SIROCCO ADDAPTS

A Data-Driven Automated Pipeline & Tracking System

Administration Guide

Version 1.0.2

Rishi Nag
rn202@cam.ac.uk

4/8/2011

Introduction

This document details the install and administration procedure for SIROCCO ADDAPTS.

The document describes

- An overview of the system
- Licencing Information
- The Installation Process

A separate User Guide is provided for end users to process samples.

About SIROCCO

SIROCCO is an EU-funded collaboration over 20 research institutes investigating the role of Silencing RNAs as organisers and coordinators of complexity in eukaryotic organisms.

The goals of the SIROCCO project are

- to exploit RNA silencing mechanisms in order to improve human and plant health
- to reveal how siRNAs play a role in developmental biology and disease defense
- to provide fundamental insights into the genetic networks that underlie normal and diseased embryonic development and adult growth

For further details on the SIROCCO project and more bioinformatics tools, check the website at http://www.sirocco-project.eu .

Licence

Preparation

Download ADDAPTS

The latest version of the ADDAPTS software can be downloaded from http://www.sirocco-project.eu/pipeline/, along with the latest copy of this Installation Guide and and User Guide.

The software is provided as tar.gz file.

In addition to this software, you will be required to install supporting software packages which are used by ADDAPTS to run and process data.

Gather your information

If you are installing this software you will have some idea of the species you will be dealing with.

In this guide Wild Type *Arabidopsis thaliana* will be provided as a reference. Decide on an organism and its ecotype. So, for purposes of this installation the organism is *Arabidopsis thaliana* and the ecotype is Wild Type.

Note that ADDAPTS can handle multiple genomes and ecotypes in a single installation.

Decide on the organisations involved, including one designated as a sequencing centre.

You will also need to decide on an install location for the data that gets processed and for the software itself to be installed to.

Overview

SIROCCO ADDAPTS is essentially two components, connected together by a common database.

The first is the TRACKING system – a LIMS where the users can enter sample information and view results once the processing has completed. This is a web front end.

The other component is the PIPELINE – this processes the samples as they arrive, running pre-alignment processes such as adaptor removal, genome alignment and then conversion into various file formats suitable for use as other items.

During operation the two of these are shell scripts left to run in a loop. These could either be set up as background tasks or screen sessions.

The two are tied together via a PostgreSQL database. This document will guide you through the PostgreSQL setup procedure if it is not already installed on your machine. You can test for the installation by typing.

```
which psql
```

Note that the block above indicates the style in which command line fragments will be highlighted in the document. Command line fragments will assume the current working directory is the main install directory containing the pipeline and tracing subdirectories.

Initial configuration occurs within configuration template files. The two main configuration files are YAML template files. For more information on YAML see http://www.yaml.org/. Follow up configuration is perfomed using the front end for the database.

Initial Configuration

SIROCCO ADDAPTS is designed to be run on a Linux system. The following set of commands will download ad install the pre-requistes to a Debian based system such as Ubuntu or Debian.

The first step is to create a user to 'be' the pipeline.

Add a Linux User 'pipe'

You may wish to process data under a specific pipeline user. This will aid in the future releases where the release will fit with linux task management procedures.

We create a user 'pipe' with the password 'pipe' for our example. Do this using from the command line using '`sudo useradd pipe`' and entering password and other information as required.

Install The Software

Decide on the install location for the software. It can be run from the pipe users home directory, in which case run the command:

```
tar -xvfz sirocco-addapts-1.0.tar.gz /home/pipe/addapts-1.0
```

Note that the directory used in command line examples is the install directory. e.g. /home/pipe/addapts-1.0 in the example above.

General pre-requisites

There are a number of modules required by the software, these are detailed here.

```
sudo apt-get install postgresql
sudo apt-get install libcatalyst-perl
sudo apt-get install libcatalyst-modules-perl
sudo apt-get install postgresql-plperl-8.4
sudo apt-get install libcatalyst-engine-apache-perl
sudo apt-get install libdbix-class-schema-loader-perl
sudo apt-get install libhtml-mason-perl
sudo apt-get install pkg-config
sudo perl -MCPAN -e 'install BioPerl'
sudo perl -MCPAN -e 'install ExtUtils::PkgConfig'
sudo perl -MCPAN -e 'install Catalyst::View::Graphics::Primitive'
sudo perl -MCPAN -e 'install Chart::Clicker'
sudo perl -MCPAN -e 'install Mouse'
sudo perl -MCPAN -e 'install Catalyst::Model::DBIC::Schema'
sudo perl -MCPAN -e 'install CatalystX::Component::Traits'
sudo perl -MCPAN -e 'install DBIx::Class::ResultSet::Data::Pageset'
sudo perl -MCPAN -e 'install Text::CSV'
sudo perl -MCPAN -e 'install Text::CSV_XS'
sudo perl -MCPAN -e 'install Test::Files'
sudo apt-get install gcc
sudo apt-get install g++
sudo apt-get install libpopt-dev
sudo apt-get install zliblg-dev
```

CPAN elements can be problematic – if they do not install straightaway you can try the following

- re-run the command.
- Search for the download directory and then run make to complete the process. e.g. on one occassion it was necessary to change directory to ~/.cpan/build/Chart-Clicker-2.69-xx, Graphics-Primitive-0.61-xx and type 'make install' to force the module to install.

Pipeline components

ADDAPTS uses third party components to perform operations such as alignment. Options g

Note that by individual package download you can get greater control of software package versions, but this comes at greater convenience.

- PATMAN
  - Download the package from http://bioinf.eva.mpg.de/patman/

```
tar -vxzf patman-1.2.2.tar.gz
cd patman-1.2.2
make install
```

- SAMTOOLS
  - Download the package from http://samtools.sourceforge.net/
  - (OR sudo apt-get install samtools)
- (optional) BWA
  - Download the package from https://sourceforge.net/projects/bio-bwa/files/

```
tar xvfj bwa-0.5.9.tar.bz2
make install
```

  - (OR sudo apt-get install bwa)

Establish the exact paths to the programs being run. This will allow the programs to fit into process management systems such as condor in the future, and will help keep track of software versions used to process data, important for replication of data for publication. You can determine location using the command 'which'

```
which patman
which bwa
which samtools
```

Edit the file `pipeline/pipeline.yaml.template` and place the full path in the positions below the tools.

Genome Preperation (example is TAIR 10):

Download the genome in fasta format

(ftp://ftp.arabidopsis.org/home/tair/Genes/TAIR10_genome_release/TAIR9_chr_all.fas)
Decide on a location for your genomes. The default is /data/organisms and is set in pipeline/pipeline.yaml.template. The example has the genome file placed with /data/organisms at arabidopsis/TAIR_9/TAIR9_chr_all.fas

Create an index file from the genome using samtools: samtools faidx TAIR9_chr_all.fas

Edit the file pipeline/pipeline.yaml.template and place the full path in the positions below the tools.

Database Setup

A postgres user is created by default during the apt-get install process. If a password has not been set up use the following to set this up (pgpwd for this exercise).

```
sudo passwd postgres
```

Setup psql database for the item:

The pipe user needs to be setup as a PostgreSQL role (n.b. as of PostgreSQL 8.3 'roles' are used rather than users/groups). Login as the PostgreSQL user.

```
su postgres
createuser -dSR pipe
createdb -O pipe sirocco_addapts_A
createdb -O pipe sirocco_addapts_testA
createlang plperl sirocco_addapts_A
exit
```

Note that new PostgreSQL installations may get the following message - psql: FATAL: Ident authentication failed for user "pipe". In this case you will need to login as the postgres user and edit a configuration file. The location of this may vary depending on the version of PostgreSQL installed.

```
su – postgres
sudo emacs /etc/postgresql/8.4/main/pg_hba.conf
```

and add the following at the end of the opened conf file.
local all all trust
host all 127.0.0.1/32 trust

Exit the editor and then type:

```
service postgresql restart
exit
```

Check the database exists by listing, and then load the database with the schema required for ADDAPTS operation

```
psql -l
psql -U pipe sirocco_addapts_A < pipeline/etc/addaptsA.sql
```

Directories

The pipeline uses a main working directory. The default setting is to use /data/pipeline/addaptsA. You will also need to make the directory /data/pipeline/addaptsA/fastq and have these writable by the user pipe. To alter these settings edit the file pipeline/etc/prod_deploy_params.

For future-proofing the configuration has two separate directories to process.

```
'pipeline-directory': /data/pipeline
'pipeline-process-directory': addaptsA
```

In the the pipe user home directory create a subdirectory to be used for log files.

```
mkdir ~/log
```

Configuration File editing/generation

Edit pipeline/etc/prod_deploy_params with the directory and database access values above.
Edit pipeline/pipeline.yaml.template with the full program paths and genome values as shown above.

Update the configuration.

```
./update_config.sh
```

Tracking Configuration

The Tracking system is a web front end built on the Catalyst Framework.

Start the Tracking Web Front End

```
cd pipeline
PERL5LIB=../tracking/lib ./script/smallrna_web_server.pl -p 3042
```

Confirm that the front page of the Tracking System is visible when you run by going to `http://localhost:3042` in your web browser. The web front end that appears should look something like this.

An optional -d can be used to generate debug information.

This command can be run in a screen session to allow the terminal to be disconnected from the machine running the process.

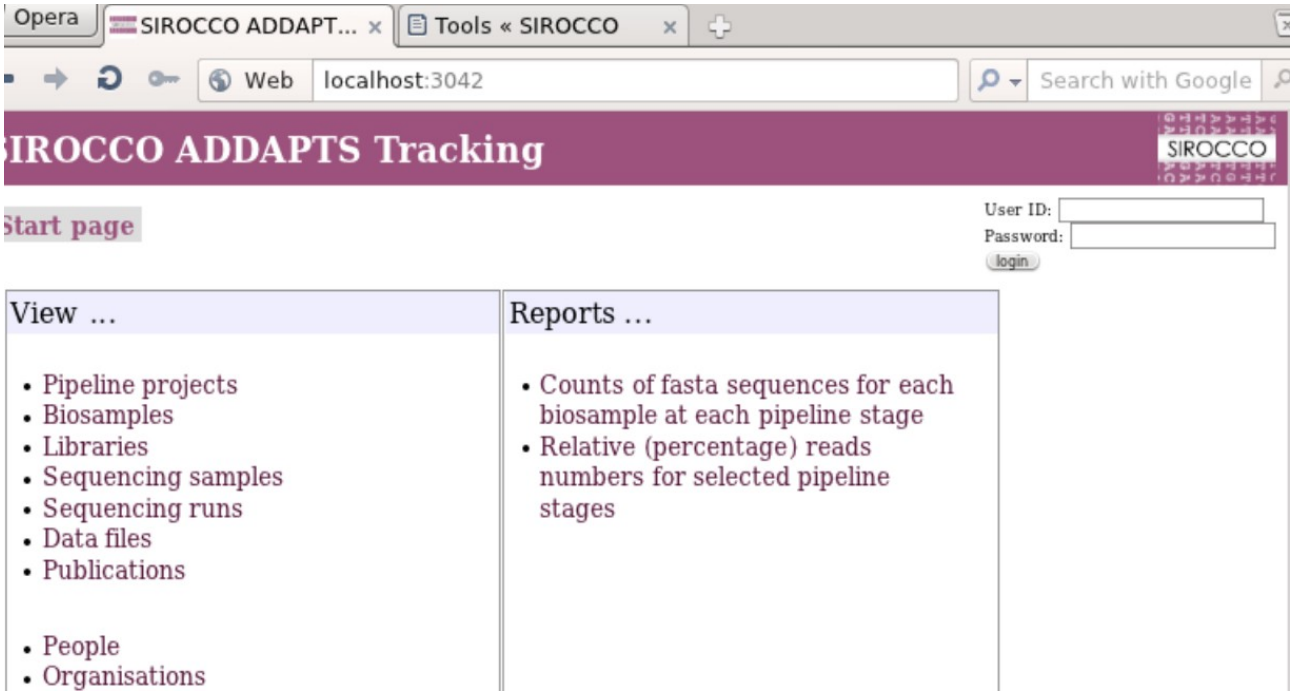
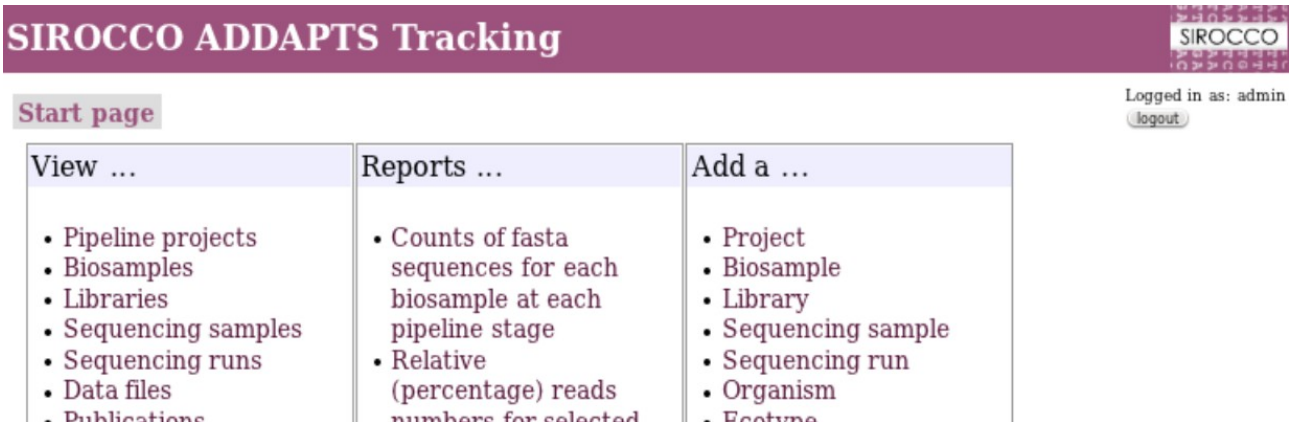


Illustration 1: SIROCCO ADDAPTS initial front page

Note that there will be some failed images – both as messages in the console and in the generated web page, these can be ignored.

Logging In

A default admin user is setup with username *admin* and password *admin99*. You can use this to login via the login box on the top right hand side of the screen. This is stored as a plain text password in the PostgreSQL database and can be edited using psql.



You can now add administer the tracking aspect of SIROCCO ADDAPTS via the web interface.

In particular, now you have logged in a new extra column will appear with various options to Add items to the setup.

Organisations and Personnel

By clicking on 'Add a...' -> Organisation you may add various organisations. It is intended groups and institutions, funding bodies and sequencing centres you may partner with are reflected here.

In particular, **adding an organisation as a dedicated sequencing centre is essential** to the process being operational. When adding this, take a note of the sequencing centre.

You may also add personnel with the 'Add a...' -> Person option.

Non-admin personnel can browse the system but only users with admin roles may adjust values and add new items.

Configuring Biological Data: Organisms, Ecotypes and Tissues

To configure the pipeline aspect the system must first be set up with the

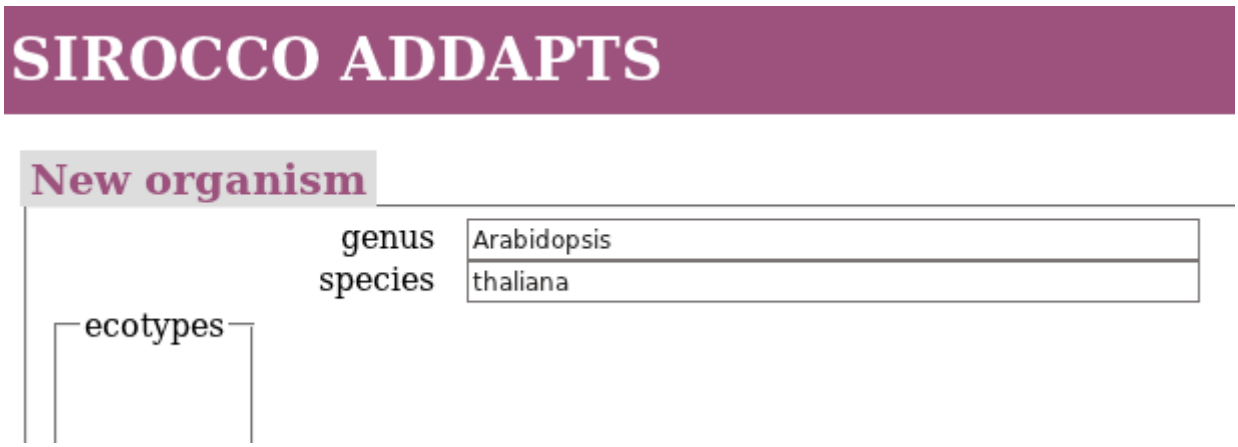
The next step is to add an organism. In the right hand 'Add' column click 'Organism'.

To continue our example with Arabidopsis thalina we will specify this.

In the `pipeline/pipeline.yaml.template` there should be something resembling the following at the bottom. This may have been changed in the steps above to reflect the species you may be dealing with.

```
databases:
  root: /data/organisms
  organisms:
    Arabidopsis_thaliana:
      database_files:
        at-genome-tair9: arabidopsis/TAIR_9/TAIR9_chr_all.fas
```

The organism name as listed here (Arabidopsis_thaliana) is of the form *genus_species*. Enter those values for the database as shown in the following figure, and then click Submit.



After this you can return the the home page by clicking the banner at the top of the page. It is worth discussing with experimentalists the range of ecotypes and tissue types they require.

Under Add - Ecotype add the new ecotypes into the system. (e.g. Wild Type, C24)

Under Add – Tissue types add tissue sample types (e.g. root, bud).

Configuring Data Processing: process_conf

This is the 'Database Driven' aspect of the ADDAPTS acronym.

The pipeline is designed to be modular in terms of what can be used to process the data at each stage. Each step of processing is a process_conf value in the database. Each of these will usually have an input process configured. The system is preloaded with standard process_conf values.

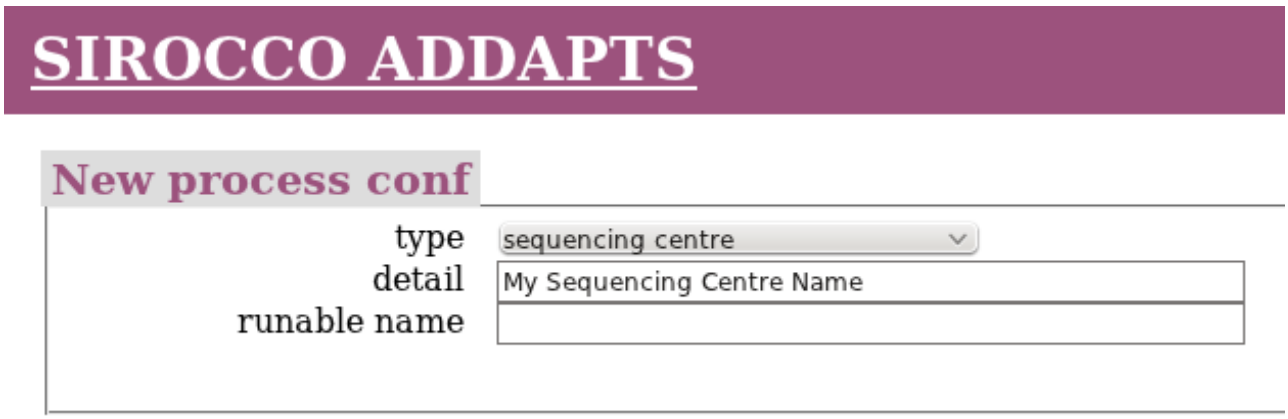
To complete pipeline configuration for your genome and sequencing centre the following steps need to be completed.

Note: it is essential the sequencing centre, organism and ecotype have been added before proceeding.

The first new Process Conf will be for the Sequencing Centre as you added earlier in the 'Organisations and Personnel' section.

From the homepage go to Add -> Pipeline Process Configuration.

From the dropdown menu under type choose 'sequencing centre'. Under detail type the name of the organisation doing the sequencing exactly. Leave the 'runable name' element blank, and click 'Submit'.



As you have seen The 'New process_conf' screen consists of fields for

- type
- detail
- runnable name

The first 'sequencing centre' process we added did not require any input data (it is responsible for the 'process' of the sequencing centre analysing data).

For the next two elements, after the add process occurs you will have to specify the type of file to be accessed by this process.

To do this,

- From the homepage go to the Add -> Pipeline Process Configuration page.
- Enter data as specified in the first three columns of the following table
- Click Submit
- On the next screen click on 'Create a input configuration for this ProcessConf...'
- A new page will pop up prompting for information which should be provided as in the last four columns for the corresponding row.
- Repeat this process (i.e. two new process_conf values) for each Organism/Ecotype pair.

Type	Detail	Runable Name	Content type	Format type	Ecotype	Biosample type
patman alignment	component: at-genome-tair9	SmallRNA::Runable::PatmanAlignmentRunable	non_redundant_reads	fasta	Wild Type Arabidopsis thaliana	
sam to bam converter		SmallRNA::Runable::SAMToBAMRunable	aligned_reads	sam	Wild Type Arabidopsis thaliana	

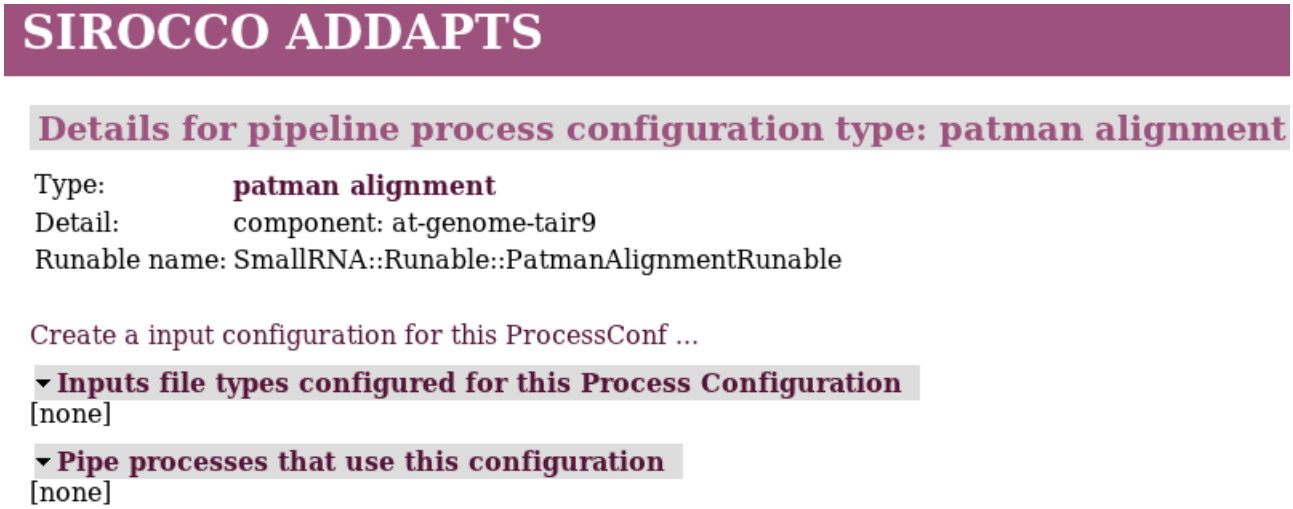


Illustration 2: Results of Add ProcessConf

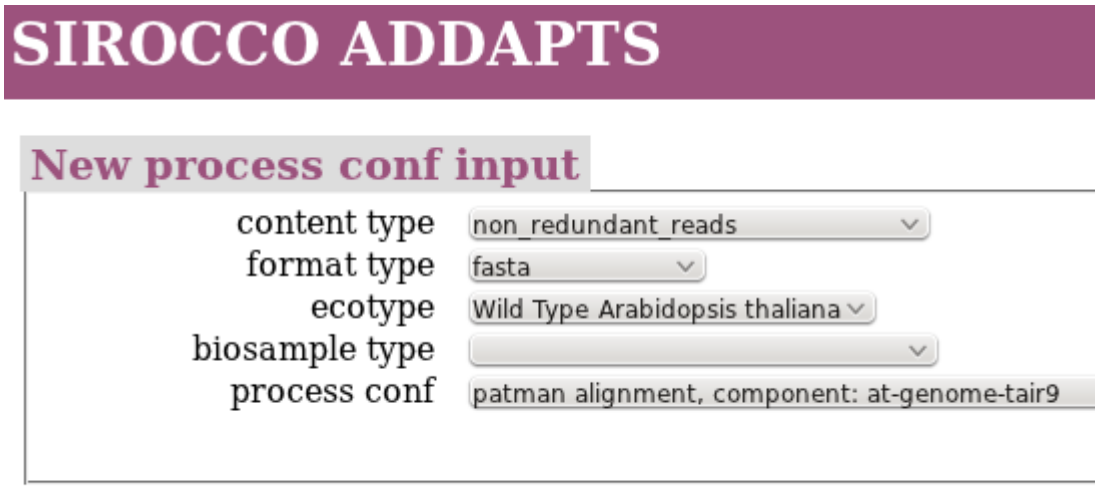


Illustration 3: Add Process Conf Input Data

Pipeline Configuration

Test Procedure

To verify successful tools and ADDAPTS installation, run the pipeline test procedure. This is a suite of tests to run the pipeline. Set the pipeline test configuration file up in pipeline/t/test_config.yaml with paths for programs as determined in the main configuration file. (n.b. Some tests currently fail due to program upgrades since the reference data was generated).

```
createdb -U pipe test_addapts_A
cd pipeline
perl Makefile.PL
make test
```

Pipeline Start

The pipeline runs as a script which regularly checks the database for requirements.

```
cd pipeline
PERL5LIB=lib script/pipeserv.pl -h localhost pipeline.yaml 2>&1 | tee -a
logs/`date +"%Y-%m-%d"`.log
```

This can be run inside a screen session to allow the user to log off and close the terminal as required.

If you have the condor task management system installed you can run this using the following.

```
cd pipeline
PERL5LIB=lib script/pipeserv.pl pipeline.yaml 2>&1 | tee -a logs/`date
+"%Y-%m-%d"`.log
```

Processing Sequencing Data

To process a file the user must first set up a sample in the database. The process of doing this is described in the accompanying SIROCCO ADDAPTS User Guide.

The system as it stands is set up to process sequenced data as a fastq file recieved from the sequencing centre. Once such a file is recieved the system needs to be told it has arrived by

```
PERL5LIB=pipeline/lib tracking/script/seq_completed.pl pipeline/pipeline.yaml
<seq_run_id> <seq_fastq>
```

The arguments are:

- seq_run_id – the unique numerical ID of the Sequencing Run in the database to which the data file corresponds
- seq_fastq – the fastq file name, having been placed in the data directory as specified in pipeline.yaml

For example, running the command:

```
~/ADDAPTS_1.0.2$ PERL5LIB=pipeline/lib tracking/script/seq_completed.pl
pipeline/pipeline.yaml 1 at_test1.fq
```

might give the following output.

```
Config:pipeline/pipeline.yaml
Adding:at_test1.fq to 1
querying database...
Data file being added for Sequencing Run 1
Sequencing Sample found in DB with ID: 1
Molecule type: RNA
Added at_test1.fq to the database
Completed
```

Depending on how the pipeline script was started, the processes will now be underway. Process can be viewed on the pipeline logs and pipeline console output. Files will be generated in the directory specified in the configuration file.

Results can be view under the biosample under the tracking system. See the User Guide for more information.

Backup Configuration

If databases and files are not being backed up automatically, it is a good idea to arrange for this to happen.

Database backups can be performed using pg_dump. This process can be automated by setting it up as a cron task.

```
pg_dump -f addapts_backup.sql sirocco_addapts_A
```

Data file backups should also be scheduled regularly, but this will depend on your particular server arrangement. Due to volumes of data generated from sequencing it is a good idea to arrange this to be an incremental backup.

Epilogue

Congratulations! You have now installed the SIROCCO ADDAPTS pipeline.

A separate user guide is made available to the user from http://www.sirocco-project.eu/pipeline/.

If you require further assistance please email bioinformatics@plantsci.cam.ac.uk