Project Plan

**The Problem**

Inferring phylogenetic trees is clearly important as a means to further understanding in which ways Life developed, but they can also be informative as to how parts of it work. To understand the possible functions and behaviours of current sets of genes, we may find it useful to know those of ancestral genes. For these reasons, it’s clearly important to investigate ways in which inference can be made more accurate. One of these ways is partitioning, which assumes different models of evolutionary development for for separate sets of sites in an alignment of gene sequences.

Over the next few months, we will attempt to better understand partitioning by experimenting with and comparing various ways of partitioning gene datasets. In particular, we wish to compare the speed and accuracy of three tree-generation algorithms – RaxML, PhyloBayes and ‘Tree Collection’, the last one having been created by Jeremy Levy. The first two use probability models to guess the phylogenies most representative of the observed data, the last uses distance to class genes into phylogenies. To compare the accuracy of the algorithms, the results from each will be subjected to a monophyly test. A monophyly test assesses the extent to which trees correctly categorise the species involved, e.g. making sure flies are with flies, rodents with rodents etc. By using these tests of speed and accuracy, not just will we have an idea of the best algorithm, but also of the way in which genes should be partitioned - according to distances, or likelihood of matching certain models? These models may simply be based on branch length or types of evolutionary change over time.

**Project Goals and their Implementation**

• Knowing basics of biology – definitions of DNA and genes, tree topologies, distances between genes.

• Read papers introducing molecular biology to computer scientists, relevant chapters of *Molecular Evolution* and *Inferring Phylogenies*.

• Reading about phylogenetic reconstruction methods (parsimony, maximum likelihood, Bayesian method)

• Trees as ways of modelling relationships between genes, relative usefulness of tree searches, distance methods, parsimony, Bayesian statistics.

• Increasing knowledge of the Python programming language, especially concerning lists, dictionaries, file use and pipelining.

*• Beginning Python,* resources on the internet.

• Understanding the methods of the PhyloBayes, Tree Collection and RaxML programs.

• Get access to CS cluster, code repository. Read parts of the PhyloBayes, RaxML manuals and download software.

• Decide on ways to try partitioning subsets of gene sequences.

• Discuss with Christophe, Jeremy and Leon.

• Write Python program that uses pipelining to run phylogeny construction programs on 40 subsets of a large dataset. (See more specific outline below)

• Evaluate and compare the speed and accuracy of each program. This will include use of the monophyly tests, maybe algorithmic analysis as well.

• Come to a conclusion on partitioning methods, and write a report.

Vanilla RaxML - takes a PHYLIP alignment of sequences -> produces an ML tree.

PartitionFinder - takes a PHYLIP alignment and a file specifying the blocks of sites to be grouped together -> produces a best-fit partition scheme according to the standard WAG + I +G model

RaxML w/ PF partitions - takes PHYLIP alignment, and its PF scheme -> produces ML tree for the alignment and one for each set in the scheme.

MtPAN - takes NEXUS alignment -> produces clusters (or partition schemes) of different sizes

RaxML w/ MtPAN partitions -> takes PHYLIP alignment and MtPAN partitions - produces ML trees

PhyloBayes - takes PHYLIP alignment -> produces two ‘chains’ of trees until the properties/statistics of each chain converges within a defined limit, and from this a majority consensus tree is made.

Tree-Collection - takes the distance-variance matrix for a set of sequences -> produces tree