Implementation or algorithm?

Recently the value of bioinformatics has been questioned. The value can be proven but are there enough qualified and professionally trained people who know how to build good bioinformatics tools?

By Dr Christopher Hogue

At the BIO convention in New York recently, I was aghast to watch as a CEO of a venture capital firm expressed his mighty disgust with bioinformatics. His rant indicated that he felt bioinformatics had little effect or impact on drug discovery. I was stunned because I didn't know whether I agreed or disagreed with him.

At the same time I found myself clutching the Genome issues of Nature and Science. I was in awe of the enormity of these journals, especially in the amount of advertising. I marvelled at the beauty of the human genome as the 'centrefold'.

What went almost unnoticed, save for me relaying this information to you here, is that the two heavyweight human genome issues listed a total of 100 job positions in Bioinformatics. These, of course, are only the companies and institutions who are capable of paying premium advertising rates, oblivious to the fact that this expensive advertising has been fruitless for many others.

There were 45 postdoctoral positions and 55 scientist positions, 12 of which are for the prestigious 'Director of Bioinformatics' positions. As I have told many a headhunter, the sightings of highly qualified candidates for top Bioinformatics positions are more rare than sightings of extraterrestrial beings. You might as well go and hire a Martian, they are easier to find. By the way, please tell your own headhunters to stop calling me. Thank you.

While Bioinformatics jobs were long identified as being critical to genomics, we may be facing the most acute problem right now. There are not enough trainees to satisfy the demand for Bioinformatics specialists in biotechnology, pharmaceutical companies and in the emerging field of proteomics. The people out there who claim these positions, this specialty of being a Bioinformaticist are fully capable of doing a lousy job and making the CEO of your VC firm very angry.

How to satisfy this dilemma? Investors clamouring for the heads of failed Bioinformatics groups on a platter, and the steady accumulation of positions unfilled. Well they are one and the same but to get to that understanding took me a while to distil, and I'll share it with you here.

Where is the bioinformatics value?

Well most would agree that value lies in the discovery that leads to an exclusive product. The product with the highest value, ultimately, is a new therapeutic. So working backwards from discovery, each step that leads down this path, whether it be high-throughput screening, large scale genomic sequencing, or the industrialisation of protein interaction discovery follows this trend:

Sample -> Database -> Instrument -> Software -> Data -> Database -> Software -> Discovery!

Bioinformatics in entrenched deeply here. The first half of this scheme is traditionally called LIMS, which stands for Laboratory Information Management System, a kind of workflow and tracking system for biological and/or chemical samples. The second half of the system, after the generation of data, is more oriented towards making biological sense out of that data. Clearly discovery is dependent on this infrastructure.

For example, in high-throughput screening, compound libraries must be tracked in databases, assayed on instruments, and the results gathered. Other information systems that group hits in

Signs of wellness in Biotech/Pharma Bioinformatics

- CIO understands build or buy decisions
- Avoids the bleeding edge of untested technology
- Front-end (GUI) and back-end (Database) are layered and isolated
- Code is platform independent and reusable
- Understand and use bioinformatics standards
- Have libraries of code and APIs (Application Programming Interfaces)
- Using cluster/compute farms more and large symmetric multiprocessor systems less
- Projected expenditures in software/database/load sharing licenses understood
- Software has life cycle models, project management and design documentation
- Centralised source code version control in use with bug-tracking systems
- PERL used as mortar, not as bricks!
- Low turnover of personnel

chemical libraries based on similarities of chemical structure are on the second half of this scheme, and are seen to add a lot of value to the raw data, with tangible results. In genomic sequencing, the LIMS keeps track of things like PCR reactions and sequence fragment samples. Sequencers are the instruments, raw sequence the data. The data is matched, contiged into longer sequences, and similarity search engines like BLAST provide the second half of the system – the analysis and the discovery of genes and novel protein coding regions. Even a single academic scientist pursuing discovery research on the bench now uses parts of this approach, usually the right side.

There exist 'pure' Bioinformatics or Chemoinformatics companies across this scheme. Chemoinformatics systems are well known for tracking compound libraries on the left side of this scheme. Bioinformatics companies like Third Millenium build LIMSs. Companies like Proteometrics make software for mass spectrometry instruments, and lie squarely in the middle. Lion Biosciences and e-Bioinformatics focus on the biological information systems on the right side. In fact just about every Bioinformatics company can be put somewhere on this scheme. Discovery companies selling databases also fit on this scheme.

So there is value in informatics, and in Bioinformatics. Otherwise these companies would not exist, and the processes would not work. Of course not all companies are successful.

What is the gating factor for success? The discovery process is moving from the 'hunter-gatherer' model to the 'harvester'. The informatics behind the harvesting of discovery information must be able to scale. Those caught stooped over, harvesting with sickles will be mowed down by others driving combines!

Algorithm vs implementation

So the question then is 'Does your discovery engine scale?' Can you rapidly harvest information in the discovery process? Well you can ramp up sample production, add more instruments, but integration of the software and databases can eat you alive and make the CEO of your VC very, very irate.

So to state the obvious, software is the implementation of algorithms that operate on data. We know what the data looks like, sequences, mass spectra, SNPs, ESTs, 2-D gel images. Neither biological scientists nor corporate shareholders understand much about the algorithms, other than that they often are expected to turn water into wine or produce other sorts of miraculous discoveries. And the bioinformaticians, while they may be adept at making such algorithms, they often don't understand what it takes to make a good implementation.

Implementation, not algorithm is key. Having been introduced to the world of business presentations by biotech companies, I shake my head when a biotech CEO puts up a pathway interaction viewer and then discusses how their algorithms are revolutionary. Ahem! Free code made available by Sun in their Java toolkit can be manipulated by a high-school student to make a protein pathway viewer. Dykstra's algorithm, a staple of computer science textbooks for a generation, is the very algorithm that finds the shortest pathways in yeast-two-hybrid data.

Implementation is the key to scale. Algorithms, without implementations that can scale, have no value. Case in point. The most used Bioinformatics application is BLAST the Basic Local Alignment Search Tool. A tool to compare sequences. BLAST runs on just about every kind of computer, and really works well on expensive computers with lots of memory and processors. But BLAST is now lagging in implementation because suddenly, the way forward in multiprocessor computing changed.

If you didn't notice the change, well I don't blame you, it is very recent. Cluster computing. The rise of the Beowulf – commodity item clusters of computers running the Linux operating system. Think of it as part of the Linux wave. Instead of expensive multi-processor computers, scientists have discovered that software can be re-engineered to work on clusters of commodity item computers. Thin slivers of servers stacked 40 tall in a rack can hold 80 1GHz Pentium IIIs. A remarkable computing density, and when coupled together with ultrafast Gigabit Ethernet networking on fibre optic cables, you truly have a supercomputer – or a 'supercluster'. A 200 processor 'supercluster' can be incredibly cost-effective and bring to bear incredible performance to Bioinformatics software. I bought three of them for my company, MDS-Proteomics, and we are using them for mass spec data analysis. We also want to run BLAST, but the trouble is one has to change the implementation to get it to work.

There are about four distinct strategies to map BLAST on to a supercluster, and one is best. Companies like Blackstone Technology (www.computefarm.com) and SGI have already taken the public domain BLAST code, altered the implementation in different ways, and now have versions that run on clusters. Yet these companies know I have a build or buy decision to make. Spend a programmer-year to reimplement BLAST to work on my clusters with the implementation I prefer, or buy their 'quickie' implementation, priced competitive to a person-year. Tough decision.

Also factor into it the cost of infrastructure like Platform Computing's Load Sharing Facility, pricey operating system enhancement software that most companies use on compute farms. An expenditure that would be unnecessary if you had the right implementation of all your tools for a cluster computer.

The problem for these companies rewriting BLAST is, if I decide to make the build decision and put the results back into the public domain from whence BLAST came, their products are going to lose value very quickly! This is nothing new, a staple of molecular structure software, Biosym^{TM} lost profitability as free tools like Swiss-PDB-Viewer provided similar benefits. Some academics take twisted pleasure out of devaluing commercial implementations by making good ones freely available. Given the human resources problem, others will still decide to pay for an implementation that is fully supported.

So is the value then in the BLAST algorithm, or the BLAST implementation? The BLAST algorithm endures as a free tool, even as the implementation changes. Is it the science of the search algorithm, or the engineering of the clustered database sub-system? Is it understanding sequence statistics or understanding how message passing between CPUs on separate cluster nodes with their own isolated RAM running separate instances of Linux?

Well, for the short term, implementation is the value, not algorithm. The implementation to scale BLAST on a cluster is technology that offers any company real savings in costs for high-performance computing. Unless an algorithm has an entrenched intellectual property ring of patents around it, outperforms everything else, and has never been made available as source code, it has little value. Few so-called proprietary algorithms are worth writing home about.

So if the immediate value proposition in Bioinformatics is in the implementation, how do we get people who are good at implementation? This is the real human resource dilemma. Bioinformatics, as a shotgun marriage of biology and computer science, lacks the engineering discipline required for understanding what a good implementation really means. Algorithms drive the academic discovery process. Nobody gets a Nature or Science paper out of a good implementation. So implementation does not get taught.

Nascent degree programmes emerging in academia to train bioinformaticians stress algorithm and leave implementation to blow in the whimsical wind of Perl scripts, instead of training people about industry standard methods for making multiprocessor systems or cluster systems operate. So if the value is in the implementation – then perhaps we need to consider it more carefully in both how we select people to work in companies, and in how we train them.

Having been involved heavily in training people under the Canadian Genetic Disease Network Bioinformatics Workshop series (bioinformatics.ca) it takes a tremendous effort to create a trained individual capable of being a leader in bioinformatics. The industry must make available its best people to spend time on training others, and to support pre-competitive initiatives in training. Those who train bioinformaticians must spend more time themselves studying implementations and parallel computing and learn the material so they can teach others.

So train people to make good implementations, and your systems will work and scale, and your biologists and venture capitalists will be happy. Stop crowing about your algorithms because they are already old and, to those in the know, fairly obvious. Good implementations are far less obvious. They have good short term value. But remember, in the end, even the implementation value is fleeting. The real value in Bioinformatics is in the speed it offers in identifying your product first. Speed is necessarily a function of scale and the quality of your data. Implementations that are engineered to scale will dominate the discovery process. DDW

***Dr Christopher Hogue** is currently Chief Information Officer and Co-founder of MDS Proteomics. Prior to joining MDS Proteomics, he was a scientist in bioinformatics at the Samuel Lunenfeld Research Institute, Mount Sinai Hospital in Toronto, Canada where he developed the Biomolecuar Interaction Network Database and studied protein folding using Beowulf clusters. Concurrently, he was an Assistant Professor of Biochemistry at the University of Toronto, where he taught bioinformatics courses as well as courses in proteomics and protein structure and function.*