Projects: a timeline

- Today: basic info
- Thu, Feb 19: **Project kick-off**
- Fri, Feb 20: **Present project plan & diaries**
- Tue, Mar 3: **Exchange of experiences**
- Mon, Mar 9: **Seminar on poster preparations**
- Fri, Mar 20: **Poster session**

1 month

Organising the project

- Read W Stafford Noble's paper!
  - PLoS Comp Biol, 2009
- Maintain an individual online project diary!

OPEN ACCESS Freely available online

PLoS COMPUTATIONAL BIOLOGY

Education

A Quick Guide to Organizing Computational Biology Projects

William Stafford Noble[1,2]*

1 Department of Genome Sciences, School of Medicine, University of Washington, Seattle, Washington, United States of America, 2 Department of Computer Science and Engineering, University of Washington, Seattle, Washington, United States of America

Introduction

Most bioinformatics coursework focuses on algorithms, with perhaps some components devoted to learning programming skills and learning how to use existing bioinformatics software. Unfortunately, for students who are preparing for a research career, this type of curriculum fails to address many of the day-to-day organizational challenges associated with performing computational experiments. In practice, the principles behind organizing and documenting computational experiments are often learned on the fly, and this learning is strongly influenced by personal predilections as well as by chance interactions with collaborators or colleagues.

The purpose of this article is to describe one good strategy for carrying out computational experiments. I will not describe profound issues such as how to formulate hypotheses, design experiments, or draw conclusions. Rather, I will focus on relatively mundane issues such as organizing files and directories and documenting progress. These issues are important because poor organizational choices can lead to significantly slower research progress. I do not claim that the strategies I outline here are optimal. These are simply the principles and practices that I have developed over 12 years of bioinformatics research, augmented with various suggestions from other researchers with whom I have discussed these issues.

Principles

The core guiding principle is simple: Someone unfamiliar with your project should be able to look at your computer files and understand in detail what you did and why. This "someone" could be any of a variety of people: someone who read your published article and wants to try to reproduce your work, a collaborator who wants to understand the details of your experiments, a future student working in your lab who wants to extend your work after you have moved on to a new job, your research advisor, who may be interested in understanding your work or who may be evaluating your research skills. Most commonly, however, that "someone" is you. A few months from now, you may not remember what you were up to when you created a particular set of files, or you may not remember what conclusions you drew. You will either have to then spend time reconstructing your previous experiments or lose whatever insights you gained from those experiments.

This leads to the second principle, which is actually more like a version of Murphy's Law: Everything you do, you will probably have to do over again. Inevitably, you will discover some flaw in your initial preparation of the data being analyzed, or you will get access to new data, or you will decide that your parameterization of a particular model was not broad enough. This means that the experiment you did last week, or even the set of experiments you've been working on over the past month, will probably need to be redone. If you have organized and documented your work clearly, then repeating the experiment with the new data or the new parameterization will be much, much easier.

To see how these two principles are applied in practice, let's begin by considering the organization of directories and files with respect to a particular project.

File and Directory Organization

When you begin a new project, you will need to decide upon some organizational structure for the relevant directories. It is generally a good idea to store all of the files relevant to one project under a common root directory. The exception to this rule is source code or scripts that are used in multiple projects. Each such program might have a project directory of its own.

Within a given project, I use a top-level organization that is logical, with chronological organization at the next level, and logical organization below that. A sample project, called `msms`, is shown in Figure 1. At the root of most of my projects, I have a `data` directory for storing fixed data sets, a `results` directory for tracking computational experiments peformed on that data, a `doc` directory with one subdirectory per manuscript, and directories such as `src` for source code and `bin` for compiled binaries or scripts.

Within the `data` and `results` directories, it is often tempting to apply a similar, logical organization. For example, you may have two or three data sets against which you plan to benchmark your algorithms, so you could create one directory for each of them under `data`. In my experience, this approach is risky, because the logical structure of your final set of experiments may look drastically different from the form you initially designed. This is particularly true under the `results` directory, where you may not even know in advance what kinds of experiments you will need to perform. If you try to give your directories logical names, you may end up with a very long list of directories with names that, six months from now, you no longer know how to interpret.

Instead, I have found that organizing my `data` and `results` directories chronologically makes the most sense. Indeed,

Citation: Noble WS (2009) A Quick Guide to Organizing Computational Biology Projects. PLoS Comput Biol 5(7): e1000424. doi:10.1371/journal.pcbi.1000424

Editor: Fran Lewitter, Whitehead Institute, United States of America

Published July 31, 2009

Funding: The author received no specific funding for writing this article.

Competing Interests: The author has declared that no competing interests exist.

* E-mail: william-noble@u.washington.edu

Learning outcomes

From t

This is
studen

1. **des**
   inv
2. **expl**
   from h
3. **choose**
   biolog
4. **apply** t
   experiments.
5. **interpret** the results of these analyses in a biologically or medical relevant context.
6. **reflect** over the choice of methods and tools and how it influences the outcome of the analyses

This goal is examined through the project work, where we in the poster presentation or in the project diary should find evidence for any of the following levels of understanding. The student:

E: gives a biological explanation of the results in the project work

C: describes the limitations of the projects in terms of its insight into the project's biological or medical question(s)

A: suggests relevant improvements of experimental or data analysis procedures to better

This goal is examined through the project work, where we in the poster presentation or in the project diary should find evidence for any of the following levels of understanding. The student:

E: gives rational to why the particular tools were selected

C: discusses realistic and relevant means to improve the selected toolchain

A: implemented means to improve the selected toolchain in a relevant way

Grading

- Goal 5: F, E, C, A
- Goal 6: F, E, C, A

		Goal 5			
		A	C	E	F
Grade of Goal 6	A	A	B	C	F
	C	B	C	D	F
	E	C	D	E	F
	F	F	F	F	F

Groups

Grupp 1 OE
Enrichetta Mileti
Marco Salvatore
Yunzhang Wang

Grupp 2 LK
Bo Zhang
Hoi Yi Stephanie Yau
Mikael Falk

Grupp 3 OE
Matilda Berkell
Matthew The
Sauvagya Manna

Grupp 4 LA
Hanna van Ooijen
Mark Högqvist
Ronnie Rodrigues Pereira

Grupp 5 LA
Guillermo Carrasco
Sofia Bergström
Yim Wing Chow

Grupp 6 LK
Angeliki Maraki
Anandashankar Anil
James Ericson