter novos módulos.

A possibilidade de ser extensível foi, aliás, a chave do tremendo sucesso do PHP 3.0. Outra característica chave introduzida nesta versão foi o suporte à sintaxe para Orientação a Objetos, ainda que bastante limitada. Para realçar a reestruturação da linguagem, suas versões foram lançadas sob um novo nome, que removeu a impressão do limitado uso pessoal que o PHP/FI possuía até então. Seu nome foi nomeado simplesmente para “PHP”, com o significado para o acrônimo PHP: *Hypertext Preprocessor*. No final de 1998, o PHP contava com uma base de dezenas de milhares de usuários e centenas de milhares de páginas na internet relatando sua instalação. Em seu auge, o PHP 3.0 foi instalado em aproximadamente 10% dos servidores web da internet. Esta versão foi oficialmente lançada em Junho de 1998, depois de ter passado aproximadamente nove meses em testes públicos.

- **Versão 4** – O desenvolvimento de aplicações robustas apenas se tornou viável por causa das novas características do PHP 3.0 e o suporte a uma variedade de banco de dados de terceiros e APIs. Entretanto, esta versão não foi projetada para trabalhar com aplicações muito complexas de forma eficiente. Uma nova *engine*, criada e lançada em meados de 1999, denominada “Zend Engine” (redução dos nomes de Zeev e Andi), foi desenvolvida tendo como objetivo a melhoria dos pontos críticos apresentados até então pela linguagem.

O PHP versão 4.0, baseado nesta *engine*, e acompanhado de uma série de novas características, foi oficialmente lançado em Maio de 2000, quase dois anos após o seu predecessor. Além da melhora do desempenho, o PHP 4.0 incluiu outras características chave, como o suporte para vários servidores web, sessões HTTP, *buffer* de saída, maneiras mais seguras de manipulação de dados de entrada de usuários, dentre outras melhorias. Nesta versão também foi incluído um dígito intermediário para um controle mais granular de versão, a exemplo das versões 4.1.X da linguagem. Até então, em suas versões anteriores, era adotado apenas o padrão 3.X. Na versão 4.1.0 apareceram as variáveis superglobais `$_GET` e `$_POST` permitindo que a diretiva `register_globals` fosse desabilitada. A última alteração significativa da versão foi apresentada na 4.3.0, que introduziu a interface de linha de comando CLI – Command Line Interface.

- **Versão 5** – Versão que significou uma quebra de paradigmas da linguagem, em que finalmente ela passa a suportar a Orientação a Objetos de forma consistente. Baseada na *Zend Engine 2*, essa versão foi lançada oficialmente em Julho de 2004. Também trouxe como novidades o suporte melhorado da manipulação de arquivos XML (via extensão SimpleXML), manipulação de webservices SOAP e REST, suporte melhorado ao MySQL via extensão

MySQLi, novas bibliotecas SQLite, Tidy e aperfeiçoamento com a integração entre a linguagem Perl. Além disso, melhorou o gerenciamento de memória e descontinuou o suporte ao sistema operacional Windows 95.

- **Versão 6** – O futuro está em dar suporte ao Unicode para ajudar na interpretação de vários caracteres, a fim de facilitar o desenvolvimento e manutenção de sistemas globais e com suporte a multiidiomas. Além disso, a nova versão terá características como:
 1. O fim das diretivas `register_globals` e da família `magic_quotes_*`.
 2. Possível remoção da diretiva `safe_mode`.
 3. O lançamento de erro do tipo `E_STRICT` para o retorno por referência a objetos.
 4. Fim do suporte a *Zend Engine* I.
 5. A descontinuidade ao suporte às variáveis da família `HTTP_*_VARS`.

Essas são algumas das futuras novidades que virão nesta nova versão, sendo que sua data de lançamento ainda não está definida. Outras alterações durante o desenvolvimento do PHP 6 poderão ser consultadas (em inglês) na íntegra no site <http://php.net/~derick/meeting-notes.html>.

1.3 Benefícios do PHP

O objetivo da linguagem não é só gerar conteúdo dinâmico para a internet. Ela pode fazer muito mais do que isso, como pode ser comprovado nos tópicos a seguir.

1.3.1 Custo

O PHP não tem custo de licença justamente por ser um software livre. Isto significa que ele pode ser instalado em qualquer máquina, para qualquer número de usuários, sem que isto denote a violação de alguma lei de direito autoral. Seu modelo de licenciamento é o da GPL – General Public License (Licença Pública Geral), largamente utilizado em softwares livres. Para maiores detalhes, consulte o site <http://php.net/license>.

Seu suporte é feito pelas comunidades que crescem dia a dia, tanto no Brasil quanto ao redor do mundo. Estas comunidades são uma das melhores maneiras de se obter suporte, trocar experiências e discutir idéias sobre a linguagem. Ao final do livro são fornecidos links de páginas da internet das principais comunidades em atividades em diversos países.

Apesar de o PHP ser um software livre, a Zend, que é empresa criada por Andi e Zeev, fornece ferramentas como IDE – Integrated Development Environment (Ambiente Integrado de Desenvolvimento), servidores e outros produtos a fim de facilitar o contato com a linguagem. Além disso, ela também oferece a certificação oficial da linguagem, a *Zend Certification*. A Borland e sua subsidiária, denominada CodeGear, também possuem um IDE proprietário, chamado “Delphi para PHP”. Para os programadores Delphi, essa ferramenta pode facilitar o desenvolvimento, já que seu IDE conta com componentes conhecidos destes programadores, como DBGrid, DBNavigator etc. Além disso, o projeto Eclipse da IBM fornece um IDE para desenvolvimento totalmente gratuito, no mesmo estilo do IDE Eclipse para Java. Este último, denominado Eclipse PDT (PHP Development Tools), em nada fica devendo aos melhores IDEs pagos do mercado.

1.3.2 O código PHP é embutido no HTML

O PHP é fácil de utilizar, seja para gerar HTML ou para ser inserido dentro deste. Veja um exemplo prático a seguir:

ola.php

```
<html>
  <head>
    <title>
      Testando o PHP
    </title>
  </head>
  <body>
    <?php
      echo '<b>Alô Mundo. Seja bem-vindo ao PHP.</b>';
    ?>
  </body>
</html>
```

Resultado:

Alô Mundo. Seja bem-vindo ao PHP.

Quando um servidor solicita a página `ola.php`, o usuário verá impresso no navegador a mensagem mostrada anteriormente como resultado. Nada de impressionante até aqui: uma página HTML tradicional contendo uma mensagem. Se usuário visse o código-fonte da página, o navegador mostraria algo como demonstrado na Figura 1.1.



Figura 1.1 – Marcação HTML: o código PHP não é mostrado.

Aqui está a grande sacada! Por ser pré-interpretado no servidor como descrito no início do capítulo, o usuário só consegue ler as tags HTML e não o código PHP em si.

1.3.3 Outras vantagens da linguagem

Além do custo e de trabalhar perfeitamente com HTML, podemos ainda citar sobre o PHP os seguintes benefícios:

- Manipulação de arquivos como texto plano, do tipo PDF, documentos DOC, entre outros, incluindo sua criação, exclusão, alteração etc.
- Geração de imagens dinâmicas para validação de formulários.
- Criptografia de dados.
- Definição de cookies e sessões.
- Definição de interfaces para webservices.
- Manipulação de arquivos XML.
- Suporte a vários bancos com acesso nativo como: DB2, Firebird, Informix, Interbase, MySQL, Oracle, PostgreSQL, SQL Server etc.
- Suporte a vários padrões e interfaces como: COM, CORBA, POP3, IMAP etc.

Além das características citadas anteriormente (que também podem ser encontradas em outras linguagens), listamos a seguir alguns pontos que podem ajudá-lo na escolha do PHP como sua plataforma de desenvolvimento:

- Independência de plataforma: roda em Windows, Linux, Unix, Mac etc.
- Suporte a Orientação a Objetos consistente na versão 5.
- Curva de aprendizado reduzida para quem está aprendendo a linguagem.
- Sintaxe semelhante com a de linguagens consagradas como C e Java.
- Integração com vários servidores web como Apache, IIS, Xitami, entre outros.
- Servidores de hospedagem tanto gratuitos como pagos em grande quantidade no Brasil e no mundo.
- Documentação oficial em português.
- Possibilidade do desenvolvimento também para desktop, usando a mesma linguagem. Hoje já é possível desenvolver com PHP em modo visual utilizando janelas a exemplo do Java, Delphi, C++, Visual Basic etc. Podemos criar aplicações para trabalhar tanto em rede ou *stand-alone* independente do sistema operacional. Isso só foi possível graças à evolução da linguagem e à inclusão da biblioteca GTK. Para conferir o que o PHP pode oferecer neste sentido, tome como ponto de partida a página <http://gtk.php.net>. Este endereço também conta com diversos exemplos disponíveis.

Estes são alguns fatores adicionais que o PHP pode oferecer e que podem influenciar sua decisão na adoção desta linguagem. Os fatos estão aí e mostram que a linguagem amadureceu e está pronta para o seu próximo projeto.

Quanto à utilização do PHP como plataforma de desenvolvimento, veja a Figura 1.2 auditada pela Netcraft, que mostra que mais de 19 milhões de domínios utilizam a linguagem. A Netcraft é uma empresa inglesa que realiza monitorias e pesquisas de mercados relacionadas com a internet, incluindo sistemas operacionais, servidores web, entre outros. Seus relatórios e artigos, considerados *benchmarks* pelo mercado, são publicados em seu site <http://netcraft.com> com regularidade.

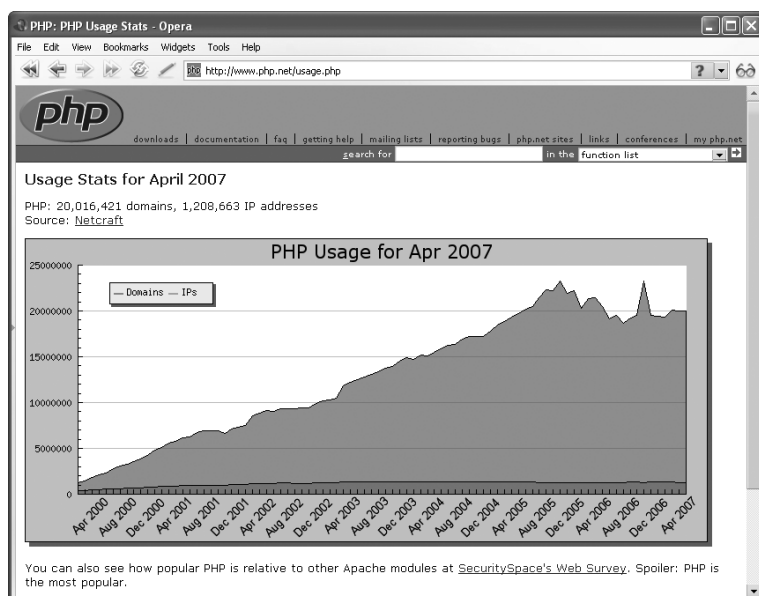


Figura 1.2 – Números do PHP no mundo. Fonte Netcraft – Abril/2007.

1.4 Ambiente do PHP

Como já foi dito, o PHP é amplamente utilizado principalmente para desenvolvimento web, seja na plataforma Unix/Linux quanto na Windows. Vamos descrever a seguir os ambientes de configuração da linguagem nessas duas plataformas.

1.4.1 Introdução sobre LAMP e WAMP

Os termos LAMP e WAMP serão amplamente utilizados no decorrer deste livro. De forma simplificada, eles especificam padrões de ambientes freqüentemente configurados para a linguagem PHP.

Apesar da inexistência de um modelo rígido para sua constituição, estes modelos contêm alguns itens básicos como: servidor web Apache, servidor de banco de dados MySQL e um interpretador para uma ou mais linguagens de programação, entre elas o PHP, Python ou Perl. Justamente a partir da junção destes componentes com a escolha do sistema operacional da máquina (Windows ou Linux) é que são originados os referidos acrônimos LAMP e WAMP. Especificamente neste livro, trataremos sobre a linguagem PHP. De forma concisa temos:

- **LAMP** – junção das iniciais de Linux, Apache, MySQL e PHP.
- **WAMP** – junção das iniciais de Windows, Apache, MySQL e PHP.

Por serem constituídos a partir de diferentes aplicativos, todos eles independentes entre si, pode-se afirmar que estes ambientes são modulares por essência. Tais componentes serão descritos detalhadamente mais adiante. Por ora destacamos a origem destes termos.

Apesar de serem constituídos a partir da mesma base de aplicativos, não existe uma regra explícita para sua distribuição na máquina, como, por exemplo, uma estrutura de arquivos e diretórios padronizada, tampouco detalhes de suas configurações internas. Na prática, os termos LAMP e WAMP referem-se a modelos estruturais lógicos para o ambiente PHP. Isto quer dizer que duas máquinas que contenham cada uma seu próprio LAMP, por exemplo, não necessariamente terão as mesmas estruturas internas de arquivos e parametrizações.

Não tão comum como estes dois termos, um terceiro, chamado AMP (ou XAMP), também pode ser encontrado em sites da internet e livros: nada mais é do que uma generalização dos anteriores. Refere-se a um ambiente com o servidor web Apache, banco de dados MySQL, linguagem PHP, sem especificar seu sistema operacional. Como descrito, em se tratando de um ambiente com tais características, podemos tomar a liberdade de não explicitar este sistema, visto que na maior parte das aplicações não haverá diferenças na utilização do ambiente, sendo ele baseado no Windows ou Linux. É o que se chama de abstração do sistema da máquina.

É importante ressaltar que os componentes do termo AMP são disponibilizados individualmente na internet e de forma gratuita, sem a necessidade da aquisição de licenças proprietárias de seus respectivos softwares. Isto quer dizer que seus componentes de base representam custo zero de compra e utilização. Especificamente o LAMP, constituído com base no sistema operacional Linux (também disponibilizado livremente na internet), forma um ambiente genuinamente livre de licenças proprietárias comerciais. Justamente por este e outros motivos que é o ambiente mais freqüentemente utilizado para a linguagem PHP, sendo também o adotado como padrão para a elaboração dos exemplos práticos deste livro e recomendado aos leitores pelos seus autores.

A escolha dos componentes AMP não é devido unicamente a sua disponibilidade gratuita na internet. É principalmente pelo fato de que todos estes itens já foram (e ainda são) constantemente testados a exaustão, corrigidos e melhorados ao longo de suas existências. Pode-se com certeza afirmar que eles são robustos, consistentes e já passaram e foram aprovados pelos mais diferentes testes de performance, em diversas configurações de máquinas e de sistemas. Além disso, são continuamente discutidos e aperfeiçoados por suas respectivas comunidades, desenvolvedores e programadores distribuídos ao redor do mundo.

Mais adiante será feita uma exposição completa sobre as diferentes formas de instalação e configuração deste ambiente. Será adotada uma configuração de um ambiente básico para o desenvolvimento do PHP, que será utilizada para todos os exemplos e práticas descritas no decorrer do livro.

1.4.2 Papel de cada componente envolvido no ambiente

- **Sistema operacional Windows ou Linux** – Formam a camada de base dos demais aplicativos instalados na máquina. Em se tratando dos ambientes LAMP e WAMP, o mais comumente utilizado (inclusive neste livro) é o primeiro deles, baseado no sistema Linux. Muitos fatores justificam esta escolha, entre eles: apresenta ótimos índices de performances para este tipo de ambiente; pode ser configurado com grande flexibilidade a partir de suas distribuições, de diversas maneiras distintas, de acordo com os requisitos da máquina e dos sistemas; possui uma vasta gama de comunidades que o apóiam e o mantêm; é seguro; pode ser administrado remotamente; pode ser instalado com ou sem um ambiente gráfico (este último não é necessário ao LAMP). É disponibilizado na internet pela licença GNU GPL, em diversas distribuições ou “sabores”, entre eles: Red Hat, Debian, Suse Linux, Slackware, Ubuntu, Fedora etc.
- **Servidor web Apache** – Desenvolvido pela Apache Software Foundation, é um servidor web de código aberto, disponibilizado livremente na internet. De acordo com a Netcraft, o servidor web Apache detinha no início de 2007 um *market share* de aproximadamente 60% para o mercado de servidores web. Atualmente conta com versões para as principais plataformas existentes, entre elas o Linux e o Windows, sendo altamente recomendado para o primeiro sistema por questões de desempenho. Possui uma vasta gama de características, configurações e ferramentas, entre elas: um poderoso sistema de registro de logs, alto nível de segurança, fina granularidade de configuração, implementação de *hosts* virtuais, suporte a SSL (Secure Sockets Layer), entre outros. Seu site oficial é <http://httpd.apache.org>.
- **MySQL** – Desenvolvido pela MySQL AB, é um servidor de banco de dados de código aberto, disponibilizado a partir de diferentes formas de licenças na internet, incluindo versões gratuitas. A mais recente delas, a 5.0, é bastante robusta e possui as principais funcionalidades esperadas deste tipo de serviço, incluindo: integridade referencial, *stored procedures*, *triggers*, *views*, suporte a campos BLOB (Binary Large Objects) etc. Seu site oficial é <http://mysql.org>.

- **PHP** – É uma das linguagens de scripts mais utilizadas para os ambientes de sistemas para a internet. Possui atualmente inúmeras comunidades, grupos de discussões e entusiastas ao redor do mundo. Sua mais recente versão recebeu incremento de novas funcionalidades e suporte mais consistente ao paradigma da programação Orientada a Objetos, extensões para webservices, novas bibliotecas internas etc. Seu site oficial é <http://php.net>.

1.4.3 Juntando todas as partes

A Figura 1.3 ilustra um típico ambiente para um servidor LAMP / WAMP. Repare que todos os componentes listados no tópico anterior podem ficar residentes dentro deste servidor.

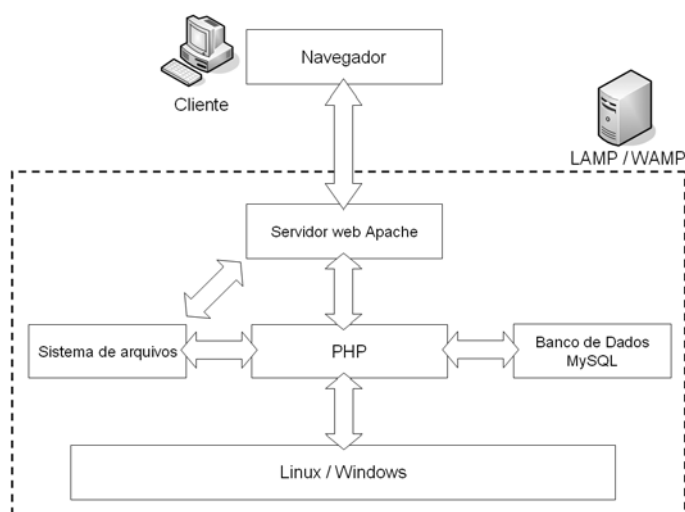


Figura 1.3 – Estrutura básica de um servidor LAMP / WAMP.

O único componente deste servidor que é visível ao mundo externo é o servidor web Apache. É ele quem escuta, trata de todas as requisições do tipo HTTP e HTTPS geradas pelos navegadores clientes e gerencia todo o seu processamento e resposta. Por ser teoricamente a única porta de entrada aos sistemas externos, sua segurança torna-se fator crítico de atenção aos administradores de sistemas, visto que grande parte dos ataques de crackers visa justamente descobrir uma maneira de burlar este serviço e obter acesso ao servidor.

Caso a requisição recebida pelo Apache seja um arquivo do tipo HTML ou figura, ele consulta o seu sistema de arquivos e, se este estiver disponível, é retornado ao cliente. Por outro lado, se o tipo solicitado tiver uma extensão especial, que deva ser tratado por algum sistema externo ao Apache (como, por exemplo, arquivos do tipo

CGI (Common Gateway Interface), PHP, Perl, Python etc.), esta requisição é delegada ao seu responsável. Este irá efetuar o devido tratamento, como, por exemplo, realizar um cálculo aritmético, e retornará a resposta ao Apache. Este por sua vez, de posse da requisição processada pelo sistema externo, entregará a resposta ao navegador do cliente. Eventualmente este agente necessitará realizar consultas a uma base de dados relacional, razão pela qual na Figura 1.3 o sistema do PHP apresenta uma conexão com o banco de dados MySQL.

1.5 Instalação básica de um ambiente LAMP e WAMP

Existem diversas maneiras distintas para se instalar um ambiente LAMP ou WAMP. Este pode ser instalado desde forma rápida e prática, a partir de pacotes totalmente pré-configurados, disponibilizados na internet, até a instalação e configuração manual de cada um de seus componentes.

Como todo o restante do livro abordará exemplos práticos num servidor AMP, é altamente recomendável que o leitor possua um ambiente próprio para testes dos seus scripts e aplicativos. Ele pode estar disponível em sua própria máquina ou a partir de um servidor de sua rede, onde este possua acesso aos arquivos de scripts. Caso ainda não o possua, serão apresentados nos próximos tópicos exemplos práticos para sua instalação e configuração.

A seguir serão descritas três formas distintas para a realização desta tarefa. Elas são suficientes para a utilização e prática de todos os exemplos propostos neste livro. É importante ressaltar que, como boa prática de segurança, tais ambientes nunca deverão ser utilizados em ambientes de produção. Para isto, seriam necessárias configurações e instalação de softwares adicionais neste servidor para sua segurança e manutenção, o que foge do escopo deste livro.

1.5.1 Pacote pré-configurado

Existem alguns projetos para manutenção e pré-configuração de ambientes LAMP e WAMP. Na grande maioria dos casos, eles podem ser baixados gratuitamente em seus sites oficiais pela internet.

A grande vantagem da instalação de um servidor por este método é em relação ao tempo demandado e a facilidade em sua execução. É a opção mais recomendada para usuários inexperientes no ambiente, ou para os que desejam montar um servidor para desenvolvimento em alguns minutos para alguma tarefa específica ou pessoal.

Ao invés de instalar e configurar cada um dos componentes envolvidos, estes pacotes oferecem ao usuário uma configuração básica pré-montada. A desvantagem deste método reside também justamente neste ponto: nem sempre o usuário poderá alterar uma configuração com facilidade ou a atualização individual de um componente nem sempre será possível. Neste caso, é necessário que os responsáveis pelo projeto atualizem e disponibilizem uma nova versão de seu produto.

Diversos exemplos de opções disponíveis poderiam ser utilizados atualmente, tanto para plataformas Linux quanto Windows. A seguir, um breve comentário sobre algumas delas:

- **XAMPP** – Interessante projeto AMP, que disponibiliza versões para mais de uma plataforma de seu pacote, entre elas: Windows, Linux, Mac OS X e Solaris. Possui uma série de extensões pré-configuradas para o PHP e para o Apache. Contém atualmente o PHP 5.2, Apache 2.2, MySQL 5.0, SQLite, além de uma série de módulos, extensões e aplicativos, como, por exemplo, para controle e gerenciamento do banco MySQL, o phpMyAdmin. Seu site oficial é <http://apachefriends.org/en/xampp.html>.
- **Vertrigo** – Versão disponível apenas para Windows. Disponibiliza um ambiente completo e de fácil manutenção para o usuário. É considerado um dos melhores pacotes para esta finalidade. Com um organizado painel de administração, disponibiliza ao usuário ferramentas para configuração das principais diretivas do PHP, MySQL e Apache, atalhos para rápida visualização e controle de seus arquivos de configuração, comandos de inicialização e encerramento de serviços, entre outros. Contém atualmente o PHP 5.2, Apache 2.2, MySQL 5.0, além de aplicativos de controle do banco MySQL, o phpMyAdmin e SQLite. Seu site oficial é <http://vertrigo.sourceforge.net>. As figuras 1.4 e 1.5 mostram algumas telas da ferramenta.

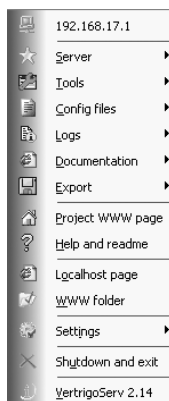


Figura 1.4 – Painel de administração do Vertrigo.

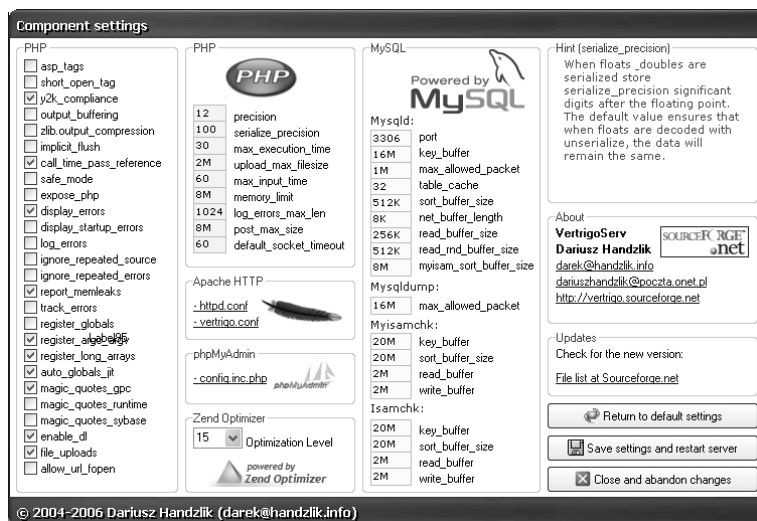


Figura 1.5 – Configuração das diretivas de cada componente.

- **EasyPHP** – Outra opção disponível para a plataforma Windows que também contém os principais aplicativos encontrados anteriormente. Seu site oficial é <http://easyphp.org>.

1.5.2 Máquina virtual pré-configurada

Esta opção baseia-se no conceito de virtualização de máquinas. Ela consiste no procedimento de rodar um ou mais servidores em um ambiente totalmente isolado da máquina real. Estas máquinas podem contar com sistemas operacionais, aplicativos e configurações distintas e independentes entre si, incluindo a máquina real (chamada de hospedeira ou *host machine*) que os suporta, através do aplicativo de virtualização.

Como diferentes “máquinas” e ambientes podem ser configurados através desta técnica, ela é interessante na medida em que um aplicativo pode ser testado em diversas combinações diferentes de hardware e software, antes de ser disponibilizado em um ambiente de produção. No caso de sistemas criados com a linguagem PHP, tais testes podem ser efetuados inclusive entre sistemas operacionais distintos. Existem atualmente no mercado diversos softwares que oferecem esta funcionalidade, entre eles o Xen (<http://xensource.com>), o Microsoft Virtual PC (<http://microsoft.com/windows/virtualpc>) e o VMware Server (<http://vmware.com>).

Uma das maiores vantagens na utilização destas máquinas virtuais é que algumas delas disponibilizam em seus sites máquinas virtuais pré-montadas e configuradas para inúmeros propósitos diferentes. Um destes casos é o VMware Server, que

oferece na página <http://vmware.com/appliances> algumas destas máquinas, chamadas *Virtual Appliances*, baseadas na plataforma Linux. Existe a possibilidade de baixar máquinas com ambientes LAMP completos, inclusive com aplicações funcionais baseadas no PHP. Alguns exemplos são:

- **rPath LAMP Appliance** – oferece um ambiente “enxuto”, com o servidor web Apache, MySQL, PostgreSQL, PHP, Python, Perl, além de uma interface web completa para administração destes aplicativos. Disponível em <http://vmware.com/appliances/directory/465>.
- **Mambo Virtual Appliance** – ambiente web com o aplicativo Mambo Server CMS, escrito em PHP, voltado para gerenciamento de conteúdo de sites da internet. Disponível em <http://vmware.com/appliances/directory/520>.
- **SugarCRM Appliance** – ambiente web com o aplicativo SugarCRM, escrito em PHP, voltado para gestão comercial de clientes. Disponível em <http://vmware.com/appliances/directory/211>.
- **MediaWiki Appliance** – sistema Linux com distribuição Fedora 5, com o aplicativo MediaWiki, para gerenciamento colaborativo de conteúdo de sites da internet, entre eles a enciclopédia eletrônica Wikipédia. Disponível em <http://vmware.com/appliances/directory/180>.

A desvantagem da técnica de virtualização com o uso de máquinas virtuais pré-montadas é semelhante à apresentada no item anterior. Nem sempre será possível atualizar ou modificar um de seus componentes de maneira individual. Geralmente estas atualizações são feitas pelos respectivos responsáveis por cada um destes projetos.

1.5.3 Instalação manual

Esta é certamente a forma mais flexível de instalação do servidor, uma vez que cada componente do ambiente LAMP ou WAMP é instalado e parametrizado de maneira manual e individual pelo próprio usuário. Ao contrário das duas opções anteriores, cabe a esta pessoa a manutenção, integração e atualização deste ambiente.

A seguir será exposto um processo prático de instalação de um ambiente LAMP a partir da distribuição Linux Debian “Sarge” versão 3.1, com a utilização da interface de linha de comando. Considera-se já instalado um conjunto mínimo de utilitários para a execução desta tarefa, como, por exemplo: o APT (Advanced Package Tool) do Debian, para controle dos pacotes do sistema e o TAR para empacotamento de arquivos. Para o procedimento ser efetuado é necessário o acesso à conta de supe-

usuário do sistema. Ao final da operação, este ambiente LAMP contará com os seguintes componentes e versões: Apache 2.2.6, PHP 5.2.4 e MySQL 5.0.45. Tais versões são as mais recentes encontradas nos seus respectivos sites oficiais no momento de escrita deste livro.

1.5.3.1 Preparação do ambiente

O usuário deverá ter uma noção básica sobre os principais comandos do Linux para efetivar o procedimento, pois como este assunto foge do escopo do livro, não serão tratados seus detalhes envolvidos. Caso seja necessário, o leitor poderá consultar livros e sites da internet que contenham documentação mais detalhada para esta finalidade.

Esta instalação pode ser feita a partir de uma máquina real ou virtual (verificar tópico anterior, com a utilização do VMware Server) com o Linux Debian. É importante notar que, assim como os itens anteriores, esta máquina tem o propósito apenas para testes do ambiente. Ela nunca deverá ser utilizada num ambiente real de produção. Além disso, os procedimentos descritos são meramente ilustrativos para concepção do sistema, podem eventualmente variar de acordo com a versão do sistema e dos componentes utilizados e não abrangem por completo todas as opções disponíveis para os seus módulos.

Para iniciarmos o procedimento de configuração do sistema, será definida uma conta de usuário, identificado por `desenv`, associado ao grupo `desenv`, que será responsável pelo download dos pacotes, pré-configuração, compilação e utilização do sistema. O superusuário `root` executará as rotinas de instalação destes pacotes e controle sobre a inicialização e encerramento de serviços.

Poderão ser utilizados o navegador Lynx ou o utilitário GNU Wget para realizar o download pela internet de todos os componentes. Os sites oficiais destes dois utilitários são, respectivamente, <http://lynx.browser.org> e <http://gnu.org/software/wget>. Caso seu sistema não os possua, eles poderão ser baixados e instalados a partir do utilitário `apt-get`, um dos componentes do pacote do APT do sistema Debian.

Para efeito ilustrativo, os trechos de códigos expostos neste livro utilizarão um prefixo com o caractere cifrão (\$) para representar comandos do usuário. As demais linhas são representações de saídas de texto do sistema.

O diretório base será adotado com o nome de `www`. Ele será criado no diretório `home`, um dos padrões na grande maioria dos sistemas Linux / Unix. Dentro do `www`, serão criados os diretórios `public_html` e `sources`, respectivamente para nossos códigos-fonte e arquivos de componentes baixados pela internet. Em seguida, serão concedidos usuário e grupo `desenv` para toda a estrutura em `www`.