



Application of MyCoRe with Tamino4 on Sun Solaris8

User Guide

Lili Tan
lilitan@tiscali.de
tan@hrz.uni-essen.de

Computer Center, University of Essen
Schützenbahn 70, 45127 Essen, Germany

Created on October 10, 2003

TABLE OF CONTENTS

1	Introduction	3
2	Organizing Mycore Environment.....	3
2.1	MindTerm: Download, Installation and Usage.....	4
2.2	WS-FTP pro: Download, Installation and Usage.....	4
2.3	MySQL: Download, Installation and Usage.....	5
3	Preparing Mycore Codes	6
3.1	Check out/Download Mycore Codes:	6
3.2	Tansport Mycore Codes from Windows to Sun Solaris via WS-FTP	6
3.3	Configuration of Mycore Codes	6
4	Implementation Steps of Mycore Appication.....	7
4.1	Ant Build MyCore Jar for Tamino DB	7
4.2	Clean up Mysql and Tamino DB if needed	7
4.3	Loading Documents for Mycore Sample Application	8
4.4	Building and Deploy Web Application.....	9

LIST OF FIGURES

<i>Number</i>	<i>Page</i>
Figure 1: MinderTerm Interface.....	4
Figure 2: WS-FTP Interface.....	5

1 Introduction

This user guide tells you how to perform Mycore-Sample-Application combined with the implementation of Tamino4 on Sun Solaris Solaris8 64-bit (SunOS 5.8). It is written for open source project “MILESS/ MyCoRe” (<http://www.mycore.de>), and especially for those who want to implement XML native database to the mycore sample application. Tamino4 is a software product for XML: DB native database, which developed by Software AG (<http://www.softwareag.com/tamino/default.htm>). It enables XML documents to be stored, queried, and managed in native XML format.

This guide includes four parts: Introduction, Organizing Mycore Environment, Preparing Mycore Codes, and Implementation Steps for Mycore Application. This guide is also based on the implementation on the Sun Solaris machine, which described below in the Computer Center of University Essen:

Sun OS:

Sunfire2 5.8 Generic_108528_22

Sun 4u Sparc SUNW, Sun-Fire-280R

2 Organizing Mycore Environment

To perform Mycore application on Solaris, the following software is required:

- MindTerm : use to access Sun Solaris system
- WS-FTP pro: use to transfer files between Windows and Unix, Sun Solaris system
- Tamino4: for the MyCoRe XML persistence. S. the separate installation guide
- MySql: for the Storage of Mycore Classification.
- Ant Build: to do Ant Build about Mycore codes. Download Ant: 1.5.3-1, installation and usage can be referred to: <http://www.apache.org/dist/ant/binaries/>
- JDK 1.4: Download, installation and usage can be referred to: <http://java.sun.com>
- Useful Command line Usage: e.g.: s.
http://www-rohan.sdsu.edu/~corpora/corpus_course/unix-commands.html

Download, installation of MindTerm, WS-FTP pro, and MySql are introduced below:

2.1 MindTerm: Download, Installation and Usage

Download of AppGate MindTerm, e.g. “mindterm_2.4-bin” can be achieved from <http://www.appgate.com/mindterm/download.php>, unzip it in a folder on windows system. Then click on the unzip file “minterm.jar”, you can the MindTerm interface.

Usage: To connect to the remote server, please choose setting, input the of the server name, your Username, and then “Connect” button on the MindTerm Interface (s. Figure1: MindTerm Interface):

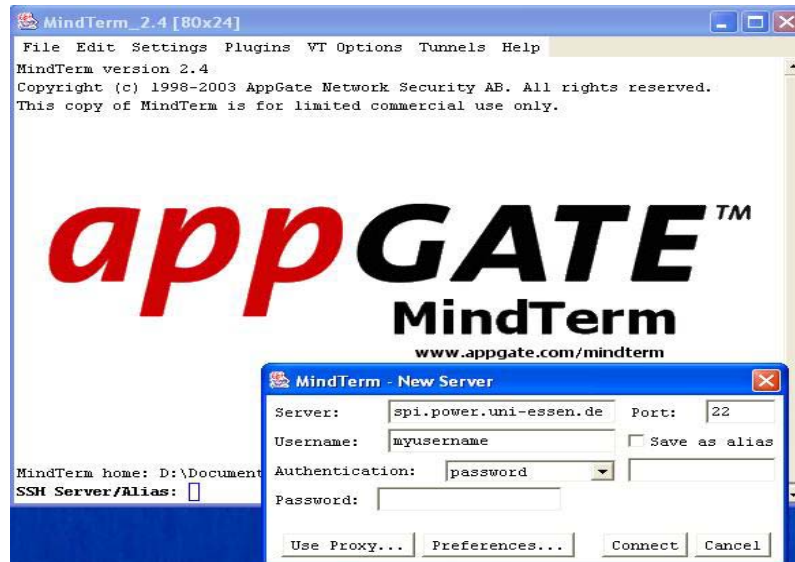


Figure 1: MinderTerm Interface

Password is also required then to connect the server.

To connect to Sun Solaris, input: “rsh: name-your-SunSolarismaschine”, login name and password. Ten you reach Sun Solaris through MinderTerm Interface.

2.2 WS-FTP pro: Download, Installation and Usage

Download WS-FTP pro 8.02 installation file “F_x86t32.exe” from <http://www.ipswitch.com/>.

Run it in Windows machine, input your login password, and then you get the WS-FTP interface. It provides an easy way to transport files between Windows and Sun Solaris machine (s. Figure 2: WS-FTP Interface).

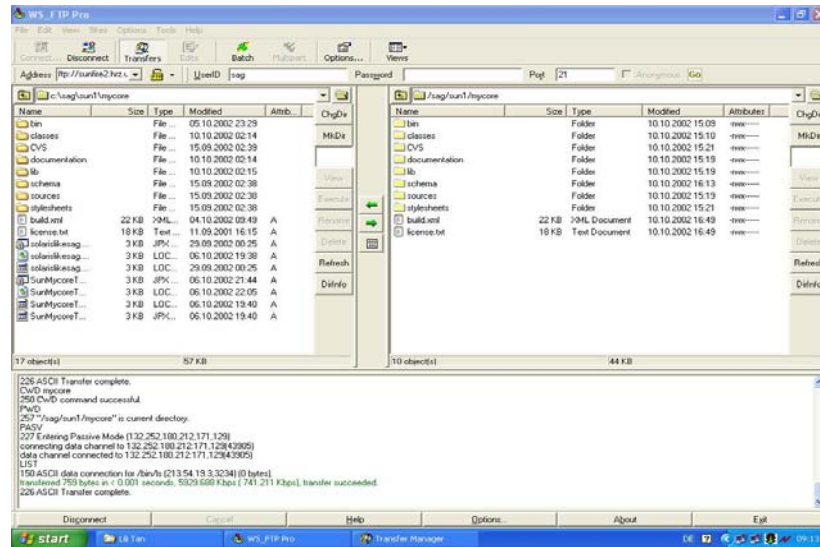


Figure 2: WS-FTP Interface

2.3 MySQL: Download, Installation and Usage

Download mysql zip file from <http://www.mysql.com/downloads/mysql-3.23.html>

Install it with default.

Usage:

- Access MySQL: SunSoarsisMachineName/ PathToMySQL: *bin/mysqld_safe*\$
- SunSoarsisMachineName/ PathToMySQL: *bin/mysql*
- Connect to MySQL: → *mysql> connect yourdatabasename;*
- Delete Table: → *mysql> drop table yourtablename;*

You have to input ";" at the end of each command line. In Sun Solaris, You have to input the capital letter of the name of the tables.

For MyCoRe Project developers, it is often to delete the following tables for new loading of classification:

- *mysql> drop table MCRCATEGORY;*
- *mysql> drop table MCRCATEGORYLABEL;*
- *mysql> drop table MCRCLASS;*
- *mysql> drop table MCRCLASSLABEL;*
- *mysql> drop table MCRFSNODES;*
- *mysql> drop table MCRLINKCLASS;*
- *mysql> drop table MCRLINKHREF;*

- Exit MySQL: → *mysql> exit;*

3 Preparing Mycore Codes

3.1 Check out/Download Mycore Codes:

The most current Mycore codes can be downloaded as zip file or check out from CVS repository from <http://www.mycore.de> into Windows machine. On windows machine, one can not only build the Mycore application on the resent machine, but can also transport the codes on Sun Solaris machine. Using CVS for checkout for download of zip file on Windows from CVS repository, please check the part of “User Guide: Application of MyCoRe with Tamino4 on WindowsXP Professional”.

3.2 Tansport Mycore Codes from Windows to Sun Solaris via WS-FTP

Via WS-FTP Interface, you can transport the Mycore Codes, which have been achieved from above in Windows machine, to Sun Solaris machine (s. Figure 2: WS-FTP Interface).

3.3 Configuration of Mycore Codes

- Point out XML store type and their path and name of Jar files in “build.properties” of the folder mycore/bin/, e.g.:

```
# Tamnio under Windows and Sun Solaris
MCR.XMLStore.Type=taminoxmldb
MCR.XMLStore.BaseDir=/sag/sun1/LibTaminoAddedtoMycore
MCR.XMLStore.Jars=tamino_xmldb.jar TaminoAPI4J.jar
```

- Point out your schema path in “mycore.properties” , e.g. :
MCR.parser_schema_path=/sag/sun1/dlwww/mycore-sample-application/schema
- Switch the persistence configuration to XML Tamino DB features in “mycore.properties.privates” file: e.g.:

```
#####
# The persistence configuration for Tamino XML:DB
#####

#The name of the Java class of the persistence layer
MCR.persistence_taminoxmldb_class_name=org.mycore.backend.taminoxmldb.MCRXMLDBPersistence
MCR.persistence_taminoxmldb_query_name=org.mycore.backend.taminoxmldb.MCRXMLDBQuery
MCR.persistence_taminoxmldb_encoding=ISO-8859-1

MCR.persistence_taminoxmldb_driver=com.softwareag.tamino.xmldb.api.base.TDatabase

#### on WinXP, plesase set your host name, port name and Tamino Database Name
#MCR.persistence_taminoxmldb_database_url=xmldb:tamino://localhost/tamino/Mycore_Sample_Application
#MCR.persistence_taminoxmldb_TaminoDATABASE_URI=http://localhost/tamino/Mycore_Sample_Application
####

#### on Sun Solaris, plesase set your host name, port name and Tamino Database Name
MCR.persistence_taminoxmldb_database_url=xmldb:tamino://sunfire2.hrz.uni-essen.de:8080/tamino/Mycore_Sample_Application
MCR.persistence_taminoxmldb_TaminoDATABASE_URI=http://sunfire2.hrz.uni-essen.de:8080/tamino/Mycore_Sample_Application
####

MCR.persistence_taminoxmldb_database=tamino
```

- Point out your JDBC parameter to connecting to Mysql in “mycore.properties.privates” file, e.g. :

```
MCR.persistence_sql_database_url=jdbc:mysql://localhost/test?user=
```

- Point out your internal file system filestore path in “mycore.properties.privates” file, e.g. :

```
MCR.IFS.ContentStore.FS.BaseDirectory=/sag/sun1/mycore-sample-application/filestore
```

4 Implementation Steps of Mycore Appication

4.1 Ant Build MyCore Jar for Tamino DB

It is supposed that your Mycore application’s environment are good prepared and mycore source code are ready configured. Implementation Steps show you how to use Ant build to compile sources codes and more until you get the web application of Mycore sample.

- Start “MinderTerm” by: clicking miderterm.jar on your window system, give your user name and password
- Reach the Sun Solaris go the directory where the “mycore” is by:

```
rsh sunfire2
```

```
login name
```

```
password
```

```
cd path-to-your-mycore
```

- Check MYCORE_HOME by: *echo \$MYCORE_HOME*,
If it is not in the directory you want to work with, change it by, e.g.:
export MYCORE_HOME=/sag/sun1/mycore
- Check and build Jar file in mycore path by:

```
ant info
```

```
ant clean
```

```
ant usage
```

```
ant JAR
```

4.2 Clean up Mysql and Tamino DB if needed

- Clean up Mysql and Tamino databse. If you are at the first time to buid Tamino Mycore sample application, skip this part and go directly to the next part.

Clean up Mysql:

Access Mysql: SunSoarsisMachineName/ PathToMysql: *bin/mysqld_safe\$*

SunSoarsisMachineName/ PathToMysql: *bin/mysql*

Connect to Mysql: → *mysql> connect yourdatabasename;*

Delete Table: → *mysql> drop table yourtablename;*

You have to input ";" at the end of each command line. In Sun Solaris, You have to input the capital letter of the name of the tables.

For MyCoRe Project developers, it is often to delete the following tables for new loading of classification:

```
→ mysql> drop table MCRCATEG;  
→ mysql> drop table MCRCATEGLABEL;  
→ mysql> drop table MCRCLASS;  
→ mysql> drop table MCRCLASSLABEL;  
→ mysql> drop table MCRFSNODES;  
→ mysql> drop table MCRLINKCLASS;  
→ mysql> drop table MCRLINKHREF;  
Exit Mysql: → mysql> exit;
```

Clean up Tamino database, need only to perform:

```
ant clean.taminocollection.
```

Because a Java class has been developed, which plays special attention for convenience of the Mycore developers, allowing without going every once and while between mysql and ant.

- Create databases for Mysql and Tamino with the names, which you have specified in your private property file. Also you have to start the database in MySql and Tamino before loading documents.

```
ant JAR
```

4.3 Loading Documents for Mycore Sample Application

- Loading Documents:

Go to directory mycore-sample-application, perform:

```
ant info, check if paths OK
```

```
ant usage
```

```
ant schema
```

```
ant users
```

```
ant load.classifications
```

ant tamino, then all of the XML contents for persistence are to be loaded into Tamino.

You can also perform the followings to see every loading step in detail:

```
ant taminoschema
```

```
ant load.taminoschema
```

```
ant load.legalentities
```

ant load.documents

ant load.derivates.

4.4 Building and Deploy Web Application

- Build Web application / War File by:

ant webapp / ant war

You can either start the web application with Sun One Studio, or copy the war file “mycoresample” into a Tomcat directory “webapps”. By stop and new start the Tomcat, you can achieve the Mycore sample application Mask via a browser in corresponding address.