

Extending XAMPP with PHP/Java Bridge

written by
Peter J. Siegel
January 2006

<http://www.peter-siegel.de>

This paper's intention

XAMPP is a fine adjusted software solution based on Apache Web Server, MySQL, PHP and Perl. It has been developed by the ApacheFriends Project at <http://www.apachefriends.org>. The intention behind the great idea of creating XAMPP is a preconfigured package that is easy to handle and up-to-date. It guarantees a minimal waste of time during installation and configuration.

The PHP/Java Bridge offers an elegant solution to connect PHP based applications to Java classes running in a virtual machine. It's an easy way to let PHP processes communicate with corresponding Java methods running in a thread within a JVM. For example, it is possible to integrate an Apache Lucene provided service into a web application with PHP powered interface. Additionally, PHP based web services can be connected to a J2EE based middleware. More information and downloadable packages are available on the project homepage <http://php-java-bridge.sourceforge.net>.

This paper is designed as a short manual that shows how XAMPP could be extended easily by PHP/Java Bridge. It has been written during my private experimental experiences connecting PHP and Java. So all described steps have led to a smooth integration of PHP/Java Bridge into a running system based on XAMPP and the SuSE Linux distribution. This paper is dedicated to anybody who doesn't want to spend much time searching the web for hints and tips. However, this howto-manual doesn't raise a claim on completeness. No liability for any loss or damage will be assumed.

pre-requisites

software components

System	Linux, Distribution SuSE 9.2 Professional
XAMPP	Version 1.5.0 available at http://www.apachefriends.org Package: ◆ xampp-linux-1.5.0.tar.gz
PHP/Java Bridge	Version 3.0.4 available at http://php-java-bridge.sourceforge.net/ Package: ◆ php-java-bridge-3.0.4-1.FC4.i386.rpm (compiled rpm for Fedora Core 4) Source: ◆ php-java-bridge_3.0.4-tar.bz2

Installing XAMPP

The XAMPP 1.5.0 package contains following components:

- Apache 2.0.55
- MySQL 5.0.15
- PHP 4.4.1
- PHP 5.0.5
- Perl 5.8.7
- ProFTPD 1.2.10
- phpMyAdmin 2.6.3-pl3
- OpenSSL 0.9.8a
- GD 2.0.1
- Freetype2 2.1.7
- libjpeg 6b
- libpng 1.2.7
- gdbm 1.8.0
- zlib 1.2.3
- expat 1.2
- Sablotron 1.0
- libxml 2.4.26
- libxslt 1.1.8
- Ming 0.2a
- Webalizer 2.01
- pdf class 009e
- ncurses 5.8
- mod_perl 2.0.1
- FreeTDS 0.63
- gettext 0.11.5
- IMAP C-Client 2004e
- OpenLDAP (client) 2.3.11
- mhash library 0.8.18
- mcrypt library 2.5.7
- cURL 7.13.1
- SQLite 2.8.9 (in case of PHP5: 2.8.14)
- phpSQLiteAdmin 0.2
- eAccelerator 0.9.3
- libapreq 2.04-dev
- FPDF 1.5.3

Step 1: Install the XAMPP package

Unpack the XAMPP package into directory '/opt'. An automatically created subdirectory '/opt/lampp' is filled with the package content. Apply the following console entry:

```
# tar xvfz xampp-linux-1.5.0.tar.gz -C '/opt'
```

Note: The base directory in the proceedings of this document is defined as '/opt/lampp'.

Step 2: Create a startup script for XAMPP

The following startup script should be saved into directory '/etc/init.d'. At system startup, different services are going to be executed by start scripts. These scripts are originally located in directory '/etc/init.d' and they are registered by a soft link in at least one run level directory from 'rc0.d' to 'rc6.d'. All soft linked start scripts in one associated run level directory are called by the entry the run level master script '/etc/init.d/rc'.

Startup script: /etc/init.d/lampp

```
#!/bin/sh
# Written in 2005 by PJS
#
# Author:
#       Peter J. Siegel <peter@peter-siegel.de>, 2005
#
# /etc/init.d/lampp
#
# Description:    Start the XAMPP environment
#
### END INIT INFO

XAMPP_DIR=/opt/lampp

case "$1" in
    start)
        echo "Starting LAMPP"
        $XAMPP_DIR/lampp start
        ;;

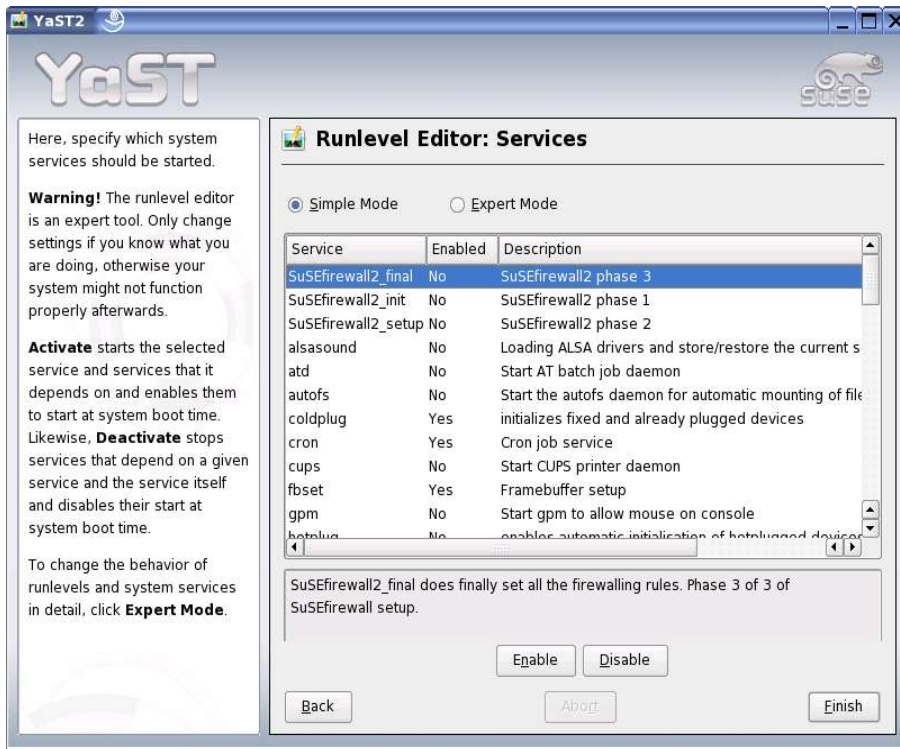
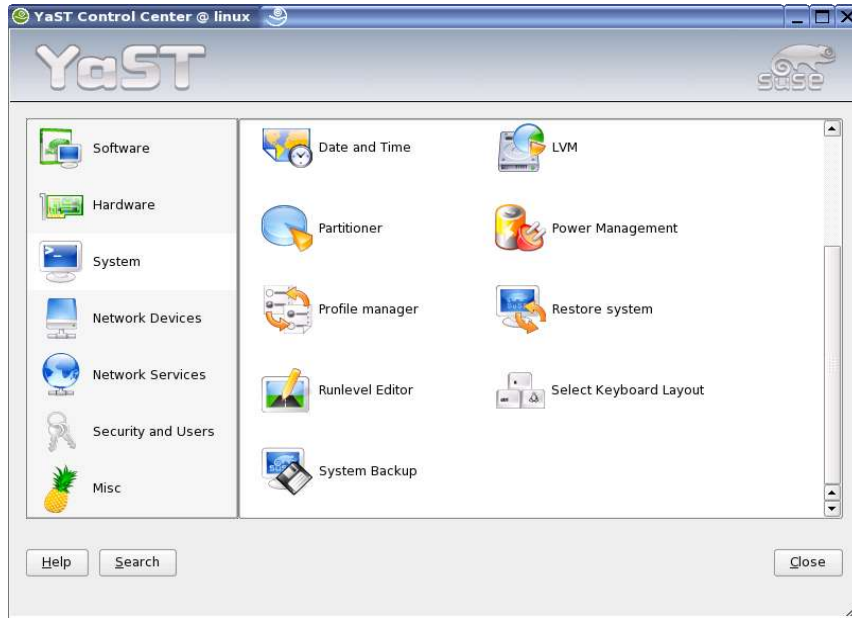
    stop)
        echo "Shutting down LAMPP"
        $XAMPP_DIR/lampp stop
        ;;

    restart)
        ## Stop the service and regardless of whether it was
        ## running or not, start it again.
        $0 stop
        $0 start
        ;;

    *)
        echo "Usage: $0 {start|stop|restart}"
        exit 1
        ;;
esac

exit 0
```

The start script can be executed automatically on system bootup. This can be set up with the runlevel editor in YaST:



Step 3: Securing XAMPP

Execute following command line:

```
# /opt/lampp/lampp start
# /opt/lampp/lampp security
```

An interactive configuration of passwords for XAMPP, MySQL phpMyAdmin and ProFTPD should be fulfilled with these lines:

```
XAMPP: Quick security check...
XAMPP: Your XAMPP pages are NOT secured by a password.
XAMPP: Do you want to set a password? [yes]
XAMPP: Password:
XAMPP: Password (again):
XAMPP: Password protection active. Please use 'lampp' as user name!
XAMPP: MySQL is accessable via network.
XAMPP: Normally that's not recommended. Do you want me to turn it off? [yes]
XAMPP: Turned off.
XAMPP: The MySQL/phpMyAdmin user pma has no password set!!!
XAMPP: Do you want to set a password? [yes]
XAMPP: Password:
XAMPP: Password (again):
XAMPP: Setting new MySQL pma password
XAMPP: Setting phpMyAdmin's pma password to the new one.
XAMPP: MySQL has no root password set!!!
XAMPP: Do you want to set a password? [yes]
XAMPP: Password:
XAMPP: Password (again):
XAMPP: Setting new MySQL root password.
XAMPP: Change phpMyAdmin's authentication method.
XAMPP: The FTP password is still set to 'lampp'.
XAMPP: Do you want to change the password? [yes]
XAMPP: Password:
XAMPP: Password (again):
XAMPP: Reload ProFTPD...
XAMPP: Done.
```

Launching the XAMPP page in a web browser under Echnaton's IP address will show a login window like this:



Launching the phpMyAdmin page in a web browser under the server address "http://192.168.10.3/phpmyadmin" will show a login window like this:



Installing PHP/Java Bridge with a precompiled package

For convenience it's a better solution to download and install the compiled package "php-java-bridge-3.0.4-1.FC4.i386.rpm". Despite it is made for the Fedora Linux distribution you can use it for SuSE Linux 9.x, too.

Step 1: Package Installation

Perform the package installation by following command line:

```
# rpm -i php-java-bridge-3.0.4-1.FC4.i386.rpm --nodeps
```

The 'nodeps' option should prevent any dependency messages that may belong to missing PHP packages. Every needed PHP component is just installed with XAMPP that is unknown for the RPM package manager.

Step 2: Directory Changes

After package installation the files for PHP/Java Bridge are installed into these directories:

```
/etc/init.d/php-java-bridge  
/etc/php.d/java.ini  
/usr/lib/php/modules/JavaBridge.jar  
/usr/lib/php/modules/RunJavaBridge  
/usr/lib/php/modules/java.so  
/usr/lib/php/modules/libnatcJavaBridge.so  
/usr/sbin/php-java-bridge
```

The documentation is saved into "/usr/share/doc/php-java-bridge-3.0.4".

However, XAMPP is located in an own directory tree “/opt/lampp”. The PHP/Java Bridge can be connected as an extension for XAMPP, so all important files listed above have to be moved into the referring extension directory under “/opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/”. This is achieved by following command lines:

```
# mv /usr/lib/php/modules/JavaBridge.jar
    /opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/JavaBridge.jar

# mv /usr/lib/php/modules/RunJavaBridge
    /opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/RunJavaBridge

# mv /usr/lib/php/modules/java.so
    /opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/java.so

# mv /usr/lib/php/modules/libnatcJavaBridge.so
    /opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/libnatcJavaBridge.so

# mv /usr/sbin/php-java-bridge /opt/lampp/bin/php-java-bridge
# chmod 755 /opt/lampp/bin/php-java-bridge

# mv /usr/share/doc/php-java-bridge-3.0.4 /opt/lampp/php-java-bridge-3.0.4
```

Step 3: New Configuration Entries In File “php.ini”

The PHP/Java Bridge extension and its configuration need to get loaded by PHP as part of the XAMPP framework. For this reason some additional entries have to be made in the PHP configuration file “/opt/lampp/etc/php.ini”, written at the end of section “Dynamic Extensions”:

```
;;;;;;;;;;;;;
; php-java-bridge-3.0.4 ;
;;;;;;;;;;;;;

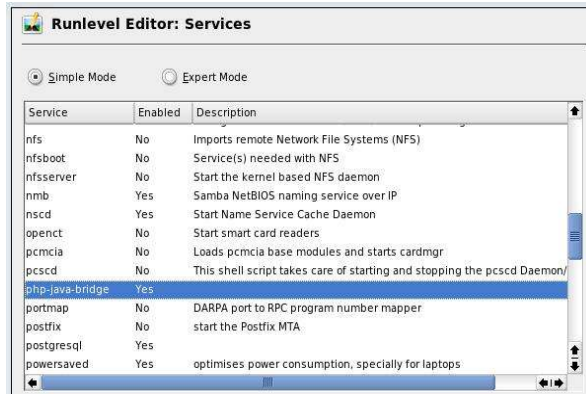
extension = java.so
[java]
java.log_level="3"
java.log_file="/opt/lampp/logs/php-java-bridge.log"
java.java_home="/opt/lampp/java_home"
java.java="/opt/lampp/java_home/bin/java"
java.wrapper="/opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/RunJavaBridge"
```

The directory “/opt/lampp/java_home” is a soft link that points to a directory containing the installed JRE. On most systems it is set in “\$JAVA_HOME”. Following command line creates this soft link:

```
# ln -s $JAVA_HOME /opt/lampp/java_home
```

Step 4: Modifications In The Backend Start Script

The RPM installation creates a start script for the PHP/Java Bridge backend. It is linked automatically with appropriate run levels to start at system bootup. The start script is named “/etc/init.d/php-java-bridge”.



The start script has to be modified to the changed directories. All changes are highlighted.

```
#!/bin/bash
#
# php-java-bridge      Startup script for the PHP/Java Bridge
#
# chkconfig: - 84 16
# description: The PHP/Java Bridge allows PHP scripts to access java based \
#               applications running in a java application server
# processname: php-java-bridge
# pidfile: /var/run/php-java-bridge.pid
# config: /etc/php.d/java.ini

# Source function library.
# . /etc/rc.d/init.d/functions

prog=/opt/lampp/bin/php-java-bridge
RETVAL=0

start() {
    echo -n $"Starting $prog: "
    exec $prog &
    RETVAL=$?
    echo
    [ $RETVAL = 0 ] && touch $prog
    return $RETVAL
}

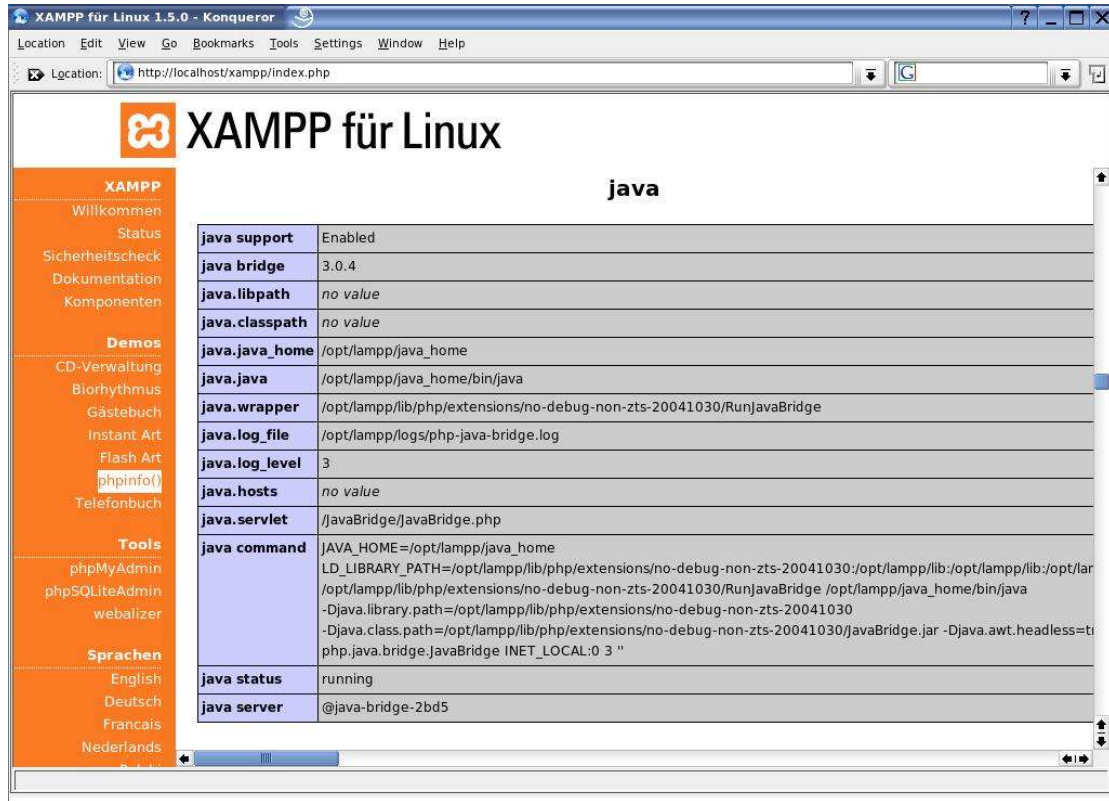
stop() {
    echo -n $"Stopping $prog: "
    killproc $prog
    RETVAL=$?
    echo
    [ $RETVAL = 0 ] && rm -f $prog.pid
}

# See how we were called.
case "$1" in
    start)
        start
        ;;
    stop)
        stop
        ;;
    status)
        status $prog
        RETVAL=$?
        ;;
    restart)
        stop
        start
        ;;
    *)
        echo $"Usage: $prog {start|stop|restart|status}"
        exit 1
esac

exit $RETVAL
```

Step 5: Result Check

Select the `phpinfo()` function on the local XAMPP page. Scroll down to the Java section, that should give information like this:



The screenshot shows the XAMPP für Linux interface in a Konqueror browser window. The page title is "XAMPP für Linux" and the sub-section is "java". The Java configuration is displayed in a table format:

java support	Enabled
java bridge	3.0.4
java.libpath	no value
java.classpath	no value
java.java_home	/opt/lampp/java_home
java.java	/opt/lampp/java_home/bin/java
java.wrapper	/opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/RunJavaBridge
java.log_file	/opt/lampp/logs/php-java-bridge.log
java.log_level	3
java.hosts	no value
java.servlet	/javaBridge/javaBridge.php
java command	JAVA_HOME=/opt/lampp/java_home LD_LIBRARY_PATH=/opt/lampp/lib/php/extensions/no-debug-non-zts-20041030:/opt/lampp/lib:/opt/lampp/lib:/opt/lampp/lib:/opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/RunJavaBridge /opt/lampp/java_home/bin/java -Djava.library.path=/opt/lampp/lib/php/extensions/no-debug-non-zts-20041030 -Djava.class.path=/opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/javaBridge.jar -Djava.awt.headless=true php.java.bridge.JavaBridge INET_LOCAL:0 3 "
java status	running
java server	@java-bridge-2bd5

Execute the “`ps -A`” shell command that shows all running processes. The threads belonging to XAMPP are extended by an additional Java thread:

```
11250 ?          00:00:00 proftpd
11260 pts/3      00:00:00 mysqld_safe
11287 pts/3      00:00:00 mysqld
11290 ?          00:00:00 httpd
11291 ?          00:00:00 java
```

Installing PHP/Java Bridge by source code compiling

Precompiled packages for the PHP/Java Bridge are built against a certain PHP version. If the Java shared object (`java.so`) depends on a PHP version that differs from the installed one (e.g 5.0 vs. 5.1.1) following error might be rising up:

```
PHP warning: PHP startup: java.so: Cannot open shared object file - no such file or
directory in Unknown on line 0
PHP warning: PHP startup: (null): Unable to initialize module
Module compiled with module API=20041030, debug=0, thread-safety=0
PHP   compiled with module API=20050922, debug=0, thread-safety=0
These options need to match
```

Like the error message says clearly: The API versions at compile time have to be the same. You can achieve this by compiling the PHP/Java Bridge source code against the installed PHP version.

Compile the PHP/Java Bridge source

Assuming you have extracted the source tree packed in the tar file with following command line

```
# cat php-java-bridge_3.0.4.tar.bz2 | bunzip2 | tar xvf -
```

and change to the unpacked source directory with

```
# cd php-java-bridge-3.0.4
```

Start the compilation with

```
# /opt/lampp/bin/phpize &&  
./configure --disable-servlet  
--with-java=/opt/lampp/java_home  
--with-php-config=/opt/lampp/bin/php-config
```

The PHP/Java Bridge source doesn't have a "config.m4" script that is needed to create the makefile by executing the configuration script "./configure". If the makefile is built successfully, start the compilation and installation with the command line

```
# make && make install
```

All built files that belong to the PHP/Java Bridge are saved in subdirectory "modules" in the source directory "php-java-bridge-3.0.4". They should be copied into the XAMPP extensions directory "/opt/lampp/lib/php/extensions/no-debug-non-zts-20041030/".

Finally, perform step 3 on page 7.

