MMBase Mini–HOWTO: Installation on Debian Woody

Table of Contents

MMBase Mini−HOWTO: Installation on Debian Woody

Casper Joost Eyckelhof `<joost@dnd.utwente.nl>`
University of Twente

v0.1, 01 October 2001

This document briefly describes how to set up MMBase on a Debian Gnu/Linux (Woody) system while using as much default packages as possible.

---

1. Introduction

1.1 Homepage

If you got this document from a Linux HOWTO mirror site or a CD−ROM, you might want to check back to the MMBase Mini−HOWTO home page to see if there's a newer version around.

This document only covers a very small portion of MMBase installation, for the official documentation visit MMBase home page.

1.2 Disclaimer

No liability for the contents of this documents can be accepted. Use the concepts, examples and other content at your own risk.

All copyrights are held by their respective owners, unless specifically noted otherwise. Use of a term in this document should not be regarded as affecting the validity of any trademark or service mark.

Finally, this is my first HOWTO, my first experience with linuxdoc and I am no expert on MMbase either. I am just trying to share my solutions to some problems I encountered while installing MMBase on a Debian system.

1.3 Copyright

1.4 Acknowledgements

I would like to thank a number of people for helping me during my quest for a clean installation. Although I do not know their names, their nicknames on irc are: *keesj, jdg, flax, scb2*. I hope they forgive me for not trying to find out their real names.

I would also like to thank my employer, the *University of Twente* for allowing me to write this document during working hours.

---

2. What is MMBase?

On their website, the authors say the following about it:

- MMBase is an opensource publishing system. MMBase can be used to create big websites that can be maintained easily.
- MMBase separates content from layout, in this way information can be reused easily.
- MMBase is very unique in the way it maintains its information. MMBase creates an object cloud that is a digital representation of the 'real' world objects that are of interest for you.

In this document I will describe how to get MMBase running on a Debian Woody System, using as many standard packages as possible. Though it is very easy to run MMBase using their automatic installation tools, you will end up with their version of a webserver, servlet engine and database.

After a few days of playing with my own install, I think I can help others by describing which choices I made to get things running. I do not claim this is the only way, or the best way! But it worked for me :)

The version of MMBase used is 1.4. Ofcourse everything might be different with another version.

MMbase can run together with various webservers, databases, servlet engines and JRE's. The programs/packages/versions I chose are:

- Apache (1.3.19)
- MySQL (3.23)
- Tomcat (3.2.3)
- Java (1.3)
- mod−jk (3.2.3)

Attention: The version numbers are those that were available in the Woody distribution at the time of writing. Except for Java, which came from a different location. See Non standard package(s).

---

3. Prerequisites

You will need a working Debian Woody installation, including apt, and make sure the following packages are installed and working correctly.

3.1 Standard packages

- apache
- tomcat
- mysql−server
- libapache−mod−jk
- imagemagick

If these packages are not installed, you can do so by typing

```
apt−get install packagename
```

where packagename is one of the above.

Note: naturally you will need al dependencies too, but apt will take automatically care of that.

3.2 Non standard package(s)

While writing this document, Woody only contains java 1.1, but for MMBase version 1.2 or higher is needed. I installed the packages from Blackdown. A list of Blackdown mirrors is at http://www.blackdown.org/java−linux/mirrors.html.

I added the following to my /etc/apt/sources.list , but the path might be different for different mirrors:

```
deb ftp://ftp.nluug.nl/pub/os/Linux/java/jdk/debian woody non-free
```

After that you can just type

```
apt-get install j2sdk1.3
```

Note: Although it must be possible to have more JRE's installed next to eachother, I wouldn't recommend it if it's not absolutely neccesary.

---

4. Getting MMBase and setting up some things

4.1 Getting MMBase

Now that you have all supporting packages up and running (you *did* follow the instructions in the previous section, didn't you?) get yourself a copy of MMBase. As said before, this document describes the works for MMbase version 1.4.

Download MMBase from http://www.mmbase.org/releases/MMBase−1.4.zip with a browser, or even easier:

```
wget http://www.mmbase.org/releases/MMBase-1.4.zip
```

In this document I will assume that MMbase will be unpacked in /usr/local/mmbase/ so do something like this

```
mkdir /usr/local/mmbase
unzip MMBase-1.4.zip -d /usr/local/mmbase
```

4.2 Getting mm.mysql

You will also need mysql.jar (JDBC driver for MySQL) which can be downloaded from http://mmmysql.sourceforge.net/.
Put the file in your /usr/local/mmbase/lib/ directory and create a symbolic link to mysql.jar.

```
cd /usr/local/mmbase/lib
wget http://mmmysql.sourceforge.net/dist/mm.mysql-2.0.2-bin.jar
ln -s mm.mysql-2.0.2-bin.jar mysql.jar
```

4.3 Creating a database for MMBase

MMbase needs a database and a user that has full rights on that database. The user seems to needs rights via localhost and the real hostname. This was not entirely clear to me.

Issue the following commands on the MySQL prompt. I chose a database called MMBase, a user named mmbaseuser and the password secret.

```
CREATE DATABASE MMBase
USE MMBase
```

```
GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP ON MMBase.* TO mmbaseuser@'%' IDENTIFIED BY 'secret';
GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP ON MMBase.* TO mmbaseuser@localhost IDENTIFIED BY 'secret'
GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP ON MMBase.* TO mmbaseuser@realhostname IDENTIFIED BY 'secret'
flush privileges;
```

4.4 Setting MMBase as tomcat ROOT

This probably isn't the cleanest way to do it, but we need to let tomcat know about MMBase, which I did by making a symbolic link from /var/lib/tomcat/webapps/ROOT to /usr/local/mmbase/html.

```
localhost# ls -la /var/lib/tomcat/webapps/ROOT
lrwxrwxrwx    1 root     root           22 Sep 24 12:01 /var/lib/tomcat/webapps/ROOT -> /usr/local/mmbase/html
```

---

5. Configuration files

There are various configuration files that need some adjustment or have to be created.

5.1 apache

In my setup, I created a virtual host to handle all MMBase request, but using a few *Alias* directors it cannot be too hard to include MMBase somewhere in your normal web tree.

```
...
LoadModule jk_module /usr/lib/apache/1.3/mod_jk.so
...
<VirtualHost x.y.z.a:80>
   ServerName mmbase.idonotexist.com
   DocumentRoot /usr/local/mmbase/html
   # mod_jk specific
   <IfModule mod_jk.c>
      JkMount /*.jsp ajp12
      JkMount /*.shtml ajp12
      JkMount /servlet/* ajp12
      JKMount /mmservlets/* ajp12
      JKMount /*.db ajp12
      <Directory "/usr/local/mmbase/html/">
          Options Indexes FollowSymLinks
      </Directory>
      <Location "/mmbase/WEB-INF/">
         AllowOverride None
         deny from all
      </Location>
   </IfModule>
</VirtualHost>

#Config for mod_jk :
<IfModule mod_jk.c>
   AddModule mod_jk.c
   JkWorkersFile /usr/share/tomcat/conf/workers.properties
   JkLogFile /usr/share/tomcat/logs/mod_jk.log
   JKLogLevel info
</IfModule>
```

5.2 tomcat

Make sure that in the file /usr/share/tomcat/conf/workers.properties the line workers.java_home=/usr/lib/j2sdk1.3 appears, and not a reference to another java version.

The file /etc/tomcat/server.xml needs 2 additions when using virtual hosts. If you are not, you can omit the Host−directive.

```
...
<!-- Request Interceptor for mmbase-jumpers -->
        <RequestInterceptor
        className="org.mmbase.servlet.jumpersInterceptor" />
...
<Host name="mmbase.idonotexist.com" >
           <Context path=""
           docBase="/usr/local/mmbase/html" />
</Host>
```

Both /var/lib/tomcat/webapps/ROOT/WEB−INF/web.xml and /etc/tomcat/web.xml have the same content. This must be wrong but it solved a problem at my system and I haven't seen any negative side effects yet. In theory only /var/lib/tomcat/webapps/ROOT/WEB−INF/web.xml should be edited, for that is the application specific configuration. The /etc/tomcat/web.xml is a global tomcat configuration file. If anybody knows what I'm doing wrong, please tell me, so I can update it in the next revision.

```
<?xml version="1.0"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN" "http://java.sun.com/j2ee/dtds/web-app_2_2.dtd">

<web-app>
      <context-param>
          <param-name>mmbase.config</param-name>
          <param-value>/usr/local/mmbase/config/default/</param-value>
      </context-param>

      <!-- Performance -->
      <servlet>
      <servlet-name>performance</servlet-name>
      <servlet-class>org.mmbase.servlet.performance</servlet-class>
      <load-on-startup>1</load-on-startup>
      </servlet>

      <!-- servscan -->
      <servlet>
      <servlet-name>servscan</servlet-name>
      <servlet-class>org.mmbase.servlet.servscan</servlet-class>
      <load-on-startup>2</load-on-startup>
      </servlet>

      <!-- servdb -->
      <servlet>
      <servlet-name>servdb</servlet-name>
      <servlet-class>org.mmbase.servlet.servdb</servlet-class>
      <load-on-startup>3</load-on-startup>
      </servlet>

      <servlet-mapping>
      <servlet-name>performance</servlet-name>
      <url-pattern>/performance.html</url-pattern>
      </servlet-mapping>
```

```
      <servlet-mapping>
      <servlet-name>servscan</servlet-name>
      <url-pattern>*.shtml</url-pattern>
      </servlet-mapping>

      <servlet-mapping>
      <servlet-name>servdb</servlet-name>
      <url-pattern>/img.db</url-pattern>
      </servlet-mapping>

      <servlet-mapping>
      <servlet-name>servdb</servlet-name>
      <url-pattern>/xml.db</url-pattern>
      </servlet-mapping>

      <servlet-mapping>
      <servlet-name>servdb</servlet-name>
      <url-pattern>/dtd.db</url-pattern>
      </servlet-mapping>

      <taglib>
        <taglib-uri>http://www.mmbase.org/mmbase-taglib-0.8</taglib-uri>
        <taglib-location>/WEB-INF/mmbase-taglib.tld</taglib-location>
      </taglib>

      <!-- add default.jsp and index.shtml to the default index files -->
      <welcome-file-list>
         <welcome-file>default.jsp</welcome-file>
         <welcome-file>index.jsp</welcome-file>
         <welcome-file>index.shtml</welcome-file>
         <welcome-file>index.html</welcome-file>
      </welcome-file-list>
</web-app>
```

The debian specific start/stop file was edited to get the CLASSPATH right and to make sure that java 1.3 was used. This too is an ugly hack, and should be addressed somewhere else, but hey! it works :)
Add the following to /etc/init.d/tomcat

```
...
JDK_DIRS="/usr/lib/jdk1.1 /usr/lib/ibm-java/jdk118 /usr/lib/j2sdk1.3 /usr/lib/j2re1.3"
CLASSPATH="/usr/lib/j2re1.3/lib/rt.jar:/usr/local/mmbase/lib/mysql.jar:/usr/local/mmbase/build/classes/:/usr/local/mmbase/lib/log4j.jar:$CLASSPATH"
export CLASSPATH
...
```

5.3 MMBase

Although many things have to be configured on MMbase itself, I have added a few subjects to this mini−HOWTO.

The database access should be set in /usr/local/mmbase/config/default/modules/jdbc.xml :

```
<property name="url">
   jdbc:mysql://localhost/MMBase?user=mmbaseuser&password=secret
</property>
```

The builder for images contains a path to convert that is not Debian−compliant. Change /usr/local/mmbase/config/default/builders/images.xml

```
<properties>
        <property name="ImageConvert.ConverterCommand">bin/convert</property>
        <property name="ImageConvert.ConverterRoot">/usr/</property>
        <property name="ImageConvertClass">org.mmbase.module.builders.ConvertImageMagick</property>
        <property name="MaxConcurrentRequests">3</property>
</properties>
```

6. Getting more help

There are various sources for more help on most of the subject covered in this mini−HOWTO. All packages have their own documentation. Some links that really helped me are:

- Apache: http://httpd.apache.org/docs/
- mod_jk: http://jakarta.apache.org/tomcat/tomcat−3.2−doc/mod_jk−howto.html
- Tomcat: http://jakarta.apache.org/tomcat/tomcat−3.2−doc/uguide/tomcat_ug.html
- MMBase: Click on Documentation on http://www.mmbase.org/
- Java: http://java.sun.com/

Another great source for help is channel #mmbase on ircnet, usually there are some developers there.