**Data Management Plan (2 pg)**

The sustainability of protein function prediction resources is very much contingent on the hardware and servers on which it is stored and run. The existing infrastructure at Yale and Royal Holloway University of London has served investigators well for over a decade, but we aim to improve the current setup by making it reliable and robust for supporting all the proposed tools as well as more accessible to the scientific community. To this end, we intend to make use of new technologies such as cloud computing.

Specifically, we intend to use Amazon Web Services (AWS) for distributing most of the newly developed resources, and intend to make use of the Amazon Elastic Compute Cloud EC2 (processing) and S3 (storage). AWS EC2 enables flexible, resizable online resources, and would serve as a sensible means for distributing the contents and services of the protein function prediction resources, as it provides high performance computing, processing resources which adjust to user demand, reliability, and greater security.

Here, we summarize various components of the proposed resources, along with the means by which we intend to disseminate each:

i) Source code, as used in constructing the various software components will be made available through open access repositories, such as sourceforge, github, or google code.

ii) Web-services (LoREgic, AP-Tool): each of the servers would be encapsulated and made available as a virtual machines (see below for a description and the advantages of virtual machines), which may be downloaded from our servers, and then stored locally by the user, or uploaded by the user to AWS EC2.

iii) Databases: the regulatory network dataset will be bundled up into a single virtual machine, and distributed in a similar manner. We intend to move this dataset to AWS S3 for storage, and would periodically (about once a month) make backups of this data locally.

As mentioned, the various servers, as well as the large dynamic datasets, would each be converted into a separate virtual machine (VM). A VM encapsulates an entire piece of software (even as large and complex as an entire operating system), and may easily be packaged up for easy storage and distribution. A VM behaves like an autonomous computer within a real computer. One primary advantage of using VMs is that they may be run on many different systems (such as Windows, Linux, or OSX). Upon execution, a VM has the appearance of booting a new computer and operating system (a process called virtualization), and it is this process which precludes the inconvenience and needed time of having to install and configure software on the host’s operating system, thereby enabling developers (in this case, us) to build autonomous, customized, turnkey, ready-to-go execution environments with a complete operating system and all required software packages, libraries, and data files packaged into a single VM. These properties make VMs suitable as a vehicle for distributing the resources through cloud-based systems making them easily and broadly available for the scientific community. Users would run the VMs locally, and for large-scale deployment, they may be exported to AWS EC2. This encapsulation strategy also contributes to the sustainability of the resources themselves.

AP to add resource development for the phenotype prediction tools