WebPerMoS - 0.2 Beta – Installation Manual

1. Unzip the distribution.

For Linux :

```
$ tar –zxvf  webpermos.tar.gz
```

For Windows :

Use WinZip to unzip the archive.

This is the organization of the directories which you will see.

```
<base_dir>
   |
   |--DistributeSystem
   |    |
   |    |--DelegateWindows
   |    |
   |    |--RC
   |    |
   |    |--
   |
   |--Development
        |
        |--Delegate-Linux
        |       |
        |       |--build
        |       |
        |       |--com
        |       |
        |       |--...
        |
        |--Delegate-Windows
                |
                |--build
                |
                |--com
                |
                |--...
```

base_dir will be "webpermos" in case of Linux and "WebPerMoS" for Windows.

2. Setting Up RC

RC has been developed using PHP. MySQL has been used a back-end database. The pre-requisite for the installing RC are as follows.

1. PHP installation with MySQL and GD Library enabled.
2. MySQL as the database.

(Following three libraries are required for producing graphical output with PHP.)

3. zlib
4. libpng
5. GD Graphics Library

Step 1 :

All the PHP file with name starting with "jpgraph" indicate the JpGraph Library for displaying the results. Place these files in the directory named *"jpgraph"* and place all the rest php files in <RC> directory. In the distribution, you can see this structure. The source code tree will look like

```
<RC>
  |
  |--login.php
  |
  |--register.php
  |
  |--....
  |
  |--jpgraph
      |
      |--jpgraph_line.php
      |
      |--jpgraph_bar.php
      |
      |--....
      |
```

Step 2 :

Most of the PHP pages access database. Some code needs to be changed to reflect the specific database parameters. Search all the files

under <RC> for the following two statements. Files under *"jpgraph"* directory need not be searched.

```
$dbconn = mysql_connect("server_name", "user_name", "password" );
$result = mysql_select_db("database_name", $dbconn) or die("failure");
```

Here "server_name" is the name of the host on which database server (MySQL) is running. "user_name" and "password" are the MySQL user name and password respectively. "database_name" is the database to used for storing all the data. Change these parameters to reflect the local configurations. Please ensure that the database has been created.

Step 3:

Now various tables need to be created. Create the table by executing tabc.php. This PHP script can be executed by giving its URL to a browser.

3. Configuring Delegate for RC

The Delegate has been developed in Java. The Delegate has to be configured for the particular RC. Some changes need to be done in the source code of the Delegate before creating binaries for distribution.

Step 1:

Changes have to be done in following files

- ConfigDelegate.java (in "Development/Delegate-Linux/source" directory) for Linux
- ConfigDelegate.java (in "Development/Delegate-Windows/source" directory) and ConfigDelegateUser.java (in "Development/Delegate-Windows/form" directory) for Windows.

ConfigDelegate.java/ConfigDelegateUser.java creates rcparam.conf file in which the parameters to access RC are stored. rcparam.conf gets created when the Delegate runs for the first time. In this file, various URLs of RC are specified. Delegate uses these URLs to communicate with RC.

```
out.write ("UpResURL = http://www.yourserver.com/path/to/resupload.php ");
```

This line prints the location of interface for uploading the results to RC. Change this to reflect the correct URL. (You need to change only *"yourserver.com/path/to"* part.) Similarly search for parameters changes have to made for the parameters *"AliveURL"*, *"JobDLURL"* and *"RegURL"*.

Step 2 : Create Distribution for Linux Delegate

The prerequisite for building distribution is Ant.

a. Change Directory

$cd Development/Delegate-Linux

(The "source" directory contains the Java source code for the Delegate, "com" directory contains library classes and "docs" contains help files.)

b. Create the Delegate jar file with
$ant clean
$ant

(First command removes earlier versions, if any, and second creates a fresh jar file.)

c. Now create the distribution
$./maketar.sh WebPerMoS-<version>.jar
where WebPerMoS-<version>.jar is the newly created jar file which is in the "jar" directory.

This will create PerMoS.tar.gz which can now be distributed.

Step 3 : Create Distribution for Windows Delegate

a. Change directory to Development/Delegate-Windows

(The "source" directory contains Java source code for the Delegate, "form" contains Java source code for the GUI part of the Delegate, "com" contains library classes and "docs" contains help files.)

b. Create jar file by executing "ant clean" followed by "ant" command in that directory. First command removes earlier versions, if any, and second creates a fresh jar file.

c. Copy the above jar file to "DistributionSystem/DelegateWindows/lib" directory.

d. Modify following parameter in "DistributeSystem/DelegateWindows/conf/wrapper.conf" directory

wrapper.java.classpath.2=../lib/WebPerMoS-<version>.jar

Change the last component with the proper jar file name.

e. Zip the contents of "Distribution" directory to create PerMoS.zip file which is ready for distribution.

Step 4: Distribute Delegate Software
Distribute the Delegate to the participants of the System.

Step 5 : Starting the Delegate
This will have to be done by every participant who runs a Delegate. Instructions for this are given in Readme.txt in the distribution of Delegate.