BIREME / PAHO / WHO

Latin American and Caribbean Center on Health Sciences Information

VHL Model

Installation manual of pre-required software to VHL applications

Version 2.0 rev. 1

Sao Paulo - July 2010

Installation manual of pre-required software to VHL applications

Card catalog

BIREME / PAHO / WHO (Brazil)

Installation manual of pre-required software to VHL applications. / BIREME / PAHO / WHO. Sao Paulo : BIREME / PAHO / WHO, July 2010.

41 p.

1. User manual. 2. Information access. 3. Information systems. 4. Information management. 5. Public health. 6. Public Health services. I. BIREME II. Title

Warning - Any mention in this document to companies, institutions, persons or products are not an endorsement or recommendation given by BIREME / PAHO / WHO, thus it does not mean a preference to a similar one, cited or not.

BIREME / PAHO / WHO

Latin American and Caribbean Center on Health Sciences Information

Rua Botucatu 862 V Clementino

This document was produced with the Documents Conformation Methodology (NorDoc) developed by BIREME.

Table of contents

Abbreviations used

- **ANSI**. American National Standards Institute.
- **ASCII**. American Standard Code for Information Interchange.
- **BIREME**. Latin American and Caribbean Center on Health Sciences Information.
- **BVS**. Biblioteca Virtual em Saúde (*see* VHL).
- **CGI**. Common Gateway Interface.
- **DTD**. Document Type Definition.
- **HTML**. HyperText Markup Language.
- **HTTP**. HyperText Transfer Protocol.
- **iAH**. Interface for Access on Health Information.
- **ISO**. International Organization for Standardization.
- **PAHO**. Pan American Health Organization.
- **SGML**. Standard Generalized Markup Language.

- **UMLS**. Unified Medical Language System.
- **UNESCO**. United Nations Educational, Scientific and Cultural Organization.
- **UNIFESP**. Universidade Federal de São Paulo
- **URL**. Universal Resource Locator.
- **VHL**. Virtual Health Library.
- **WHO**. World Health Organization.
- **XHTML**. eXtensible HyperText Markup Language.
- **XML**. eXtensible Markup Language.
- **XSL**. eXtensible Stylesheet Language.
- **XSLT**. eXtensible Stylesheet Language Transformations.

How to Use this Manual

This manual was developed to help system analysts and librarians specialized in IT with the installation and configuration of the software required by programs and applications within the VHL Model.

The first two chapters explain how to install and configure Apache and PHP under Windows. The third chapter is addressed only to analysts as we assume that users are knowledgeable about the Linux operating system and about configuration, compilation and installation of software on this platform.

Preface

About BIREME

Year after year, BIREME has been following its mission of being a center dedicated to scientific and technical health information for the region of Latin America and the Caribbean. Founded in Brazil in 1967, under the name of Regional Medicine Library (which the acronym BIREME comes from), it has always met the growing demand for up-to-date scientific literature from the Brazilian health systems and the communities of healthcare researchers, professionals and students. Then, in 1982, its name changed to Latin-American and Caribbean Center on Health Sciences Information so as to better express its dedication to the strengthening and expansion of the flow of scientific and technical health information across the region, but kept the acronym.

Networking, based on decentralization, on the development of local capacities, on sharing information resources, on developing cooperative products and services, on designing common methodologies, has always been the foundation of BIREME's technical cooperation work. It has been like this that the center established itself as an international model that fosters professional education with managerial and technical information with the adoption of information and communication paradigms that best meet local needs.

The main foundations that gave origin and which support the existence of BIREME are following:

- ✓ access to scientific and technical health information is essential for the development of health;
- ✓ the need to develop the capacity of Latin American and Caribbean countries to operate their sources of scientific-technical health information in a cooperative and efficient manner;
- ✓ the need to foster the use and to respond to the demands for scientific-technical health information from governments, health systems, educational and research institutions.

BIREME, as a specialized center of the Pan-American Health Organization (PAHO)/ World Health Organization (WHO), coordinates and conducts technical cooperation activities on the management of scientific information and knowledge with the aim of strengthening and expanding the flow of scientific health information in Brazil and in other Latin American and Caribbean countries as a key condition for the development of health, including its planning, management, promotion, research, education, and care.

The agreement that supports BIREME is renewed every five years by the members of the National Advisory Committee of the institution (PAHO, Brazilian Ministry of Health, Brazilian Ministry of Education and Culture, Secretary of Health of the State of São Paulo, and Federal University of São Paulo – Unifesp). The latter provides the physical infrastructure necessary for the establishment of the institution.

In 2004 the institution took on the responsibility of becoming a knowledge-based institution.

The Virtual Health Library (VHL)

With the rise and consolidation of the internet as the prevailing means of access to information and communication, BIREME's technical cooperation model evolved, as of 1998, to build and develop the Virtual Health Library (VHL) as a common space for the convergence of the cooperative work of producers, intermediaries, and users of information. The VHL promotes the development of a network of sources of scientific and technical information with universal access on the internet. For the first time there has been a real possibility of equal access to health information.

To BIREME, the Virtual Health Library is a model for the management of information and knowledge, which includes the cooperation and convergence between institutions, systems, networks, and initiatives of producers, intermediaries, and users in the operation of networks of local, national, regional and international information sources favoring open and universal access.

Today, every country in Latin America and the Caribbean (Region) participates either directly or indirectly in the cooperative products and services offered by the VHL, which includes over 1,000 institutions in more than 30 countries.

The VHL is simulated in a virtual space of the internet formed by a collection or network of health information sources in the Region. Users of different levels and locations can interact and navigate in the space of one or many information sources, regardless of where they are. Information sources are generated, updated, stored and operated on the internet by producers, integrators, and intermediaries, in a decentralized manner, following common methodologies for their integration in the VHL.

The VHL organizes information in a structure that integrates and interconnects reference databases, specialist directories, events and institutions, a catalogue of the information resources available on the internet, collections of full texts with a highlight for the SciELO (*Scientific Electronic Library Online*) collection of scientific journals, selective information dissemination services, information sources to support education and decision-making, news, discussion lists, and support to virtual communities. The space of the VHL is, therefore, a dynamic and decentralized network of information sources based on which it is possible to retrieve and extract information and knowledge to support health decision-making processes.

The Virtual Health Library can be visualized as a distributed base of scientific and technical health knowledge that is saved, organized and stored in electronic format in the countries of the Region, universally accessible on the internet and compatible with international databases.

Simple installation of APACHE 1.3.xx in Windows environments

About this chapter

This chapter is about the installation of APACHE for use with VHL products in Windows environments. As a pre-requisite, Windows NT, 2000 or above should be installed.

Although this procedure works correctly in older Windows versions, such installations are not regularly assessed by BIREME's team, and therefore their use is not recommended.

VHL products work with versions 1.3.xx, but version 2.xx has not yet been assessed for all products.

Your queries and suggestions may be addressed to bvs.technical.support@listas.bireme.br.

Obtaining the software

The Web server may be obtained at the APACHE site

http://www.apache.org/.

Installing APACHE

In the Windows environment APACHE can be installed with an installer, which simplifies the whole process. If you have no experience with this type of program, it is advisable to perform the installation as proposed by the system.

Click twice on the APACHE installer and follow the instructions. The correct installation of the system should generate the following configuration on your computer:

1. The APACHE programs will be installed at
   C:\PROGRAM FILES\APACHE GROUP\APACHE
2. Unless otherwise indicated, APACHE will be installed as a Windows automatic service; otherwise you will need to start the server manually.
3. In any case, access to APACHE's configuration and control may be effected through the following route:
   Start > Programs > Apache HTTP Server >
4. The important folders for installation are:
   a) **C:\PROGRAM FILES\APACHE GROUP\APACHE\htdocs**, where the documents that the server will offer to visitors are stored.
   b) **C:\PROGRAM FILES\APACHE GROUP\APACHE\conf**, where the file httpd.conf is located. This file effects the configuration of the server.
   c) **C:\PROGRAM FILES\APACHE GROUP\APACHE\cgi-bin**, where the specific programs used by the server to access other programs or data are stored.

Configuring APACHE

The detailed information to configure APACHE can be found in the program's documentation. From VHL's point of view, no major changes to the product's standard installation are required. The use of PHP has to be configured to prepare the system for the installation of VHL products.

The configuration procedure basically consists of editing the file httpd.conf, adding some lines to it.

Before beginning the configuration, make sure that the Web server is working correctly. If the APACHE installation is successful, the URL **http://localhost/** should show a page indicating that your server is working.

Where to find help

The VHL is a collective construction, and all the information on the use of its products is available from the technical support site, especially in the forums of each product. For more information visit the site http://bvsmodelo.bvsalud.org/ or direct your queries in writing to the support team of BIREME/PAHO/WHO at bvs.technical.support@listas.bireme.br.

Simple installation of PHP 4.4.x in Windows environments

About this chapter

This chapter addresses the installation of PHP for use with VHL products in Windows environments. It requires Windows NT, 2000 or above installed.

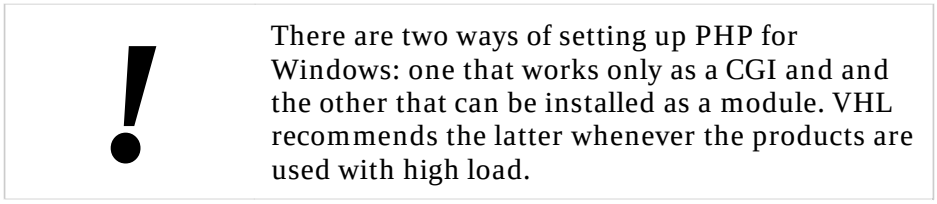
Although this procedure work correctly in older Windows versions, such installations are not regularly assessed by the BIREME team and their use is therefore not recommended.

The use of PHP in VHL products is targeted at the handling of data and processing of XML documents for their presentation through transformations using XSL files.

Your queries may be directed to bvs.technical.support@listas.bireme.br.

Obtaining the software

The latest PHP version is available from **http://www.php.net/**.



Installing PHP

Installing PHP in a Windows environment is very simple. Just decompress the file into a folder and copy the extensions that PHP is going to use into the system's directories.

Following are the steps for a PHP installation compliant with VHL products:

a) Create a "**php**" folder in the root directory of the hard disk, namely, "**c:\php**";
b) Decompress the file that contains the PHP components into the directory created in (a);
c) Copy the DLL files of the directory **c:\php\dlls*.dll**, into the system extension directory (usually **WINDOWS\SYSTEM32** or **WINNT\SYSTEM32**). Only the DLL files that handle XML documents are required, but it seems easier to prepare the installation for future adjustments. In any case, the unused files do not affect the system and are controlled through the PHP configuration file (php.ini);
d) Copy the file **c:\php\php4ts.dll** into the system directory, namely **C:\WINDOWS\SYSTEM32** or **C:\WINNT\SYSTEM32**;
e) Use the **php.ini-dist** file as a basis for PHP configuration. Review the "PHP Configuration" section;
f) Save the file created in (e) as "**php.ini**" into the following Windows directory: (**C:\WINNT** or **C:\WINDOWS**);
g) Configure your Web server to operate programs written in PHP. The VHL uses APACHE as a Web server, and an example of configuration for this server is in the "APACHE with PHP" section.

Configuring PHP

The detailed documentation on PHP configuration can be found at **c:\php\install.txt**. In this section we will address only the procedure for configuring the aspects required for the operation of VHL programs.

The procedure basically consists of editing the file php.ini-dist, adjusting some parameters and saving it as php.ini into the Windows directory.

1. Make a copy of the file **`c:\php\php.ini-dist`**. Save it as **`c:\windows\php.ini`**;
2. Edit the file **`c:\windows\php.ini`** (you may use the NOTEPAD) and adjust the following parameters (look for the appropriate line in the document):
   - **`register_globals = On`**
   - **`default_charset = "iso-8859-1"`**
   - **`extension_dir = "c:\php\extensions"`**
   - **`extension=php_xslt.dll`**

Apache with PHP

Before beginning the configuration make sure that the Web server is working properly. Use your browser and go to the address of the Web site to check this.

If APACHE has been correctly installed, the URL **`http://localhost/`** should show a page indicating that your server is working.

To configure APACHE, edit the file **`httpd.conf`**, available at the folder <*path_to_apache*>\conf\ for example:

`C:\APACHE\conf\`.

Edit this file with a text editor (for example NOTEPAD), and lines will be added that will allow APACHE to handle PHP programs.

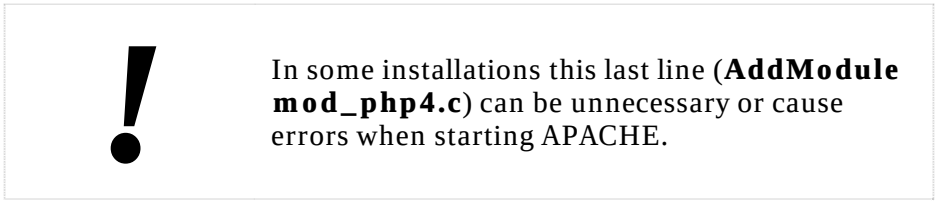
The configuration file is structured in sections. Make sure that the lines that you include are in the correct section.

As mentioned above, PHP can work in CGI mode or as an APACHE module (but not in both modes). The primary difference lies in the fact that the second option behaves better with heavier loads.

PHP as an APACHE module

For this option, configure the following lines in the file **`httpd.conf`**:

1. Add the module to the list of Apache APIs:
   `LoadModule php4_module "c:/php/sapi/php4apache.dll"`
2. A line that includes the type of php file into the MIME table:
   `AddType application/x-httpd-php .php`
3. A line that includes the Apache support file for PHP:
   `AddModule mod_php4.c`



PHP in CGI mode

The CGI option requires the inclusion of three lines in the configuration file.

1. A line of alias for scripts indicating the php path
   `ScriptAlias /php/ "c:/php/"`
2. A line that includes the type of php file into the MIME table
   `AddType application/x-httpd-php .php`
3. A line that indicates which executable file that has to be called to process php-type files
   `Action application/x-httpd-php "/php/php.exe"`

Testing the installation

The last step for configuring PHP is ensuring that APACHE is processing the PHP files correctly.

To this effect, we will create a new file called “info.php”.

1. Make sure that the file effectively has the extension php and not another one (Windows hides the extensions of the files that it defines as “known”).
2. The file info.php should contain a line with the following text:
   `<? phpinfo(); ?>`
3. Save the file into the directory of documents configured in your Web server. In an APACHE standard installation, this directory is called “htdocs”.
4. Restart the Web server
   `Start > Programs > Apache HTTP Server > Restart Apache`
5. Use your internet browser to visit the URL:

`http://localhost/info.php`

This page should show information on your PHP installation, as shown on the picture below.

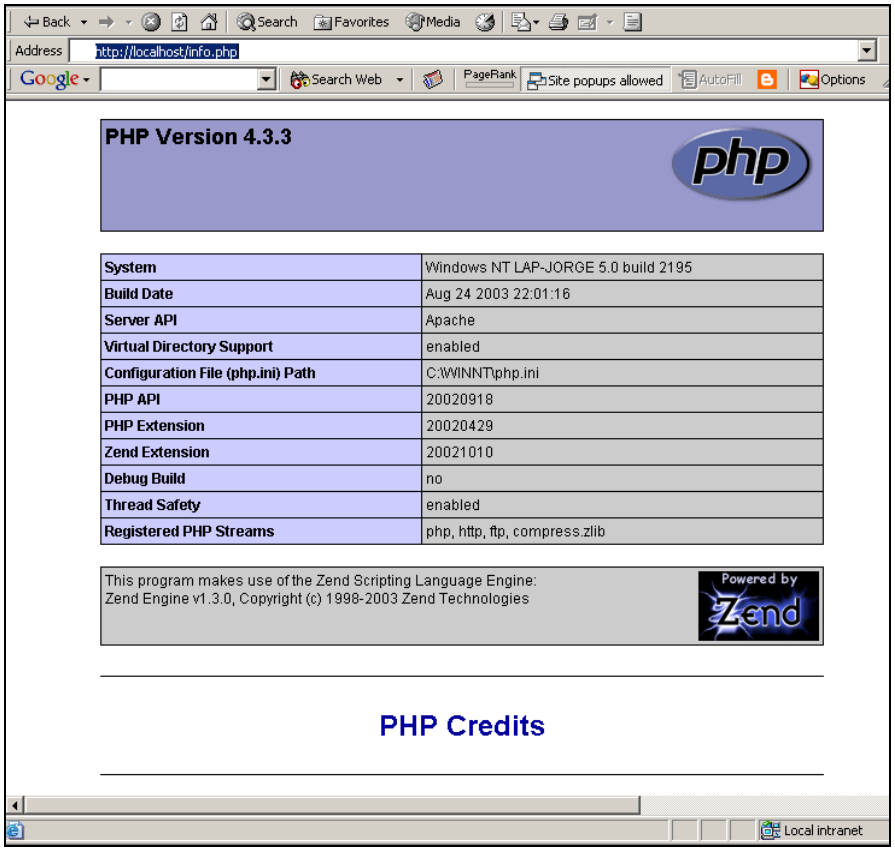


Figure 1: Page that should appear after configuring PHP with APACHE

Where to find help

The VHL is a collective construction. All the information on the use of its products is available from the technical support site, especially at the forums of each product. For more information visit the site http://bvsmodelo.bvsalud.org/ or address your queries to the support team at bvs.technical.support@listas.bireme.br (BIREME/PAHO/WHO).

Simple installation of APACHE 2.2.x in Windows environments

About this chapter

This chapter is about the installation of APACHE for use with VHL products in Windows environments. As a pre-requisite, Windows NT, 2000 or above should be installed.

Obtaining the software

The Web server may be obtained at the APACHE site, **http://www.apache.org/**.

Installing APACHE

In the Windows environment APACHE can be installed with an installer, which simplifies the whole process. If you have no experience with this type of program, it is advisable to perform the installation as proposed by the system.

Click twice on the APACHE installer and follow the instructions. The correct installation of the system should generate the following configuration on your computer:

1. The APACHE programs will be installed at
   C:\PROGRAM FILES\Apache Software Foundation\Apache2.2
2. Unless otherwise indicated, APACHE will be installed as a Windows automatic service; otherwise you will need to start the server manually.
3. In any case, access to APACHE's configuration and control may be effected through the following route:
   Start > Programs > Apache HTTP Server 2.2
4. The important folders for installation are:
   a) **C:\PROGRAM FILES\Apache Software Foundation\Apache2.2\htdocs**, where the documents that the server will offer to visitors are stored.
   b) **C:\PROGRAM FILES\Apache Software Foundation\Apache2.2\conf**, where the file httpd.conf is located. This file effects the configuration of the server.
   c) **C:\PROGRAM FILES\Apache Software Foundation\Apache2.2\cgi-bin**, where the specific programs used by the server to access other programs or data are stored.

Configuring APACHE

The detailed information to configure APACHE can be found in the program's documentation. From VHL's point of view, no major changes to the product's standard installation are required. The use of PHP has to be configured to prepare the system for the installation of VHL products.

The configuration procedure basically consists of editing the file httpd.conf, adding some lines to it.

Before beginning the configuration, make sure that the Web server is working correctly. If the APACHE installation is successful, the URL **http://localhost/** should show a page indicating that your server is working.

Where to find help

The VHL is a collective construction, and all the information on the use of its products is available from the technical support site, especially in the forums of each product. For more information visit the site http://bvsmodelo.bvsalud.org/ or direct your queries in writing to the support team of BIREME/PAHO/WHO at bvs.technical.support@listas.bireme.br.

Simple installation of PHP 5.3.x in Windows environments

About this chapter

This chapter addresses the installation of PHP for use with VHL products in Windows environments. It requires Windows NT, 2000 or above installed.

Obtaining the software

The latest PHP version is available from **http://www.php.net/**.

> **!** There are two ways of setting up PHP for Windows: one that works only as a CGI and and the other that can be installed as a module. VHL recommends the latter whenever the products are used with high load.

Installing PHP

Installing PHP in a Windows environment is very simple. Just decompress the file into a folder and copy the extensions that PHP is going to use into the system's directories.

Following are the steps for a PHP installation compliant with VHL products:

a) Create a "**php**" folder in the root directory of the hard disk, namely, "**c:\php**";
b) Decompress the file that contains the PHP components into the directory created in (a);
c) Use the **php.ini-production** file as a basis for PHP configuration. Review the "PHP Configuration" section;

Configuring PHP

The procedure basically consists of editing the file php.ini-production, adjusting some parameters and saving it as php.ini into the Windows directory.

a) Make a copy of the file **c:\php\php.ini-production**. Save it as **c:\php\php.ini**;
b) Edit the file **c:\php\php.ini** (you may use the NOTEPAD) and adjust the following parameters (look for the appropriate line in the document):

```
extension_dir = "/php/ext"

extension=php_xmlrpc.dll

extension=php_xsl.dll

short_open_tag = On
```

Apache with PHP

Before beginning the configuration make sure that the web server is working properly. Use your browser and go to the address of the website to check this.

If APACHE has been correctly installed, the URL **http://localhost/** should show a page indicating that your server is working.

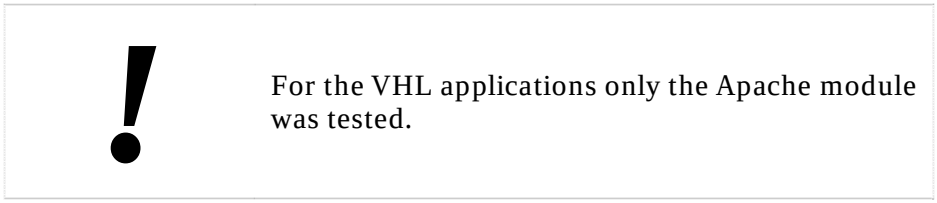
To configure APACHE, edit the file **httpd.conf**, available at the folder <*path_to_apache*>\conf\ for example:

C:\APACHE\conf.

Edit this file with a text editor (for example NOTEPAD), and lines will be added that will allow APACHE to handle PHP programs.

The configuration file is structured in sections. Make sure that the lines that you include are in the correct section.

As mentioned above, PHP can work in CGI mode or as an APACHE module (but not in both modes). The primary difference lies in the fact that the second option behaves better with heavier loads.



PHP as an APACHE module

For this option, configure the following lines in the file **`httpd.conf`**:

1. Add the module to the list of Apache APIs:
   `LoadModule php5_module "c:\php\php5apache2_2.dll"`
2. A line that includes the type of php file into the MIME table:
   `AddType application/x-httpd-php .php`
3. A line to instruct the PHP engine the path to the php.ini file:
   `PHPIniDir "C:/php”`
4. Delete the following line inside <Directory /> block:

   `Deny from all`

Testing the installation

The last step for configuring PHP is ensuring that APACHE is processing the PHP files correctly.

To this effect, we will create a new file called “info.php”.

1. Make sure that the file effectively has the extension php and not another one (Windows hides the extensions of the files that it defines as “known files”).
2. The file info.php should contain a line with the following text:
   `<? phpinfo(); ?>`
3. Save the file into the directory of documents configured in your web server. In an APACHE standard installation, this directory is called “htdocs”.
4. Restart the Web server
   `Start > Programs > Apache Software Foundation > Restart Apache`
5. Use your internet browser to visit the URL:
   `http://localhost/info.php`

This page should show information on your PHP installation.

Where to find help

The VHL is a collective construction. All the information on the use of its products is available from the technical support site, especially at the forums of each product. For more information visit the site http://bvsmodelo.bvsalud.org/ or address your queries to the support team at bvs.technical.support@listas.bireme.br (BIREME/PAHO/WHO).

Configuration of Apache 1.3.xx, PHP 4.4.x, Sablotron and Expat in Linux environments

About this chapter

This chapter addresses the installation of Apache, PHP, Sablotron and Expat for use by VHL products in Linux environments. The standard Linux operating system has to be installed in your computer, preferrable without graphic interface (optimized), without the default web server that comes with the distribution and without SELinux option.

The latter is necessary to guarantee that the PHP can include a dynamic library in the Apache and load it without restrictions.

In order to be able to compile and link all software described here, it is suggested the server administration guarantees that the following libraries and tools are available in the system:

Software	ANSI C Compiler	Perl5 Interpreter	Flex v.2.5.4	Bison v.1.28(*) / 1.35 / 1.75	iconv	Expat
Expat	X					
Sablotron	X				X	X
PHP	X		X	X		
Apache	X	X				

The use of PHP in VHL products aims at allowing the handling of data and the processing of XML documents for their presentation through transformations using XSL files.

You may address your queries to bvs.technical.support@listas.bireme.br.

Example of proven versions

1. Apache 1.3.xx
2. PHP 4.4.0
3. Sablotron 0.98
4. Expat 1.95.2 (o 1.95.6)

All of these software programs should be compiled in the server. Auto-install packages such as RPM and others should not be used. The lines recommended for compilation are the following:

Unpacking in the standard source code area:

```
cd /usr/local/src
tar xvfzp apache_1.3.37.tar.gz
tar xvfzp php-4.4.0.tar.gz
tar xvfzp expat-1.95.2.tar.gz
tar xvfzp Sablot-0.98.tar.gz
```

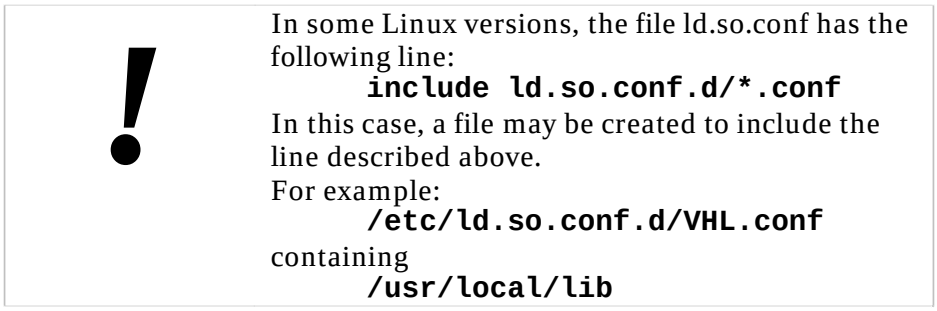
Configuring and installing Expat

```
cd expat-1.95.2
./configure
make
make install
```

Updating the paths to the libraries

```
vi /etc/ld.so.conf
```

Include the paths to the library **/usr/local/lib**.




```
ldconfig
```

Commands to maintain compatibility with previous Red Hat versions

```
sync
sync

cd ..
```

Configuring and installing Sablotron

```
cd Sablot-0.98
./configure
make
make install
```

Updating the paths to libraries

```
ldconfig
sync
sync

cd ..
```

Configuring and installing Apache

```
cd apache_1.3.37
```

Note that the following execution line consists of three lines of parameters.

```
./configure --prefix=/usr/local/apache\
  --enable-rule=SHARED_CORE\
  --enable-module=so

make
make install


cd ..
```

Starting the Apache service

Testing the configuration

```
/usr/local/apache/bin/apachectl configtest
```

Starting the service

```
/usr/local/apache/bin/apachectl start
```

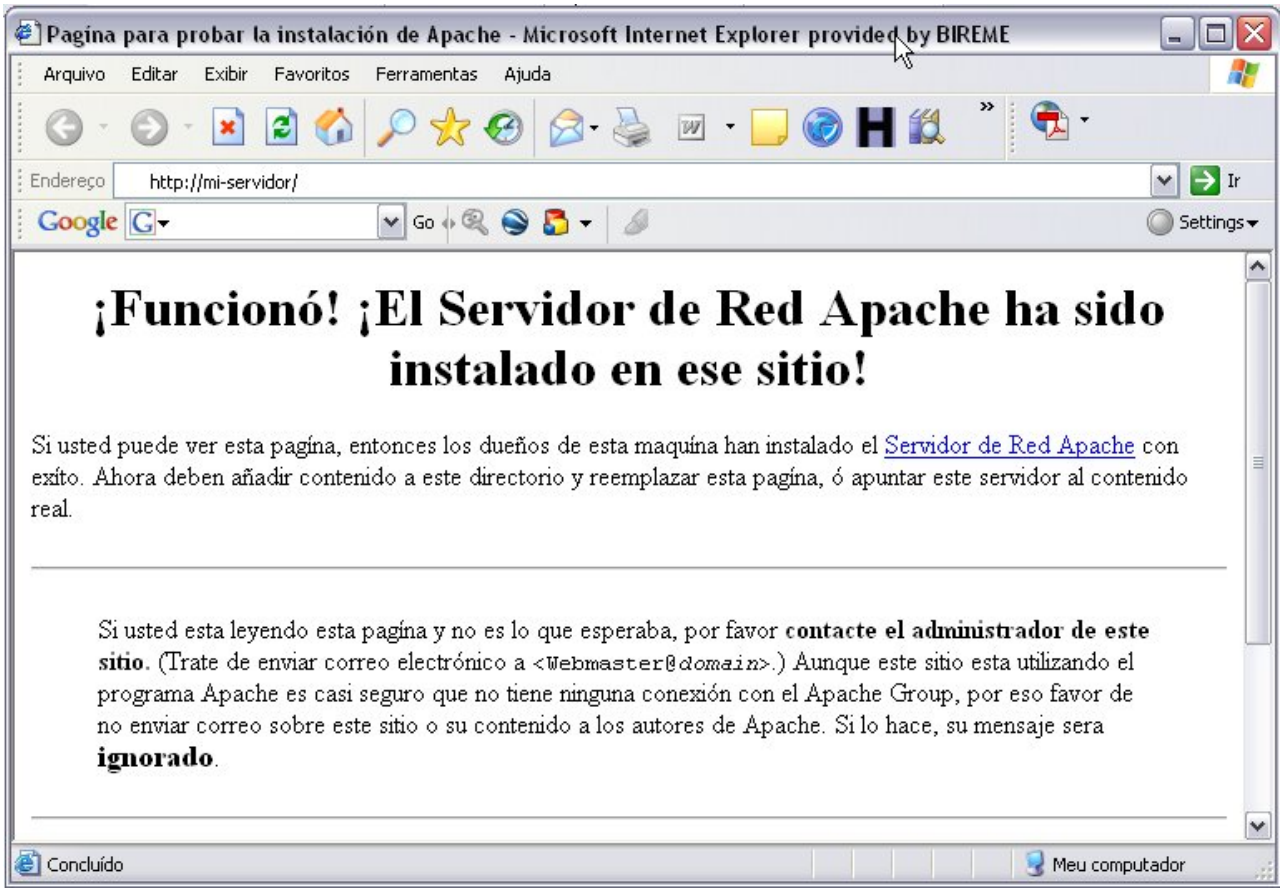
Checking the installation status

Open a window of your favorite internet browser and test the URL with the server's IP.

For example:

```
http://200.100.1.100/
```

The result should be a screen like the following:



Configuring and installing PHP

```
cd php-4.4.0
```

Configuring the LDFLAGS variable

This parameter is required to inform the GCC on how the PHP code should be compiled.

```
export LDFLAGS=-lstdc++
```

Beginning the configuration of PHP

Note that the execution line below consists of multiple lines of parameters.

```
./configure --prefix=/usr/local/php\
  --with-apxs=/usr/local/apache/bin/apxs\
  --enable-track-vars\
  --enable-sockets\
  --enable-xslt\
  --with-xslt-sablot\
  --enable-ftp

make

make install
```

Editing the file httpd.conf

```
vi /usr/local/apache/conf/httpd.conf
```

1. Check if the module has been added to the Apache API list as shown below:
   `LoadModule php4_module        libexec/libphp4.so`

2. Add a line that includes the type of php file into the MIME table:
   `AddType application/x-httpd-php .php`

> **!** Some Apache versions may require the inclusion of the line below in the file httpd.conf:
> **`AddModule mod_php4.c`**

Editing the PHP configuration file

Some PHP versions create the file **`php.ini`** in the subdirectory **`/usr/local/php/lib`** o **`/usr/local/php/lib/php`**. If the version used has not created this file, copy the file php.ini-dist from the installation directory (**`php-4.4.0`**) into **`/usr/local/php/lib`** as **`php.ini`**.

The file then has to be edited to change/enable some parameters.

```
vi php.ini
```

1. Activate the variable register_globals:
   `register_globals = On`

2. Remove the commentary from the line:
   `default_charset = "iso-8859-1"`

Starting the Apache service again

```
/usr/local/apache/bin/apachectl start
```

Testing the installation

The last step for configuring PHP is ensuring that APACHE is processing the PHP files correctly.

To this effect, we will create a new file called “info.php”.

1. The file info.php should contain a line with the following text:
   ```
   <? phpinfo(); ?>
   ```
2. Save the file into the directory of documents configured in your web server. In an APACHE standard installation, this directory is called “htdocs”.
3. Restart the web server
   ```
   /usr/local/apache/bin/apachectl start
   ```
4. Use your internet browser to visit the URL:
   ```
   http://localhost/info.php
   ```

This page should show information on your PHP installation.

Where to find help

The VHL is a collective construction. All the information on the use of its products is available from the technical support site, especially at the forums of each product. For more information visit the site http://bvsmodelo.bvsalud.org/ or address your queries to the support team at bvs.technical.support@listas.bireme.br (BIREME/PAHO/WHO).

Configuration of Apache 2.2.xx, PHP 5.3.x in Linux environments

About this chapter

This chapter addresses the installation of Apache and PHP for use by VHL products in Linux environments. The standard Linux operating system has to be installed in your computer, preferrable without graphic interface (optimized), without the default web server that comes with the distribution and without SELinux option.

The latter is necessary to guarantee that the PHP can include a dynamic library in the Apache and load it without restrictions.

In order to be able to compile and link all software described here, it is suggested the server administration guarantees that the following libraries and tools are available in the system:

Software	ANSI C Compiler	Perl5 Interpreter	Flex v.2.5.4	iconv
PHP	X		X	X
Apache	X	X		

The use of PHP in VHL products aims at allowing the handling of data and the processing of XML documents for their presentation through transformations using XSL files.

You may address your queries to bvs.technical.support@listas.bireme.br.

Example of proven versions

1. Apache 2.2.11
2. PHP 5.3.x

All of these software programs should be compiled in the server. Auto-install packages such as RPM and others should not be used. The lines recommended for compilation are the following:

Unpacking in the standard source code area:

```
cd /usr/local/src
tar xvfzp apache_2.2.11.tar.gz
tar xvfzp php-5.3.x.tar.gz
```

Configuring and installing Apache

```
cd apache_2.2.11
```

Note that the following execution line consists of three lines of parameters.

```
./configure --prefix=/usr/local/apache\
   --enable-so
   --enable-rewrite

make
make install


cd ..
```

Starting the Apache service

Testing the configuration

```
/usr/local/apache/bin/apachectl configtest
or
/usr/local/apache/bin/httpd -t
```

Starting the service

```
/usr/local/apache/bin/apachectl start
or
/usr/local/apache/bin/httpd -k start
```

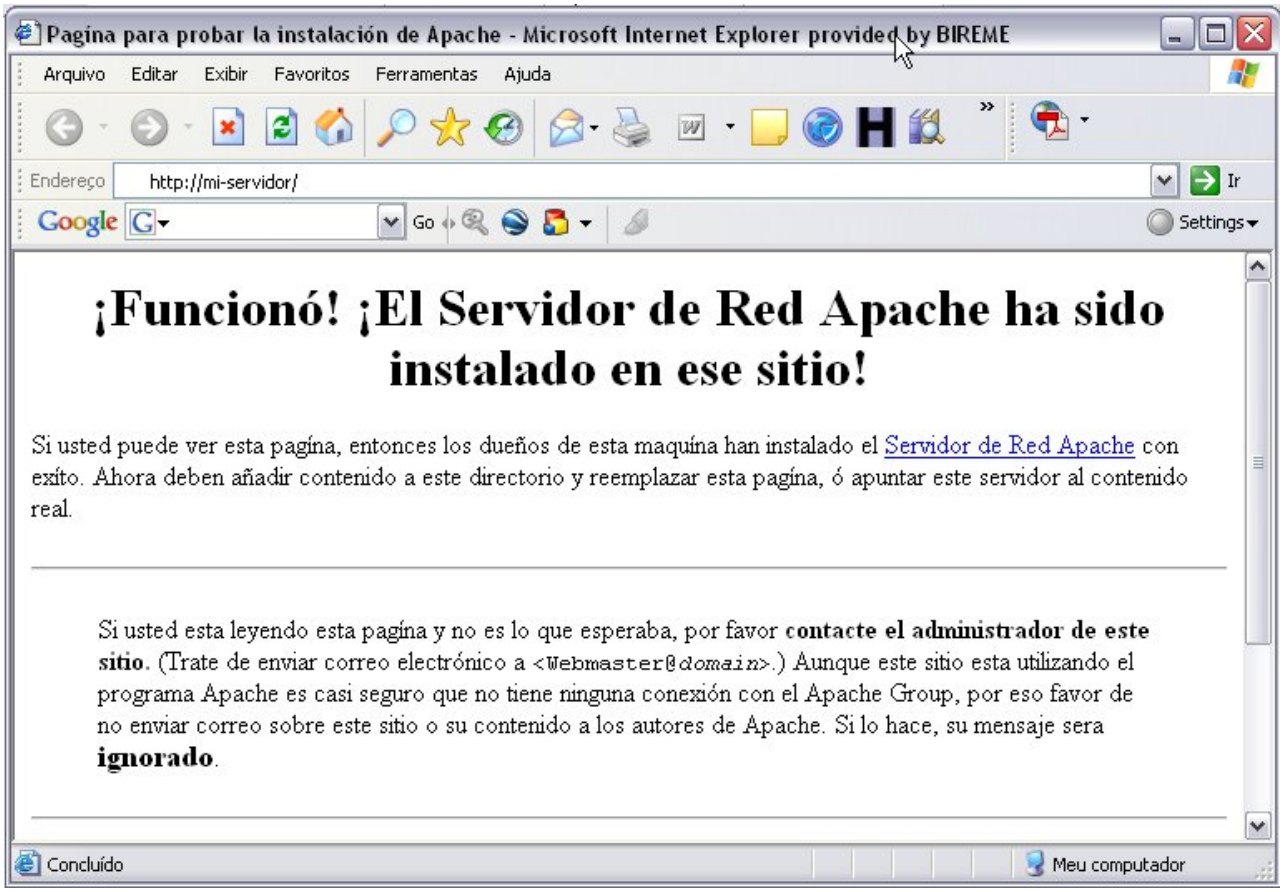
Checking the installation status

Open a window of your favorite internet browser and test the URL with the server's IP.

For example:

```
http://200.100.1.100/
```

The result should be a screen like the following:



Configuring and installing PHP

```
cd php-5.3.x
```

Beginning the configuration of PHP

Note that the execution line below consists of multiple lines of parameters.

```
./configure --prefix=/usr/local/php \
--with-apxs2=/usr/local/apache/bin/apxs --with-gd --with-zlib \
--enable-mbstring --with-iconv --with-iconv-dir --enable-gd-native-ttf \
--enable-ftp --with-xsl --enable-wddx --with-gettext \
--with-kerberos --enable-soap --with-libxml-dir --with-xmlrpc \
--with-libxml-dir=/usr

make
make install
```

In the case an error occurs during PHP compiling related to a missing library, please verify that the following items are installed:

Example for SUSE

Zypper search gcc

Zypper search make

Zypper search libxml2

Zypper search libxml2-dev

Zypper search libpng12-dev

Zypper search libxslt1-dev

Example for Fedora

yum search gcc

yum search make

yum search libxml2

yum search libxml2-dev

yum search libpng12-dev

yum search libxslt1-dev

In case any other library is missing you should consider installing it with the appropriate command available in your operating system and/or distro.

Editing the file httpd.conf

```
vi /usr/local/apache/conf/httpd.conf
```

1. Check if the module has been added to the Apache API list as shown below:
   ```
   LoadModule php5_module        modules/libphp5.so
   ```
2. Add a line that includes the type of php file into the MIME table:
   ```
   AddType application/x-httpd-php .php
   ```

Editing the PHP configuration file

Some PHP versions create the file **`php.ini`** in the subdirectory **`/usr/local/php/lib`** o **`/usr/local/php/lib/php`**. If the version used has not created this file, copy the file `php.ini-production` from the installation directory (**`php-5.3.x`**) into **`/usr/local/php/lib`** as **`php.ini`**.

The file then has to be edited to change/enable some parameters.

```
vi php.ini
```

Activate the register_globals variable when you create the Virtual Host.

```
php_flag register_globals on
```

Starting the Apache service again

```
/usr/local/apache/bin/apachectl start
or
/usr/local/apache/bin/httpd -k start
```

Testing the installation

The last step for configuring PHP is ensuring that APACHE is processing the PHP files correctly.

To this effect, we will create a new file called “info.php”.

1. The file info.php should contain a line with the following text:
   ```
   <? phpinfo(); ?>
   ```
2. Save the file into the directory of documents configured in your web server. In an APACHE standard installation, this directory is called “htdocs”.
3. Restart the web server
   ```
   /usr/local/apache/bin/apachectl start
   ```

4. Use your internet browser to visit the URL:
   `http://localhost/info.php`

This page should show information on your PHP installation.

Where to find help

The VHL is a collective construction. All the information on the use of its products is available from the technical support site, especially at the forums of each product. For more information visit the site http://bvsmodelo.bvsalud.org/ or address your queries to the support team at bvs.technical.support@listas.bireme.br (BIREME/PAHO/WHO).

Bibliographic references

1. Apache Software Foundation. *The Apache HTTPD Server Project*. Available at: <http://httpd.apache.org/>. Accessed on: November 6th, 2006.

2. The PHP Group. *PHP: Hypertext Preprocessor*. Available at: <http://www.php.net/>. Accessed on: November 6th, 2006.

3. Ginger Alliance s.r.o. *Sablotron: XSLT, DOM and XPath processor*. Available at: <http://www.gingerall.org/sablotron.html>. Accessed on: November 6th, 2006.

4. Source Forge Net. *The Expat XML Parser*. Available at: <http://expat.sourceforge.net/>. Accessed on: November 6th, 2006.

Glossary

- **Application**. Program used to execute tasks in connection with an application, such as the creation or edition of texts, drawings, animations, layout, etc. E.g.: text processor, database manager, Internet browser, etc.
- **Backup**. Procedure used to duplicate one or more files and/or directories in another storing device (tape or disc), thus producing a backup copy that may be restored in the event of accidental deletion or physical damage to the original data.
- **Browser**. Internet page navigator, such as Internet Explorer and Netscape Navigator.
- **CGI**. The Common Gateway Interface is a standard for interfacing external applications with information servers, such as HTTP or Web servers.
- **File**. In computing, a set of data that may be saved into some type of storing device. The data files are created by applications, such as a text processor for example.

- **SGML**. Metalanguage standard of the ISO (International Organization for Standardization) used for the definition of languages of marking of electronic texts, making possible the interchange and the distribution of documents in the most varied formats, from one same source of data.

- **TCP/IP Protocol**. Standard that defines the method of communication between digital equipment. It employs a single number of identification.

- **URL**. Standard defined for the addressing of data contents via the TCP/IP protocol. Internet browsers use the URL to access Web pages.

- **XML**. Language created to allow the arrangement of data in a structured and hierarchical manner, thus facilitating data communication between different systems and platforms.

- **XSL**. Language created to allow the navigation, selection and capture of data of an XML file.