Network Analysis Tools
Web server installation

Sylvain Brohée
sbrohee@ulb.ac.be

Karoline Faust
kfaust@ulb.ac.be

Jacques van Helden
jvhelden@ulb.ac.be

Laboratoire de Bioinformatique des Génomes et des Réseaux (BiGRe)
Laboratory of Genome and Network Biology
Université Libre de Bruxelles, Belgium
http://www.bigre.ulb.ac.be/

November 30, 2009

Contents

Description

This document describes the installation procedure for the web server of the **Network Analysis Tools** (*NeAT*).

It assumes that you already installed the perl scripts, java tools and the genomes, as described in the *RSAT* installation guide and that it is working properly on the command line and as webserver. To this, please refer to the *RSAT* webserver install guide.

Chapter 1

Web server installation

1.1 Installing a local web server

As the Regulatory Sequence Analysis Tools, *NeAT* includes a web server, which offers a user-friendly interface for biologists. The main server is available for academic users at *http://rsat.ulb.ac.be/neat/*. A few additional mirrors have been installed in different countries.

1.1.1 Web server pages

The web pages are located in the directory *rsa-tools/public_html*. This directory contains both the HTML help pages, and the PHP and CGI scripts.

1.1.2 Apache modules

The *NeAT* interface mainly relies on PHP (and CGI only for the roc-stats tool). These modules should be installed on the web server, and activated in the Apache configuration files. The installation and configuration of CGI is described in manual of the web server of *RSAT*.

To perform the following steps, you might dipose of the administrator permissions.

PHP module for Mac OSX

If your server is running under Mac OSX, you need to install a recent version (at least v5) of the php module, which can be found at the following site. *http://www.entropy.ch/software/macosx/php/*

PHP module for LINUX

Generally, PHP5 is included with the Linux distribution or can easyly be installed using the installer (YAST, YUM, etc). Take care that the PHP5 Module for Apache 2.0 (apache2-mod_php5) is installed.

PHP5 can also be installed manually from the PHP website (*http://www.php.net/downloads.php*).

Modification of php.ini

In order for the server to work with *NeAT*, you have to edit the file *php.ini* which
is the main configuration file of PHP. Depending on your PHP installation, this file
might be in different directory. In computers running Suse 10.3, this file is in the
/etc/php5/apache2/ directory. In this file, you must modify the following fields accord-
ingly.

```
soap.wsdl_cache_enabled=0
max_execution_time = 2400
max_input_time = 6000 ;
memory_limit = 600M
error_reporting  =  E_ALL & ˜E_NOTICE
post_max_size = 300M
upload_max_filesize = 200M
default_socket_timeout = 3000
upload_tmp_dir = "/tmp/php/"
```

Pay attention to the location of php5 extension libraries. Indicate the correct direc-
tory, e.g.

```
extension_dir=/usr/lib/php5/extensions
```

You must change permissions so that the directory */tmp/php/* for temporary upload
is writable by eveverybody. To this please type

```
mkdir /tmp/php
chmod 777 php
```

1.1.3   Tomcat

A part of *NeAT* is using axis web services, JSP and Java servlet pages and needs Tomcat
or an equivalent servlet engine to run. Tomcat can be easily installed on SUSE with
yast and is usually located in */usr/share/tomcat* after installation. On MacOS, it is
already installed by default. Make sure to install at least Tomcat version 5. We noticed
however that Tomcat version 6 is more stable. We will refer from now on to the Tomcat
root directory as $CATALINA_HOME.

Configuration of Tomcat

If you would like to use the Tomcat manager, make sure to configure the file tomcat-
users.xml located in $CATALINA_HOME/conf.
    Add a manager role with a special login and password, e.g.

```
<tomcat-users>
  <role rolename="tomcat"/>
  <role rolename="manager"/>
  <user username="tomcat" password="tomcat" roles="tomcat"/>
  <user username="metheadmin" password="mypassword" roles="manager"/>
</tomcat-users>
```

Make sure that tomcat-users.xml cannot be read by anyone else than tomcat or root.

By default, Tomcat takes a lot of memory. You can reduce this amount by modifying the file tomcat.conf in $CATALINA_HOME/conf. Adjust the variable JAVA_OPTS as follows:
JAVA_OPTS="-Xmx800m"

You can start Tomcat on SUSE with
/etc/init.d/tomcat6 start
and stop it with
/etc/init.d/tomcat6 stop

More advice for the installation and configuration of Tomcat on unix systems can be found at: *http://linux-sxs.org/internet_serving/book1.html*

Optional: Mod_jk

The installation of Mod_jk is optional. If you do not want to install it, skip this section.

Mod_jk forwards requests for defined contexts from Apache to Tomcat. Users that have a strict firewall configuration blocking the Tomcat port will be able to access the web services through the Apache default port.

To install and configure mod_jk, you can follow the steps below:

1. Install the mod_jk module from *http://tomcat.apache.org/connectors-doc/* or via installation systems like yast.

2. In the Apache configuration folder (e.g. */etc/apache2/conf.d*), add two configuration files: workers.properties and tomcat.conf

3. Content of workers.properties

```
# Define 1 real worker using ajp13
  # this coud be al list in the format
  # worker.list=worker1, worker2, worker3, worker4
  worker.list=worker1
  # Set properties for worker1 (ajp13)
  worker.worker1.type=ajp13
  worker.worker1.host=localhost
  worker.worker1.port=8009
```

4. Content of tomcat.conf

```
# Update this path to match your modules location
LoadModule jk_module /usr/lib/apache2/mod_jk.so
# Where to find workers.properties
# Update this path to match your conf directory location
JkWorkersFile /etc/apache2/conf.d/workers.properties
# Where to put jk shared memory
# Update this path to match your local log directory
# JkShmFile /var/log/apache2/mod_jk.shm
# Where to put jk logs
```

```
# Update this path to match your logs directory location
JkLogFile /var/log/apache2/mod_jk.log
# Set the jk log level [debug/error/info]
JkLogLevel error
# Send everything for context to worker named worker1 (ajp13)
JKMount /be.ac.ulb.bigre.graphtools.server/* worker1
JKMount /metabolicpathfinding/* worker1
```

5. Reload Apache (on SUSE */etc/init.d/apache2* reload).

An alternative to mod_jk is mod_proxy, which requires less configuration.

Preparation of folders for Tomcat

In Linux, a tomcat user is created if Tomcat is installed via yast. Make sure that this user has read, write and execution rights for the following folders:
$RSAT/contrib/REA
$RSAT/contrib/kwalks/bin
$RSAT/public_html/data/KEGG
$RSAT/public_html/data/Stored_networks
$RSAT/java/lib

If any of these folders does not yet exist, create them.
For installation of REA and kwalks see the section on third-party programs in the RSAT install guide.

1.1.4   Java tools server

The graphtools server contains the java web services of NeAT.

Installation of Java tools server

The graphtools server is stored as a war file in *$RSAT/java/web*.

1. Open the RSAT configuration file *RSAT_config.props* located in *$RSAT*. Set the value of the parameter *neat_java_ws* to *web_services/GraphAlgorithms.wsdl*

2. If mod_jk or equivalent has not been installed, open the RSAT configuration file and set the parameter *tomcat_port* to the port on which Tomcat runs (by default 8080).

3. There are two ways to deploy a war file. If you do not use the Tomcat manager, make sure Tomcat can open the war file with the command:
chown tomcat:tomcat be.ac.ulb.bigre.graphtools.server.war Then place the war file *be.ac.ulb.bigre.graphtools.server.war* in $CATALINA_HOME/webapps and start Tomcat. If you use the Tomcat manager, open (*http://localhost:8080/manager/html* and go to option *Select WAR file to upload*. After having selected the war file to upload, click *Deploy*.

4. Go to *$CATALINA_HOME/webapps/be.ac.ulb.bigre.graphtools.server/WEB-INF*

5. Open *serverConfig.txt* and set the value of RSAT_ROOT to the value of $RSAT.

6. Run install.sh with the following commands:
   chmod 755 install.sh
   ./install.sh

You are done. You may read the section "Configuration remarks" for additional information. If you want to install the metabolic pathfinder, continue with section "Metabolic Pathfinder and Pathway extraction".

Configuration remarks

The directory *$RSAT/public_html/data/Stored_networks* allows to store graph files for longer time (since *$RSAT/public_html/tmp* is cleaned regularly).

Note that the Kegg network provider accesses the metabolic database to add attributes such as compound name or EC number to networks. In order to install the database, see the installation of the metabolic pathfinder. The name of this database, its IP address, its owner and password may be modified. These parameters may be set in *serverConfig.txt* located at *$CATALINA_HOME/webapps/be.ac.ulb.bigre.graphtools.server/WEB-INF*.

Update the KEGG network provider

The KEGG network provider places organism-specific KGML files in *$RSAT/public_html/data/KEGG* in a folder with the KEGG organism name (e.g. sce). You may place KGML folders for organisms yourself or you may let the program download required KGML files on the fly. In the latter case, make sure that the KGML version is set correctly. To change the KGML version displayed in the web interface,
go to *$CATALINA_HOME/webapps/be.ac.ulb.bigre.graphtools.server/WEB-INF*, open *serverConfig.txt* and modify the value of the parameter KGML_VERSION.

The Kegg network provider queries a metabolic database in order to annotate the network with certain compound and reaction attributes. To install and update the metabolic database, use the KeggLigandDataManager command line tool. For details, type: *java graphtools.parser.KeggLigandDataManager -h*

The KEGG network provider is also using the file rpairs.tab located in *$RSAT/data/KEGG* to convert reaction into RPAIR graphs. To update this file, use the KeggLigandDataManager command line tool.

Finally, the KEGG network provider displays a list of KEGG organisms. To obtain the organism list for the recent KEGG version, use the MetabolicGraphProvider command line tool. For details on this tool, type: *java graphtools.util.MetabolicGraphProvider -h* Place the updated list in the folder *$RSAT/data/KEGG* to replace the old list.

1.1.5   Metabolic Pathfinder and Pathway extraction

After successful installation of the Java web services, you may install metabolic pathfinder and Pathway extraction, which are clients of the pathfinder and pathwayinference web services respectively.

Requirements of metabolic pathfinder and pathway extraction

Metabolic pathfinder and pathway extraction have a number of additional requirements.

1. Dot
   Dot is needed to draw graphs and can be obtained freely from *http://www.graphviz.org/*.

2. Postgres

   (a) Installation
       Postgres (version 8.2 or later) is needed to store KEGG and MetaCyc data.
       Postgres can be obtained freely from *http://www.postgresql.org/*. For Ma-
       cOS, you can use Darwinports to install postgres. On SUSE, you may
       install it with yast (install postgresql and postgresql-server). Usually, a
       postgres user is created during installation of postgres.

   (b) Postgres configuration
       You may need to configure the postgres server. You can simply allow
       all users on your machine (but not from outside) to access all postgres
       databases. This can be achieved by modifying the pg_hba.conf file located
       in the postgres home directory. Paste the following in this file:

       ```
       host    all         all          127.0.0.1/32            trust
       # IPv6 local connections:
       host    all         all          ::1/128                 trust
       host    all         all          127.0.0.1               trust
       ```

       For more details on postgres configuration, check the postgres manual on
       the pg_hba.conf and pg_ident.conf files.

   (c) Start the server
       On SUSE, you may start the postgres server with */etc/init.d/postgresql start*.
       On MacOS, you may start the postgres server using a command similar to
       pg_ctl -D /usr/local/pgsql/data/ -l logfile start.

   (d) Obtain the data file
       Download the postgres backup file *metabolicdb_dump_day_month_year.backup*
       from the data section of the official NeAT web site.

   (e) Load data into postgres
       Load the *metabolicdb_dump_day_month_year.backup* file into postgres as
       follows:

       i. Start postgres by typing the following on command line:
          psql -U postgres

    ii. In postgres, do:
        create user metabolic with password 'metabolic';
        create database "metabolicdb" with owner "metabolic" encoding='UTF8';

    iii. Quit postgres and type the following command in one line:
        pg_restore -d metabolicdb metabolicdb_dump_day_month_year.backup
        -U postgres

Installation of metabolic pathfinder and pathway extraction

Either place the war file *metabolicpathfinding.war* located in *$RSAT/java/web* in $CATALINA_HOME/webapps and then start Tomcat or use the Tomcat manager to deploy the war file.

Configuration of metabolic pathfinder and pathway extraction

1. Go to the folder *$CATALINA_HOME/webapps/metabolicpathfinding/WEB-INF*.

2. If you are not the rsat user, set the RSAT environment variable, e.g.:
   export RSAT=/home/rsat/rsa-tools Run the configuration script with the following commands:
   chmod 755 configureWebxml.pl
   ./configureWebxml.pl

3. Reload the metabolic pathfinding web application using the Tomcat manager or on command line, restart Tomcat.

Update of metabolic pathfinder

The metabolic pathfinder contains by default data from KEGG version 49.0. In order to update it, you can follow the steps below:

1. Load KEGG LIGAND compound and reaction file into the metabolic database. The command line tool *KeggLigandDataManager* can be used for this. If present, delete previous KEGG data in the database. You may change the name, the location, owner and password of the metabolic database. In this case, change the default values in the web.xml file located at
   $CATALINA_HOME/webapps/metabolicpathfinding/WEB-INF.

2. Generate the preloaded networks with the *KeggLigandDataManager* command line tool. Place those networks in *$RSAT/data/Stored_networks* replacing the old ones.

3. In point 2, example networks have been generated as well.
   Copy them to *$CATALINA_HOME/webapps/metabolicpathfinding/networks*, replacing the old ones.

4. Set the parameter *keggVersion* in the web.xml file to the updated KEGG version.

5. Reload the metabolicpathfinding web application in tomcat.

Update of pathway extraction

This section only describes how to update MetaCyc data. See the section on the update of metabolic pathfinder for the KEGG data.

1. Load the MetaCyc OWL file into the metabolic database. The command line tool *MetabolicXMLFilesParser* can be used for this. If present, delete previous MetaCyc data in the database. You may change the name, the location, owner and password of the metabolic database. In this case, change the default values in the web.xml file located at
   $CATALINA_HOME/webapps/metabolicpathfinding/WEB-INF.

2. Generate the preloaded networks with the *MetabolicGraphProvider* command line tool. Place those networks in *$RSAT/data/Stored_networks* replacing the old ones.

3. In point 2, an example network has been generated as well.
   Copy it to *$CATALINA_HOME/webapps/metabolicpathfinding/networks*, replacing the old one.

4. Set the parameter *metacycVersion* in the web.xml file to the updated MetaCyc version.

5. Reload the metabolicpathfinding web application in tomcat.

1.1.6   Web services

Edit the WSDL file

The web interface consists in web services that are called by the PHP web pages. Your computer must thus act as web service server.

   First, edit the file *RSATWS.wsdl* located in the *$RSAT/public_html/web_services/* directory. At the very end of the file, the line

```
<soap:address location="http://rsat.bigre.ulb.ac.be/rsat/web_services/RSATWS.cgi"/>
```

   must be replaced by

```
<soap:address location="url_of_the_cgi_file_on_your_server"/>
```

   The URL can be *http://127.0.0.1/rsa-tools/web_services/RSATWS.cgi*.

Edit the *NeAT* config file

Edit the *RSAT_config.props* present in the main RSAT directory and edit the following fields so that they correspond to your local configuration.

```
neat_supported=1
neat_ws=web link to the WSDL file on your computer
(e.g. http://127.0.0.1/rsa-tools/web_services/RSATWS.wsdl)
```

Change permissions of the temporary files and log files directories

The directories *$RSAT/public_html/logs/* and *$RSAT/public_html/tmp/* must be writable. So, change the permissions

```
chmod 777 \$RSAT/public\_html/logs/
chmod 777 \$RSAT/public\_html/tmp/
```