

Software

Table of contents

1 ProteinParser.....	2
2 Ant backup script.....	2
3 openMVC.....	3

1. ProteinParser

ProteinParser is a novel command line based software tool for automatically producing reports on the acid frequency within protein alignments. The tool is written in Java, a platform neutral language, and requires only the Java 2 runtime environment for execution. A [ClustalW file](#) is taken as input, along with a tolerance value between 0-100. The tolerance value dictates how often a acid must be present, at a particular position, within a number of proteins before its frequency is reported by the tool. You can download the software [here](#). The associated [Javadocs](#) are also available. The [source code](#) is also made available under the GNU General Public License Version 2.

NEW: ProteinParser 1.1 now generates FASTA output.

You must have a Java JRE installed to run the ProteinParser. The program has been tested exclusively with JRE 5, see java -version output below.

```
java version "1.5.0"  
Java(TM) 2 Runtime Environment, Standard Edition (build 1.5.0-b64)  
Java HotSpot(TM) Client VM (build 1.5.0-b64, mixed mode)
```

To run protein parser perform the following.

- Save the [jar file](#) to a directory.
- Run the program in DOS/Bash by typing the following. (all on one line)

```
java -classpath ProteinParser.jar ProteinParser  
/home/ronan/autoedit/protein2.txt consensus_name 50
```

note: /home/ronan/autoedit/protein2.txt should be the path to your ClustalW file and 50 is the tolerance i've set which can be 0-100

note: As of version 1.1 there is an extra parameter required to run ProteinParser. This parameter names the generated FASTA output.

2. Ant backup script

I've written a nice backup script for [ant](#) which can be used to backup your work. You can download it [here](#). It does the following.

- 1. Zips up some selected directories.
- 2. Backups the selected zip files to a local directory
- 3. Emails the smaller zip files to a mail account
- 4. SCP's the zip files to another server

Of course you can choose what the script does by selecting different targets. To automate the

running of the ant script you could add something like this to your cron jobs on linux.

```
00 19 * * 1-5 cd /home/username/eclipse/workspace/Backup/;ant > /home/username/eclipse/workspace/Backup/cron_backup.log
```

Note you will need to download some jars to get the SCP and email targets working. The jars must be on your classpath before running ant. The SCP jar is available from JCraft [here](#). The JavaMail and JavaBeans(TM) Activation Framework jars are available from SUN.

3. openMVC

openMVC was my final year undergraduate project undertaken at [Dublin Institute of Technology, Kevin Street, Dublin](#) and represents over nine months of research and development work. Although I provide all the code and documentation as is, interested parties are more than welcome to drop me an email to discuss its finer points. openMVC is GPL'ed so all GPL license rules apply.

This project is an investigation into the advantages and disadvantages of a development paradigm that combines many existing standardised technologies. These technologies are used to build a framework that consists of finely grained, loosely bound components which provide for excellent maintainability. The proposed framework is an evolution of the Model-View-Controller (MVC) pattern.

The lack of standardised approaches in the development of web-based systems is an ongoing issue for the developers of commercial software. To address this issue we propose a hybrid development framework for web-based solutions that combines much of the best attributes of existing frameworks but utilises open, standardised W3C technologies where possible. This framework called openMVC is an evolution of the Model-View-Controller (MVC) pattern. An implementation of openMVC has been built over a 5-tier architecture using Java and .NET.

[openMVC Documentation](#)

[openMVC C# Code](#)

[openMVC Java Code](#)