

Matisse[®] Installation Guide for Linux

January 2017



Matisse Installation Guide for Linux

Copyright © 2017 Matisse Software Inc. All Rights Reserved.

This manual and the software described in it are copyrighted. Under the copyright laws, this manual or the software may not be copied, in whole or in part, without prior written consent of Matisse Software Inc. This manual and the software described in it are provided under the terms of a license between Matisse Software Inc. and the recipient, and their use is subject to the terms of that license.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and FAR 52.227-19.

The product described in this manual may be protected by one or more U.S. and international patents.

TRADEMARKS: MATISSE and the MATISSE logo are registered trademarks of Matisse Software Inc. All other trademarks belong to their respective owners.

PDF generated 7 January 2017

Contents

Installing Matisse on Linux	4
System Requirements	4
Hardware	4
Distribution	4
Software Requirements	4
Installing Matisse DBMS on Linux	4
Uninstall Matisse	4
Install Matisse	5
After the installation	5
Installing Matisse In-Memory on Linux	6
Uninstall Matisse	6
Install Matisse	6
After the installation	7
Installing Matisse Lite on Linux	7
Install Matisse Lite	7
After the installation	8
Installation Notes	8
Installing the Matisse ODBC Driver for Linux	8
Setting Up	9
Testing the mt_odbc Utility	9
Managing Matisse Services	10
Port Monitors	10
Server Manager Listener	10
Managing Databases Autorestart	11
Autorestart	11
Autoshutdown	12
Troubleshooting your Matisse Installation	12
Install Directory	12
Enterprise Manager	13
Matisse Port Monitor	13
Remote Server Access	13

Installing Matisse on Linux

System Requirements

Before installing Matisse on Linux, you must have the following system requirements:

Hardware

- ◆ At least 64 MB of RAM; 128 MB RAM is recommended
- ◆ 100 MB disk space
- ◆ x86 (32 bit), x86_64 (64 bit) AMD or Intel chip families

Distribution

- ◆ The most popular Red-Hat based Linux distributions
- ◆ The most popular Debian-based Linux distributions
- ◆ Any Linux distribution, where Matisse has not been tested, requires Linux kernel 2.6.18 on systems based on x86 (32-bits) or x86_64 (64-bits) chip families.

Software Requirements

- ◆ The Matisse Enterprise Manager requires that you have the Java Runtime Environment (JRE) installed on your machine. We recommend Java 7 or higher.
- ◆ If you plan to develop applications with the Matisse Java binding you must install JDK 1.7.0 or higher.
- ◆ For developing applications in C, C++ you will need gcc 4.1 or higher, and g++ 4.1 or higher.

Installing Matisse DBMS on Linux

This section guides you through the installation of Matisse DBMS on Red-Hat based Linux distributions as well as on Debian-based Linux distributions.

Uninstall Matisse

Before installing, you need to stop all the running Matisse databases and uninstall your previous installation of Matisse. This will preserve your databases and any file that you may have created or updated in the installation directories.

Red-Had based To uninstall Matisse from your system, use the following commands.

```
> rpm -q matisse
    matisse-9.x-y
> rpm -e matisse-9.x-y
```

Debian based To uninstall Matisse from your system, use the following commands.

```
> dpkg -l matisse
||/ Name      Version      Description
+++-----
ii matisse 9.x-y       Matisse Post-Relational SQL Database

> sudo dpkg -r matisse
```

Install Matisse To install Matisse in your system, you need to become the root user.

Red-Had based To install Matisse on your Red-Had based system, use the following command.

```
> rpm -ihv matisse-9.x-y.i386.rpm
```

The file name may vary depending on the version. This installs matisse in the default directory `/usr/local/matisse`. We refer to this directory as `<install_dir>` throughout this document.

You may choose another directory with the following command:

```
> rpm -ihv --prefix <your_install_dir> matisse-9.x-y.i386.rpm
```

Debian based To install Matisse on your Debian-based system, you need to first convert the RPM file into a Debian installation file using `Alien`. The Matisse installation package contains pre/post-installation scripts that need to be converted as well.

If `Alien` is not yet installed on your machine, run the following command.

```
> sudo apt-get install alien
```

To proceed with the installation use the following sequence of commands.

```
> sudo alien -k --scripts matisse-9.x-y.i386.rpm
> sudo dpkg -i matisse_9.x-y_i386.deb
```

After the installation

At this stage, you do not need to be logged as root. Verify that you have the Java Runtime Environment installed on your machine. For instance:

```
> which java
/opt/jdk1.7.0_02/bin/java
```

Set the proper environment variable to run Matisse. You may execute one of the following environment setup scripts depending on your shell:

```
> . <install_dir>/mt_env.sh(sh, bash, ksh shells)
> source <install_dir>/mt_env.csh(csh, tcsh shells)
```

Then you can launch the Matisse Enterprise Manager to create or start your databases:

```
> mt_emgr &
```

Open the file `readme.html` located in `<install_dir>` for a quick start with using Matisse.

Installing Matisse In-Memory on Linux

This section guides you through the installation of Matisse In-Memory on Red-Hat based Linux distributions as well as on Debian-based Linux distributions.

CAUTION: Matisse In-Memory is only available for Linux 64-bit distributions

Uninstall Matisse

Before installing, you need to stop all the running Matisse databases and uninstall your previous installation of Matisse. This will preserve your databases and any file that you may have created or updated in the installation directories.

Red-Had based To uninstall Matisse from your system, use the following commands.

```
> rpm -q matisseim
    matisseim-9.x-y
> rpm -e matisseim-9.x-y
```

Debian based To uninstall Matisse from your system, use the following commands.

```
> dpkg -l matisseim
||/ Name      Version      Description
+-+-----+-----+-----+
ii matisseim 9.x-y       Matisse Post-Relational SQL
Database

> sudo dpkg -r matisseim
```

Install Matisse

To install Matisse in your system, you need to become the root user.

Red-Had based To install Matisse on your Red-Had based system, use the following command.

```
> rpm -ihv matisseim-9.x-y.x86_64.rpm
```

The file name may vary depending on the version. This installs matisse in the default directory `/usr/local/matisse`. We refer to this directory as `<install_dir>` throughout this document.

You may choose another directory with the following command:

```
> rpm -ihv --prefix <your_install_dir> matisseim-9.x-  
y.x86_64.rpm
```

Debian based To install Matisse on your Debian-based system, you need to first convert the RPM file into a Debian installation file using `Alien`. The Matisse installation package contains pre/post-installation scripts that need to be converted as well.

If `Alien` is not yet installed on your machine, run the following command.

```
> sudo apt-get install alien
```

To proceed with the installation use the following sequence of commands.

```
> sudo alien -k --scripts matisseim-9.x-y.x86_64.rpm  
> sudo dpkg -i matisseim_9.x-y_x86_64.deb
```

After the installation

At this stage, you do not need to be logged as root. Verify that you have the Java Runtime Environment installed on your machine. For instance:

```
> which java  
/opt/jdk1.7.0_02/bin/java
```

Set the proper environment variable to run Matisse. You may execute one of the following environment setup scripts depending on your shell:

```
> . <install_dir>/mt_env.sh(sh, bash, ksh shells)  
> source <install_dir>/mt_env.csh(csh, tcsh shells)
```

Then you can launch the Matisse Enterprise Manager to create or start your databases:

```
> mt_emgr &
```

Open the file `readme.html` located in `<install_dir>` for a quick start with using Matisse.

Installing Matisse Lite on Linux

This section guides you through the installation of Matisse Lite on any Linux distributions.

Install Matisse Lite

To install Matisse in your system, select the location you want Matisse Lite to be installed and untar the compressed tar file.

```
> tar zxvf matisselite900x32.tar.gz
```

After the installation

Verify that you have the Java Runtime Environment installed on your machine. For instance:

```
> which java
/opt/jdk1.7.0_02/bin/java
```

Run once the post installation script to update the environment setup scripts (`mt_env_lite.sh`, `mt_env_lite.csh`):

```
> sh <install_dir>/sys/mt_postinstall_lite.sh <install_dir>
```

Set the proper environment variable to run Matisse. You may execute one of the following environment setup scripts depending on your shell:

```
> . <install_dir>/mt_env_lite.sh(sh, bash, ksh shells)
> source <install_dir>/mt_env_lite.csh(csh, tcsh shells)
```

Then you can launch the Matisse Enterprise Manager to create or start your databases:

```
> mt_emgr_lite &
```

Open the file `readme.html` located in `<install_dir>` for a quick start with using Matisse.

Installation Notes

◆ Compatibility:

Matisse 9.0 client/server dialog is incompatible with all platforms running previous versions of Matisse. Therefore, you should ensure that your client applications are linked with the latest 9.0 client library.

◆ Troubleshooting:

For troubleshooting, or for a more detailed description of the installation process, please refer to the last section of this document.

◆ Problems/feedback:

Please report any problem or feedback to our support at support@matisse.com via our feedback page at www.matisse.com.

Installing the Matisse ODBC Driver for Linux

The Matisse ODBC driver included with the standard Matisse installation currently supports the open source Driver Managers iODBC (at www.iodbc.org) and unixODBC (at www.unixodbc.org), with ODBC 2.0 API functions.

We do not detail here the unixODBC installation, as it is very similar to the iODBC installation.

Setting Up

First, the iODBC driver manager must be installed and the path for the driver manager must be in the `LD_LIBRARY_PATH` of your shell configuration file.

Visit the iODBC website (at www.iodbc.org) and download the source for the iODBC open source ODBC driver manager, and compile it on your machine. It only takes a few minutes to configure and install.

Since the default installation directory of the driver manager is `/usr/local/iodbc`, and the driver manager library is in `/usr/iodbc/lib`, in your shell configure file (`.cshrc` or `.bashrc` for example) add the following to your `LD_LIBRARY_PATH`:

```
>LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/usr/local/iodbc/lib
```

In your home directory you need to write a file called `.odbc.ini` that has the information needed by both the driver manager and the driver itself when connecting to the database.

In the following example, Matisse is installed in the default directory `/usr/local/matisse`, so the ODBC driver library (`libmatisseODBC.so.9.x`) would be in the `/usr/local/matisse/lib/` directory. For instance if the database name is `accounts`, and the host computer being connected to is `cornsilk`, the format of the file needs to be as follows:

```
# Identifier for connection information to a specific
# database - called a DSN or data source name
[ODBC Data Sources]
matissedsn=MATISSE DATABASE

# DSN description contains path to driver manager library
# and connection information
[matissedsn]
Driver=/opt/matisse8/lib/libmatisseODBC.so.9.x
Database=accounts
HostName=cornsilk
```

Testing the mt_odbc Utility

To run the program `mt_odbc`, after starting your database, type:

```
> mt_odbc <data source name>
```

In the `.odbc.ini` file described previously, the data source name is `matissedsn`, so the command would be:

```
> mt_odbc matissedsn
```

After successful connection to your database, you may run any SQL statement. For instance to discover the classes in your database:

```
mt_odbc> SELECT MtName FROM MtClass;
MtName
```

```
-----  
MtClass  
...
```

Managing Matisse Services

The `mt_services.sh` shell script located in the `sys` directory of a Matisse installation holds the commands to start/stop the Matisse services after a reboot/shutdown of the machine. Logged as root, the DBA can update the file to activate/deactivate Matisse Services. By default, Matisse portmon for TPC/IP and for local socket (AF_UNIX) are activated. Matisse Server Manager Listener (SMListener) which requires Java 7 or higher to be installed and accessible on the machine is deactivated.

Port Monitors

Matisse port monitor allows client applications to connect to any Matisse server with a single port number. Matisse portmon for TPC/IP and for local socket (AF_UNIX) are activated. You need at least one portmon activated on the machine. If you only intend to run your application and the database server locally on the same machine, you may only need to active `local` connections.

```
start() {  
    [...]  
    # Start portmon services  
    ${MATISSE_HOME}/bin/mt_pmadm -s -p mtlocal -t local &&  
    success "mt_portmon local startup" || failure "mt_portmon  
    local startup"  
    RETVAL=$?  
    [...]  
}  
  
stop() {  
    [...]  
    # Stop portmon services  
    (${MATISSE_HOME}/bin/mt_pmadm -k -p mtlocal || killproc  
    mt_portmon) && success "$base mtlocal shutdown" || failure  
    "$base mtlocal shutdown"  
    [...]  
}
```

Server Manager Listener

Matisse Server Manager Listener (SMListener) manages remote operation requests on a local network. This service is optional and therefore is not activated by the default installation.

To activate the SMListener, you need to uncomment the JVM path (JAVA_JVM_PATH) and the library path (LD_LIBRARY_PATH) in the `sys/mt_services.sh` shell script and update the JVM path with the library path of your Java 7 installation.

```
# Matisse Server Manager Listener service
#JAVA_JVM_PATH=/opt/tools/j2se/jre/lib/i386/
#LD_LIBRARY_PATH=${JAVA_JVM_PATH}/server:${JAVA_JVM_PATH}/
client:${JAVA_JVM_PATH}:${LD_LIBRARY_PATH}
#
you also need to uncomment the script of the start() and
stop() function in the sys/mt_services.sh shell script.
start() {
    [...]
    # Matisse Server Manager Listener service
    #daemon ${MATISSE_HOME}/bin/mt_smlistener &
    #RETVAL=$?
    [...]
}

stop() {
    [...]
    # Matisse Server Manager Listener service
    #killproc mt_smlistener && success "$base mt_smlistener
shutdown" || failure "$base mt_smlistener shutdown"
    [...]
}
```

Managing Databases Autorestart

The `mt_databases.sh` shell script located in the `sys` directory of a Matisse installation holds the commands to start/stop the database automatically after a reboot/shutdown of the machine. Logged as root, the DBA can update the file to list the databases to be automatically started/stopped.

Autorestart

To automatically restart a database after a reboot, you need to uncomment the script of the `start()` function in the `sys/mt_databases.sh` shell script and update the script with your database name.

```
start() {
#
# To start a MATISSE database, uncomments the script below
# and list your databases
#
```

```

## if [ -f ${MATISSE_HOME}/bin/mts -a
${MATISSE_HOME}/bin/mt_server ] ; then
##     echo "Delaying Matisse databases startup for 3 sec
until portmons are started"
##     sleep 3
##     echo "Starting Matisse database <your_db_name>"
##     ${MATISSE_HOME}/bin/mt_server -d <your_db_name> start
## fi
}

```

Autoshutdown

To automatically stop a database during a shutdown, you need to uncomment the script of the `stop()` function in the `sys/mt_databases.sh` shell script and update the script with your database name.

```

stop() {
#
# To stop a MATISSE database, uncomments the script below
# and list your databases
#
## if [ -f ${MATISSE_HOME}/bin/mts -a
${MATISSE_HOME}/bin/mt_server ] ; then
##     echo "Stopping Matisse database <your_db_name>"
##     ${MATISSE_HOME}/bin/mt_server -d <your_db_name> stop
## fi
}

```

Troubleshooting your Matisse Installation

After following the standard Matisse installation procedure you should be able to run Matisse. In particular, the Matisse Port Monitor is automatically started by the installer, and set to be stopped and restarted during shutdown/reboot. It is also automatically stopped and removed by the uninstaller.

You may check the following guidelines if you experience any problem or if you want more control over the port monitor.

Install Directory

The commands `mt_env.sh` and `mt_env.csh` will set the `MATISSE_HOME`, `MATISSE_CFG`, `MATISSE_LOG` environment variables to `<install_dir>`. These variable should point to local filesystem paths.

If `<install_dir>` is a NFS directory, you will have to define new values to the `MATISSE_HOME`, `MATISSE_CFG`, `MATISSE_LOG` environment variables.

You may also verify that the `matisse` executables have write permission in the `log` directory and in the `config` directory.

```
> chmod a+w <install_dir>/log
> chmod a+w <install_dir>/config
```

Enterprise Manager

The Matisse Enterprise Manager is a Java application which requires Java 7 or higher to be installed and accessible on the machine. The commands `mt_env.sh` and `mt_env.csh` will set the path to the JVM whenever possible.

Matisse Port Monitor

The port monitor allows client applications to connect to any Matisse server with a single port number. To check if the port monitor is properly installed and started, first log as root, then verify (or add) the following line in `/etc/services`:

```
mtportmon      7421/tcp # Matisse port monitor
```

You may verify that it is running with `mt_pmadm`:

```
> mt_pmadm -L -p mttcp
11:17:33 PMADM-I-PMSTATE, Port Monitor is enabled
11:17:33 PMADM-I-TRPTYPE, Transport type: tcp
11:17:33 PMADM-I-NOSVCLIST, No services registered
```

If you do not have a port monitor running, login as root and set your environment variables, then start the port monitor. Then verify that it is properly running with the `-L` option or by checking the log file:

```
# mt_pmadm -s -p mttcp -t tcp
# cat <install_dir>/log/mttcp.myhost.log
17:27:00 MATISSE Portmon Version 9.0.0.0
17:27:00 PORTMON-I-ISTATE, Initial state: enabled
17:27:00 PORTMON-I-INITCOMPLETE, Initialization complete
```

Remote Server Access

If you cannot connect to a remote database server because of errors such as `INVTRANSPORT` or `SRVCONFAILED` while you can establish local connections without any problem, you may need to double check your firewall settings. Please refer to the [Matisse® Server Administration Guide](#) for more details on updating your firewall settings.