

Towards the development of computer systems that express individuality

Neil McBride
School of Computing
De Montfort University
Leicester
LE1 9BH

Abstract

Individuality is a key property of many systems, particularly biological. It can be defined as the sum of the characteristics which differentiate one instantiation of a system from another within a given domain. Individuality enables biological systems to be robust, adaptable and to form complex systems and societies. However, individuality is generally not a property of artificial systems, in particular information technology artifacts. Indeed, quality practices often focus on avoiding individuality. The ambition expressed in this paper is to make some progress towards developing computer systems which can express individual properties, develop adaptability and express emergent behaviour. The paper looks to biology for inspiration in the development of computer systems with individuality. In particular the concepts of systems biology are explored. Examples from molecular systems biology – non-coding RNAs, signal transduction systems and regulatory networks – are used to illustrate the idea that complex structure and function in biological systems result from complex dynamic networks. A view of systems is taken that they are networks of interactions between elements, which may themselves be systems, changing in space and time, and resulting in observable behaviour. The ideas that arise from this view are briefly explored. This leads to the tentative proposal of some principles which will need to be addressed in designing computer systems with individuality. Such systems will need to be designed within a well-defined domain, planned and portrayed as networks, designed with a key focus on the relationships between elements in the networks and tested for emergent behaviour. It is concluded ideas from systems biology provide good material for advancing systems thinking but that a deeper analysis of networks in systems biology is required to develop metaphors in detail that can be used as a basis for computer system structure.

Keywords

Individuality, emergent behaviour, systems biology, computer system design.

Introduction

Individuality is a property of most biological systems. Small variations of characteristics support variety without compromising basic structure and function. Individuality in biological systems is not avoided, but embraced, giving rise to rich societies of organisms and robust and adaptable systems.

However, in artificial artefacts and systems, individuality is avoided and considered to be damaging to quality. Machines are constructed so that there is no variation since variation may cause uncertainty as to the performance of the machine. Uniformity, defined function, simplicity and the control of uncertainty are required. Manufactured machines or systems must be identical.

This machine-metaphor-based approach has its limitations when it comes to computer systems. Those limits are becoming increasingly apparent as the complexity of computer systems and the diversity of connections between them increases. It is becoming more difficult to continue design, construction and testing, and to apply quality mechanisms to ensure uniformity. Computer scientists battle to avoid the emergent behaviour that may result in complex systems and to retain control over complex systems.

This paper considers why individuality as a system property should be welcomed in computer systems and suggests some initial guidelines for promoting individuality. To understand individuality, the causes of individuality in biological systems are considered and the resulting ideas examined as a basis for developing networked, relationship-based computer systems which express individuality.

The Problem

Computer systems are limited machine-based artefacts. As systems, they do not adapt well. Any changes in environment have to be anticipated and programmed. They break easily because there is little tolerance to change. Deviations from the set of algorithms or processes are not allowed, since deviation leads to unacceptable consequences in the breakdown of the program.

Typical commercial systems limit the flexibility of business processes. Once the business processes are defined, it is difficult to change them. They may fix ways-of-doing-things in the organisation, since the organisation cannot stray from the way the computer system carries out the business processes without incurring significant cost. Computer systems are generally not self-adjusting. If they have any such capacity, it is limited.

They are not very user configurable and are certainly unable to mirror the individuality of human users. They have very limited evolvability and little ability to develop or change in response to the environment.

The use of the descriptive metaphor of a computer system as a machine carries with it connotations of conformability. Quality in such systems is viewed as lack of variation. Such ideas derive from quality in manufacturing where defect-free involves the removal of variation. This lack of variation may lead to inadequacies in computer systems which allowing some individuality would overcome. Lack of variation means

- Uniform lack of resistance to virus exposure. All computer systems of the same type are open to the same virus attack.
- No new solutions to problems. All computer systems solve a problem in the same way.
- Spread of errors. If one computer system is wrong, then all computer systems which conform to the design parameters of that computer will also be wrong.
- A loss of richness of alternative solutions and approaches.
- A lack of added value in cooperation because of the uniformity of the artificial systems.
- A lack of adaptability and response to changing information in large networks which it communicates with.

Computer systems do not express individual properties unless these properties are put in from the outside. A user or developer must make that system more individual by changing parameters, changing the interface etc. Any individuality is being imposed from the outside and isn't coming from the inside.

The Ambition

The ambition expressed in this paper is to develop computer systems which can express individual properties, develop adaptability and express emergent behaviour. This gap in the properties of computer and artificial systems is made clear when artificial systems are compared with natural systems. For example, robot dogs are programmed to behave in the same way. Unless a particular dog is altered, it will have the same algorithms and behave the same way in a particular situation as all other robot dogs. There is no intrinsic tendency to individuality because all robot dogs are the same machine.

Real dogs are all individuals. While there is an underlying pattern of doggie behaviour, dogs will show preferences and subtle differences of behaviour in response to their environment. The range of responses arises from individuality expressed or constrained within a domain.

If individuality could be achieved in a commercial information system it may help in culturally aligning the system to the organization it is serving and getting a match to the individuality of the organization. Individuality in software agents may enable better co-operation and provide a richer set of services in the context of a service-oriented architecture. Individuality in computer system interfaces may help in enabling the user to be effective. Individuality at the system interface to may create variety in the computer network. Systems that can give more tailored responses may be able to cooperated better to produce more flexible, responsive networks.

The expression of individuality could significantly expand the adaptability of computer systems and reduce their brittleness. Systems with individuality may respond better to threats, adapt to different environments and act as more responsive agents in larger networks.

Perhaps this structural ambition is part of a wider processual ambition which seeks to find alternative approaches to computer systems understanding and development. I would suggest that individuality will not result from current linear approaches to systems development and may require a more non-linear, networked approach.

What is Individuality?

A working definition of individuality can be offered as:

Individuality is the sum of the characteristics which differentiate one instantiation of a system from another within a given domain.

It is expressed as variations in an observable outcome or structure. Furthermore, individuality is bounded. It shows up within the context of a defined system with describable generic characteristics. Individuality is not random nor of infinite variety, but constrained within a system and its rules. It is constrained within a domain.

A nose is the same structure in the middle of everyone's face. Barring an accident, or a catastrophic birth defect, the nose will be a recognisable structure on every human. A protrusion with two nostrils used for breathing is a recognisable domain structure. However, within the domain of a nose a significant amount of variation is possible – size, shape, pointedness, hairiness, colour can all vary to establish individuality as a sum of the variable characteristics. Similarly, a fingerprint or a retina has definable generic characteristics; but within their domains they may be infinitely variable as far as we know.

It should be noted that we need to distinguish individuality from identity. While a set of manufactured products may each have identity, each car with its own vehicle number, for example, they may not be individuals unless there is variation. Variation in car colour may be taken as making those cars individuals, but individuality has a connotation of variation at a more extensive and higher level; such that a number of characteristics vary. Each system has identity, established by its existence as an entity but may be identical and indistinguishable from others. Individuality is the norm in biological systems, but a rarity in artificial systems.

In biological systems, it is clear that individuality is an emergent phenomenon, hence any study of individuality in systems – with the aim of creating individuality in computer systems - must examine all the variable characteristics as well as researching their origins.

Individuality may emerge from complex interactions. It may also be suggested that individuality increases with the complexity of the system. Worms seem to have less individuality than humans, although that may be dependant on the point of view. At a molecular level, worms may be just as individual as humans. Indeed, individual cells of the same type may express individuality. Therefore the observation of individuality may be significantly a matter of subjective interpretation.

Value of Individuality

In biological systems, in society, in organizations, and in communities, individuality is essential. In establishing variation within species and within domains, an ability to meet changing conditions and new challenges is developed. This is not simply ‘the survival of the fittest’, but the obtaining of a better fit to a changing landscape through co-operation and collaboration, through the creation of a pool of problem-solving abilities.

Individuality in biological systems is an essential outcome of sexual reproduction. Cloning may produce weaker individuals; in-breeding increases defects, but recombining different characteristics produces hybrid vigor, expressed as individuality.

Individuality within the elements of a system enables a richer more responsive system – which can respond to more environmental change. There is greater support for cooperation in which individuals bring their different characteristics to bear collaboratively on problem. Since individuals are different, with different detailed characteristics, strengths and gifts, the networked system of which they are part will benefit from cooperation rather than competition. Cooperation between individuals may give rise to more robust systems.

In biological systems, individuality enables richer communication between systems. Rich communication will lead to more complex structures, which uniformity of system elements could not support. Each individual system (for example, protein, gene, cell, organ, person) functions within a community of systems which itself constitutes a system. The complexity of the community is helped by the individuality of each member system. Hence, a community of individuals, in contrast to a community of identical elements, can produce more complex organisations, a wider range of complex behaviour and may store more information. A greater variety of structures will also be generated.

Since individuality is a property of most biological systems we should question why we avoid it in artificial systems. Individuality in computer systems may encourage cooperation and help develop adaptability. It may be easier for a group of computers expressing individuality to respond to a problem than a group of identical computers with the same code, and the same programmed solution to a problem. Individuality may give rise to creative potential and enhance the development of novel solutions. Variation should be supported within a generic application through the expression of individuality.

The problem of individuality

In computer systems design, requirements for the machine are defined on behalf of a group of users. If some users want different requirements, then an alternative specification must be produced and effectively, two different or variant systems are produced. Hence, designing individuality into systems may be complex and uneconomic. Each individual design results in a separate new specification requiring maintenance. The individuality is then not built into the system specification but is an add-on. Individuality is then an additional layer which suppresses the generic rather than something that emerges at a deeper level and is embedded in the generic.

The key question is then how can a system allow individuality while retaining the generic function within a given domain? How can limited designs give rise to individuality? In computer systems the design of individual systems must involve constructing individual screens, programs, database calls for each individual system. This can give rise to different systems for each individual variation, in which each system is equally brittle and the creation of individuality results in the proliferation of software and increased maintenance problems. The individuality is neither dynamic nor emergent. It is then seen as spurious extra effort which inhibits standardization.

Using the traditional programming paradigm it is impracticable or even impossible to design, develop and deliver different requirements for every user. The general response of developers and IT departments is to make requirements as generic as possible and to seek standardization of platforms and applications. Hence individuality in computer systems is mostly limited to cosmetic alterations done by end-users themselves, which may contribute little to developing flexible responses to business problems.

Biologically Inspired Computing

Since individuality is a common property of biological systems, but is rare in artificial systems and particularly IT artifacts, the study of how individuality emerges in biological systems may identify some ideas and methods for producing individuality in computer systems. Hence this present study comes under the academic umbrella of biologically inspired computing.

Ideas from biology have inspired computer scientists and information systems professionals for some time. Genetic algorithms and genetic programming are based on the principles of evolution and genetic selection. Neural networks, artificial swarms, and fractal gene networks have all been applied in computing (De Castro and Von Zuben, 2005). Work has been done to create artificial immune systems for computer systems based on the workings of the human immune system. The autonomic system has been used as a metaphor for developing self-regulating computer systems. Other biologically inspired computing studies have examined the visual cortex and self-repair mechanisms.

Since it is recognized that biological systems have a lot of properties including robustness and adaptability that computer systems do not, increasing work is being done in investigating the value of biological concepts to computing. We will follow that route:

seeing what is understood about the development of individuality in biological systems and applying that to artificial systems, particularly computer systems.

The Problem of Individuality in Biological Systems

It would seem to make sense that specifying individuality in computer systems is going to require a large amount of specification and programming. The more different two systems are, surely the larger will be the specifications to define those differences. Hence large systems with a large amount of individuality will have massive specifications.

If we look to how structure and function are specified in biological systems in the DNA, we would expect to find larger amounts of DNA i.e. large specifications for complex biological organisms. The specification for a human, in terms of proteins coding sequences and size of genome should be much greater than a mouse. However, that is not the case. Genome size does not correlate with the complexity of the organism. The rice genome is large than the human genome.

The problem of individuality in biological systems is put into stark relief when the extent of protein coding sequences in the genome is considered. Following the human genome sequencing project, original estimates of 100,000 expressable genes, encoding for proteins, was scaled down drastically to a current level of 20-25,000. The number of genes in humans is only twice that of roundworms or fruitflies. Mice and men share 99% of protein coding sequences.

The expression of complexity in the phenome that leads to individuality cannot arise from solely from the number of expressed genes. Many of these are very critical from development and are conserved across wide ranges of species. The basic problem here is that the central dogma of molecular biology – DNA to protein to structure – does not explain species differences or individuality. There has to be a different mechanism.

Of the 3 million sequence differences between individuals, only 0.3% occur in protein coding sequences (Mattick, 2003). The rest occur in non-coding DNA, the so-called junk DNA, which, far from being junk, contains large amounts of control sequences. It is in the non-coding control sequences that the specification of individuality lies. At a species level, the ratio of non-coding to coding DNA increases in a predictable manner with increasing biological complexity (Taft and Mattick, 2003).

Hence biological structures and their variation and behaviour emerge from dynamic interactions within networks of control elements, the nature of which will be discussed briefly below. Biological systems may be considered as networks of interactions, developing and constantly changing in time and space which give rise to expressed structure and function.

It may be that the solution to developing individuality in computer systems lies in their development and representation in terms of networks of interacting elements, where development of function results from the changing relationships between the elements

and emergent information is held, suspended in the networks themselves and not in the specification.

Such ideas are not new, but are increasing in circulation in biological sciences as the limits of a reductionist approach to biology are discovered. The triumph of biology in determining the genetic code and the sequence of the human genome, gives rise to new problems as the complexity of development is revealed. In biology a new branch of systems biology is emerging which tries to address problems of structure and function in a holistic manner.

In considering approaches to individuality in computer systems, this paper will next outline some properties of a networks approach to systems through a series of network hypotheses, then briefly highlight some examples of networks from systems biology and finally explore some guidelines for the development of network-based computer systems which might express individuality.

Network Hypotheses for Systems

A network hypothesis for systems could be tentatively suggested as follows:

A system is a network of interactions between elements, which may themselves be systems, changing in space and time, and resulting in observable behaviour.

A complex system arises from networks of interactions between elements of the system. The pattern of dynamic interactions in the network determines the resulting structure of the outcome, whether an organization, an organism or an artifact. Networks are always dynamic. Observed structure, both artificial and natural arises from stopping or taking a snapshot of a network at a particular time.

Due to the dynamic, developing nature of networks, the information required to encode or represent the network may be less than the information the network holds. Networks which constitute systems will carry information which is not apparent in by studying individuals (see Merali, 2004). Emergent information can be held in the network as it develops. It is possible to precisely define a network structure in a set of control instructions so that the network will be consistently reproduced every time that set of instructions is executed.

Small changes in interactions, arising from changes in the network program may have large effects. Small changes early in the dynamic progression of a network will have catastrophic effects, resulting in entirely different structures or failures to produce structures or processes. System defects will emerge from dysfunctional relationships within the network. However, as the network progresses its development, the capacity to absorb mistakes without damaging core structure or process increases. Large networks will absorb breakages.

Dynamic development networks have some give (springiness, wobble) that enables variation without losing the resulting domain structure. The springiness of the network gives rise to robustness and reduces brittleness.

Individuality arises from small variations in the dynamic network. Multiple small variations accumulate to give rise to emergent variation. More complex dynamic networks have more propensity for the expression of individuality and for catastrophic failure.

Individuality may be more likely to emerge from variations in interactions late in the development of a dynamic network. Key interactions take place at the interfaces between individual and between systems. Variations in these interactions because of individuality in the participants will give rise to more complex robust networks.

To illustrate these concepts further, we now turn to some examples in systems biology.

Systems Biology

The recognition that an understanding of the genome structure of organisms does not provide an explanation of development and function has led to the development of systems biology which seeks to understand the complex networks of interactions that give living cells their structure and function (Kitano, 2002).

Reductionism can only go so far in explaining the effects of individual components on cell function. Systems biology studies the underlying network structure of organisms, rather than focusing on individual components. Empirical methods enable the analysis of a range of cellular components in microarrays simultaneously which can lead to increased understanding of complex networks. Metabolic networks can be simulated, as has been done with the methane network of methanosarcina, for example (Feist et al, 2006). Networks can be characterized, described in terms of graphs (see Barabasi and Olivai, 2004).

Proponents of systems biology recognize the universality of complex networks in technologies such as the Internet and societies. In biology these dynamic networks provide the basis for the development of dynamic pathways resulting in metabolism and cell development.

These interacting networks provide layers for phenotypic expression on top of the genomic layer. In addition they add informational value and provide a platform for the level of complexity required for individuality to emerge. This individuality results from the flexibility or give in the networks. System networks may not be rigid and hence control mechanisms will not be as brittle as computer systems tend to be. Brief descriptions of three biological examples, non-coding networks, signal transduction networks and regulatory networks will illustrate the possibilities of phenotypic diversity. If small variations in networks may give rise to species, even smaller variations in interactions in networks will give rise to individuality.

Non-coding RNA networks

The discovery that non-coding DNA, that is DNA that is not transcribed into protein, accounts for a vast majority of the genome has been known for some time. The central dogma of molecular biology, that DNA is transcribed to RNA which is translated to proteins, which give rise to structure, led to the idea that non-coding DNA was an irrelevancy in cell development and was called junk DNA, unnecessary padding between the real working DNA sequences.

Recent work, described by Mattick (2003, 2004), indicates that much of the so-called junk DNA is transcribed into RNA. Indeed, 75% of transcripts produced from the DNA are non-coding. Such non-coding RNA is itself processed in networked molecular pathways and is developmentally regulated. It has high sequence complexity and some sequences are clearly fundamental to development since they are conserved over large evolutionary distances. Mattick (2004) suggests that this non-coding RNA is not evolutionary junk but forms the primary control architecture underpinning differentiation and development. In *C.elegans* developmental timing is controlled through a network of RNA-RNA interactions. Non-coding RNA may be involved in a range of activities including the regulation of chromatin architecture and gene silencing.

Hence how the organization ultimately develops and behaves is determined by networks of interactions involving non-coding RNAs as well as expressed proteins. Non-coding RNAs may allow a greater amount of variability than protein networks.

Non-coding RNA networks may help in the expression of limited individuality within a stable architecture. Phenotypic and individual characteristics may be seen as emergent phenomena, resulting from the added informational value derived from interactive, dynamic molecular networks.

Signal Transduction Systems

Many communication effects in cells are triggered by molecular signals such as hormones. Developmental signals are transmitted into cells through receptors in the cell membranes. The signal protein binds to a receptor protein, changing its structure such that the receptor connects with intracellular pathways, triggering off a cascade of protein to protein interactions which spreads through a network to result in cellular changes right down to the switching off or on of transcription in the DNA and hence connecting to non-coding RNA networks of interaction.

Targets within signaling networks may be affected by concentrations of signaling components. Indeed, concentration changes can rewire the network (Fontana, 2002). The network coordinates through messaging which may involve amplification and feedback control mechanisms. Kinases and phosphatases are ubiquitous signaling agents within cells that participate in complex networks. Complex behaviour of signal transduction systems

arises from changing interaction among its proteins. Interactions also involve modifications that occur to proteins within these networks.

Here dynamic behaviour in a biological system is being achieved through an evolving set of interactions within a network. Small changes in that network can result in different responses to the same signal by different cells and systems.

Regulatory Networks

Many genes encode transcription factors, which are protein with entirely control functions, rather than structural involvement. Even in the comparatively small pool of expressed genes, some are primarily concerned with control.

In biological development, cis-regulatory elements, regions of DNA, regulate the expression of genes located on the same strand. Each element receives multiple inputs from other genes in the network. Operative cis-regulatory elements produce refined spatial patterns, resulting in complex information processing devices.

In sea urchin embryos, the spatial triggers are signaling ligands produced by other cells. In the sea urchin Endo16 regulatory system a large polyfunctional protein is secreted into the embryonic and larval midgut. Its distribution is temporal, starting in the mesoderm and spreading throughout the gut to end up only in the midgut.

Two control networks have been identified (Davidson et al, 2002), involving nine sequence specific transcription factors, and also involving linear amplification, switches between networks and logic operations. The first stage regulatory network involves 40 genes, each verifiable at the DNA sequence level and each involved in a developmental process which is time-specific. There are steady changes in the regulatory networks which are successively overlain with new regulatory patterns.

Here structure arises from dynamic processes in networks. The structure is only derived from the processes, is dynamic and is maintained by the processes.

These three examples from system biology emphasize the prevalence of networks as a means of enabling structure and function to develop and as the basis for the execution of that structure and function. Such networks give rise to clearly reproducible and robust structure and function. The implementation, execution or running of the network is repeatable. And yet these systems networks also enable flexibility and adaptability to changing circumstances. The networks can flex or rewire to subtly change structure and function. Such networks could also give rise to individuality in that the robustness of the network is tempered by the tolerances of small perturbation which also results in slightly different outcomes in structure and function.

Overall Properties of Network

It can be seen then that networks change over time. There is temporal process, A network is always changing, always dynamic. However, the programmed nature of the network gives rise to reproducible structure which emerges from the development of the network. A nose is always recognizable as a nose, although it differs from individual to individual and over each individual's life time. A person nose is different as a baby and as an elderly person.

Network interactions give rise to emergent structure which includes individual aspects. The importing of individuality into computer systems may involve the development of networks of interactions. In the final section of this paper we begin to consider some principles or guidelines which might be involved in the development of individuality in computer systems.

Designing individuality into networked computer systems

If we are to design individuality into a computer system a number of aspects will have to be considered. The following briefly outlines four relevant aspects.

Defining the Domain

Since individuality occurs within the constraints of a domain, the extent and nature of that domain must be defined. We must define what is part of the system and what is outside the system. Hence we must define system boundaries. Individuality will result from controlled variation in parts of the system, not in the addition of new elements from outside the system. It will be important to identify interfaces and signals in and out of the system.

Taking a Network View

The designing of a computer system with individuality will involve designing the underlying network of interactions. Hence the system may be modeled as a network of interactions.

Such a design process will involve understanding the properties of networks, building up complex networks from simpler networks and studying relationships and the properties of relationships between individuals.

However, since individuality is emergent behavior, it will only be apparent in complex networks. Hence it will be difficult to plan and predict individuality based on the simpler networks which are aggregated to create the larger networks. This adds a level of design complexity and problems.

The problem of designing networks is further exacerbated by the dynamic nature of networks since behavior will change and further interactions generate as the network is constructed. Individuality may be less likely to emerge from static networks.

Taking a Relational View

Systems are set of relationships. Without relationship there is no system. No element of a system whether physical, biological, theological can have any definable existence without relationship. Designing a computer system is a case of designing relationships. Understanding the relationships in any complex system is key to supporting that system with information technology.

In taking a relational view of computer systems it will be of value to concentrate on information transfer between individuals. Furthermore, the information in and character of the relationship between individual elements should be a focus of attention. For example, in traditional database systems, the information held in the relationship is seen as very secondary to the information in the entity. The information carried by relationships in a system is of key importance to the development of that system and the emergence of behaviour.

Testing the Network

At whichever level of granularity has been selected, it may be more practical to start with small numbers of networked interactions and test for emergent individuality. As the network is developed, sets of relationships in small networks may be encapsulated and treated as individuals in a larger network or hierarchy.

Testing may involve characterizing the emergent behavior required and tweaking network to achieve it. The behaviour of the system must be molded, like molding a pot on the potter's wheel. Small changes are made to the elements and relationships in the network, and the effect observed. From a learning cycle, the effect of changes can be predicted such that the network can be proactively changed.

Testing is about removing errors. In a network those errors are likely to be in relationships. Dysfunctional relationships must be identified and restored.

Conclusions

Individuality is of such core importance in biological systems that it would be surprising if it did not have relevance in other types of systems, particularly computer systems. This paper has looked at the development of understanding of networks in biological systems as a possible basis for individuality in computer systems. Hence the paper is really within the discipline of biologically-inspired computing.

Progress in systems thinking may be made by considering systems in biology. The recent expansion in understanding of the molecular biology of organisms, the recognition that the central dogma does not tell the whole story and the rise of systems biology all provide food for thought in considering system interactions and the development of systems.

The recognition that molecular systems for development, signaling and control are in the form of complex non-random networks (Barabasi and Olivai, 2004) leads us to an analysis of such dynamic structure as a basis for computer systems. Networks and relationships may be crucial building blocks for biological systems and are already gaining credence in the understanding of systems in general.

By taking a relationship-based, networked view of systems, principles can begin to be formed about how individuality might be developed in systems. However, a deeper analysis of networks in systems biology is required to develop metaphors in detail that can be used as a basis for computer system structure. It may be that, rather than designing the final structure of a system, a program or design for the development of a system is built, as a basis for the system developing itself.

A great deal of work is needed to turn fuzzy concepts, perhaps based more on a gut feeling that this is the right way to go, into concrete proposals for developing systems based on networks. And yet if we don't start thinking in different directions we will be trapped with the limitations of current approaches to computer systems and will have no way of coping with the increasingly overwhelm complexity of today's computer systems.

References

Barabasi, A-L and Olivai, Z.N. (2004) Network Biology: Understanding the call's functional organization. *Nature Reviews Genetic.* 5, 101- 113.

Davidson,E.H., et al (2002) A genomic Regulatory Network for Development. *Science* 295, 1669-1678.

De Castro,L.N. and Von Zuben, F.J. (2005) Recent Developments in Biologically Inspired Computing. Idea Group, Hershey.

Feist, AM, Scholten,J.C.M., Palsson,B.O, Brockman,F.J. and Ideker,T.G. (2006) Modeling methanogenesis with a genome-scale metabolic reconstruction of *Methanosarcina harkeri* *Nature Reviews Molecular Systems Biology.*

Fontana,W (2002) Self-rewiring signaling networks: organization as distributed control. URL: <http://www.santafe.edu/~walter/Reserach/reserach/node5.html>.

Kitano, H. (2002) Systems Biology: A Brief Overview. *Science* 295, 1662-1664

Mattick, J.S. (2003) Challenging the dogma: the hidden layer of non-protien coding RNAs in complex organizations. *Bioessays*, 10, 930-939.

Mattick, J.S. (2004) RNA regulation: A new genetics? *Nature Review Genetics* 5, 316-322.

Merali,Y. (2004) Complexity and Information Systems. In Mingers,J and Willcocks,L.(Eds) *Social Theory and Philosophy for Information Systems.* 407-446. John Wiley.

Taft,R.J. and Mattick,J. (2003) Increasing biological complexity is positively correlated with the relative genome-wide expansion of non-protein coding DNA sequences. *Genome Biology.* 5, 1-10.