

The following transcript is part of an ISCB educational discussion thread

**Deanne Taylor PhD**  
Department of Biostatistics  
Harvard School of Public Health  
655 Huntington Avenue  
Boston, MA 02115  
[dtaylor@hsph.harvard.edu](mailto:dtaylor@hsph.harvard.edu)

### **Background**

I work at Harvard but I also manage the continuing education Masters Degree program in Bioinformatics at Brandeis University. I have just completely re-designed the Brandeis Bioinformatics Graduate Program from scratch, so I've been thinking about this very kind of question. For the redesign, we used the combination of the "summit" that John Wooley had over in UCSD back in 2005 I think it was, plus input from industry bioinformatics directors and our own faculty, as well as my own feelings about what bioinformaticists should know, plus talking to colleagues. I think the work I did pretty much helped sum up for me what people thought would be important.

### **Skills**

The central-most skills I believe that a bioinformatician needs can be summed up by "biotechnology". Basically, a thorough physical and biological understanding of genomics and biotechnology, then a heavy dose of statistics and probability, and a strong understanding of basic compsci, including a scripting language, a sci/comp language like C, C++ or derivatives like Matlab, familiarity with the common methods of large-scale data and sequence analysis currently used in the field, and in a more subtle way -- how to manipulate objects and common data structures (important across the board!) and familiarity with information technology subjects if not expertise.

I think the central core of any bioinformatics education should rely on the area encompassed at the junction of biology, biotechnology and IT. A bioinformaticist can choose to act in a service (maintenance/etc) capacity, but I think they should also be ready to step up and offer solutions to data analysis problems that are not immediately apparent to classically-trained biologists and computer scientists. Therefore, they're in a place I think where biophysicists still live -- one has to be able to cross the line between pure IT, pure software development, and biological questions.

The industry leaders we interviewed (Novartis, Wyeth, etc.) told us they did not want bioinformaticists who were pure programmers with a hint of biology -- they wanted biotech/biology-heavy technologists with good strong IT backgrounds that did not eclipse their biological knowledge or technological skills in biology. Part of our biology/biotech-heavy education reflects that need as industry as well as academia hire our graduates.

### **Example curriculum.**

Our new curriculum at Brandeis consists of four */prerequisite/* courses or equivalents -- Gen/Org/Biochem, Mol Cell Bio, Intro Stats & Prob, and Intro to Programming (including data struct).

Then, all students enter into the six core courses of the 36-credit Program (in any order except #3 and #4 have to be taken in series):

1. [Macromolecular Biophysics](#)
2. [Biological Sequence Analysis](#)
3. [Statistical Programming and Data Analysis](#)

4. Data Mining and Modeling
5. Genomes and Genetics
6. Protein Sequence, Structure and Function

Our bioinformatics electives consist of the following:

7. Management, Modeling and Warehousing of Biological Data (essentially a relational database course with specifics applied to biological data and master databases)
8. Design and Analysis of Microarray Experiments
9. Python Programming for Bioinformatics
10. Perl Programming for Bioinformatics (currently satisfied by advanced Perl courses in the Software Engineering program)
11. Molecular Biology Laboratory (intensive hands-on laboratory course)
12. Drug Discovery and Development
13. Proteomics: Analytical and Computational Principles
14. Computational Systems Biology
14. Advanced Topics in Computational Biology
15. Thesis Credit (2009 possibly)
16. New proposed elective in Text mining and ontological analysis (2008 possibly)

There are also options for students to take additional electives in the other solidly-technical Rabb School programs, such as Project Management, Software Engineering, etc.

### **Saras Saraswathi**

MS in CS and in Bioinformatics ( ODU,NTU)

PhD student at BCB, IAState, USA

Chair - ISCB-SC-RSG

Member, ISCB-Student Council

### **Skills**

ONE of the MOST important subjects is of course BIOLOGY along with statistics, math, software engineering, relational databases. I would recommend more than basic knowledge in Biology. Know as much as you can.

### **Kevin Karplus**

Professor of Biomolecular Engineering,

University of California, Santa Cruz

Undergraduate and Graduate Director, Bioinformatics

Board of Directors & Chair of Education Committee, ISCB

karplus@soe.ucsc.edu

<http://www.soe.ucsc.edu/~karplus>

I like to separate the field into tool users and tool builders. There need to be a lot more tool users than tool builders, but the skill sets needed are different.

Both need to have a fairly solid base in biology and statistics, particularly knowledge of the types of data they will be analyzing and how the experiments that produce the data can go wrong. They also need to understand what sorts of questions may be answered with the data.

The tool builders need more software engineering, knowledge of algorithms, and programming skill. The tool users may need more biology, more understanding of databases, but less programming skill.

I think we need to train almost every biologist to be an effective tool user, but for the next 20 years there will probably be a market for effective tool users to work with research biologist who have not learned the tools themselves.

Our program at UCSC focuses more on the tool builders than the tool users. We've found that it is easier to add knowledge of biology to people who are already competent programmers than to add programming skills to people who are already competent biologists. I think this has to do mainly with the difficulty of teaching programming---I don't know anyone who can really teach someone to program well. The good programmers seem to arise more from selection processes than training processes.

**Prof dr Antoine H.C. van Kampen**

Scientific Director Netherlands Bioinformatics Centre ([www.nbic.nl](http://www.nbic.nl))

Bioinformatics Laboratory, Academic Medical Center (AMC)

Biosystems Data Analysis, University of Amsterdam

I share most of the points of view of Deanne. But have few questions/remarks:

- What is the duration of the master programme, 2 years? Does this give sufficient time to acquire all these skills? One difficulty i'm experiencing is that you cannot expect students to become sufficiently good in statistics in only 1 course (of a few days/weeks). Similar problem for becoming a good software/database developer. And what about the time left if we would focus too much on BIOLOGY as stated by Saraswathi?

- What is the background of the students that enrol in your masters program? Informatics? Biology? How do you deal with different backgrounds?

- Within our institute there was discussion whether we should train students to become 'hard core bioinformaticians' who are able to develop their own algorithms, statistics, software, databases, etc or whether we should focus on training biologists to use available software/databases/etc. These two types of bioinformaticians would require completely different programmes. This is in line with point the Kevin is making. I think we should focus on the tool builders.

- Another question concerns the teaching material. What do you use? It is my experience that there is especially a lack of good, worked out, biological cases demonstrate the use of bioinformatics.

- I do not have the same experience as Kevin w.r.t. the fact that it is more easier to train biologist to develop software (in contrast to training a informaticians in biology). With the people in my lab i have observed that biologists really have a hard time doing core bioinformatics. However, on the other hand biology is much harder to pick up since this requires so much background knowledge. But how much biology should a bioinformatician really know. I agree, there should be a strong basis but in general a bioinformatician will collaborate with a biologist and both together have the expertise to perform a project successfully.

One serious problem that we encounter here in the Netherlands is the very low number of students (sometimes less than five) that enroll in the master programmes of dutch universities. How is that situation in the US? Or, for those who are following the discussion, in other countries?

**Panayiotis (Takis) Benos, Ph.D.**

Assistant Professor

Department of Computational Biology, SOM and

Department of Human Genetics, GSPH | Tel. : +1 412 648 3315  
University of Pittsburgh, | Fax : +1 412 648 3163  
3501 Fifth Ave., Suite 3064 BST3 | e-mail: [benos@pitt.edu](mailto:benos@pitt.edu)  
Pittsburgh, PA 15260, U.S.A. | URL: <http://athena.hgen.pitt.edu/benos/>

Although i share some views with the people participated in the discussion previously, i would like to mention that i do not see computational biology simply as "tool building" or "tool sing". I rather prefer to see it as an integral part of biology, only instead of knowing how to do southern, western, PCR, etc to answer your **\*biological\*** question, you know how to do Gibbs sampling, forward-backward, MCMC and (structure) homology modeling.

Now what training does one need for that? Undoubtedly, quantitative skills (math, stat, physics, CS, engineering, etc) are more important than biology knowledge, at least in the beginning. Thus, this is where the student selection and training should focus (see below). However, good biological knowledge is essential for someone to excel in the field. Therefore, courses should pay attention to that too.

And how do you do all that in one or two years of coursework? I think the answer is that you need to start with people that come from interdisciplinary background. An ideal student for example, will be someone that has a double major, or major in quantitative science and minor in biology (for example). In this way one does not have to teach these students the ABC of statistics, but rather start directly with Markov models, HMMs, etc.

### **Kevin Karplus**

Professor of Biomolecular Engineering,  
University of California, Santa Cruz  
Undergraduate and Graduate Director, Bioinformatics  
Board of Directors & Chair of Education Committee, ISCB  
[karplus@soe.ucsc.edu](mailto:karplus@soe.ucsc.edu)  
<http://www.soe.ucsc.edu/~karplus>

I can't speak for Deanne, but the Master's program at UCSC is about 2 years, which is fairly typical for an MS in the US. We assume that students have already had some background.

### **Example curriculum**

The graduation requirements can be found at  
<http://www.soe.ucsc.edu/programs/binf/graduate/requirements.html>

#### *Summary of requirements (included van Antoine van Kampen)*

The program in bioinformatics is a multidisciplinary program sponsored by the Department of Biomolecular Engineering. The program offers B.S., M.S., and Ph.D. degrees in bioinformatics and a minor in bioinformatics. Bioinformatics combines mathematics, science, and engineering to explore and understand biological data from high-throughput experiments, such as genome sequencing, gene expression chips, and proteomics experiments. The program builds upon the research and academic strengths of the faculty in the Center for Biomolecular Science and Engineering.

#### Course requirements

Both M.S. and Ph. D. students must complete nine 5-credit courses (seven core courses and two electives; see below) and a 3-credit research and teaching course. In addition, M.S. students must complete 4 seminar credits, while Ph. D. students must complete 8 seminar credits. Also, M.S. students must complete two (1-credit or 2-credit) research project courses (such as BME 220L, BME 230L, BME 297F, or BME297), and Ph.D. students must complete three research lab rotations (BME 296) with different supervisors.

Core courses (5-credit)—7 are required:

- Biomolecular Engineering 100+L, Introduction to Bioinformatics + Laboratory
- Biomolecular Engineering 220, Protein Bioinformatics
- Biomolecular Engineering 230, Computational Genomics
- Biomolecular Engineering 80G, Bioethics in the Twenty-First Century: Science, Business, and Society or Philosophy 245 Brave New World: Ethical Issues in Genetics

One graduate-level course, approved by the faculty, in each of the following three areas:

- Statistics (Engineering 206 recommended)
- Biology (Biology 200B recommended)
- Chemistry (Chemistry 200B recommended)

Electives (5-credit)—2 are required:

The electives should be graduate-level courses selected with approval of the faculty to ensure a coherent, balanced program. For M.S. students, 5 units of independent research (297) or thesis research (299) may count as electives toward the degree requirements upon approval of the faculty. For Ph.D. students, independent or thesis research cannot be counted as electives.

Students must choose their electives with faculty guidance and approval to balance their preparation and make up for deficiencies in background areas. In addition to fulfilling background needs, students may choose to emphasize one of the breadth areas: molecular biology, biochemistry, statistics, computational biology, genetics, computer science, computer engineering, applied mathematics, cell biology, and computer graphics/visualization. Alternatively, students may take a cross-sampling of the electives to achieve a broad knowledge base.

Other curriculum requirements:

- Biomolecular Engineering 200, Research and Teaching in Bioinformatics (3 credits)
- Seminars:  
M.S. students: a minimum of two seminar courses, including at least one quarter of the 2-credit BME 280, Bioinformatics Seminar (formerly CMPE 280B).  
Ph.D. students: a minimum of four seminar courses, including at least two quarters of the 2-credit BME 280, Bioinformatics Seminar (formerly CMPE 280B).
- Research experience:  
M.S. students: a minimum of two research project courses (BME 220L, BME 230L, BME 297F, or BME 297)  
Ph.D. students: three quarters of laboratory rotations (BME 296), generally completed within the first 12 months; at least one rotation must be with a faculty supervisor who does wet-lab research, though the rotation project may be purely computational.

Thesis and dissertation requirements

In addition to completing the course requirements, students must fulfill the following thesis or dissertation requirements:

M.S. students must submit a written thesis proposal to a faculty member by the end of the fourth academic quarter. If the faculty member accepts the proposal, he or she will become the student's advisor and will be in charge of supervising the writing of the master's thesis. When the thesis is completed, it will be submitted to and must be accepted by a faculty review committee consisting of the thesis advisor and at least two additional readers. The committee must include a School of Engineering faculty member, may include participants from the Division of Physical & Biological Sciences and from industry as appropriate, and must be approved by the bioinformatics program director. Students are required to present the thesis project in a public seminar.

Ph.D. students are encouraged to select a faculty research advisor by the beginning of the second year and must select one by the end of the second year. A written dissertation proposal is required before the end of the third year. A qualifying committee is then formed, consisting of the advisor and three additional members approved by the bioinformatics program director and the campus graduate dean. The student must

submit his or her written dissertation proposal to all members of the committee and to the graduate assistant one month in advance of the examination. The dissertation proposal is publicly and formally presented in an oral qualifying examination given by the qualifying committee.

Ph.D. candidates will submit the completed dissertation to a reading committee at least one month prior to the dissertation defense. The reading committee, formed upon advancement to candidacy, consists of the dissertation supervisor and two readers appointed by the bioinformatics program director upon the recommendation of the dissertation supervisor. The candidate will present his or her research in a public seminar. The seminar will be followed by a defense of the dissertation to the reading committee and attending faculty, who will then decide whether the dissertation is acceptable or requires revision.

We do not have an official core set of undergraduate courses that students must have had, but we don't generally admit students unless they have had some bio, some programming, and some statistics, plus a solid background in at least one of the relevant fields. Students with a strong background in one field, but none in others are encouraged to take community college courses (which are very cheap) to fill in some of the deficit before applying.

Our PhD program has essentially the same coursework requirements as the master's but typically takes 5-7 years, with the extra time spent on the PhD thesis.

We do require all PhD students to do 3 lab rotations, at least one of which is with a supervisor who runs a wet lab (most often someone outside the department, though both Haussler and Lowe have wet labs, and we have 3 newer faculty just starting wet labs).

**Deanne Taylor PhD**  
Department of Biostatistics  
Harvard School of Public Health  
655 Huntington Avenue  
Boston, MA 02115  
[dtaylor@hsph.harvard.edu](mailto:dtaylor@hsph.harvard.edu)

The Masters degree program is typically done part-time. It is a two-year equivalent. How soon a student earns the Masters degree is usually up to them (three years is typical).

The classes are 10 weeks long. Every course actually has some statistics in it -- either completely in the cases of the prereq in statistics, or the two core courses: Statistical programming, Data Mining. Or built into the course material, which makes up just about every other course in the Program.

### Skills

Our goal is to suffuse the courses not only with their individualized content (per subject) but also the unifying skills -- thus, every class -- even the biology-heavy classes, has both some aspect of applied statistics and computer programming, even if it's several scripting exercises at the minimum. So, while many of our subjects are not titled or described as computer programming or statistics courses, we have enough applied science within them to make it necessary for them to use these skills to pass the courses.

The idea we use (as above) is to teach that biology and biotech heavily mixed with stats and compsci. So, for instance, with the human genome project on the topic of computation, we spend time discussing the history of the computational applications to the Project, including applications such as Phred, Phrap, Gigassembler, RepeatMasker, and Genscan. Algorithms are discussed both from a higher point of view as they tie into the biology, as well as "this is how the algorithms are executed". Same goes for BLAST and BLAT -- we study the statistical methods

behind BLAST, and how the program executes (roughly in the case of BLAST). So, there's a lot of biology there, as well as compsci. In this small example, students use these tools, but they also are required to understand how they work, from a deeper point of view than a typical user.

What is the background of the students that enroll in your masters program? Informatics? Biology? How do you deal with different backgrounds?

First, I have to say this; although students may come from vastly different backgrounds the subject matters we teach are generally unknown to most of them -- whether they come from biology or they come from informatics. The only way someone could come into the program with a real advantage is if they were already bioinformaticians, we don't get many of those.

We deal with evening the playing field for different backgrounds by requiring those four prereqs mentioned in the earlier letter, before they even start the Masters program, and then we ALSO help the students choose their electives. For instance, I give all incoming pure-biology people the Introduction to Bioinformatics Scripting and Programming course, which gives them an introduction to Perl programming (ala O'Reilly) as well as an introduction to basic Java so they gain concepts like objects and classes, etc. It's only 12 weeks for this particular course, but it's a high-content class that I've developed to get them thinking like a programmer as soon as possible. Additionally, most students have to take an introduction to statistics course -- which also introduces them to mathematics such as linear algebra. The programming and stats are used all through the core courses, so they'll keep on applying them so they don't lose the knowledge. If they need additional programming, we have full-semester Perl, Java, or C++ courses we can enroll them in here at Brandeis as part of our Software Engineering courses.

The Informatics people come in and take the two introduction courses to biology. Like statistics, there is biology all through all the courses, so they are reinforced through the Program as well.

> Within our institute there was discussion whether we should train students to become 'hard core bioinformaticians' who are able to develop their own algorithms, statistics, software, databases, etc or whether we should focus on training biologists to use available software/databases/etc. These two types of bioinformaticians would require completely different programmes. This is in line with point the Kevin is making. I think we should focus on the tool builders.

We have decided to turn informaticists who are (traditional) tool-builders already into bioinformaticists and we are also seeking to turn the biologists into tool-builders too. They'll still have a slightly different track. We sit with each student and work out the best track for them.

We do hope they become in some part tool-builders, or we try to make them all tool-builders but an integral part of tool-building for us is understanding the genomics/biology/stats/etc. One of the directors of bioinformatics in our meeting said something like, "If the tool-builder does not deeply understand the genomics/biology, they're the same as a non-bioinformatics tool builder, at which case one does not need a bioinformatician. I have lots of people who can build tools, I have very few people who can build a tool to do a new analysis type that empowers the biology".

Another important thing we asked ourselves is -- what kinds of tools does the community need? Granted, there are plenty of niches that still need to be filled -- but in some niches we don't NEED more tools. The more forward-looking question is, where will bioinformatics be in six years? Because that's the kinds of tool-builders we need to start training now. Our program is designed a bit around "the next six years" concept. There are some proposed electives I haven't mentioned that will also fill in those gaps.

>Another question concerns the teaching material. What do you use? It is my experience that there is especially a lack of good, worked out, biological cases demonstrate the use of bioinformatics.

For "cases" there are no real cases that are well-worked out, because biology is messy. Then again, there are plenty of publications that one can refer to and recapitulate their analyses. It's easy from public data, anyway. For instance in microarray analysis, there are no real biological cases that can be tested for absolute truth unless they're artificial, such as spike-in gene analysis, or if they're somehow trending rather than truth, such as seeing certain pathways appear as a kind of validation since they were established already in other experiments. Tool-builders **\*should\*** understand the standards that are given, for instance the "affycomp" package in Bioconductor. In fact, Bioconductor is full of case analyses waiting to happen as they are all packages that were developed for a need.

The field is still in flux and the students have to learn what is the latest and the greatest. Lucky for us there are some methods that will have to do until something better comes along. The best gift you can give your students is a connection to the "community" consensus solutions and what their limitations are, so they can improve on them if they are inspired to.

For classwork, I use, for the most part, classical and current bioinformatics literature. For instance, when we start with BLAST, we read the original BLAST papers. We also use the Durbin book for biological sequence analysis, etc. I rarely use the books published as bioinformatics text books, though I've found there are certain texts I suggest -- for instance Terry Brown's Genomes 3 is very useful in teaching people about genomics in general as an introductory text or a concurrent text with the literature.

I write all my own homework exercises. Most of the learning in our classes are by doing. I use public data -- mostly Ensembl or Genbank (and the students better know how to use those databases by the end of the first course or they're in trouble). :)

>I do not have the same experience as Kevin w.r.t. the fact that it is more easier to train biologist to develop software (in contrast to training a informaticians in biology). With the people in my lab i have observed that biologists really have a hard time doing core bioinformatics.

Depends on how they are trained we do not design our courses to be strictly for informaticists who need to know biology, so both sides are almost coming in equally but from different directions in most of our courses. For instance, very few informaticians (ok, none) have read the Durbin book, and I guarantee that most biologists have not either we treat bioinformatics as a different but related field to both informatics and biology, so both sides are coming into new things.

> However, on the other hand biology is much harder to pick up since this requires so much background knowledge. But how much biology should a bioinformatician really know. I agree, there should be a strong basis but in general a bioinformatician will collaborate with a biologist and both together have the expertise to perform a project successfully.

I find that classically trained biologists can only "talk" so far into the genomics or biotechnology. For instance, I taught a classically trained biologists how to use R to analyse microarray chips. He did not understand the core biophysics of hybridization so it was difficult to describe stochastic noise to him, therefore the underlying issues with normalization were lost on him. Our bioinformaticians should know some physics as well as biology, as well as chemistry. They cannot rely on biologists to let them know what will work, at some point both will be stumbling in the dark.

Additionally, most biologists who are actively in research projects are in niche fields -- as they can get pretty deep. No bioinformaticist is going to know that much biology. The bioinformaticist however should understand the biology that is deeply associated with the biotechnology that is typically analysed, so this means genomics, transcription, translation, protein interaction data, protein structure, systems, signal transduction, post-translational modification, biophysical characteristics of macromolecules -- etc. The knowledge should be deeply in touch with the methods that are being used in the workplace.

> One serious problem that we encounter here in the Netherlands is the very low number of students (sometimes less than five) that enroll in the master programmes of dutch universities. How is that situation in the US? Or, for those who are following the discussion, in other countries?

Yes, we have small numbers of students. This is typical. The same goes for biophysics. It's a niche educational market -- we do get many people taking classes, however, that do not want a Masters degree, so there's a lot of room for education outside of the Program.

**Lisa Vawter, Ph.D.**  
Head, Bioanalysis  
Bioinformatics  
Sanofi-Aventis  
Bridgewater, NJ  
tel: 908-231-2285  
fax: 908-231-2285

I write from the perspective of a bioinformatician working in industry.

IT skills are good to have, but these are not generally the focus of what bioinformaticians in industry do. Increasingly, database and programming tasks are taken by people in specialized scientific computing departments. It is much more important for bioinformaticians to be good scientists, to understand formulating and testing hypotheses and interpreting data than it is for them to write code. Some programming skills are necessary, but these should not be the emphasis of a program that seeks to qualify students for employment in industry.

As you know, industry requires problem-solving skills, the ability to work with groups, the ability to communicate, the ability to infer solutions from partial data and under time pressure, and the attitude that if something that is less than exciting needs to be done, then that is exactly what will be done. Clear thought and attitude cannot be taught. Coding, to the extent that it is required, can be.

Most companies have departments dedicated to computing infrastructure and additionally have specialist scientific computing departments that function as that interface. If your students are too code-oriented and are educated only at a masters level, they are likely to be employed primarily as contractors writing code for bioinformaticians rather than as bioinformaticians.

**Prof dr Antoine H.C. van Kampen**  
Scientific Director Netherlands Bioinformatics Centre ([www.nbic.nl](http://www.nbic.nl))  
Bioinformatics Laboratory, Academic Medical Center (AMC)  
Biosystems Data Analysis, University of Amsterdam

Skills

One issue that was not yet (explicitly) raised concerns new developments such as GRID and e-science and maybe new informatics/IT developments such as service oriented architectures (SOA; e.g., web-services). How much of this should the bioinformatician know? Or should this be left to the informatician. It is my opinion that the bioinformatician plays a crucial role on the interface between informatics and life sciences and that he should also facilitate the translation of the new informatics technologies to daily life sciences (of bioinformatics) practice.

**Lakshmanan Iyer**  
lax.iyer@tufts.edu

You are right about the bioinformatician facilitating the translation of new informatics techniques to daily life sciences. An appropriate choice of a project in the curriculum should help the student get the necessary exposure to the emerging technologies. Of course, one project would not cover all topics. However, it would give the student an experience to learn about how to understand, translate and implement novel technologies. This would mean that as a future bioinformatician he/she would be able to collaborate productively with the research computing folks in their organization rather than watching the development from the side lines.

Finally, I believe that a choice of a good project in the curriculum goes a long way in the training. It could help develop all the skills that has been discussed in this thread, such as, asking the right question, problem solving, providing solutions etc...

**Fran Lewitter, Ph.D.**  
Director, Bioinformatics and Research Computing  
Whitehead Institute  
Nine Cambridge Center  
Cambridge, MA 02142-1479  
Tel: 617-258-5087  
Fax: 617-258-5578  
Email: [lewitter@wi.mit.edu](mailto:lewitter@wi.mit.edu)  
Web: <http://web.wi.mit.edu/bio>

I have read with much interest the discussion of what makes a good bioinformatician. I am the director of a bioinformatics core facility at Whitehead Institute for Biomedical Research. There are four full time bioinformaticians in the group - two bioinformatics scientists (Ph.D. level) and two bioinformatics analysts (masters level). We collaborate with the biological scientists in the twenty labs on numerous projects and co-author papers. In addition, our mission is to educate biologists in the area of bioinformatics. Everyone in my group is trained in both the biological sciences and computational biology.

In the last year I have conducted several job searches. It was interesting to see the skill sets of people applying for positions in my group. There were numerous applicants who had recently completed one of the many masters programs in Bioinformatics across the US and in the Boston area. What stood out most was the candidates' lack of experience. Most masters programs do not require any practical experience. Rather, it is mainly course work. Even if there is an internship associated with a degree program, the projects students work on are rather narrow. People list a long menu of applications they have used on their resumes but when interviewing them, they seem to have little clue about how the applications work. We have been lucky and, for the most part, have hired people who are fast learners and get up to speed rather quickly.

So, what makes a good bioinformatician for my environment? I look for bright people with good problem solving skills and with a passion for learning. A background in biological sciences and computational methods is also required.

What do we do for bioinformatics education for biologists? My philosophy is to, at a minimum, train biologists while they are here to at least get familiar with the vocabulary and concepts in bioinformatics. For example, these postdocs will go off to head up labs and will likely hire someone who can do bioinformatics for them. It's important for the future PIs to have some idea about the breadth and depth of bioinformatics although they themselves may not do it. That said, we also like to train people in the labs to do actual bioinformatics work on their own. We have taught a number of courses over the years - see <http://jura.wi.mit.edu/bio/education/index.html> - some being semester long but more recently we have taught 6 hour minicourses. These short courses have been popular and get people started in a particular area.

In the biology Ph.D. programs at MIT and other schools, computational biology courses are taught but often seem to be more geared towards specific computational biology research areas rather than being more applied. If we think about what Temple Smith said in his Keynote at ISMB, biologists of the future will be the ones applying computational tools to their research. We're not there yet, but the newer biology grad students and postdoc realize how important it is to learn to program and apply bioinformatics tools to their research.

I am interested in hearing from others about their experiences teaching biologists computational methods.

**Saras Saraswathi**

MS in CS and in Bioinformatics ( ODU,NTU)

PhD student at BCB, IAState, USA

Chair - ISCB-SC-RSG

Member, ISCB-Student Council

Yes. I agree with you on the matter of new informatics technologies. It is not so much as a matter of the latest technology but the sheer necessity to know the latest development.

It is imperative that we use new developments such as the GRID simply because the exponentially growing data base cannot be tackled without such measures.

Yes, we need to keep up with the technological advances so that we can reap the advantages which will help us solve the problems. Sometimes it is possible to think that without such technologies we could in fact become stranded in our pursuit of solutions to biological problems when it involves very large sets of data.

**Deanne Taylor PhD**

Department of Biostatistics

Harvard School of Public Health

655 Huntington Avenue

Boston, MA 02115

[dtaylor@hsph.harvard.edu](mailto:dtaylor@hsph.harvard.edu)

I agree, experience is very important, and it's been one of the major sticking points in our Program. One method would be a Masters thesis -- but that might not be OK with traditional faculty as it will appear our class-based Masters program is trying to usurp the territory of the research-based Masters. Or so I'm told.

Even without a thesis, there should be the ability to do intenships -- one of the limitations we've had in getting people up to speed in experience is the lack of research funding in and out of

Brandeis there is no funding and even if we were offered money, we are "adjuncts", we cannot have University standing, so we cannot have Brandeis support our grants that would give our students some hands-on experience with funding.

We've supported independent internships but only in helping students find them -- no other way at this point until we can get a Masters thesis track through and approved.

I wish there was a better way of getting bioinformatics research opportunities out to students who could benefit from them at the */Masters/* level (PhD levels are often covered by usual departmental stipends and/or teaching). Anyway, there are some instances of students in our classes who are */not/* working, and would have benefited from an internship.

Part of it is trying to get research groups to build in internship-level research into their budgets -- not all internships should have to come from industry. Doesn't seem to be in people's models.