

SAPAC Summer Scholarship
2006/2007

User-Friendly Web Interfaces for Phylogenetic Applications

Sponsored by:
**SAPAC &
APAC**

Student:
John Toubia

Supervisors:
Shunde Zhange, Julian Schwerdt and
Dr Paul Coddington

Table of Contents

Introduction

Background.....	3
Aims.....	3
Progress against the Aims.....	3

Discussion

GridSphere and GridPortlets.....	4
Phylogenetic Applications – MrBayes.....	5
– PAUP*.....	5
– BEAST.....	5
JCU Source Code.....	6
Bioportal Operations – Input File.....	7
– Submitting a Job.....	9
– Retrieving Output.....	9

Conclusion

Future Directions.....	10
------------------------	----

Additional Information

Appendix I - configuration.xml.....	11
II - globus_PAUP/BEAST.tmpl.....	15
III - instructions.html.....	17
IV - Commands in a PAUP*job.....	19
V - Input Files.....	21
VI - Screen Shots.....	27
References.....	30

Background:

SAPAC supports a number of researchers in the fields of molecular and evolutionary biology who use phylogenetic programs such as MrBayes, PAUP* and BEAST. Current users run these programs from the command line which demands a level of knowledge on UNIX and job submission systems, such as PBS, that generally is unfamiliar to most biologists. Additionally, it may take some time for a new user to simply acquaint themselves with the SAPAC high-performance computing systems.

Aim:

To develop a simple, user-friendly web-based interface (Bioportal) that allows current and future users to access the above mentioned phylogenetic programs and effortlessly run jobs on SAPAC supercomputers and the APAC National Grid. The interface will extend an existing GridSphere web portal developed by Ian Atkinson's group at James Cook University (JCU), which supports MrBayes and allows jobs to be run on clusters using PBS, or via APAC grid gateways.

Progress against the Aim:

This project has achieved the aims which were set. However, due to time restraints, only limited feedback from the various research groups that were to test the portals was received. Customisation and modification to the portals based on further feedback may be required.

Discussion

GridSphere and GridPortlets:

Portlet-based portals have become one of the most exciting areas for portal server platforms. The advent of Java™ Specification Request (JSR) 168, the portlet API, allows portal vendors to offer a common programming model that will allow developers to more rapidly plug new functionality into a portal server, making it easily available for its consumer base. The GridSphere Portal provides an advanced open source portlet-based portal. The portlet model gives users a flexible easy-to-use interface, and it gives portal developers a model to create pluggable and dynamic application support. Core GridSphere features appropriate to this project include;

- Support for the easy development and integration of "third-party portlets" that plug into GridSphere portlet container
- Flexible, XML-based portal presentation description can be easily modified to create customized portal layouts
- Built-in support for Role Based Access Control (RBAC) separating users into guests, users, admins, and super users
- Sophisticated portlet service model that allows for creation of "user services," where service methods can be limited according to user rights
- Persistence of data provided using Hibernate for RDBMS database support
- Integrated JUnit/Cactus unit tests for server-side testing of portlet services including the generation of test reports
- GridSphere core portlets offer base functionality including login, logout, user, and access control management

(Novotny, 2004)(Wehrens, 2006).

Grid Portlets builds upon the GridSphere portlet container to provide many Grid services such as job submission, file transfer, and credential management. The Grid Portlets web application enables users to upload their Grid credentials and use them to gain access to Grid services. With respect to the Bioportal, Grid Portlets is utilized by the user to delegate credentials to a portal with the Credential Retrieval Portlet, enabling sign-on to the Grid with those credentials (Grid Portlets User's Guide, 2006).

Phylogenetic Applications:

The influence of high-speed computer analysis of molecular, morphological and/or behavioural data to infer phylogenetic relationships has expanded well beyond its central role in evolutionary biology, now encompassing applications in areas as diverse as conservation biology, ecology, and forensic studies. In order to meet the demands from researchers in these fields, SAPAC has chosen three phylogenetic programs to offer to its users. They are described in short below.

MrBayes - Bayesian Inference of Phylogeny:

MrBayes (Ronquist and Huelsenbeck 2003) is a program for the Bayesian estimation of phylogeny. The version of the program used (3.1.2) is distributed with open source code and possesses the ability to spread jobs over a cluster of computers using MPI. It reads sequence data in the NEXUS file format, and outputs posterior distribution estimates of trees and parameters.

PAUP* - Phylogenetic Analysis using Parsimony:

PAUP (Swofford, D. L. 2003) is a program for phylogenetic analysis using parsimony, maximum likelihood, and distance methods. It is the most widely used software package for the inference of evolutionary trees. PAUP is a single threaded application that will only take advantage of one processor at a time. The program reads sequence data in the NEXUS file format.

BEAST - Bayesian Evolutionary Analysis Sampling Trees:

BEAST (Drummond and Rambaut, 2006) is a cross-platform program for Bayesian Markov chain Monte Carlo (MCMC) analysis of molecular sequences. The program uses a complex and powerful input format (specified in XML) to describe the evolutionary model. BEAST, like PAUP, is a single threaded application that will only take advantage of one processor at a time. The version of the program used (1.3) is distributed with open source code.

JCU Source Code:

JCU provided the source code to a GridSphere web portal which was assembled, installed and tested on ngportaltest. Minor alterations were required to get the JCU portal running and returning valid output at SAPAC. The portal supported MrBayes and allowed jobs to run on clusters using PBS, or via APAC gridways. The development of new portals to support extra phylogenetic programs was made easy, due to the generic nature of the JCU code. The steps taken were as follows:

For each separate package added to the portal;

- I. A program-configuration block customised to each program was added to the configuration.xml file (Appendix I).
- II. A template file containing the relevant script customised to each program was produced (Appendix II).
- III. Instructions.html was customised to include generic instructions for all portals (Appendix III).

Any future additions to the portal should be installed in the same manner. The interface was also slightly modified for aesthetic reasons.

At present, the Bioportal will not accept PBS jobs due to the security risk involved in Network File System (NSF) file transfers. NFS is a *stateless* protocol. This means that the file server stores no per-client information, and there are no NFS "connections" (Sun Microsystems, 2007). In order to gauge the risks associated with running a Bioportal job, a flow chart of all commands called during job submission and file retrieval was produced. This was done for both globus and PBS job submission types (only globus type shown. Appendix IV). Modification of the code may be required if globus type job submission is deemed a security threat.

Bioportal Operations:

There are three parts involved in running any job on the Bioportal:

1. Production of an input file.
2. Selecting the appropriate settings and submitting the job.
3. Retrieving the output.

Input File:

Production of the input file is placed in the user's hand. However, it must conform to a small number of conditions which are described to the user on the submission page of the portal. As mentioned previously, both MrBayes and PAUP* read sequence data in the NEXUS file format while BEAST is specified in XML.

NEXUS is a file format designed to contain systematic data for use by computer programs. The format is modular, with a file consisting of separate blocks, each containing one particular kind of information and consisting of standardized commands. Public blocks (those containing information utilized by several programs) house information about taxa, morphological and molecular characters, distances, genetic codes, assumptions, sets, trees, etc.; private blocks contain information of relevance to single programs (Maddisson *et al* 1997). With respect to the phylogenetic programs used and job submission on the Bioportal, it is the private block with which we are most concerned. The private block simply contains all the commands relevant to the program in question and is therefore referred to as the command block. These blocks are mostly used for batch jobs where the same commands are executed on a number of files. To ensure the return of valid output when performing a job on the Bioportal, certain defined commands are required in the command block of each input file for both MrBayes and PAUP* (see Appendix V for details). These requirements are made compulsory so that the elected program runs in a non-interactive way, withholding any program requests to the user, and exiting the program when the job is finished. To educate users, example command blocks for both programs are displayed on the job page with compulsory commands highlighted. Additionally, links to relevant online documentation are provided.

XML (eXtensible Mark-up Language) is not a file format in itself but rather a set of rules for defining a file format that can be easily (and reliably) read by software and still read and edited by humans. BEAST uses a particular format that has been defined according to XML rules (Drummond and Rambaut, 2006). As has been mentioned before, the BEAST XML input format is both complex and powerful (see Appendix V for an example). This has advantages in terms of flexibility in that the developers of BEAST did not have to try and predict every analysis that researchers wished to perform and explicitly provide an option for doing it. However, this flexibility means it is possible to construct models that don't perform well under the Markov chain Monte Carlo (MCMC) inference framework used. Thus, construction of a BEAST input file requires a considerable amount of knowledge in phylogenetic processes as well as XML coding. To quote the creators of the program, “BEAST is not a black-box into which you can put your data and expect an easily interpretable answer. It requires careful inspection of the output to check that it has performed correctly and usually will need tweaking, adjustment and a number of runs to get a valid answer, Sorry”. It is expected that users of the portal possess all such knowledge as only a link to the BEAST online XML format documentation is provided.

A possible addition to the portal is the BEAUti (BEAST Utility) program included in the BEAST package. It is a simple user interface for creating input files to run BEAST. It takes sequence data as NEXUS format (as used by MrBayes and PAUP), allows you to select the evolutionary model, tune the operators and configure the MCMC. It then generates an XML format file that can be used with BEAST. However, this program cannot create even a small fraction of the models that are potentially available in BEAST. This said, it does provide an easy way to set up some of the simpler and more commonly used ones and can also provide a way to generate the input XML format for subsequent editing and tweaking (Drummond and Rambaut, 2006).

Submitting a Job:

All users must perform some preliminary steps before being allowed to submit a job on the Bioportal. To begin, a user must create a proxy and delegate it to the myproxy server (assuming the user already possesses a grid certificate). After logging into the portal the user will need to retrieve and activate his/her credentials from “myproxy.apac.edu.au.” using the grid portlets credential retrieval page (see Appendix VI for sample screen shots). Once these steps are completed, the user is free to join the Bioportal group and begin the job submission process.

On entering the portlet, the user is presented with a simple interface maintaining two pages called job and files (see Appendix VI for sample screen shots). The user can alternate between the two pages by hitting the appropriate tabs in the navigation bar.

Job page

It is on this page that all options are selected and jobs are submitted. Located at the centre of the page is the job submission instruction which explains the simple procedure of submitting a job. The process of submitting a job is as follows;

1. Choose a program from the drop down list in the navigation bar.

Then, from within the input panel

2. Enter or select the command file in the command file panel.
3. Adjust the job type, number of CPU's and host in the appropriate drop down lists.
4. Select the submit job option in the action panel and hit the “Go” button.

The user also has the option to reset the form, save defaults and reload defaults from the action drop down list.

Retrieving the Output:***Files page***

Output is retrieved simply by hitting the files tab, expanding the relevant timestamped folder and selecting “output.zip”, prompting a download manager.

Conclusion

Future Directions:

Informatics has assisted evolutionary biologists in several key ways and the Bioportal will prove to be an additional, versatile tool in the study of the origin and descent of species. The portal offers biologists, an easy to use interface to commonly employed programs running on SAPAC supercomputers, without prior knowledge on UNIX commands and job submission systems, such as PBS.

Future advancements to the portal may include;

- Job Scheduling, which will give end users, application developers and managers a scheduling functionality similar to that found on local Distributed Resource Management (DRM) systems (This additional feature is currently under construction).
- Input file preparation, assisting users in converting files of biological data (nucleotide or protein) between various formats.
- The addition of extra bioinformatics programs, diversifying the usefulness of the portal.

Appendix I - configuration.xml

MrBayes program-configuration block shown in black, PAUP* program-configuration block shown in red and BEAST program-configuration block shown in blue.

```

<!--
  READ ME:
    the prefix attribute is mandatory and indicates the base of the template tree
    the name, version, pbs and max-cpus attributes are mandatory
    max-cpus is used to indicate whether parallel is allowed (set to 1 if not)
    the instructions-url attribute overrides the instructions element
    the information-url attribute overrides the information element
    the locations attribute refers to files below prefix, ending in .jsp
    the html in the inputs element _must be _valid xhtml. In addition, every
    attribute _must_ be followed by the = character (SAX requirement). For
    example,
      <select name="s" multiple>
    while being valid html, will not get through the parser. You must instead use
      <select name="s" multiple="">
    A pain in the arse I know - I haven't tried too hard to solve this yet but will
    try sometime.
-->

<group-configuration
  name="group_name"
  site-code="JCU"
  prefix="http://ngportaltest.sapac.edu.au:8080/interpreter_shell">
  <program-configuration
    name="MrBayes"
    version="3.1.2"
    locations="job,files"
    pbs="true"
    max-cpus="16"
    instructions-url="http://localhost:8080/interpreter_client/config/instructions.html"
    information-url="http://localhost:8080/interpreter_client/config/information.html">
  <!--
    custom-preparation="http://localhost:8080/test.jar/BayesPrep" -->
  <job-configuration type="PBS" template="http://localhost:8080/interpreter_client/config/pbs_bayes.tpl">

  </job-configuration>
  <job-configuration type="globus" template="http://localhost:8080/interpreter_client/config/globus_bayes.tpl">
  <configuration>
    <transfer type="stagein" suffix="script.pbs">
      <source>gsiftp://@ftphost@@@base@@@name@@@version@@@timestamp@/</source>
      <destination>file:///${GLOBUS_USER_HOME}/@name@@@version@@@timestamp@_</destination>
    </transfer>
    <transfer type="stagein" suffix="input.tgz">
      <source>gsiftp://@ftphost@@@base@@@name@@@version@@@timestamp@/</source>
      <destination>file:///${GLOBUS_USER_HOME}/@name@@@version@@@timestamp@_</destination>
    </transfer>
  </configuration>
  </job-configuration>
</group-configuration>

```

```

<transfer type="stageout" suffix="stdout">
  <source>file://${GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</source>
  <destination>gsiftp://@ftphost@@base@@@name@@@version@@@timestamp@/</destination>
</transfer>
<transfer type="stageout" suffix="output.zip">
  <source>file://${GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</source>
  <destination>gsiftp://@ftphost@@base@@@name@@@version@@@timestamp@/</destination>
</transfer>
<cleanup>file://${GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_script.pbs</cleanup>
<cleanup>file://${GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_input.tgz</cleanup>
<cleanup>file://${GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_stdout</cleanup>
<cleanup>file://${GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_output.zip</cleanup>
<extensions>
  <pbsjobscript>yes</pbsjobscript>
  <logfile>tmp/globus_log</logfile>
  <jobname>@name@_@timestamp@</jobname>
  <module>mrbayes/3.1.2</module>
</extensions>
</configuration>
</job-configuration>
<inputs>
  <table>
    <tr>
      <td><FONT COLOR="blue">MrBayes Command File:</FONT></td>
    </tr>
    <tr>
      <td><input type="file" validator="http://localhost:8080/test.jar/BayesValidator" name="input"/></td>
    </tr>
  </table>
<!-- hidden inputs _do not_ get validated -->
  <input value="hidden_value" type="hidden" name="hidden_input"/>
</inputs>
</program-configuration>
<program-configuration
  name="PAUP"
  version="4.0"
  locations="job_files"
  pbs="true"
  max-cpus="1"
  instructions-url="http://localhost:8080/interpreter_client/config/instructions.html"
  information-url="http://localhost:8080/interpreter_client/config/information.html">
  <!--
    custom-preparation="http://localhost:8080/test.jar/BayesPrep" -->
</job-configuration type="PBS" template="http://localhost:8080/interpreter_client/config/pbs_paup.tmpl">
</job-configuration>
<job-configuration type="globus" template="http://localhost:8080/interpreter_client/config/globus_paup.tmpl">
  <configuration>
    <transfer type="stagein" suffix="script.pbs">
      <source>gsiftp://@ftphost@@base@@@name@@@version@@@timestamp@/</source>

```

```

        <destination>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</destination>
    </transfer>
    <transfer type="stagein" suffix="input.tgz">
        <source>gsiftp://@ftphost@@base@/@name@/@version@/@timestamp@/</source>
        <destination>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</destination>
    </transfer>
    <transfer type="stageout" suffix="stdout">
        <source>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</source>
        <destination>gsiftp://@ftphost@@base@/@name@/@version@/@timestamp@/</destination>
    </transfer>
    <transfer type="stageout" suffix="output.zip">
        <source>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</source>
        <destination>gsiftp://@ftphost@@base@/@name@/@version@/@timestamp@/</destination>
    </transfer>
    <cleanup>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_script.pbs</cleanup>
    <cleanup>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_input.tgz</cleanup>
    <cleanup>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_stdout</cleanup>
    <cleanup>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_output.zip</cleanup>
    <extensions>
        <pbsjobscrip>yes</pbsjobscrip>
        <logfile>/tmp/globus_log</logfile>
        <jobname>@name@_@timestamp@</jobname>
        <module>java</module>
    </extensions>
</configuration>
</job-configuration>
<inputs>
    <table>
        <tr>
            <td><FONT COLOR="blue">PAUP Command File:</FONT></td>
        </tr>
        <tr>
            <td><input type="file" validator="http://localhost:8080/test.jar/BayesValidator" name="input"/></td>
        </tr>
    </table>
<!-- hidden inputs _do not_ get validated -->
    <input value="hidden_value" type="hidden" name="hidden_input"/>
</inputs>
</program-configuration>
<program-configuration
    name="BEAST"
    version="1.3"
    locations="job,files"
    pbs="true"
    max-cpus="1"
    instructions-url="http://localhost:8080/interpreter_client/config/instructions.html"
    information-url="http://localhost:8080/interpreter_client/config/information.html">
<!--
    custom-preparation="http://localhost:8080/test.jar/BayesPrep" -->

```

```

<job-configuration type="PBS" template="http://localhost:8080/interpreter_client/config/pbs_beast.tpl">
</job-configuration>
<job-configuration type="globus" template="http://localhost:8080/interpreter_client/config/globus_beast.tpl">
  <configuration>
    <transfer type="stagein" suffix="script.pbs">
      <source>gsiftp://@ftphost@@base@/@name@/@version@/@timestamp@/</source>
      <destination>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</destination>
    </transfer>
    <transfer type="stagein" suffix="input.tgz">
      <source>gsiftp://@ftphost@@base@/@name@/@version@/@timestamp@/</source>
      <destination>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</destination>
    </transfer>
    <transfer type="stageout" suffix="stdout">
      <source>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</source>
      <destination>gsiftp://@ftphost@@base@/@name@/@version@/@timestamp@/</destination>
    </transfer>
    <transfer type="stageout" suffix="output.zip">
      <source>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_</source>
      <destination>gsiftp://@ftphost@@base@/@name@/@version@/@timestamp@/</destination>
    </transfer>
    <cleanup>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_script.pbs</cleanup>
    <cleanup>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_input.tgz</cleanup>
    <cleanup>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_stdout</cleanup>
    <cleanup>file://$GLOBUS_USER_HOME}/@name@_@version@_@timestamp@_output.zip</cleanup>
    <extensions>
      <pbsjobscrip>yes</pbsjobscrip>
      <logfile>/tmp/globus_log</logfile>
      <jobname>@name@_@timestamp@</jobname>
      <module>java</module>
    </extensions>
  </configuration>
</job-configuration>
<inputs>
  <table>
    <tr>
      <td><FONT COLOR="blue">BEAST Command File:</FONT></td>
    </tr>
    <tr>
      <td><input type="file" validator="http://localhost:8080/test.jar/BayesValidator" name="input"/></td>
    </tr>
  </table>
  <!-- hidden inputs _do not_ get validated -->
  <input value="hidden_value" type="hidden" name="hidden_input"/>
</inputs>
</program-configuration>
</group-configuration>

```

Appendix II - globus_paup.templ

```
#!/bin/sh
#PBS -V
### Output file
#PBS -j oe
### Queue name
#PBS -q hydra
### Number of nodes
#PBS -l nodes=1,walltime=10:00:00
# This job's working directory
echo Running on host `hostname`
echo Time is `date`
# set all variables that we need
# we're using variables so that we can test if the replacement succeeded
CPU=#CPU#
TIMESTAMP=#TIMESTAMP#
PROGRAM_NAME=#PROGRAM_NAME#
VERSION=#VERSION#
EXE="Paup"
# unpack our job
tar xzf ~/${PROGRAM_NAME}_${VERSION}_${TIMESTAMP}_input.tgz
cd ~/${PROGRAM_NAME}/${VERSION}/${TIMESTAMP}
# run the job
echo "running $EXE"
/opt/shared/paup/4b10-icc/bin/paup -n -f input
RETURN=$?
# pack up for stage out and clean up
echo "packing output"
zip ~/${PROGRAM_NAME}_${VERSION}_${TIMESTAMP}_output.zip *
cd ~
rm -rf ~/${PROGRAM_NAME}/${VERSION}/${TIMESTAMP}
# this might fail if there are other MrBayes (3.1.2) jobs running
# but we do it anyway for cleanup
rmdir ~/${PROGRAM_NAME}/${VERSION}
rmdir ~/${PROGRAM_NAME}
# shut down the MPI environment
# TODO: maybe do this directly after the job is finished?
exit $RETURN
```

Appendix II cont... - globus_beast.tmpl

```
#!/bin/sh
#PBS -V
### Output files
#PBS -j oe
### Queue name
#PBS -q hydra
### Number of nodes
#PBS -l nodes=1,walltime=10:00:00
# This job's working directory
echo Running on host `hostname`
echo Time is `date`
# set all variables that we need
# we're using variables so that we can test if the replacement succeeded
CPU=#CPU#
TIMESTAMP=#TIMESTAMP#
PROGRAM_NAME=#PROGRAM_NAME#
VERSION=#VERSION#
EXE="BEAST"
# unpack our job
tar xzf ~/${PROGRAM_NAME}_${VERSION}_${TIMESTAMP}_input.tgz
cd ~/${PROGRAM_NAME}/${VERSION}/${TIMESTAMP}
# run the job
echo "running $EXE"
/opt/shared/beast/1.3/bin/beast input
RETURN=$?
# pack up for stage out and clean up
echo "packing output"
zip ~/${PROGRAM_NAME}_${VERSION}_${TIMESTAMP}_output.zip *
cd ~
rm -rf ~/${PROGRAM_NAME}/${VERSION}/${TIMESTAMP}
# this might fail if there are other MrBayes (3.1.2) jobs running
# but we do it anyway for cleanup
rmdir ~/${PROGRAM_NAME}/${VERSION}
rmdir ~/${PROGRAM_NAME}
# shut down the MPI environment
# TODO: maybe do this directly after the job is finished?
exit $RETURN
```

Appendix III - instructions.html

```

<table>
  <tr valign="top">
    <td>
      <fieldset>
        <legend>Job Submission Instructions</legend>
        <p>
          To start, please choose the <font color="#0000FF">Program</font> you wish to use from the drop down list
          in the Navigation Bar. </p>
        <p>
          Please enter or select the Command File you wish to run in the <font color="#0000FF">Command
          File</font> panel. <em><FONT COLOR="red"><strong>Please note that the Command file that you select must contain 1 and
          only 1 Command Block.</strong></font></em> For more details see Command Block Creation below.
        <p>
          If necessary, please also adjust the job <font color="#0000FF">Type</font>, number of <font
          color="#0000FF">CPUs</font> and <font color="#0000FF">Host</font> that you require in the appropriate panels.
        <p>
          When you have completed the above steps please select the "Submit Job" option in the <font
          color="#0000FF">Action</font> panel and hit "Go" to run the job. If you wish to start over before you have submitted, select
          the "Reset Form" option and hit the "Go" button. To save or load any personal default settings, use the "Save as Default" and
          "Load Default" options.
        <p>
          Output can be retrieved by selecting the <strong>files</strong> tab in the Navigation Bar, expanding the
          relevant timestamped folder and selecting "output.zip" which will prompt a download manager.
        <p>
          To return to this Job Submission Portlet at any time, select the <strong>job</strong> tab in the Navigation
          Bar.
      </fieldset>
    </td>
  </tr>
</table>
<table>
  <tr valign="top">
    <td>
      <fieldset>
        <legend>Command Block Creation</legend>
        <p>
          The Command block simply contains the commands as you would have given them from the command line,
          with the difference that each command line is ended with a semi-colon. Command Blocks are usually placed after the Data Block
          of a Nexus File.
        <p>
          <strong>Example MrBayes Command Block. </strong><em><font color="#FF0000">note: red font
          indicates compulsory requirements</font></em>
          <BLOCKQUOTE><font color="#FF0000">begin mrbayes;
          <br>
          set autoclose=yes nowarn=yes;</font>
          <br>
        </td>
      </fieldset>
    </tr>
  </table>

```

```

lset nst=6 rates=gamma;
<br>
<font color="#FF0000">mcmc ngen=</font>10000 samplefreq=10 file=output;
<br>
sump filename=output burnin=250;
<br>
sumt filename=output burnin=250;
<br>
<font color="#FF0000">quit;
<br>
end;</font> </BLOCKQUOTE>

```

For more information on MrBayes commands please hit the link and reference the [MrBayes online manual](http://mrbayes.csit.fsu.edu/wiki/index.php/Manual).

Example PAUP Command Block. *note: red font indicates compulsory requirements*

```

<BLOCKQUOTE>
<p><font color="#FF0000">B</font><font color="#FF0000">egin paup ;<br>
set autoclose=yes warntree=no warnreset=no; </font>
<br>
log start file=output.log replace;
<br>
NJ;
<br>
LSCORE 1/ BASEFREQ=empirical TRATIO=estimate RATES=gamma SHAPE=estimate;
<br>
SET CRITERION=like;
<br>
LSET BASEFREQ=empirical TRATIO=previous RATES=gamma SHAPE=previous;
<br>
HSEARCH ADDSEQ=random NREPS=5 SWAP=TBR;
<br>
savetrees file=output.tre replace;
<br>
<font color="#FF0000">quit;
<br>
end;</font></BLOCKQUOTE>
<p>

```

For more information on PAUP commands please hit the link and download the [PAUP Command Reference Document](http://paup.csit.fsu.edu/downl.html).

For information about BEAST XML file format, please hit the link and reference the [BEAST online XML format documentation](http://evolve.zoo.ox.ac.uk/beast/help/XML_format_documentation).

```

</fieldset>
</td>
</tr>
</table>

```

Appendix IV - Commands in a Paup Job (Globus - Non JCU user)**job.java:** (variables)

```

-program path == /Paup
-version path == /4.0
-stamp path   == /070208_094314
-base        == /data/ngportal/johntoubia/paup/4.0/070208_094314
-tmp_dir     == /data/ngportal/johntoubia/paup/4.0/070208_094314
-properties: path.cp           ==/bin/cp
             host.gridftp     ==ngdata.sapac.edu.au
             dir.mount        ==/data/ngportal
             path.mv          ==/bin/mv
             dir.persistence  =/data/ngportal
             path.mkdir       ==/bin/mkdir
             path.qsub        ==/usr/bin/qsub
             PBS_FILEPATH     == /script.pbs
             tarball_path     == /input.tgz

```

Command 1 (common_prep - 507) Temporary directory created
if(!tmp_dir.mkdirs())

Command 2 (common_prep - 544) Base template created
String template = getTemplate(jobtype, get(CPU_KEY) != null)@
/data/ngportal/johntoubia/paup/4.0/ 070208_094314/script.pbs

Util.java

Command 3 (run - 297) Processing Input - called by job.fixTemplate
properties.getProperty(PATH_CP), input.file_value.getPath(), tmp_dir.getPath() + File.separator + key
-running: /bin/cp/tmp/input10030porletupload/data/ngportal/johntoubia/
paup/4.0/070208_094314/input

Command 4 (run - 297) Make Executable - called by job.common_prep
properties.getProperty(PATH_CHMOD), "+x", tmp_dir + PBS_FILEPATH
-running: /bin/chmod +x /data/ngportal/johntoubia/paup/4.0/070208_
094314/script.pbs

Script.pbs

```

-tar zxf~/${PROGRAM_NAME}_${VERSION}_${TIMESTAMP}_input.tgz
-tar zxf ~/paup/4.0/070208_094314/input.tgz

-cd ~/${PROGRAM_NAME}/${VERSION}/${TIMESTAMP}
-cd ~/paup/4.0/070208_094314

-zip ~/${PROGRAM_NAME}_${VERSION}_${TIMESTAMP}_output.zip *
-zip ~/paup/4.0/070208_094314/output.zip

-rm -rf ~/${PROGRAM_NAME}/${VERSION}/${TIMESTAMP}
-rm -rf ~/paup/4.0/070208_094314

```

```
-rmdir ~/${PROGRAM_NAME}/${VERSION}  
-rmdir ~/paup/4.0/070208_094314
```

```
-rmdir ~/${PROGRAM_NAME}  
-rmdir ~/paup
```

Command 5 (run - 297) Zipping output - called by job.submit(GSSCredential)

```
"tar", "-C", orig_tmp_dir.toString(), "--exclude", config.getName() + version_path + stamp_path +  
tarball_path, "-zcf", tmp_dir + tarball_path, config.getName() + version_path + stamp_path  
-running: tar -C /data/ngportal/johntoubia --exclude paup/4.0/070208_  
094314/input.tgz -zcf /data/ngportal/johntoubia/paup/4.0/070208_  
094314/input.tgz paup/4.0/070208_094314
```

Globus_Job

Command 6 - destroying job

```
-job.destroy()  
-/data/ngportal/johntoubia/paup/4.0/070208_094314/.globus.xml
```

Portlet.java

Command 7 (doFilesAction 1150) Downloading file

```
-sudo /bin/chmod/ -R 755 data/ngportal/johntoubia/paup/4.0/070208_  
094314
```

(else, if file is not directory)

```
-sudo -u 644 data/ngportal/johntoubia/paup/4.0/070208_094314/output.zip
```

Appendix V - Input Files

Example MrBayes input file. Compulsory commands are shown in red.

```
#NEXUS
begin data;
dimensions ntax=12 nchar=898;
format datatype=dna interleave=no gap=-;
matrix
Tarsius_syrichta      AAGTTTCATTTGGAGCCACCACTCTTATAATTGCCCATGGCCT.....
Lemur_catta           AAGCTTCATAGGAGCAACCATTCTAATAATCGCACATGGCCT.....
Homo_sapiens          AAGCTTCACCGGCGCAGTCATTCTCATAATCGCCCACGGGG.....
Pan                   AAGCTTCACCGGCGCAATTATCCTCATAATCGCCCACGGACC.....
Gorilla               AAGCTTCACCGGCGCAGTTGTTCTTATAATTGCCACGGACA.....
Pongo                 AAGCTTCACCGGCGCAACCACCTCATGATTGCCCATGGACT.....
Hylobates             AAGCTTTACAGGTGCAACCGTCCCTCATAATCGCCCACGGACT.....
Macaca_fuscata        AAGCTTTTCCGGCGCAACCATCCTTATGATCGCTCACGGACT.....
M_mulatta             AAGCTTTTCTGGCGCAACCATCCTCATGATTGCTCACGGACT.....
M_fascicularia        AAGCTTCTCCGGCGCAACCACCTTATAATCGCCCACGGGCT.....
M_sylvanus            AAGCTTCTCCGGTGCAACTATCCTTATAGTTGCCCATGGACT.....
Saimiri_sciureus      AAGCTTCACCGGCGCAATGATCCTAATAATCGCTCACGGGTT.....
;
end;
[example MrBayes command block]
begin mrbayes;
  set autoclose=yes nowarn=yes;
  lset nst=6 rates=gamma;
  mcmc ngen=10000 samplefreq=10 file=output;
  sump filename=output burnin=250;
  sumt filename=output burnin=250;
  quit;
end;
```

Example PAUP* input file. Compulsory commands are shown in red.

```
#NEXUS
begin data;
dimensions ntax=12 nchar=898;
format datatype=dna interleave=no gap=-;
matrix
Tarsius_syrichta      AAGTTTCATTTGGAGCCACCACTCTTATAATTGCCCATGGCCT.....
Lemur_catta           AAGCTTCATAGGAGCAACCATTCTAATAATCGCACATGGCCT.....
Homo_sapiens          AAGCTTCACCGGCGCAGTCATTCTCATAATCGCCCACGGGG.....
Pan                   AAGCTTCACCGGCGCAATTATCCTCATAATCGCCCACGGACC.....
Gorilla               AAGCTTCACCGGCGCAGTTGTTCTTATAATTGCCACGGACA.....
Pongo                 AAGCTTCACCGGCGCAACCACCTCATGATTGCCCATGGACT.....
Hylobates             AAGCTTTACAGGTGCAACCGTCCCTCATAATCGCCCACGGACT.....
Macaca_fuscata        AAGCTTTTCCGGCGCAACCATCCTTATGATCGCTCACGGACT.....
M_mulatta             AAGCTTTTCTGGCGCAACCATCCTCATGATTGCTCACGGACT.....
M_fascicularia        AAGCTTCTCCGGCGCAACCACCTTATAATCGCCCACGGGCT.....
M_sylvanus            AAGCTTCTCCGGTGCAACTATCCTTATAGTTGCCCATGGACT.....
Saimiri_sciureus      AAGCTTCACCGGCGCAATGATCCTAATAATCGCTCACGGGTT.....
;
end;
[example PAUP* command block]
BEGIN PAUP;
  set autoclose=yes warnreset=no increase=auto;
  log file= paup.log replace;
  NJ;
  LSCORE 1/ BASEFREQ=empirical TRATIO=estimate RATES=gamma SHAPE=estimate;
  SET CRITERION=like;
  LSET BASEFREQ=empirical TRATIO=previous RATES=gamma SHAPE=previous;
  HSEARCH ADDSEQ=random NREPS=5 SWAP=TBR;
  savetrees file= paup.tre replace;
  quit;
END;
```

Appendix V cont... - Input Files**Example BEAST input file.**

```

<?xml version="1.0" standalone="yes"?>
<beast>
  <!-- The list of taxa analyse (can also include dates/ages). -->
  <!-- ntax=12 -->
  <taxa id="taxa">
    <taxon id="Tarsius_syrichtha"/>
    <taxon id="Lemur_catta"/>
    <taxon id="Homo_sapiens"/>
    <taxon id="Pan"/>
    <taxon id="Gorilla"/>
    <taxon id="Pongo"/>
    <taxon id="Hylobates"/>
    <taxon id="Macaca_fuscata"/>
    <taxon id="M_mulatta"/>
    <taxon id="M_fascicularis"/>
    <taxon id="M_sylvanus"/>
    <taxon id="Saimiri_sciureus"/>
  </taxa>
  <!-- The sequence alignment (each sequence refers to a taxon above). -->
  <!-- ntax=12 nchar=898 -->
  <alignment id="alignment" dataType="nucleotide">
    <sequence>
      <taxon idref="Tarsius_syrichtha"/>AAGTTTCATTGGAGCCACCACTCTTAT.....
    </sequence>
    <sequence>
      <taxon idref="Lemur_catta"/>AAGCTTCATAGGAGCAACCATTCTAATAA.....
    </sequence>
    <sequence>
      <taxon idref="Homo_sapiens"/>AAGCTTCACCGGCGCAGTCATTCTCAT.....
    </sequence>
    <sequence>
      <taxon idref="Pan"/>AAGCTTCACCGGCGCAATTATCCTCATAATCGCC.....
    </sequence>
    <sequence>
      <taxon idref="Gorilla"/>AAGCTTCACCGGCGCAGTTGTTCTTATAATTG.....
    </sequence>
    <sequence>
      <taxon idref="Pongo"/>AAGCTTCACCGGCGCAACCACCCTCATGATTGA.....
    </sequence>
    <sequence>
      <taxon idref="Hylobates"/>AAGCTTTACAGGTGCAACCGTCCTCATAAT.....
    </sequence>
    <sequence>
      <taxon idref="Macaca_fuscata"/>AAGCTTTCCGGCGCAACCATCCTTA.....
  </alignment>

```

```

</sequence>
<sequence>
  <taxon idref="M_mulatta"/>AAGCTTTTCTGGCGCAACCATCCTCATGA.....
</sequence>
<sequence>
  <taxon idref="M_fascicularis"/>AAGCTTCTCCGGCGCAACCACCCTTA.....
</sequence>
<sequence>
  <taxon idref="M_sylvanus"/>AAGCTTCTCCGGTGCAACTATCCTTATA.....
</sequence>
<sequence>
  <taxon idref="Saimiri_sciureus"/>AAGCTTCACCGGCGCAATGATCCTA.....
</sequence>
</alignment>
<!-- The unique patterns for all positions -->
<!-- npatterns=413 -->
<patterns id="patterns" from="1">
  <alignment idref="alignment"/>
</patterns>
<!-- A prior assumption that the population size has remained constant -->
<!-- throughout the time spanned by the genealogy. -->
<constantSize id="constant" units="substitutions">
  <populationSize>
    <parameter id="constant.popSize" value="0.3" lower="0.0" upper="300.0"/>
  </populationSize>
</constantSize>
<!-- Generate a random starting tree under the coalescent process -->
<coalescentTree id="startingTree">
  <taxa idref="taxa"/>
  <constantSize idref="constant"/>
</coalescentTree>
<treeModel id="treeModel">
  <coalescentTree idref="startingTree"/>
  <rootHeight>
    <parameter id="treeModel.rootHeight"/>
  </rootHeight>
  <nodeHeights internalNodes="true">
    <parameter id="treeModel.internalNodeHeights"/>
  </nodeHeights>
  <nodeHeights internalNodes="true" rootNode="true">
    <parameter id="treeModel.allInternalNodeHeights"/>
  </nodeHeights>
</treeModel>
<coalescentLikelihood id="coalescent">
  <model>
    <constantSize idref="constant"/>
  </model>

```

```

    <populationTree>
        <treeModel idref="treeModel"/>
    </populationTree>
</coalescentLikelihood>
<!-- The strict clock (Uniform rates across branches) -->
<strictClockBranchRates id="branchRates">
    <rate>
        <parameter id="clock.rate" value="1.0"/>
    </rate>
</strictClockBranchRates>
<!-- The HKY substitution model (Hasegawa, Kishino & Yano, 1985) -->
<hkyModel id="hky">
    <frequencies>
        <frequencyModel dataType="nucleotide">
            <alignment idref="alignment"/>
            <frequencies>
                <parameter id="hky.frequencies" dimension="4"/>
            </frequencies>
        </frequencyModel>
    </frequencies>
    <kappa>
        <parameter id="hky.kappa" value="1.0" lower="0.0" upper="100.0"/>
    </kappa>
</hkyModel>
<!-- site model -->
<siteModel id="siteModel">
    <substitutionModel>
        <hkyModel idref="hky"/>
    </substitutionModel>
</siteModel>
<treeLikelihood id="treeLikelihood">
    <patterns idref="patterns"/>
    <treeModel idref="treeModel"/>
    <siteModel idref="siteModel"/>
    <strictClockBranchRates idref="branchRates"/>
</treeLikelihood>
<operators id="operators">
    <scaleOperator scaleFactor="0.75" weight="1">
        <parameter idref="hky.kappa"/>
    </scaleOperator>
    <scaleOperator scaleFactor="0.75" weight="3">
        <parameter idref="constant.popSize"/>
    </scaleOperator>
    <scaleOperator scaleFactor="0.75" weight="3">
        <parameter idref="treeModel.rootHeight"/>
    </scaleOperator>
    <uniformOperator weight="30">
        <parameter idref="treeModel.internalNodeHeights"/>
    </uniformOperator>
</operators>

```

```

</uniformOperator>
<subtreeSlide size="0.03" gaussian="true" weight="15">
  <treeModel idref="treeModel"/>
</subtreeSlide>
<narrowExchange weight="15">
  <treeModel idref="treeModel"/>
</narrowExchange>
<wideExchange weight="3">
  <treeModel idref="treeModel"/>
</wideExchange>
<wilsonBalding weight="3">
  <treeModel idref="treeModel"/>
  <constantSize idref="constant"/>
</wilsonBalding>
</operators>
<mcmc id="mcmc" chainLength="10000" autoOptimize="true">
  <posterior id="posterior">
    <prior id="prior">
      <coalescentLikelihood idref="coalescent"/>
    </prior>
    <likelihood id="likelihood">
      <treeLikelihood idref="treeLikelihood"/>
    </likelihood>
  </posterior>
  <operators idref="operators"/>
  <log id="screenLog" logEvery="100">
    <column label="Posterior" dp="4" width="12">
      <posterior idref="posterior"/>
    </column>
    <column label="Root Height" sf="6" width="12">
      <parameter idref="treeModel.rootHeight"/>
    </column>
    <column label="L(tree)" dp="4" width="12">
      <treeLikelihood idref="treeLikelihood"/>
    </column>
    <column label="L(coalecent)" dp="4" width="12">
      <coalescentLikelihood idref="coalescent"/>
    </column>
  </log>
  <log id="fileLog" logEvery="100" fileName="primates.log">
    <posterior idref="posterior"/>
    <treeLikelihood idref="treeLikelihood"/>
    <coalescentLikelihood idref="coalescent"/>
    <parameter idref="treeModel.rootHeight"/>
    <parameter idref="constant.popSize"/>
    <parameter idref="hky.kappa"/>
  </log>
  <logTree id="treeFileLog" logEvery="100" nexusFormat="true" fileName="primates.trees">

```

```
        <treeModel idref="treeModel"/>
        <posterior idref="posterior"/>
    </logTree>
</mcmc>
<report>
    <property name="timer">
        <object idref="mcmc"/>
    </property>
</report>
</beast>
```

Appendix VI - Bioportal Screen Shots – Credential retrieval page

SAPAC PORTAL [Logout](#)
Welcome, john toubia

Welcome Administration **Grid** BioPortal

Registry **Credentials** Resources Files Jobs

Credential Manager Portlet

This credential can be retrieved from myproxy.apac.edu.au.

Label:* (Required: Label to display for credential in portal)
 johntoubia

User Name:* (Required: -l or --username option to myproxy-init)

Credential Name: (Optional: -k or --credname option to myproxy-init)

Credential Lifetime: 8100 (in seconds)

Passphrase:* (Required: Your credential repository password)

February 16, 2007

SAPAC PORTAL [Logout](#)
Welcome, john toubia

Welcome Administration **Grid** BioPortal

Registry **Credentials** Resources Files Jobs

Credential Manager Portlet

The following credentials can be retrieved from myproxy.apac.edu.au.

Credential	Certificate	Status	Time left	
johnt	/C=AU/O=APACGrid/OU=SAPAC/CN=John Toubia	Active	1 hours 48 minutes 25 seconds	<input type="button" value="Deactivate"/>

Passphrase:

February 16, 2007

Appendix VI cont... - Bioportal Screen Shots - Job page

BIOPORTAL

Logout
Welcome, john.toubia

Welcome Administration Grid **BioPortal**

InterpreterClient

MrBayes 3.1.2

Program: MrBayes 3.1.2 job files

MrBayes 3.1.2 Inputs

MrBayes Command File:

Type: CPUs: Host:
PBS 1 hydra.sapac.edu.au

Action:

Job Submission Instructions

To start, please choose the [Program](#) you wish to use from the drop down list in the Navigation Bar.

Please enter or select the Command File you wish to run in the [Command File](#) panel. **Please note that the Command file that you select must contain 1 and only 1 Command Block.** For more details see [Command Block Creation](#) below.

If necessary, please also adjust the job [Type](#), number of [CPUs](#) and [Host](#) that you require in the appropriate panels.

When you have completed the above steps please select the "Submit Job" option in the [Action](#) panel and hit "Go" to run the job. If you wish to start over before you have submitted, select the "Reset Form" option and hit the "Go" button. To save or load any personal default settings, use the "Save as Default" and "Load Default" options.

Output can be retrieved by selecting the [files](#) tab in the Navigation Bar, expanding the relevant timestamped folder and selecting "output.zip" which will prompt a download manager.

To return to this Job Submission Portlet at any time, select the **job** tab in the Navigation Bar.

Command Block Creation

The Command block simply contains the commands as you would have given them from the command line, with the difference that each command line is ended with a semi-colon. Command Blocks are usually placed after the Data Block of a Nexus File.

Example MrBayes Command Block. *note: red font indicates compulsory requirements*

```
begin mrbayes;
set autoclose=yes nowarn=yes;
lset nst=6 rates=gamma;
mcmc ngen=10000 samplefreq=10 file=output;
sump filename=output burnin=250;
sumt filename=output burnin=250;
quit;
end;
```

For more information on MrBayes commands please hit the link and reference the [MrBayes online manual](#).

Example PAUP Command Block. *note: red font indicates compulsory requirements*

```
Begin paup ;
set autoclose=yes warntree=no warnreset=no;
log start file=output.log replace;
NJ;
LSCORE 1/ BASEFREQ=empirical TRATIO=estimate RATES=gamma SHAPE=estimate;
SET CRITERION=like;
LSET BASEFREQ=empirical TRATIO=previous RATES=gamma SHAPE=previous;
HSEARCH ADDSEQ=random NREPS=5 SWAP=TBR;
savetrees file=output.tre replace;
quit;
end;
```

For more information on PAUP commands please hit the link and download the [PAUP Command Reference Document](#).

For information about BEAST XML file format, please hit the link and reference the [BEAST online XML format documentation](#).

February 16, 2007

Appendix VI cont... - Bioportal Screen Shots - Files page

BIOPORTAL

[Logout](#)
Welcome, john.toubia

Welcome Administration Grid **BioPortal**

InterpreterClient

MrBayes 3.1.2

Program: MrBayes 3.1.2 | job | files

JavaScript Tree Menu

MrBayes 3.1.2 Files

- 070201_095625
- 070201_095941
- 070201_102227
- 070201_112421
- 070201_114249
- 070201_115336
- 070201_135006
- 070201_135403
- 070202_134202
- 070202_135606
- 070202_140017
- 070202_144835
 - script.pbs
 - input
 - data
 - input.tgz
 - .properties
 - stdout
 - output.zip
- 070202_145352
- 070205_161710
- 070205_163321
- 070206_114245
- 070206_120156
- 070207_121321
- 070207_132543
- 070208_105805
- 070209_103303
- 070209_163543
- 070214_140718
- 070215_092414

February 16, 2007

References:

- Wehrens, O. 2006, *Gridsphere – A Guide*, [Online] viewed 8 February 2007, [Available] <http://www.gridisphere.org/gridisphere/html/gridsphereguide.pdf>
- Novotny, J. 2004, Developing grid portlets using the GridSphere Portal framework, [Online], viewed 10 February 2007, [Available], <http://www-128.ibm.com/developerworks/grid/library/gr-portlets/>
- Grid Portlets User's Guide, 2006, GridSphere Portal Framework, [Online], viewed 10 February 2007, [Available], <http://www.gridisphere.org/gridisphere/docs/gridportlets/docbook/UserGuide/UserGuide.html>
- Huelsenbeck, J.P. and Ronquist, F. 2003, MrBayes: A program for the Bayesian inference of phylogeny, [Online], viewed 10 February 2007, [Available], <http://golab.unl.edu/teaching/SBseminar/manual.pdf>
- Swofford, D.L. 2003, PAUP*, [Online], viewed 10 February 2007, [Available], <http://paup.csit.fsu.edu/about.html>
- Drummond, A. and Rambaut, A. 2006, BEAST version 1.4, [Online], viewed 10 February 2007, [Available], http://evolve.zoo.ox.ac.uk/beast/help/Main_Page
- Drummond, A. and Rambaut, A. 2006, Introduction to XML format, [Online], viewed 10 February 2007, [Available], http://evolve.zoo.ox.ac.uk/beast/help/Introduction_to_XML_format
- Drummond, A. and Rambaut, A. 2006, BEAUti ReadMe, [Online], viewed 10 February 2007, [Available], http://evolve.zoo.ox.ac.uk/beast/help/BEAUti_ReadMe
- Maddison, D.R., Swofford, D.L. and Maddison, W.P. 1997, NEXUS: an extensible file format for systematic information. *Syst Biol.* 1997 Dec;46(4):590-621.