

The NCIA3 Installation Environment

Paul Koppel
 Washington University School of Medicine
 Mallinckrodt Institute of Radiology
 Electronic Radiology Laboratory
pkoppel@wustl.edu
 July 2008

Table of Contents

Introduction.....	2
Creating a Xen virtual machine.....	2
Figure 1 virt-manager interface with Fedora 8.....	3
Figure 2 ncia3 virtual machine console.....	4
Install Java, Junit, and Ant.....	4
Modify .bashrc.....	5
Install JDK - use the non-rpm method	5
Install Junit.....	5
Install Ant.....	6
Create tomcat and jboss users.....	7
Figure 3 user manager showing users.....	7
Create ncia group.....	8
Figure 4 user manager showing groups.....	8
Install Tomcat.....	8
Tomcat Startup.....	9
Figure 5 service configuration.....	10
Install Java Image I/O API 1.0.....	10
Install MySQL.....	11
MySQL Startup.....	11
Create Database Accounts.....	12
Install MySQL Connector/J.....	12
Install CSMUPT and NCIA.....	13
Jboss startup.....	14
NFS Mounting tomcat webapps and downloaded data.....	14
Appendix A .bashrc for building ncia3.....	16
Appendix B tomcat startup script.....	16
Appendix C jboss startup script.....	19
Appendix D resetting ncia database.....	20
Appendix E migrating a Xen virtual machine.....	24

Introduction

This document describes many of the steps used to create the environment for installing ncia3 - The National Cancer Center Imaging Archive 3.0 that is now running at the Mallinckrodt Institute of Radiology at Washington University School of Medicine in St. Louis, MO. Although ncia3 installation has become easier with the automated installer scripts, there is still much that needs to be done in order to prepare the environment first. This document is therefore not a rehash of the NCIA installation documents (which have greatly improved over time) but a supplement that goes into some of the other nuts and bolts details – such as creating a virtual machine to host ncia3, environmental variables needed for building and running some of the support applications, and start-up scripts.

Creating a Xen virtual machine

The ncia3 virtual machine (guest operating system) is based on CentOS 5.2, and runs on top of Fedora 8 (base operating system). The hardware consists of a Dell PowerEdge 1950 with 8 GB RAM and two 146 GB drives running as RAID 1. The Poweredge 1950 supports full hardware virtualization. This option was used in creating the virtual machine ncia3.

In order to create a Xen-based virtual machine, the Xen kernel must be running. For example, in the base operating system:

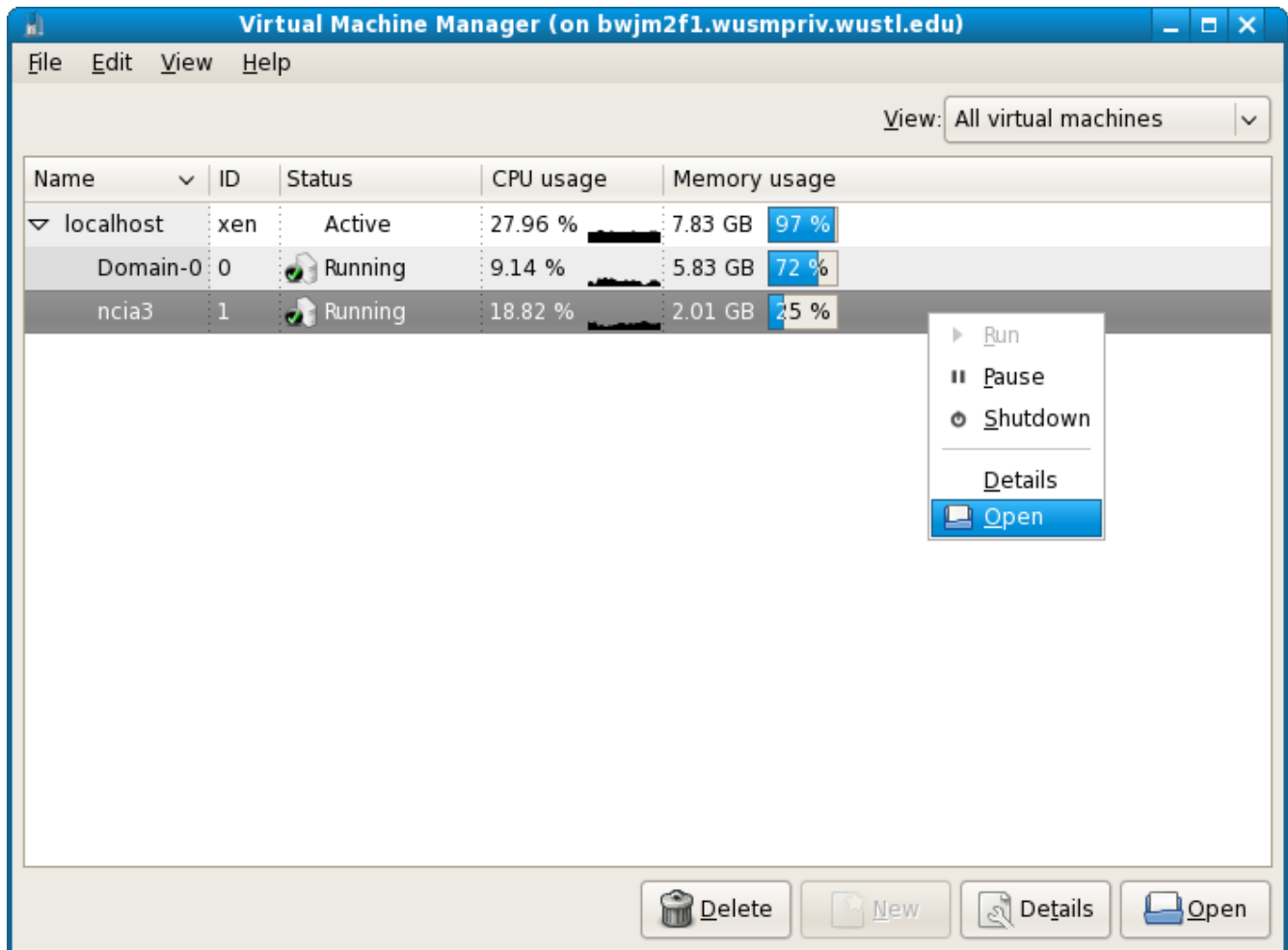
```
[root@bwjm2f1 koppel]# uname -r  
2.6.21.7-3.fc8xen
```

In addition, the virtualization packages must be installed:

```
[root@bwjm2f1 koppel]# rpm -qa | grep virt  
virt-viewer-0.0.2-2.fc8  
libvirt-python-0.4.2-1.fc8  
python-virtinst-0.300.2-4.fc8  
libvirt-0.4.2-1.fc8  
virt-manager-0.5.3-2.fc8
```

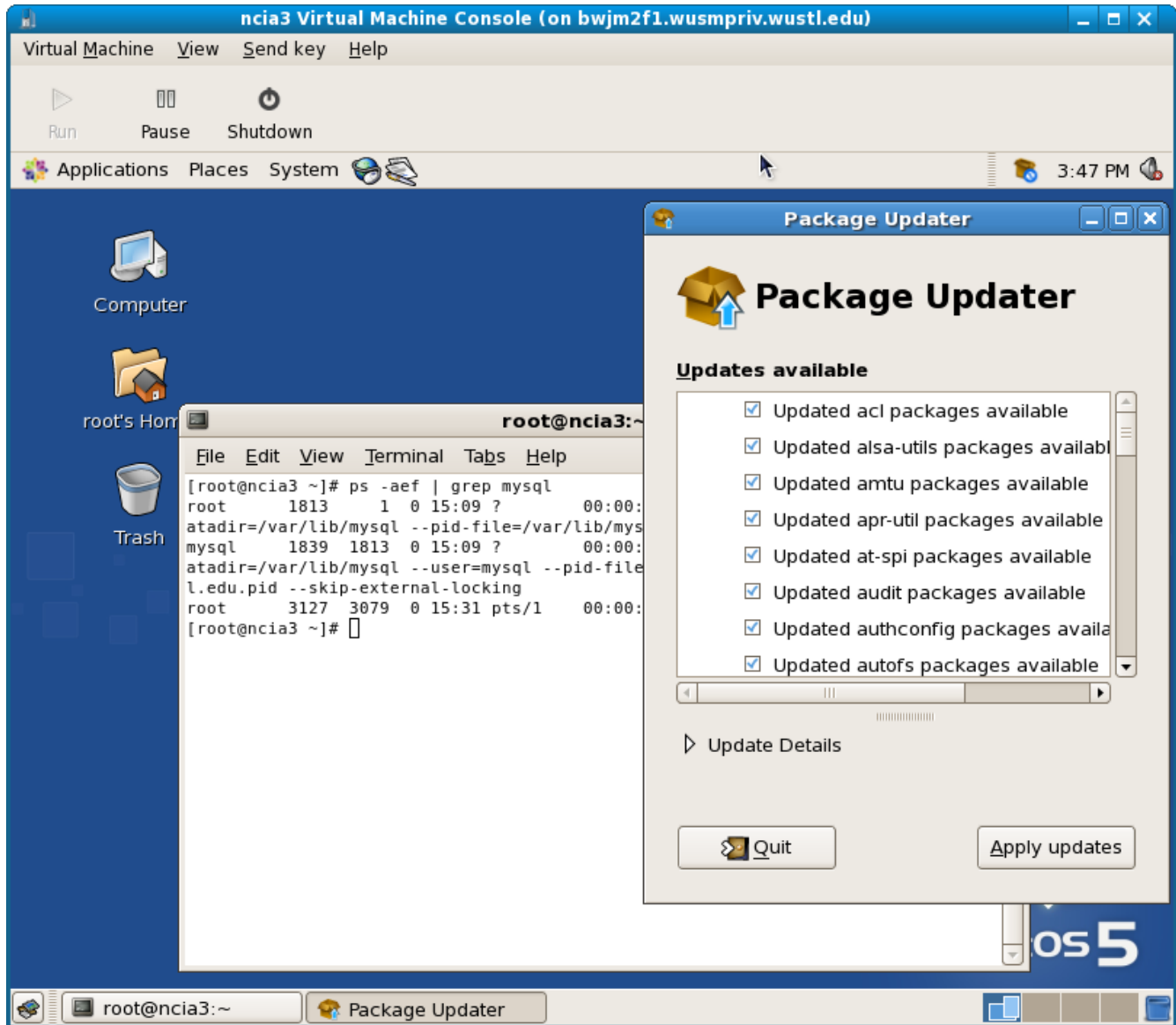
The Xen kernel and virtualization packages can all be easily installed from the software updater. The basic tool for creating and managing virtual machines is the *virt-manager* which comes with all of the recent Red Hat distributions and Red Hat clones such as CentOS.

Figure 1 *virt-manager* interface with Fedora 8



Once ncia3 is “opened”, the console is accessible. The system looks and behaves like any other real, physical server – even though it is virtualized:

Figure 2 ncia3 virtual machine console



Note that ncia3 happens to be the hostname of the virtualized system and also is used to refer to the software application itself - National Cancer Center Imaging Archive 3.0 Life Science Distribution (LSD) bundle.

Install Java, Junit, and Ant

Since the ncia3 application relies on java software, installing java is the first step. Ant is a java based build tool which uses junit – a framework to write repeatable tests and is used in building the ncia application.

Modify .bashrc

The default .bashrc login script needs to be modified so that the path to the new java, junit, and ant will be used, and not those that come with the operating system. Appendix A shows the .bashrc file that is used for building ncia3.

Install JDK - use the non-rpm method

Because the Dell Poweredge 1950 is a 64 bit, machine, a 64 bit java was used (the java that is installed with CentOS was left in place – but not used to run the ncia3 software). After downloading java from www.java.sun.com (your home directory is a good choice for download location), become root and create a new java directory. To install the java from Sun:

```
[koppel@ncia3 ~]$ chmod a+x jdk-1_5_0_14-linux-amd64.bin    (make install script executable)
[koppel@ncia3 ~]$ su root
Password:
[root@ncia3 koppel]# mkdir -p /usr/java
[root@ncia3 koppel]# cd /usr/java
[root@ncia3 java]# /home/koppel/jdk-1_5_0_14-linux-amd64.bin    (run the install script)
[root@ncia3 java]# ln -s /usr/java/jdk1.5.0_14/bin/java /usr/local/bin/java
[root@ncia3 java]# which java
/usr/java/jdk1.5.0_14/bin/java
```

The last step above is to make sure that the java that is being used is from Sun and not Red Hat.

Install Junit

Junit is needed by ant. After downloading from www.junit.org, install (as root) in /usr/java:

```
[root@ncia3 java]# pwd
/usr/java
[root@ncia3 java]# unzip /home/koppel/junit4.4.zip

[koppel@ncia3 ~]$ echo $JAVA_HOME
/usr/java/jdk1.5.0_14
[koppel@ncia3 ~]$ echo $CLASSPATH
/usr/local/mysql-connector-java-5.0.8/mysql-connector-java-5.0.8-
bin.jar:/usr/java/junit4.4/junit-4.4.jar:/usr/java/junit4.4
[koppel@ncia3 ~]$ java org.junit.runner.JUnitCore org.junit.tests.AllTests
JUnit version 4.4
```

Time: 44.992

OK (329 tests)

[koppel@ncia3 ~]\$

Install Ant

Ant is needed to run the ncia build scripts. After downloading from www.apache.org:

```
[koppel@ncia3 ~]$ tar xvzf apache-ant-1.7.0-src.tar.gz
```

```
[koppel@ncia3 ~]$ su root
```

Password:

```
[root@ncia3 koppel]# set | grep JAVA
```

```
JAVACMD=/usr/java/jdk1.5.0_14/bin/java
```

```
JAVA_HOME=/usr/java/jdk1.5.0_14
```

```
[root@ncia3 koppel]#
```

```
[root@ncia3 koppel]# set | grep ANT
```

```
ANT_HOME=/usr/local/ant
```

```
[root@ncia3 koppel]# cd apache-ant-1.7.0
```

```
[root@ncia3 apache-ant-1.7.0]# ./build.sh -Ddist.dir=/usr/local/apache-ant-1.7.0 dist
```

```
[root@ncia3 apache-ant-1.7.0]# cd /usr/local
```

```
[root@ncia3 local]# ln -s apache-ant-1.7.0 ant
```

```
[root@ncia3 local]# echo $PATH
```

```
/usr/kerberos/sbin:/usr/java/jdk1.5.0_14/bin:/usr/local/ant/bin:/usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:/home/koppel/bin
```

```
[root@ncia3 local]# ant -version
```

```
Apache Ant version 1.7.0 compiled on February 26 2008
```

```
[root@ncia3 local]#
```

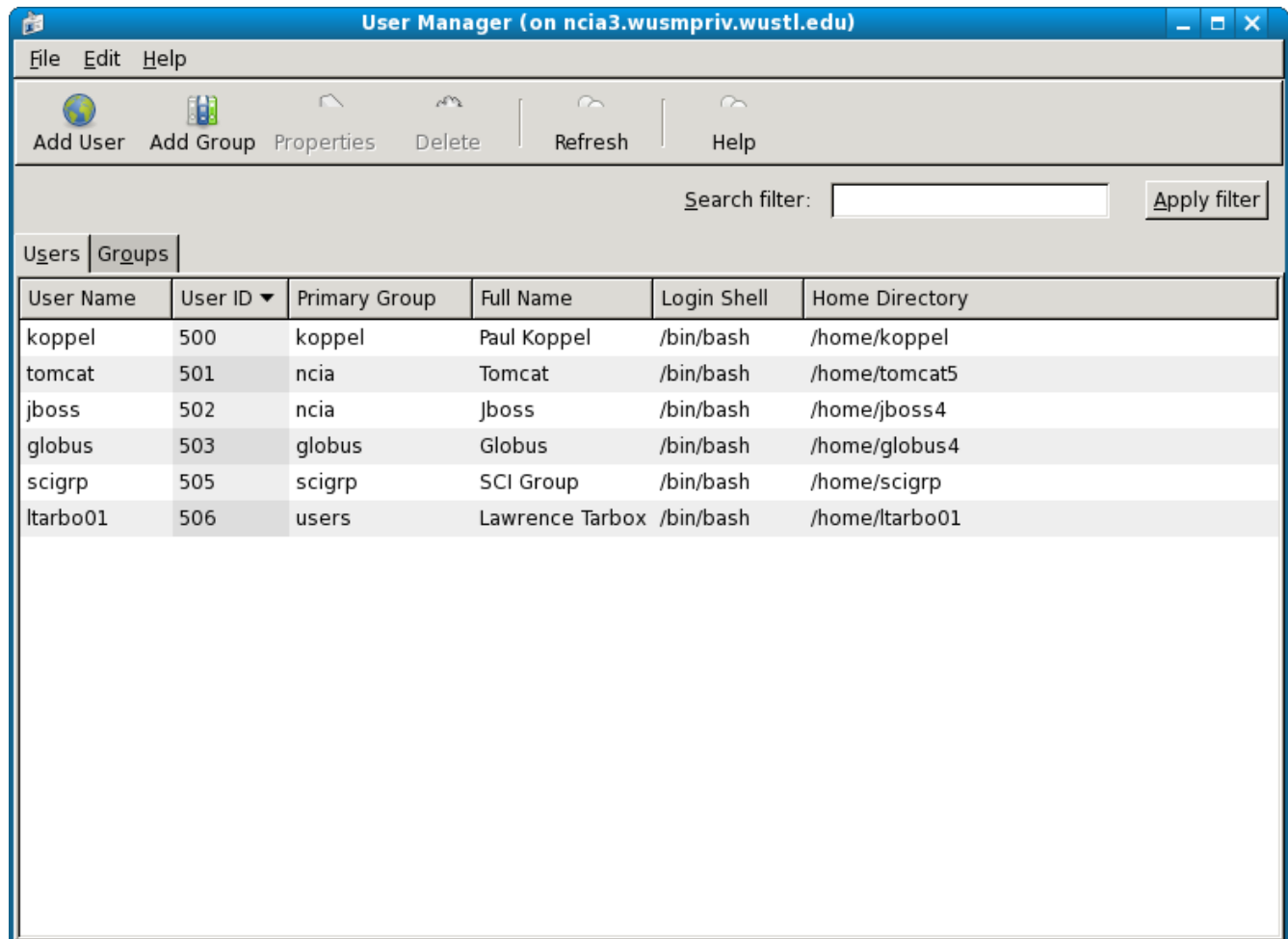
Create tomcat and jboss users

If these users do not exist, create them using *useradd* command or the graphical interface tool:

```
[root@ncia3 koppel]# /usr/bin/system-config-users &
```

which will bring up the following screen:

Figure 3 user manager showing users

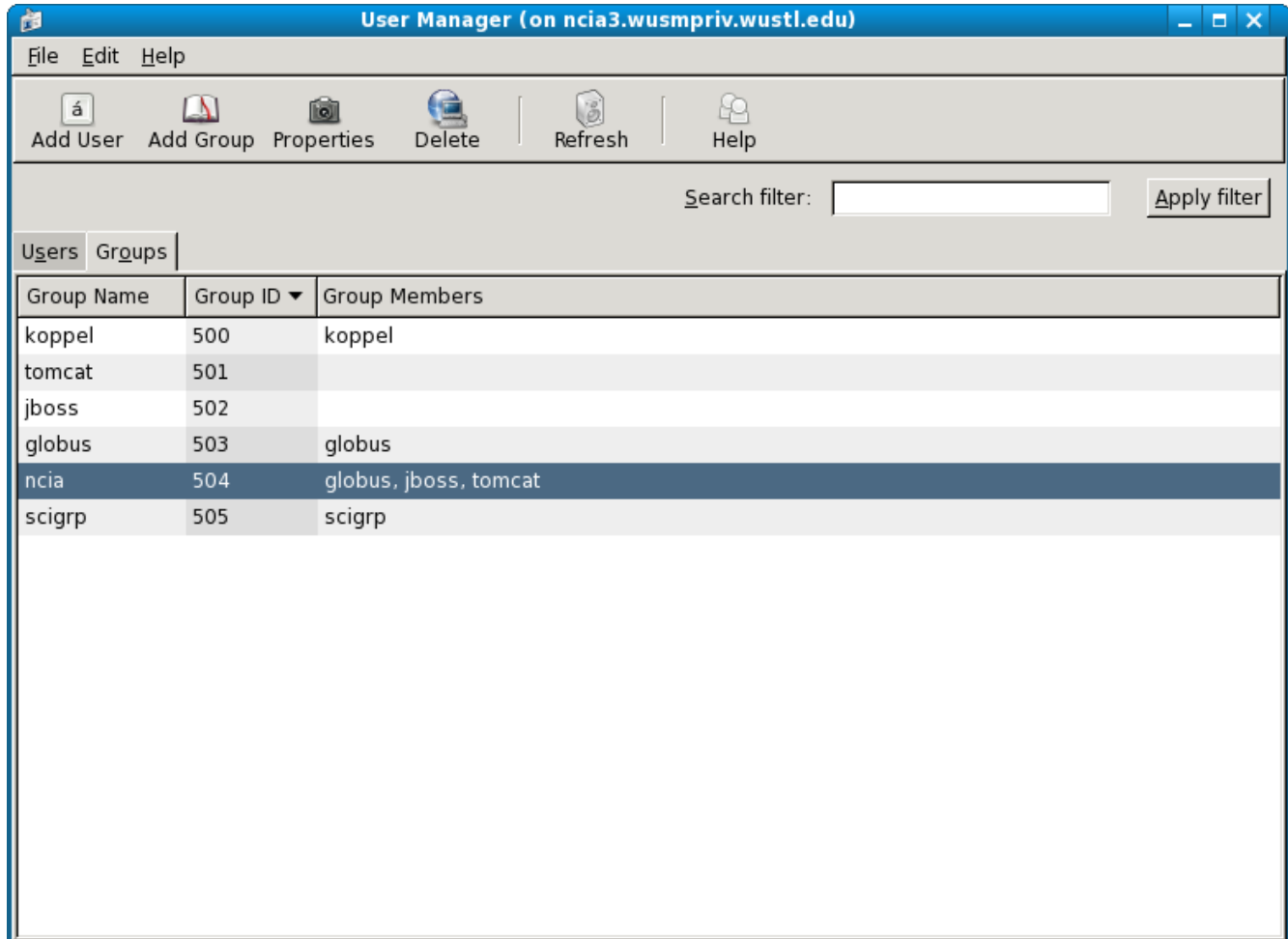


The tomcat and jboss users have a `.bashrc` file that is identical to the one shown in Appendix A. The mysql user will be created later as a consequence of the rpm installation of mysql – so we don't create a mysql user here.

Create ncia group

From the command line, add an ncia group (for example as root: `groupadd -g 504 ncia`) or use the graphical interface above. Make sure to add the jboss and tomcat users to the ncia group. Although a globus user is also part of the ncia group, it is not really used.

Figure 4 user manager showing groups



Install Tomcat

After downloading `apache-tomcat-5.5.25.tar.gz` from www.apache.org, tomcat was installed using the following steps:

```
[koppel@ncia3 ~]$ tar xvfz apache-tomcat-5.5.25.tar.gz
```

```
[koppel@ncia3 ~]$ su root
```

Password:

```
[root@ncia3 koppel]# mv apache-tomcat-5.5.25 /usr/local
[root@ncia3 koppel]# cd $CATALINA_HOME/bin
[root@ncia3 bin]# pwd
/usr/local/apache-tomcat-5.5.25/bin
[root@ncia3 bin]# tar xvfz jsvc.tar.gz
[root@ncia3 bin]# cd jsvc-src
[root@ncia3 jsvc-src]# autoconf
[root@ncia3 jsvc-src]# chmod a+x configure
[root@ncia3 jsvc-src]# ./configure
[root@ncia3 jsvc-src]# make
[root@ncia3 jsvc-src]# cp jsvc ..
[root@ncia3 init.d]# cd /usr/local
[root@ncia3 local]# chown -R tomcat:ncia apache-tomcat-5.5.25
[root@ncia3 local]# chmod -R 775 apache-tomcat-5.5.25
```

Tomcat Startup

To startup the tomcat application server at system boot, modify the example script that comes with the tomcat download.

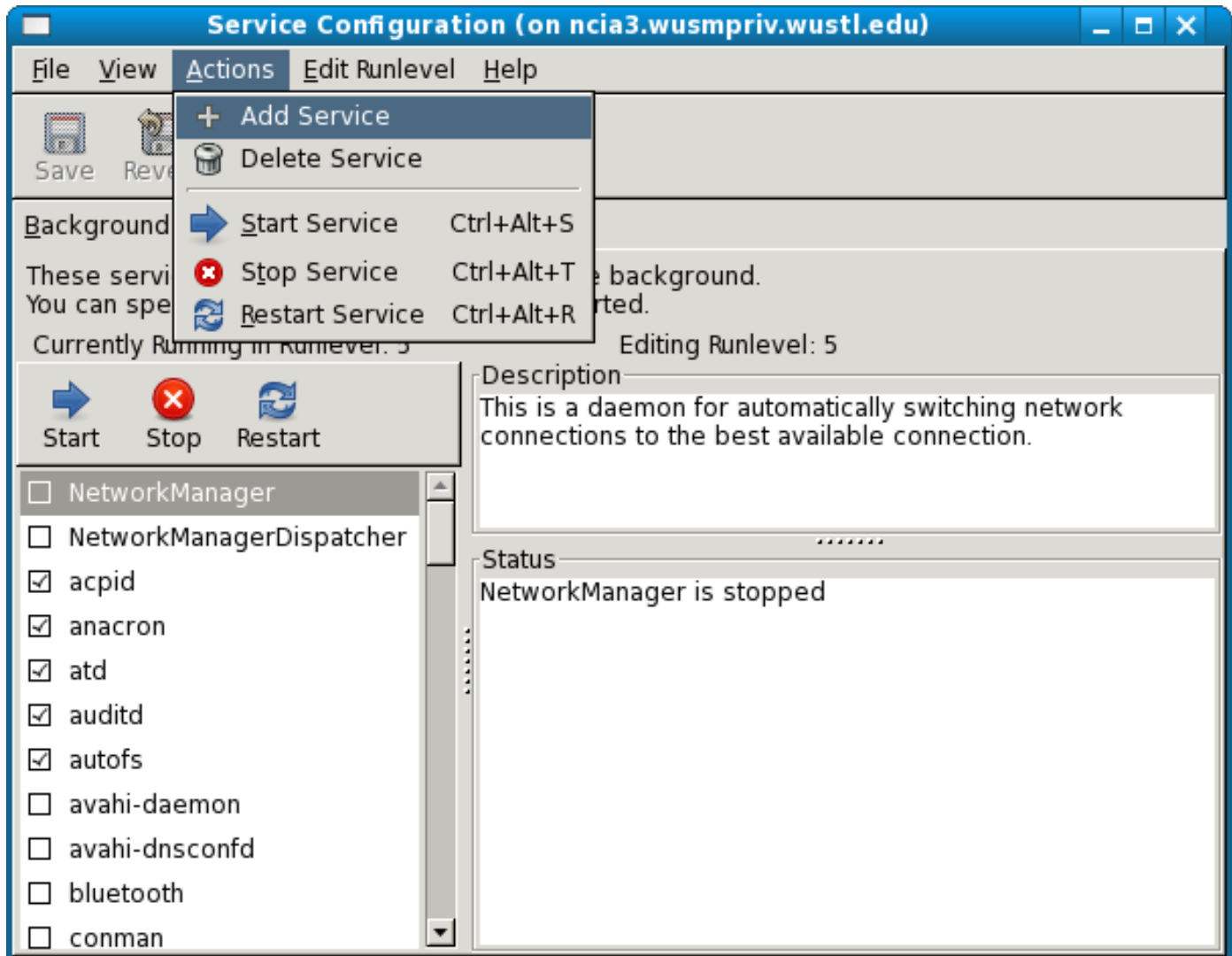
```
[root@ncia3 native]# pwd
/usr/local/apache-tomcat-5.5.25/bin/jsvc-src/native
[root@ncia3 native]# cp Tomcat5.sh /etc/init.d/tomcat
[root@ncia3 native]# cd /etc/init.d
[root@ncia3 init.d]# chmod a+x tomcat
[root@ncia3 init.d]# chown root:root tomcat
```

Appendix B shows the modified start-up script `/etc/init.d/tomcat`. Make sure that tomcat can be started up manually:

```
[root@ncia3 init.d]# /etc/init.d/tomcat start
```

and that you can browse to <http://localhost:8080> to see the apache-tomcat test page. The tomcat application server is used for MIRC and needs to be running at <http://localhost:58081> instead of the default port of 8080. To change the port at which tomcat is listening, modify the tomcat server.xml file located in `/usr/local/apache-tomcat-5.5.25/conf`. After this modification, restart tomcat. It may be necessary to adjust the local firewall policy on the virtual server operating system. As root, use the

/usr/sbin/system-config-services program to add the tomcat startup script to the virtual machine boot sequence:

Figure 5 service configuration

Install Java Image I/O API 1.0

The `jai_imageio.jar` library is used by Tomcat (and MIRC). Download the CLASSPATH version – such as `jai_imageio-1_0_01-lib-linux-i586.tar.gz`. After expanding, put a copy in the tomcat instances `server/lib` directory:

```
[koppel@ncia3 ~]$ tar xvzf jai_imageio-1_0_01-lib-linux-i586.tar.gz
```

```
[koppel@ncia3 ~]$ su root
```

Password:

```
[root@ncia3 koppel]# cd jai_imageio-1_0_01 /lib
```

```
[root@ncia3 lib]# cp -p jai_imageio.jar /usr/local/apache-tomcat-5.5.25/server/lib
```

```
[root@ncia3 lib]# chown tomcat:ncia /usr/local/apache-tomcat-5.5.25/server/lib/jai_imageio.jar
```

Install MySQL

The mysql version that was bundled with the operating system lagged the version available from www.mysql.com, so the mysql server and client rpms were downloaded and installed. On the ncia3 system, it was also necessary to install *perl-DBI* as a prerequisite using the package manager. As root:

```
[root@ncia3 koppel]# rpm -Uvh MySQL-server-community-5.0.51a-0.rhel5.x86_64.rpm
```

```
[root@ncia3 koppel]# rpm -Uvh MySQL-client-community-5.0.51a-0.rhel5.x86_64.rpm
```

Installing the mysql server rpm also creates a login shell account for a user named mysql. Modify or create if necessary a configuration file:

```
[koppel@ncia3 ~]$ cat /etc/my.cnf
```

```
[mysqld]
```

```
lower_case_table_names=1
```

```
max_allowed_packet=64M
```

```
[mysqldump]
```

```
max_allowed_packet=64M
```

```
[mysql]
```

```
max_allowed_packet=64M
```

```
[koppel@ncia3 ~]$
```

MySQL Startup

To startup the mysql database server at system boot, use the example script that comes with the mysql download - `/usr/share/mysql/mysql.server`. No modifications were needed to the startup script. For example:

```
[root@ncia3 koppel]# cp -p /usr/share/mysql/mysql.server /etc/init.d/mysql
```

```
[root@ncia3 koppel]# chown root:root /etc/init.d/mysql
```

```
[root@ncia3 koppel]# chmod 755 /etc/init.d/mysql
```

Make sure that the mysql server is listed in the `/usr/sbin/system-config-services` program. Add the mysql startup script to the virtual machine boot sequence in exactly the same way as was done for the tomcat startup and that mysql can also be started up manually:

```
[root@ncia3 init.d]# /etc/init.d/mysql start
```

Create Database Accounts

The ncia installation scripts require two database users which should be created manually:

nciaadmin

nciauser

With mysql server now running, login to the server as root:

```
[koppel@ncia3 ~]$ mysql -u root
```

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 118

Server version: 5.0.51a-community MySQL Community Edition (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

```
mysql> grant all privileges on *.* to 'nciaadmin'@'localhost' identified by 'password' with grant option;
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> grant all privileges on *.* to 'nciauser'@'localhost' identified by 'password' with grant option;
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> quit
```

Bye

Install MySQL Connector/J

The MySQL Connector/J provides connectivity for client applications developed in the Java programming language via a JDBC driver. To install Connector/J from a binary distribution, download `mysql-connector-java-5.0.8.tar.gz` from www.mysql.com and extract to `/usr/local`. For example:

```
[koppel@ncia3 ~]$ su root
```

Password:

```
[root@ncia3 koppel]# cd /usr/local
```

```
[root@ncia3 local]# tar xvfz mysql-connector-java-5.0.8.tar.gz
```

The `$CLASSPATH` environmental variable that was setup earlier in the `.bashrc` example (Appendix A) is:

```
[root@ncia3 koppel]# echo $CLASSPATH
```

```
/usr/local/mysql-connector-java-5.0.8/mysql-connector-java-5.0.8-
```

```
bin.jar:/usr/java/junit4.4/junit-4.4.jar:/usr/java/junit4.4
```

Verify that \$CLASSPATH is correct:

```
[root@ncia3 koppel]# file /usr/local/mysql-connector-java-5.0.8/mysql-connector-java-5.0.8-bin.jar
/usr/local/mysql-connector-java-5.0.8/mysql-connector-java-5.0.8-bin.jar: Zip archive data, at least
v1.0 to extract
```

Install CSMUPT and NCIA

Download the two zip files and the installation guide from the NCICB GForge Life Sciences Distribution Project at https://gforge.nci.nih.gov/frs/group_id=450 :

```
installation_guide_lsd_r1_0_0_ga.pdf
upt_distribution_3_2_0_lsd_r1_0_0_ga.zip
ncia_distribution_3_0_0_lsd_r1_0_0_ga.zip
```

Since installation of csmupt and ncia is covered extensively in other NCI documents, these installations will not be detailed here. Each of the zip files needs to be expanded – for example (as root):

```
[root@ncia3 local]# pwd
/usr/local
[root@ncia3 local]# mkdir upt
[root@ncia3 local]# mkdir ncia
[root@ncia3 local]# cd /usr/local/upt
[root@ncia3 upt]# unzip /home/koppel/upt_distribution_3_2_0_lsd_r1_0_0_ga.zip
[root@ncia3 local]# cd /usr/local/ncia
[root@ncia3 ncia]# unzip /home/koppel/ncia_distribution_3_0_0_lsd_r1_0_0_ga.zip
```

The csmupt tool and ncia application require separate JBoss instances. These instances will be created as a result of running the two install scripts – so it is not necessary to install Jboss manually. To specify the location of the csmupt Jboss server, modify the upt-install.properties file in /usr/local/upt so that *upt.home=/usr/local/apps/upt*. Similarly, to specify the location of the ncia jboss server, modify the ncia-install.properties file in /usr/local/ncia so that *ncia.home=/usr/local/apps/ncia*. After installation, the two Jboss servers are in /usr/local/apps:

```
[root@ncia3 apps]# pwd
/usr/local/apps
```

```
[root@ncia3 apps]# ls *
```

```
ncia:
```

```
jboss-4.0.4.GA
```

```
upt:
```

```
jboss-4.0.4.GA
```

```
[root@ncia3 apps]#
```

The two JBoss servers listen on different ports:

```
http://localhost:45210/ncia
```

```
http://localhost:46210/upt
```

Change the ownership and mode of the apps directory recursively:

```
[root@ncia3 init.d]# cd /usr/local
```

```
[root@ncia3 local]# chown -R jboss:ncia apps
```

```
[root@ncia3 local]# chmod -R 775 apps
```

Jboss startup

A script that starts up both instances of Jboss is shown in Appendix C - /etc/init.d/jboss:

```
[root@ncia3 init.d]# chmod a+x jboss
```

```
[root@ncia3 init.d]# chown root:root jboss
```

This script also needs to be listed in the `/usr/sbin/system-config-services` program. Add the jboss startup script to the virtual machine boot sequence in exactly the same way as was done for the tomcat and mysql startup scripts. Make sure that the JBoss servers can also be started up manually:

```
[root@ncia3 init.d]# /etc/init.d/jboss start
```

NFS Mounting tomcat webapps and downloaded data

The hard drive contained in the ncia3 virtual machine is approximately 20 GB – which will be too small for some applications involving large image data sets. For example, the FDA Lung Phantom data that is currently being uploaded to ncia3 is over 300 GB in size. The procedure below is one way to

move the tomcat webapps directory (which contains the MIRC documents and associated images) and downloaded data from the “shopping cart” to a larger disk storage array. In our case, the large disk storage array is nfs mounted to /mnt/erlbluearc – which has terabytes of capacity.

As root, stop jboss and tomcat processes:

```
[root@ncia3 koppel]# /etc/init.d/jboss stop
```

Stopping jboss service.

```
[root@ncia3 koppel]# /etc/init.d/tomcat stop
```

Use “ps” to verify that these processes are not running:

```
[root@ncia3 koppel]# ps -aef | grep jboss
```

```
root  5109 4942 0 10:16 pts/1  00:00:00 grep jboss
```

```
[root@ncia3 koppel]# ps -aef | grep tomcat
```

```
root  5111 4942 0 10:16 pts/1  00:00:00 grep tomcat
```

```
[root@ncia3 koppel]# ps -aef | grep java
```

```
root  5113 4942 0 10:16 pts/1  00:00:00 grep java
```

Copy the tomcat webapps directory to the nfs mount:

```
[root@ncia3 koppel]# cd /usr/local/apache-tomcat-5.5.25/
```

```
[root@ncia3 apache-tomcat-5.5.25]# cp -rp webapps /mnt/erlbluearc/systems/ncia3/data
```

Rename the webapps directory to MIRCwebapps (this is only for convenience):

```
[root@ncia3 data]# pwd
```

```
/mnt/erlbluearc/systems/ncia3/data
```

```
[root@ncia3 data]# mv webapps MIRCwebapps
```

On the local virtual machine hard drive, remove the webapps directory:

```
[root@ncia3 data]# cd /usr/local/apache-tomcat-5.5.25/
```

```
[root@ncia3 apache-tomcat-5.5.25]# rm -rf webapps/
```

Create a soft link to the nfs mount:

```
[root@ncia3 apache-tomcat-5.5.25]# ln -s
```

```
/mnt/erlbluearc/systems/ncia3/data/MIRCwebapps /usr/local/apache-tomcat-5.5.25/webapps
```

```
[root@ncia3 data]# pwd
/data
[root@ncia3 data]# cp -rp ncia_data /mnt/erlbluearc/systems/ncia3/data/
[root@ncia3 data]# rm -rf ncia_data/
[root@ncia3 data]# ln -s /mnt/erlbluearc/systems/ncia3/data/ncia_data /data/ncia_data
```

Appendix A .bashrc for building ncia3

This file should be modified according to the version numbers of jdk, junit, mysql-connector that are downloaded.

```
[koppel@ncia3 ~]$ cat .bashrc
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# User specific aliases and functions
#
export ANT_HOME=/usr/local/ant
export JAVA_HOME=/usr/java/jdk1.5.0_14
export JAVACMD=$JAVA_HOME/bin/java
export CATALINA_HOME=/usr/local/apache-tomcat-5.5.25
export JUNIT_HOME=/usr/java/junit4.4
export CLASSPATH=${JUNIT_HOME}/junit-4.4.jar:${JUNIT_HOME}
export PATH=${JAVA_HOME}/bin:${ANT_HOME}/bin:${PATH}
export CLASSPATH=/usr/local/mysql-connector-java-5.0.8/mysql-connector-java-5.0.8-bin.jar:${CLASSPATH}
[koppel@ncia3 ~]$
```

Appendix B tomcat startup script

Note- text in bold shows the modifications to the original.

```
[root@ncia3 koppel]# cd /etc/init.d
[root@ncia3 init.d]# cat tomcat
```

```

#!/bin/sh
#
# chkconfig: 345 98 02
# description: starts and stops the Tomcat server
#
#
#####
#
# Copyright 2004 The Apache Software Foundation.
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#   http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#####
#
# Small shell script to show how to start/stop Tomcat using jsvc
# If you want to have Tomcat running on port 80 please modify the server.xml
# file:
#
# <!-- Define a non-SSL HTTP/1.1 Connector on port 80 -->
# <Connector className="org.apache.catalina.connector.http.HttpConnector"
#     port="80" minProcessors="5" maxProcessors="75"
#     enableLookups="true" redirectPort="8443"
#     acceptCount="10" debug="0" connectionTimeout="60000"/>
#

```

```
# That is for Tomcat-5.0.x (Apache Tomcat/5.0)
#
# Adapt the following lines to your configuration
JAVA_HOME=/usr/java/jdk1.5.0_14
CATALINA_HOME=/usr/local/apache-tomcat-5.5.25
DAEMON_HOME=/usr/local/apache-tomcat-5.5.25/bin
TOMCAT_USER=tomcat

# for multi instances adapt those lines.
TMP_DIR=/var/tmp
PID_FILE=/var/run/jsvc.pid
CATALINA_BASE=/usr/local/apache-tomcat-5.5.25

CATALINA_OPTS="-Xms256m -Xmx1536m"
CLASSPATH=\
$JAVA_HOME/lib/tools.jar:\
$CATALINA_HOME/bin/commons-daemon.jar:\
$CATALINA_HOME/bin/bootstrap.jar

case "$1" in
start)
#
# Start Tomcat
#
$DAEMON_HOME/jsvc \
-user $TOMCAT_USER \
-home $JAVA_HOME \
-Dcatalina.home=$CATALINA_HOME \
-Dcatalina.base=$CATALINA_BASE \
-Djava.io.tmpdir=$TMP_DIR \
-wait 10 \
-pidfile $PID_FILE \
-outfile $CATALINA_HOME/logs/catalina.out \
```

```

-errfile '&1' \
$CATALINA_OPTS \
-cp $CLASSPATH \
org.apache.catalina.startup.Bootstrap
#
# To get a verbose JVM
# -verbose \
# To get a debug of jsvc.
#-debug \
exit $?
;;

stop)
#
# Stop Tomcat
#
$DAEMON_HOME/jsvc \
-stop \
-pidfile $PID_FILE \
org.apache.catalina.startup.Bootstrap
exit $?
;;

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

```

Appendix C jboss startup script

```

[root@ncia3 koppel]# cd /etc/init.d
[root@ncia3 init.d]# cat jboss
#!/bin/bash
#

```

```

# chkconfig: 345 85 15
# description: starts and stops jboss server
#
NCIA_HOME=/usr/local/apps/ncia/jboss-4.0.4.GA
NCIA_BIN=${NCIA_HOME}/bin
UPT_HOME=/usr/local/apps/upt/jboss-4.0.4.GA
UPT_BIN=${UPT_HOME}/bin

case "$1" in
  start)
    echo "Starting jboss service."
    su -l jboss -c "${NCIA_BIN}/run.sh > /dev/null 2> /dev/null &"
    su -l jboss -c "${UPT_BIN}/run.sh > /dev/null 2> /dev/null &"
    ;;
  stop)
    echo "Stopping jboss service."
    su -l jboss -c "${NCIA_BIN}/shutdown.sh -S > /dev/null 2> /dev/null &"
    su -l jboss -c "${UPT_BIN}/shutdown.sh -S > /dev/null 2> /dev/null &"
    ;;
  *)
    echo "Usage: $0 {start|stop}"
    exit 1
    ;;
esac

exit 0

```

Appendix D resetting ncia database

Currently, there is no easy method to delete an individual set of images from nica3. The following procedure was used in clearing out test images from nica3 before we began uploading the real, FDA Lung Phantom data. The database tables that have csmupt information are not modified – so that existing users, privileges, and their logins should be maintained. It is worth keeping a second virtual nica3 system around for testing small batches of submissions for each new collection to be uploaded based on a Field Center anonymizer file. By testing submissions on a second virtual nica3 system, deleting and or managing the images on the production archive will not be necessary.

Backup the ncia database:

```
[koppel@ncia3 ~]$ /usr/bin/mysqldump --single-transaction --user=root --opt --databases ncia
> /tmp/ncia.sql
```

Reset ncia database by clearing out database tables and associated operating system files. The mysql return output is not shown for the delete commands listed below:

```
[root@ncia3 koppel]# /etc/init.d/jboss stop
[root@ncia3 koppel]# /etc/init.d/tomcat stop
[root@ncia3 koppel]# mysql -u root
```

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 3694

Server version: 5.0.51a-community MySQL Community Edition (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

```
mysql> use ncia;
```

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

```
mysql> show full tables;
```

```
+-----+-----+
| Tables_in_ncia          | Table_type |
+-----+-----+
| all_logins              | VIEW      |
| anatomic_site           | VIEW      |
| annotation              | BASE TABLE |
| clinical_trial          | BASE TABLE |
| clinical_trial_protocol | BASE TABLE |
| clinical_trial_sponsor  | BASE TABLE |
| clinical_trial_subject  | BASE TABLE |
| convolution_kernel      | VIEW      |
```

csm_application	BASE TABLE
csm_group	BASE TABLE
csm_pg_pe	BASE TABLE
csm_privilege	BASE TABLE
csm_protection_element	BASE TABLE
csm_protection_group	BASE TABLE
csm_role	BASE TABLE
csm_role_privilege	BASE TABLE
csm_user	BASE TABLE
csm_user_group	BASE TABLE
csm_user_group_role_pg	BASE TABLE
csm_user_pe	BASE TABLE
ct_image	BASE TABLE
curation_data	BASE TABLE
curation_status	BASE TABLE
dicom_tag_info	BASE TABLE
doc_location	BASE TABLE
download_history	BASE TABLE
general_equipment	BASE TABLE
general_image	BASE TABLE
general_series	BASE TABLE
grid_node	BASE TABLE
hibernate_unique_key	BASE TABLE
image	VIEW
image_markup	BASE TABLE
img_cnt_per_series	VIEW
img_cnt_per_series_uid	VIEW
login_history	BASE TABLE
manufacturer	VIEW
manufacturer_model_name	VIEW
manufacturer_model_software	VIEW
modality	VIEW
number_month	VIEW

numeric_curation_data	VIEW
patient	BASE TABLE
project	VIEW
qa_status_history	BASE TABLE
qa_status_summary	VIEW
qa_status_summary_tmp	VIEW
quarantine	VIEW
query_history	BASE TABLE
query_history_attribute	BASE TABLE
saved_query	BASE TABLE
saved_query_attribute	BASE TABLE
saved_query_last_exec	VIEW
series_total_ann_file_size	VIEW
software_versions	VIEW
study	BASE TABLE
study_series_number	VIEW
submission_count_by_month	VIEW
submitted_annotation	VIEW
submitted_images	VIEW
submitted_non_visible	VIEW
submitted_visible	VIEW
tmp_table_1	BASE TABLE
tmp_table_2	BASE TABLE
tmp_table_3	BASE TABLE
trial_data_provenance	BASE TABLE
trial_site	BASE TABLE

+-----+-----+

67 rows in set (0.03 sec)

mysql> delete from all_logins;

mysql> delete from clinical_trial;

mysql> delete from ct_image;

mysql> delete from qa_status_history;

```
mysql> delete from general_image;
mysql> delete from general_series;
mysql> delete from download_history;
mysql> delete from login_history;
mysql> delete from study;
mysql> delete from patient;
mysql> delete from query_history_attribute;
mysql> delete from query_history;
mysql> delete from trial_data_provenance;
mysql> delete from trial_site;
mysql> exit
```

Bye

```
[root@ncia3 koppel]# cd /usr/local/apache-tomcat-5.5.25/webapps/NCIA3/documents
[root@ncia3 documents]# rm -rf 1.2*
[root@ncia3 documents]# rm -rf 1.3*
[root@ncia3 documents]# cd /usr/local/apache-tomcat-5.5.25/webapps/NCIA3/trial/quarantine
[root@ncia3 quarantine]# rm -rf *.qe
[root@ncia3 quarantine]# reboot
```

Appendix E migrating a Xen virtual machine

The *ncia3* virtual machine was initially created on a Dell PowerEdge 1950 with service tag *fwjm2f1*. It was then moved to another Dell PowerEdge 1950 with service tag *bwjm2f1*. Both 1950's have the same hardware configuration including memory, disk storage, and hardware virtualization enabled in the bios. Both 1950's also have Fedora 8 as the base operating system with Xen kernel. The *ncia3* virtual machine on *fwjm2f1* was shutdown first before migrating from physical server *fwjm2f1* to *bwjm2f1*:

```
[koppel@bwjm2f1 ~]$ su root
Password:
[root@bwjm2f1 koppel]# cd /var/lib/xen/images/
[root@bwjm2f1 images]# rsync -avz -e ssh root@fwjm2f1:/var/lib/xen/images/ncia3. img .
root@fwjm2f1's password:
receiving file list ... done

sent 20 bytes received 64 bytes 9.88 bytes/sec
```

```

total size is 20971520000 speedup is 249660952.38
[root@bwjm2f1 images]# ls
ncia3.img
[root@bwjm2f1 images]# cd /var/lib/xend
[root@bwjm2f1 xend]# rsync -avz -e ssh root@fwjm2f1:/var/lib/xend/domains .
root@fwjm2f1's password:
receiving file list ... done
domains/
domains/6d826abb-6c51-d94b-83a3-24c07760dd72/
domains/6d826abb-6c51-d94b-83a3-24c07760dd72/config.sxp

sent 54 bytes received 1179 bytes 224.18 bytes/sec
total size is 2841 speedup is 2.30

[root@bwjm2f1 xend]# /usr/sbin/xm create --config
/var/lib/xend/domains/6d826abb-6c51-d94b-83a3-24c07760dd72/config.sxp &
[1] 12215
[root@bwjm2f1 xend]# Started domain ncia3

[1]+ Done /usr/sbin/xm create --config
/var/lib/xend/domains/6d826abb-6c51-d94b-83a3-24c07760dd72/config.sxp
[root@bwjm2f1 xend]#

```

Running nmap shows an open port at 5900 which corresponds to the migrated virtual machine console. By using vncviewer (see below) it is easy to connect to the migrated console of ncia3 as in Fig. 2.

```
[root@bwjm2f1 xend]# nmap localhost
```

```

Starting Nmap 4.52 ( http://insecure.org ) at 2008-04-28 09:43 CDT
Interesting ports on localhost.localdomain (127.0.0.1):
Not shown: 1709 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
25/tcp    open  smtp

```

```
111/tcp open rpcbind
```

```
631/tcp open ipp
```

```
5900/tcp open vnc
```

```
Nmap done: 1 IP address (1 host up) scanned in 0.158 seconds
```

```
[root@bwjm2f1 xend]# vncviewer localhost:5900 &
```

```
[koppel@bwjm2f1 ~]$
```