Regulatory Sequence Analysis Tools
Installation guide

Jacques van Helden
jvanheld@scmbb.ulb.ac.be
http://www.scmbb.ulb.ac.be/~jvanheld/
Service de Conformation des Macromolécules Biologiques et de Bioinformatique,
Université Libre de Bruxelles,
Campus Plaine, CP 263, Boulevard du Triomphe, B-1050 Bruxelles, Belgium.
Tel: +32 2 650 2013 - Fax: +32 2 650 5425

February 22, 2006

Contents

1 Description

This documents describes the installation procedure for the software package **Regulatory Sequence Analysis Tools** (***RSAT***).

2 Requirements

2.1 Operating system

RSATis a unix-based package. It has been installed successfully on the following operating systems.

1. Linux
2. Mac OSX
3. Sun Solaris
4. Dec Alpha
5. cygwin (under MS Windows 98) (except for the graphical librairies, because I did not find a cygwin version of GD.pm)

RSATis not compatible with any version of Microsoft Windows and I have no intention to make it compatible in a foreseeable future. Since most programs are written in perl, part of them might run under windows, but some others will certainly not, because they include calls to unix system commands.

2.2 Perl language

The programs in ***RSAT***are written in perl. Version 5.1 or later is recommended.

2.3 Perl modules

Some perl modules are required for the graphical tools of ***RSAT***, and for some other specific programs.

GD.pm Interface to Gd Graphics Library. Used by ***XYgraph*** and ***feature-map***.

PostScript::Simple Produce PostScript files from Perl. Used by ***feature-map***.

bioperl ***RSAT***was developed independently of the bioperl project, but for some recent programs, I used bioperl classes (e.g. for reading and exporting taxonomic trees). It is thus useful to install a recent version of bioperl to take benefit of the full functionalities of ***RSAT***.

These modules can be found in the Comprehensive Perl Archive Network (*http://www.cpan.org/*). For ***bioperl***, we recommend to use the CVS distribution (*http://cvs.open-bio.org/*), which includes more recent updates.

3 Installation

For the time being, ***RSAT***is distributed as a compressed archive. In a near future, we will also distribute it via a CVS server, which will greatly facilitate the updates.

RSATcan be distributed either as a compressed archive, or via the CVS server. The CVS distribution greatly facilitates updates.

Note The CVS distribution will soon be available for external users, but we still need to configure the CVS server to accept a guest login. For the time being, the CVS distribution is still restricted to the people from the lab. Inbetween, the only distribution mode for external users is the compressed archive.

3.1 Installation from the CVS repository

Before being able to retrieve ***RSAT***from the CVS repository, you need an account on our server. For this, please contact Jacques van Helden (*jvanheld@scmbb.ulb.ac.be*).

3.1.1 First installation

The following command should be used the first time you retrieve the tools from the server:

```
cvs -d mylogin@cvs.scmbb.ulb.ac.be:/cvs co rsa-tools
```

This will create a directory *rsa-tools* on your computer, and store the programs in it. Note that at this stage the programs are not yet functional, because you still need to install genomes, which are not included in the CVS distribution.

3.1.2 Updates

Once the tools have been retrieved, you can obtain updates very easily. For this, you need to change your directory to the rsa-tools directory, and use the `cvs` command in the following way.

```
cd rsa-tools
cvs update .
```

3.2 Installation from a compressed archive

Uncompress the archive containing the programs. The archive is distributed in `zip` or `tar` format.

The `.zip` files can be uncompressed with the command ***unzip***.

```
unzip rsa-tools_yyyymmdd.zip
```

where yyyymmdd stands for the version number (delivery date).

If the ***unzip*** command is not supported on your system, you can uncompress the `.tar.gz` archive with the commands ***gunzip*** and ***tar***, which are part of the default unix installation.

```
gunzip rsa-tools_yyyymmdd.tar.gz
tar -xpf rsa-tools_yyyymmdd.tar
```

4 Adding *RSAT* to your path

1. Create an environment variable named RSAT and containing the path of rsa-tools.

   The precise command depends on your shell. To know you shell, you can type

   ```
   echo $SHELL
   ```

   Now, if we assume that ***RSAT*** have been installed in the directory

   ```
   /home/myaccount/rsa-tools
   ```

   you should type the following command.

   If your shell is bash:

   ```
   export RSAT=/home/myaccount/rsa-tools
   ```

   If your shell is csh or tcsh:

   ```
   setenv RSAT /home/myaccount/rsa-tools
   ```

2. Add the path of the ***RSAT*** perl scripts and binaries to your path.

   If your shell is bash:

   ```
   export PATH=${PATH}:${RSAT}/bin
   export PATH=${PATH}:${RSAT}/perl-scripts
   ```

   If your shell is csh or tcsh:

   ```
   set path=($path $RSAT/bin)
   set path=($path $RSAT/perl-scripts)
   rehash
   ```

   (the `rehash` command updates the list of executable programs)

If you are using a different shell than bash, csh or tcsh, the specification of environment variables is slightly different. In case of doubt, ask your system administrator how to configure your environment variables and your path.

The specification of the environment variables and paths are required each time you want to use ***RSAT***. You can add these specification to your personal profile. This file is normally found at the root of your personal account, in the file *.bashrc* if your shell is bash, or *.cshrc* if your shell is csh or tcsh. If you don't know how to proceed, ask your system administrator.

5 Initializing the directories

In addition to the programs, the installation of rsa-tools requires the creation of a few directories for storing data, access logs (for the web server), and temporary files.

The distribution includes a series of make scripts which will facilitate this step. You just need go to the rsa-tools directory, and start the appropriate make file.

```
cd rsa-tools
make -f makefiles/init_RSAT.mk init
```

6 Configuring *RSAT* for utilization on the command line

The ***RSAT*** distribution comes with a template configuration file named *RSA.config.default* and located in the *rsa-tools* directory.

Copy this file to create your own config file *RSA.config*.

```
cp RSA.config.default RSA.config
```

In principle, this default configuration file is sufficient to run the tools on the command-line.

You only need to edit it if you want to install a web server of the tools, or if you want to specify custom settings (for example the installation of additional genomes on a separate hard drive).

7 Downloading genomes

RSAT includes a series of tools to install and maintain the latest version of genomes.

7.1 Original data sources

Genomes supported on ***RSAT*** were obtained from various sources.

Genomes can be installed either from the ***RSAT*** web site, or from their original sources.

- NCBI/Genbank (*ftp://ftp.ncbi.nih.gov/genomes/*)
- ENSEMBL (*http://www.ensembl.org/*)
- The EBI genome directory (*ftp://ftp.ebi.ac.uk/pub/databases/genomes/Eukaryota/*)

Other genomes can also be found on the web site of a diversity of genome-sequencing centers.

7.2 Requirement : wget

The download of genomes relies on the application ***wget***, which is part of linux distribution. ***wget*** is a "web aspirator", which allows to downlaod whole directories from ftp and http sites. You can check if the program is installed on your machine.

```
wget -help
```

This command should return the help pages for ***wget***. If you obtain an error message ("command not found"), you need to ask your system administrator to install it.

7.3 Importing organisms from the *RSAT*main server

The simplest way to install organisms on our ***RSAT***site is to download the RSAT-formatted files from the web server. For this, you can use a web aspirator (for example the program ***wget***).

Beware, the full installation (including Mammals) requires a large disk space (several dozens of Gb). You should better start installting a small genome and test it before processing to the full installation. We illustrate the approach with one of the smallest sequenced genome: *Mycoplasma genitalium*.

7.3.1 Importing a single organism

The makefile script *makefilesinit_RSAT.mk* includes a target to install and configure a single organism on your RSAT site.

```
cd \$RSAT

# Download a single genome from the RSAT web server.
# This requires the program wget.
make -f makefiles/init_RSAT.mk download_one_genome

# Declare the newly downlaoded genome as a supported organism
make -f makefiles/init_RSAT.mk configure_one_genome
```

You can now check if the configuration file has been correctly updated by typing the command.

```
supported-organisms
```

In principle, the following information should be displayed on your terminal.

```
Saccharomyces_cerevisiae   Saccharomyces cerevisiae
```

7.3.2 Importing another organism

You can now proceed exactly in th same way to install any organism of your choice. For example, if you want to install Escherichia coli K12? you can run the ffollowing commands.

```
cd \$RSAT

# Download a single genome from the RSAT web server.
# This requires the program wget.
make -f makefiles/init_RSAT.mk download_one_genome ORG=Escherichia_coli_K12

# Declare the newly downlaoded genome as a supported organism
make -f makefiles/init_RSAT.mk configure_one_genome ORG=Escherichia_coli_K12

## Check that the new genome has bee added to the list of supported organisms
supported-organisms
```

8 Testing the command-line tools

8.1 Testing the access to perl scripts

From now on, you should be able to use the perl scripts from the command line. To test this, run:

```
random-seq -help
```

This should display the on-line help for the random sequence generator.

```
random-seq -l 200 -r 4 -a a:t 0.3 c:g 0.2
```

Should generate a random sequence.

8.2 Testing genome installation

We will now test if the genomes are correctly installed. You can obtain the list of supported organisms with the command:

```
supported-organisms
```

If this command returns no result, it means that genomes were either not installed, or not correctly configured. In such a case, check the directories in the *data/genomes* directory, and check that the file *data/supported_organisms.pl*.

Once you can obtain the list of installed organisms, try to retrieve some upstream sequences. You can first read the list of options for the ***retrieve-seq*** program.

```
retrieve-seq -help
```

Select an organism (say *Saccharomyces cerevisiae*), and retrieve all the start codons with the following options :

```
retrieve-seq -org Saccharomyces_cerevisiae \
        -type upstream -from 0 -to +2 -all \
        -format wc -nocomment
```

This should return a set of 3 bp sequences, mostly ATG (in the case of *Saccharomyces cerevisiae* at least)

8.3 Testing the graphical scripts

RSAT includes two graphical tools, ***feature-map*** and ***XYgraph***. These tools require the following perl modules:

GD.pm Interface to Gd Graphics Library.

PostScript::Simple Produce PostScript files from Perl.

To test if these modules are available on your machine, type.

```
feature-map -help
```

If the modules are available, you should see the help message of the program feature-map. If not, you will see an error message complaining about the missing librairies. In such a case, ask your system administrator to install the missing modules.

8.4 Installing third-party programs

The ***RSAT*** distribution only contains the programs developed by Jacques van Helden. A few additional programs, developed by third parties, can be integrated in the package. In order to obtain these programs, please download them from their original site.

In particular, we recommend to install the following programs.

vmatch : developed by Stefan Kurtz, is used used by the program ***purge-sequences***, for the detection of sequence repeats.

patser : developed by Jerry Hertz, is used for matrix-based pattern matching.

matrix-based pattern discovery : several other pattern discovery programs have been embedded in the ***RSAT*** program ***multiple-family-analysis***: ***consensus*** (Jerry Hertz), ***meme*** (Tim Bailey), ***MotifSampler*** (Gert Thijs), ***gibbs*** (Andrew Neuwald).

I particularly recommend the installation of ***mkvtree*** and ***vmatch*** (Stefan Kurtz), because these programs are used by the program purge-seq to discard redundant sequence fragments.

In order to add functionalities to ***RSAT***, install some or all of these programs and include their binaries path rsa-tools/bin. If you are not familiar with the installation of unix programs, ask assistance to your system administrator.

Program	author	URL
vmatch	Stefan Kurtz	http://www.vmatch.de/
patser	Jerry Hertz	ftp://ftp.genetics.wustl.edu/pub/stormo/Consensus/
consensus	Jerry Hertz	ftp://ftp.genetics.wustl.edu/pub/stormo/Consensus/
meme	Tim Bailey	http://meme.sdsc.edu/meme/website/meme-download.html
MotifSampler	Gert Thijs	http://www.esat.kuleuven.ac.be/~thijs/download.html
gibbs	Andrew Neuwald	ftp://ftp.ncbi.nih.gov/pub/neuwald/gibbs9_95/

Table 1: Programs from other developers which are complementary to the ***RSAT***package.

9 Further steps

The installation is now finished, you can start the user's guide.