# Lab-to-lab connectivity and semantics in the life sciences

Walter Fontana[*]
*Harvard Medical School*
*Systems Biology*
*250 Longwood Ave.*
*SGM 221*
*Boston, MA 02115*
*+1 617 432 5202 Tel*
*+1 617 432 5012 Fax*

Jim Karkanias[†]
*Executive Director*
*Clinical and Analytical Systems*
*U.S. Human Health, Merck*

L.G. Meredith[‡]
*CEO*
*Djinnisys Corporation*
*505 N72nd St*
*Seattle, WA 98103*
*+1 206 388 4046 Tel*
*+1 206 388 4367 Fax*

Matthias Radestock[§]
*CTO*
*LShift, Ltd.*
*Hoxton Point*
*6 Rufus St*
*London N1 6PE*
*+ 44 (0) 20 7729 7060 Tel*
*+ 44 (0) 20 7729 7005 Fax*

## Challenges for semantic information technology in the life sciences

In each domain, harnessing the opportunities afforded by the Internet proceeds according to a pattern: connectivity; schematization of data; schematization of process. The early business-to-business efforts exemplify this; being connected excited dreams of business efficiencies achieved through electronic interchange, but invariably demanded a common understanding of the data being exchanged. Likewise, such schematization efforts led to a recognition of the need to specify (and thence ratify) protocols by which the schematized data is exchanged. This history is codified in artifacts and standardization efforts like EDI, XML [3], XML schema [1] [2], and the emerging choreography standards [4].

As the Internet becomes increasingly embedded into daily practice in academic and industrial research, a similar development is necessary in making the lab-to-lab connectivity a productive reality. Very recently, Turing award winner Jim Gray was instrumental in working with the astronomy community to help them schematize and then provide web-service-based access to their data, dramatically increasing productivity in that community. The geologic and oceanographic communities are also undergoing this process. Due to the nature of biological data, however, the life sciences, unlike these communities, faces particular challenges to realize this

[*] Electronic address: walter@hms.harvard.edu
[†] Electronic address: jkarkanias@gmail.com
[‡] Electronic address: lgreg.meredith@gmail.com
[§] Electronic address: matthias@lshift.net

development.

The first challenge is the *volume* of data. Projects at the scale of the Human Genome Project, and corresponding technological advances, like high-throughput techniques, have not only produced an avalanche of data, but opened the door to methods and further investigations that will produce even more data. The volume of the data means that a post-facto annotation effort requiring human intelligence to *add* semantic data will almost certainly be impractical. It also presents serious challenges for existing search technology. Much of the interest and attention on high-performance computing in application to the life sciences can be seen as a witness to this fact.

The second challenge is the *kind* of data. Biological entities possess at all scales internal structure that engenders behavior. When networked together, such entities assume functional roles towards meeting a multitude of constraints derived from the need to represent, store and process biological information. Data about such entities is therefore not only 'syntactical', like sequence information for DNA or structure information for proteins, but essentially interactional. The high-throughput techniques responsible, in part, for the data volume are also yielding data *about* the decentralized, yet highly structured ways in which the many molecular pieces of a cell work together in producing 'organization'. Thus, while much of genomic data can be thought of as structural in nature, data in post-genomic biology will be greatly concerned with behavior, e.g. metabolic and signal pathways, and with behavior in context, e.g. gene expression (as opposed to gene sequences).

While we stipulate that some of these behaviors will be discovered by human ingenuity through experimental efforts, others, we submit, will be discovered through computational methods be they fully or only partially automated. Naturally, these methods must be kept in close contact with the established base of trusted fact and experimental result, but over time should guide the growth of that base and inform what comes to be taken for 'trusted fact'. If these methods are to employ data and models from multiple different research organizations (be they academic lab to academic lab, or academic to industrial or lab to lab within a single large company), they must rely on a common understanding, i.e. a schema, of the data exchanged. That is, to say, they must rely on a schematization of dynamics, behavior and behavior in context. In all likelihood, they will rely upon such a schematization not only to store and retrieve data about the behavioral repertoire of biological components and systems, but also to build up-to-date models on-the-fly, which may, in turn, amend the semantic information stored and inform experimental choices.

The schematization problem is thus foundational in at least two ways. If one takes enterprise efforts at data integration as a proxy, the story of the tower of babel is no myth, but a day-to-day reality. In point of fact, in that domain, the places where schematization amongst autonomous organizational entities does result in agreement is precisely at points of interchange in protocols the participation in which is of significant recognized value to all parties. But, as noted before the problem is much more serious because post-genomic data will increasingly be about system *dynamics*. The likelihood is that along with the explosion of raw biological data there will be an explosion of dynamical biological models – after all, it's much less costly to generate models than it is data. And, the point of modeling is to explore the space of possible systems to be more effective in the way we go about asking questions that generate real data. In fact, we create models in the hope that they allow us to transform data into information. Schematization of dynamics, needless to say, is notoriously difficult. If our understanding of computer programs represent any kind of proxy for the dynamics of these systems then we should expect this data to be of much greater complexity, and more to the point to present particular challenges for schematization. Likewise, semantic-based search on temporal and dynamical data is notoriously difficult and is an active field of research.

The third challenge is the *sensitivity* of the data. As much of this data is directly related to human health, much of it will be central to commercial interests, national security and personal privacy concerns. The sensitivity of data means that dependencies amongst semantic elements may be severed because some portion that data must be hidden from view. A useful semantic technology will be able to provide models that remain informative even when parts of them are intentionally obscured or under specified.

## Behavioral representations

For data about the dynamics of biochemical and biological systems, such as is captured in Systems Biology Markup Language (SBML) [7], we think computation itself provides a useful proxy to help with a requirements analysis of the problem of specifying and analyzing semantics. Specifically, in computing dynamics *is* semantics. The specification of the semantics of a computer program is a complete account of its dynamics. From this point of view, we envision an appropriate specification of the dynamics of biochemical and biological systems that need not be laboriously annotated. It will already contain its semantics. The problem is to extract and manipulate that semantic information in useful ways.

While dynamics is semantics, in the world we live in, we are sentenced to syntax. The history of logic and computation is a history of compromising between the necessity for syntax in expressing reasoning about anything and the annoying interference from spurious syntactical constructs [6]. A physicist would say the actual materials matter. A computer scientist would say the code matters. The best compromises, therefore are those in which the language employed to specify the behavior of a system have some immediately recognizable correlation to the domain to which it refers. In that case, an analysis of the *expression* representing a system will reveal something about the *system*'s dynamical (i.e. semantical) properties. Classical mathematical models, like ordinary differential equations, enjoy this property in some measure. Mathematical analysis of such systems of equations can reveal properties of the systems they represent, e.g. existence or identity of fixed points, limit cycles, stability properties, etc. In both cases, programs as well as differential equations, complexity and non-linearity place limits on the static analysis off dynamics, at which point one resorts to 'running' (simulating) the respective syntactic representations. It would seem, then, that we seek a *language* for molecular biology – or segments thereof – that would contain the semantics of molecular interactions in a fashion amenable to static analysis, e.g. model-checking, and animation, through execution.

Amongst the various accounts of the semantics of programs, one stands out as particularly promising, the mobile process calculi [8] [12]. Their formal structure illustrates the requirements for extracting and manipulating semantics from dynamical specifications. In the mobile process calculi semantics follows from, or is based upon programs as autonomous interacting entities. The fact that the mobile process calculi are, to date, the only family of computational models that exhibit all of the following properties simultaneously and also have a track record of successfully modeling chemical, biochemical and biological systems [9] [11] [10] can be traced to this semantic foundation.

- Completeness – formally, this is Turing completeness; informally, it means the model is expressive enough to write down any reasonable program.
- Compositionality – formally, the model is an algebra; informally, the model enables building larger systems out of smaller systems.
- Concurrency – the model has an explicit representation concurrent execution of autonomous, interacting computational agents.
- Cost – the model has an explicit account of resources like space and time.

Having these properties simultaneously means that dynamical data may be manipulated in a modular fashion. Independent dynamical models may be brought together to form composite models; and, the dynamics of those composite models may be analysed, that is semantic data extracted, in terms of the dynamics (read semantics) of their components. This gives rise to dramatic gains in efficiency of representation and calculation.

But, this kind of modularity also provides a framework in which to control the view of data. Some components in a modular specification may be opaque, or specified up to logical properties, without revealing detailed internal structure. Thus, for many situations it is possible to yield sensible and informative specifications without revealing sensitive data.

In this connection it is important to note that dual to these calculi, in a mathematically precise notion of duality, are a family of logics including the Hennessy-Milner Logics [8] and the recently developed Spatial Logics [5], that allow one to express behavioral and spatial properties as formulae. As noted above, these formulae serve to provide abstract specifications of components, allowing data-hiding. But, they also serve as a natural basis for a query language, and the corresponding model-checking algorithms serve as a basis for query and search engines that may be applied to individual or collections of models.

Moving from these general observations to applications in biology we might characterize – in an admittedly simplistic way – a class of work on signal pathways in the following terms. A (cell's) surface is submitted to (a biochemical soup constituting) environmental stimulus. Due to advances in technology we can see how the (cell's) interior responds (in terms of gene expression). The biologist's job in this setting is to propose a mechanism of the interior that when presented with this environmental stimulus gives rise to the observed behavior.

We may express such a situation in the process algebraic framework as follows. We construct an agent expression corresponding to the environment to which we submit the (cell's) surface, call it $E$. This is reasonable, because the environmental stimulus is under experimental control. Likewise the proposed interior mechanism to be evaluated is expressed as an agent expression, call it $P$.

Finally, we express the observed gene expression as a formulae, say $G$. We can submit to standard model-checking techniques the assertion $E \mid P \models G$, which states that $P$ in the environment $E$ respects the observations $G$.

This calculation may be used by an individual biologist in evaluating her hypothesis about a mechanism. But, it may also be used by a biologist searching for a mechanism (that has already been modeled and stored in a repository) that responds to an environment in a way that meets her requirements for observed gene expression.

## conclusion

Biological data are increasingly about the interaction of components in a network context. Intriguingly, in almost every domain of industry and research, the world wide web is driving interaction amongst individual researchers, organizations, and, increasingly computational agents. We submit that semantic questions arise at points where agents, be they molecular, computational or human, interact. We assert that activities such as

- expressing and publishing a dynamical model;
- investigating whether an expressed dynamical model enjoys an expressed property;
- search amongst bio-dynamical models for ones enjoying expressed properties

are instances of communication acts, i.e. of interaction. We argue, therefore, that a computational model of interaction, as is realized in the mobile process calculi, provides a powerful proxy for understanding the requirements of a useful semantic technology in the world wide web.

*a. Acknowledgments.* The authors thank Corrado Priami and Brian Schultz for discussions on a number of points made in this document.

[1] *Xml schema part 1: Structures*, W3C Recommendation (2001).

[2] *Xml schema part 2: Datatypes*, W3C Recommendation (2001).

[3] *Extensible markup language (xml) 1.0 (third edition)*, W3C Recommendation (2004).

[4] *Web services choreography description language version 1.0*, W3C Recommendation (2004).

[5] L. Caires and L. Cardelli, *A spatial logic for concurrency (part ii)*, Lecture Notes in Computer Science, Springer-Verlag, 2002.

[6] J.Y. Girard, Y. Lafont, and P. Taylor, *Proofs and types*, Cambridge University Press, 1989.

[7] M. Hucka, A. Finney, H. M. Sauro, and H. Bolouri, *Systems biology markup language (sbml) level 1: Structures and facilities for basic model definitions*, (2001).

[8] Robin Milner, *The polyadic $\pi$-calculus: A tutorial*, Logic and Algebra of Specification **Springer-Verlag** (1993).

[9] Corrado Priami, Aviv Regev, William Silverman, and Ehudi Shapiro, *Application of a stochastic name-passing calculus to representation and simulation of molecular processes*, Information processing letters **80** (2001), 25–31.

[10] Aviv Regev, *Representation and simulation of molecular pathways in the stochastic $\pi$-calculus*, 2001.

[11] Aviv Regev, William Silverman, and Ehudi Shapiro, *Representing biomolecular processes with computer process algebra: $\pi$-calculus of signal transduction pathways*, American Association for Artificial Intelligence Publication (2000).

[12] David Sangiorgi and David Walker, *The $\pi$-calculus: A theory of mobile processes*, Cambridge University Press, 2001.