Università degli Studi di Milano-Bicocca

SCUOLA DI DOTTORATO DI RICERCA IN SCIENZE

Dottorato di Ricerca in Informatica

XIX ciclo

Coordinatore Prof. Giancarlo Mauri

Managing Microarray Knowledge with the Semantic Web

Tesi di dottorato di **Marco Brandizi**

Tutor: Prof. Giancarlo Mauri

Cotutor: Prof. Mauro Pezzè

A.A. 2006/2007

Abstract

Modern Molecular Biology is data intensive, information intensive science. High throughput biotechnology allows to measure a great variety of biological characteristics in a single operation. Computer science is of considerable importance in handling such data and the knowledge achieved by the activity of analysing and interpreting the data.

In this thesis, we focus on data generated by means of the microarray technology. Microarrays are widely used to understand in which conditions the genes actually produce the proteins they encode, what it is called gene expression.

The Bioinformatics developments of the last years mainly concern the management of those information which allow to precisely describe the experimental activity which has been needed to produce a given microarray data set. As a result of this efforts, mature standards and tools exist which allow to represent and publicly share information about microarray experiments. This way experimental data may be analysed and interpreted by the scientific community, in a collaborative manner.

However, existing standards have limited support to the representation of the outcomes, experimental hypotheses or conclusions about an investigated biological question, which result from microarray data analysis. We argue that dealing with such kind of information would enhance the collaboration on analysing microarray data and, in general, it would improve the achievements of the scientific community working in this field.

In order to make that possible, we propose a model which addresses this issue. We represent concepts like: sets of differentially expressed genes, results from clustering algorithms, claims about the role of genes in a biological pathway. Our model is based on OWL formalisms, a Description Logics based language, which is part of what the W3C consortium calls the Semantic Web. The Semantic Web is increasingly being used in Life Sciences. In fact, it suites particularly well with the high heterogeneity of information which is present in this field. The fact it aims at improving technologies for the World Wide Web is also of great interest for the biological and medical applications. Ontologies, which are part of the Semantic Web, are also widely used in Life Sciences, since they allow to make order in a

science with a variety of related phenomena, studied from multiple point of views and using different terminologies. The model we propose for modelling microarray knowledge may actually be considered an application ontology.

In order to show how our ontology may be used in practice, we have developed a demo application based on an existing Semantic wiki tool. Semantic wikis leverage on wiki tools for building a conceptually simple interface to build Semantic Web based contents. They allow to usefully combine knowledge represented in natural language, with constructs which make use of our OWL ontology. We show the potential of our application to be a tool for managing "gene expression atlas" about specific biological topics, studied by means of microarray technology.

We also provide examples of how the OWL based knowledge base which is created from the semantic wiki, can be worthily exploited, for searching and browsing purposes. In particular, we propose an extension of a ranking algorithm for semantic networks, the Spreading Activation algorithm. Our method is based on the use of the SPARQL query language and makes use of the specific microarray knowledge we deal with. In particular, we build ranks on the basis of quality evaluations, which are either provided by users, or automatically computed.

The work is supported by the 6th European Framework Program's project "DC-THERA".

Availability: an instance of the demo described in Chapter 6 is available at the Web address: http://artemisia.leafbioscience.com:8080/mannMakna

Acknowledgements

I want to thank the many people, who have made possible to realise the work described in this thesis.

I want to thank the personnel from the laboratory of Prof. Paola Castagnoli, who have given me the opportunity to start working in the challenging and fulfilling Bioinformatics field and who has greatly contributed in achieving some knowledge about Biology and microarrays.

I also want to thank Prof. Giancarlo Mauri, for the guidance provided as PhD supervisor and for having given me the opportunity to develop this PhD project. Furthermore, I want to thank Prof. Mauro Pezzè, for the suggestions provided as PhD co-tutor.

Other thanks go to Doct. Ugis Sarkans from the European Bioinformatics Institute, for the suggestions and support provided during my staying at the EBI, as visiting student. I also want to thank Doct. Alvis Brazma, for the support provided and for having agreed on hosting me at the EBI as visiting student. My thanks Doct. Sergio Contrino as well, for the initial introduction to the EBI personnel.

I owe thanks to my colleague and friend Andrea Splendiani, who has given me fundamental support, with long discussions and suggestions about both the Semantic Web and the specific project hereby described.

I want to thank the authors of Makna, for having developed their good semantic wiki software. In particular, many thanks to Karsten Dello, for the help given me in understanding the Makna code.

(I prefer to address the following persons in Italian)

Ogni parola è inadeguata per esprimere la mia gratitudine verso Cristina, per ciò che ci lega, per come mi ha sostenuto, incoraggiato, sopportato, materialmente aiutato, in questi anni di dottorato.

Ringrazio i miei genitori, il cui amore nel crescermi ed educarmi, e il cui supporto morale e materiale, nel sostenere la mia formazione, hanno un valore incomparabilmente più alto del lavoro di dottorato qui presentato. Ringrazio mia sorella per il sostegno dato in questi anni e per il bene che mi vuole.

Table of Contents

1 Introduction

1.1 Formal knowledge in modern Life Sciences

Modern science is founded on the Galilean experimental method [1][2]: new objective knowledge is inducted from making experiments and observing reality. New hypotheses, laws or predictions, are achieved by interpreting experiment results and by applying existing knowledge.

In this thesis we focus on Life Sciences, and in particular on Molecular Biology. Modern Molecular Biology is a data intensive science. It uses experimental methods which produce a great quantity and variety of data. It follows a more up-to-date version of the feed-back process above, depicted in Figure 1.1. Both the experimental activity and the data it generates are much complex and a proper reporting of them is required. Furthermore, given the complexity and the high amount of data produced, information technologies are widely used, in particular, World Wide Web technologies.

Formal models are needed for better understanding the complexity of biological knowledge and for elaborating it with computational approaches. However, in real life scenarios, a mixture of formal and informal knowledge is actually managed by means of computers. For instance, mathematical models of bio-molecular processes are managed together with scientific articles which describe such models.

Computational models and algorithms are of great value for the scientific research in general, and specifically for the Molecular Biology. In fact, computers helps in those knowledge deductions which may be automated, starting from proper formal models, such as Logics-based models.

We have started this PhD project from the observation that more formal models would benefit the research activity in the Microarray areas. Microarray is a technology used to measure the gene expression of thousands of genes in a single measurement operation[3]. Gene expression is a fundamental aspect of how life works and, for this reason, widely studied.

Microarray experiments produce a high flow of numerical data, that may be analysed by means of a variety of approaches, including statistical methods, network analysis methods, meta-information analysis.

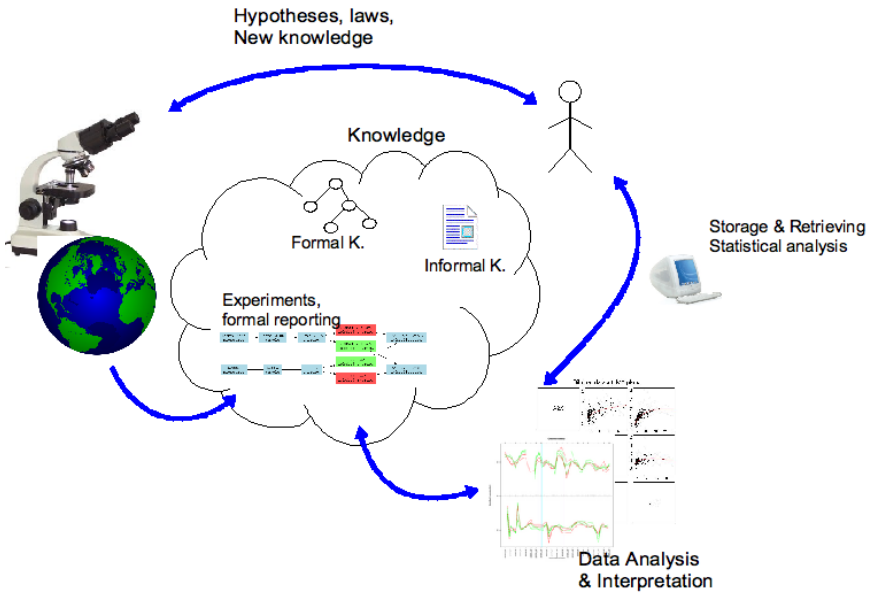


Figure 1.1: the experimental approach in modern Molecular Biology. (Sources: Wikipedia, Bioconductor)

In our work, we argue that, while well established standards exist, that allow the scientific community to share and repeat microarray experiments, less has been done toward the formal representation of the outcomes that are achieved from the microarray data analysis. These formal models, and software applications that could be based on them, would benefit the "scientific discovery loop", depicted in Figure 1.1, by both promoting the collaboration and the share of existing and new knowledge. They would also make possible to build software applications for automatic deduction of new knowledge, from existing formalised one. In order to achieve such results, it is important that such models are standard and shared by the scientific community. That is the main reason why we have explored the use of the Semantic Web technologies[4][5][6]. Because of the already mentioned biological knowledge complexity, we propose semi-formal models, where non formal contents, like articles written in natural language, is complemented by formal structures and models, such as ontology term annotations. This approach makes our formalisation goals realistic and useful.

We also argue that, in real modern world, the experimental method is not applied in isolation, by single people or single groups. Instead, modern scientific activity is run by means of worldwide collaboration, thanks to wide usage of distributed software applications, mainly Web applications, as well as by means of human artefacts, such as articles, conferences and research projects. Furthermore, many non strictly severe evaluations are made when part of the vast scientific knowledge available has to be quickly selected and focused. For instance, impact factors, author authoritativeness and the quantity of papers

which propose the same theory, are criteria which are often used to assess the relevance of the theory. Although the use of such criteria may appear as deviations from the objectivity and strictness of the original experimental approach, indeed they are an inevitable and rational navigational tool in the ocean of scientific knowledge nowadays is to be dealt with. It is for that reasons that we propose to complement the representation of that knowledge which is more directly derived by the experimental method application, with other kind of complementary information. User evaluations, weighted up by user role or expertise on a particular topic, are an of such complementary information.

In conclusion, we propose the view of the experimental method that is shown in Figure 1.2. Here we show differently formalised knowledge which is shared by a community of collaborating people. We also show that the cycle of scientific discoveries may be usefully performed by means of distributed computing. These help in both sharing and exchanging knowledge, and in the discovering process itself, when it or part of it may be computationally done, for instance by means of automatic reasoning.

The rest of this chapter provides details about the Microarray field, while the next chapter clarifies the specific issues we address in this thesis work.

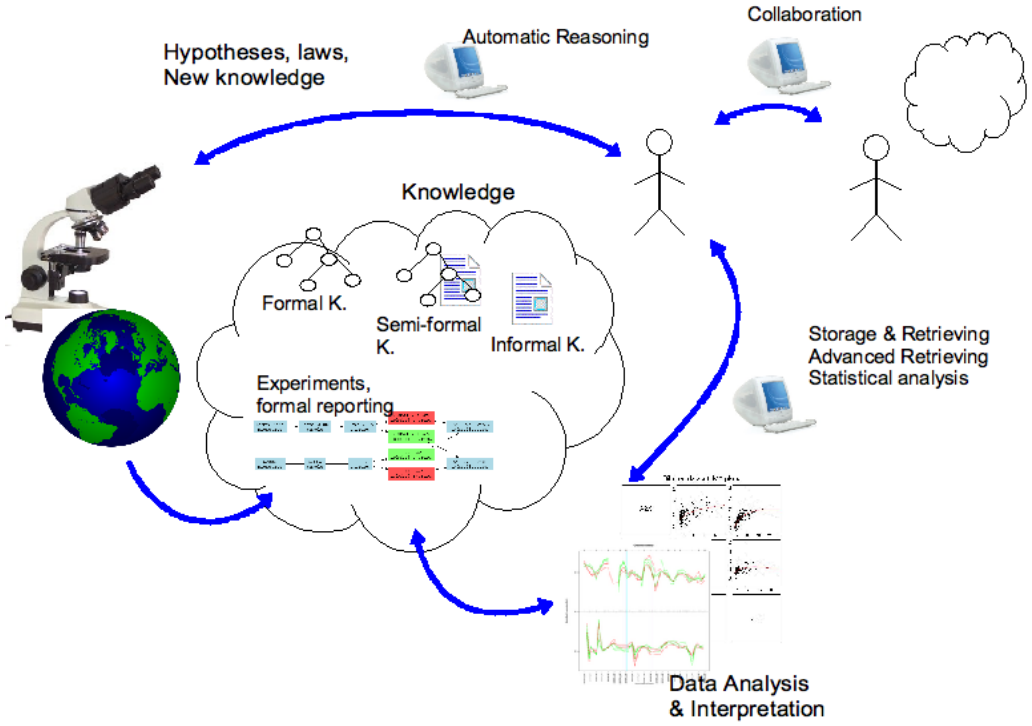


Figure 1.2: the experimental approach updated.

1.2 The basics of Gene Expression

We here report basic notions of the biology of Gene Expression process. A simplified view of such a process is provided by the so called Central Dogma of Molecular Biology [7][8]. According to it, the life is "encoded" by the chaining of four base molecules, the nucleotide, which forms the DNA molecule, the Deoxyribonucleic acid. From a Computer Science perspective, we may see the DNA as a sequence made with an alphabet of four symbols. DNA is structured in sub-string units, the genes. Genes provide the "program", the encoding, for the production of protein molecules, which are chains of another kind of base molecules, called amino acids (Figure 1.3).

DNA and genes are like a life handbook, they contains instructions about how the proteins have to be built. However, whether a given gene is or is not actually producing the protein it encodes, is something that depends on a complex set of biological conditions. The term gene expression refers to the actual production of a protein, from the gene that encodes it. More precisely, it often refer to the quantity of a given protein that is produced under a given condition[1].

All the gene expression machinery is so important in Life Sciences, because the proteins are fundamental building blocks of the life.

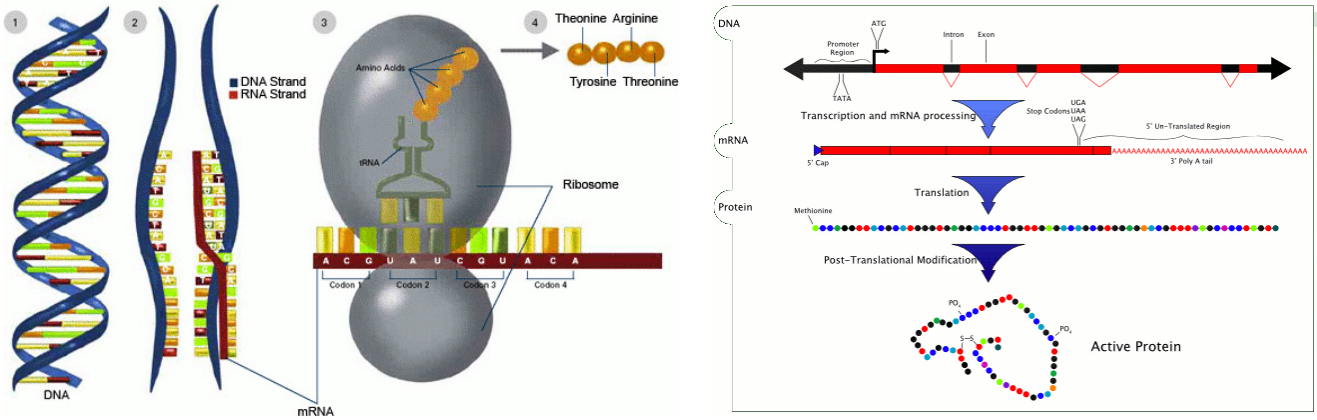


Figure 1.3: the basics of the gene expression machinery. Left: the two complementary strands that make the DNA. Genes are encoded in the strands, by means of four symbols, physically represented by four nucleotide molecules. Proteins are built by opening and reading one of the strands. Right: a gene is logically composed of different units (exons and introns). The gene expression process goes through several stages, during which different variants of RNA molecule are involved. Exons are properly assembled during messenger RNA (mRNA) transcription, which is the first stage in the transcription process. (Sources: National Institute of General Medical Sciences, Wikipedia)

Cells are made of proteins. Proteins are used as communication mean in the interaction between the cell and its outside environment, as well as in cell's internal processes, such as the cloning of DNA molecules, or the gene expression itself. Proteins play a role in many vital cellular processes, whose effects are visible at the macroscopic level, for instance the process of differentiation of stem cells that leads to the formation of tissues and living complex organisms.

Because of such paramount importance of proteins, understanding gene expression is an essential aspect of biological research.

1 Unless otherwise specified, we will refer to the simplest case of genes which encode one protein only. More complex scenarios are known and we refer the interested reader to the literature about the topic.

1.2.1 Gene Expression in detail

Figure 1.3 (right side) reports the details of the gene expression process. Here we may see that the path from genes to the expression of their encoded proteins have intermediate steps, whose most important ones are the transcription and the translation. During the transcription, one thread of DNA, representing a gene, is read and complementary sequences of RNA, a molecule which has some similarities with DNA, are built. In the following steps, amino acids are assembled according to the information depicted in the RNA, following a predefined encoding schema (a single amino acid is defined by 3-symbol RNA sequences). The resulting protein then undergoes a set of spatial changes, until a final structure is reached.

The gene expression of a protein is regulated, at the different levels shown in the figure. There are proteins that play the role of transcription factors, which bind to specific regions of a gene, the gene domains, controlling the gene actual transcription. This imply that many cellular processes are networks of interactions between proteins and genes. The study of gene expression may be important, among other reasons, to elucidate such networks. Many of these networks are feedback systems, so that, for example, the expression of a group of proteins may be auto-regulated.

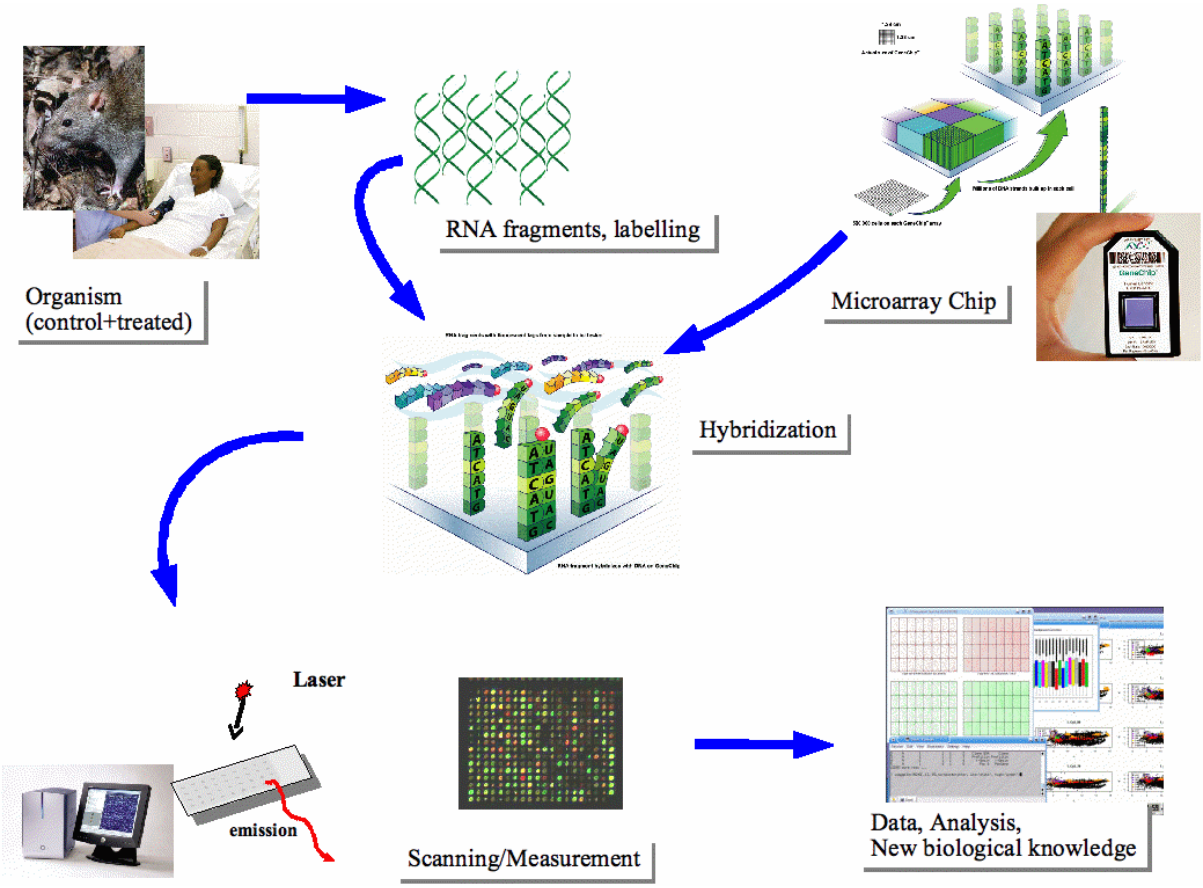


Figure 1.4: the microarray technology. DNA or RNA fragments (depending on the specific technology) with known sequences are immobilised in the chip. The chip is divided into spots, many copies of the same known sequence are immobilised in the same spot, different sequences are located in different spots. Nucleotide material is extracted from organisms being studied and treated with a fluorescent compound (labelling). Labelled extract is mixed with the microarray chip (hybridisation). The hybridised chip is laser-scanned. Roughly, the more a spot is enlightened in the final image, the more fragments of the corresponding sequence were present in the original sample. Most of nucleotide fragments are part of a single gene. With reasonable approximation, the quantity of a given fragment is correlated with the quantity of the protein which is encoded by the gene the fragment belongs to. (Sources: Wikipedia, Affymetrix, Genome Canada)

1.3 Microarrays and high throughput technologies

The microarray technology is one of the innovations that have dramatically changed the Life Science research and development in the last 10 years. The main reason for that lies on the fact that the microarrays allow to have an indirect measurement of the gene expression levels of thousands of known genes, all in one single measurement operation and using a single biological assay. This has provided new powerful investigation approaches in Molecular Biology, as well as the opportunity to study biological phenomena from a holistic point of view, which includes Systems Biology studies[9] and multi-scale studies[10].

In Figure 1.4 a schema of how microarray technology works is reported. A RNA microarray chip is based on the complementary nature of the RNA, which, from this point of view, is similar to the DNA. A RNA fragment of a given sequence will bind to another fragment that has the complementary sequence. The complementary is based on the fact that each nucleotide type, the base molecules the RNA is made of, binds only to exactly one given type of another nucleotide. In order to exploit this principle, the chip is built by immobilising multiple copies of a known sequence on a single spot. RNA fragments are extracted from the biological sample whose expression levels is being measured. The RNA material is then labelled (i.e. made to bind to) with a fluorescent compound. After labelling, a labelled extract is put in contact with the the microarray chip, an operation called hybridisation. Eventually the chip is scanned, by means of a laser beam and an optical scanner. The final result is an image like the one in Figure 1.4. Roughly, if a spot in the scanning image is much luminous, then this means that, in the original sample, it was present a high quantity of fragments having the known sequence probed by the spot. Although there are some variability factors in reality, we may assume that the presence of a given RNA sequence is proportional to the quantity of the corresponding protein it encodes. Therefore the brightness intensities in the microarray image may reasonably be assumed to correspond to gene expression levels.

1.4 Microarray applications and recent trends

As described above, the gene expression process is a central process in living beings, and microarrays are a powerful technology that is used to study gene expression and other related biological phenomena. We summarise the main kind of studies and uses that are possible with these technology. We refer to [11] [12][13] for further details.

Microarrays have are widely used in basic research. Functional characterisation of genes is a typical investigation field, which aims at identifying the function which has the product of genes having known DNA sequences. A basic approach that is used, in combination with microarray data, is the so called “guilty by association”: the expression patterns of unknown genes are compared with the pattern of known ones. Similarities allows to infer that the unknown genes are involved in functions and processes that are similar to the ones that are known for the known genes. These methods often take great advantage from the use of public information, such as banks of formally annotated genes or public literature. An interesting use of functional characterisation is drug discovery.

A different approach is considering the whole “transcriptome”, i.e.: the profile of all gene expression levels, over different experimental conditions. This is widely used for a variety of purposes. For instance, “transcriptome signatures” of patients affected by a given disease, have been elaborated, by comparison with healthy subjects, and the resulting profiles have been used as diagnostic tools. This is useful for early diagnosis, or for the characterisation of the particular type of a given disease. A similar approach consists of the identification of few genes that are expressed under a disease condition, or another target biological condition. The identified genes may be used as bio-markers for detecting the condition of interest.

Other examples microarray applications for functional genomics include: dose-response studies, pathogen-stimulation studies and toxicological studies.

Microarray data are often used in combination with other kind of biological information. Several methods have been proposed to study the activation of biological pathways under conditions of interest. In general, microarray data may be used to study several kind of molecular interactions. The so Chromatin Immuno-Precipitation approach (ChIP-on-Chip), may be used to study the interaction between transcription factors (TF) and genes. TFs are proteins that bind DNA and affect the expression of target genes. Molecular interaction studies are a demonstration of holistic, non reductionist biology, aiming at understanding the complex network of processes the life is composed of.

The same spirit inspires several integrative studies which have been performed recently, where the microarray data are used in combination with the so called "proteomics" data[14]. These are extracted by technologies like Mass Spectrometry or Nuclear Magnetic Resonance (NMR), and are more direct measures of protein amounts. On the information technologies side, projects exists for integrated data management of microarray and proteomics data[15].

A trend in microarray technology is increasing the probe density in chip devices, so that more genes may be probed at less cost. For instance, chips which target the study of Single Nucleotide Polymorphisms are being developed[16]. One possible use of these devices is the identification of chromosome mutations in cancerogenic cells[2]. The so called tiling arrays are another example of high density arrays[17], which store probes that are regularly spaced over the genome of an organism, including regions with unknown function. An example of application of tiling chips is the identification of novel exons.

We conclude this review of microarray applications by mentioning Immunology applications, one of which will be considered as use case in Chapter 6. Microarrays have been used in Immunology for pathogen characterisation, or for the study of the interaction between an infected organism and attacking pathogens. More developments may be expected for the future, for example in vaccine design, diagnostics, monitoring of food and environmental safety.

2 Chromosomes are modular units the DNA may be decomposed of, whose existence have been acknowledged on most organisms.

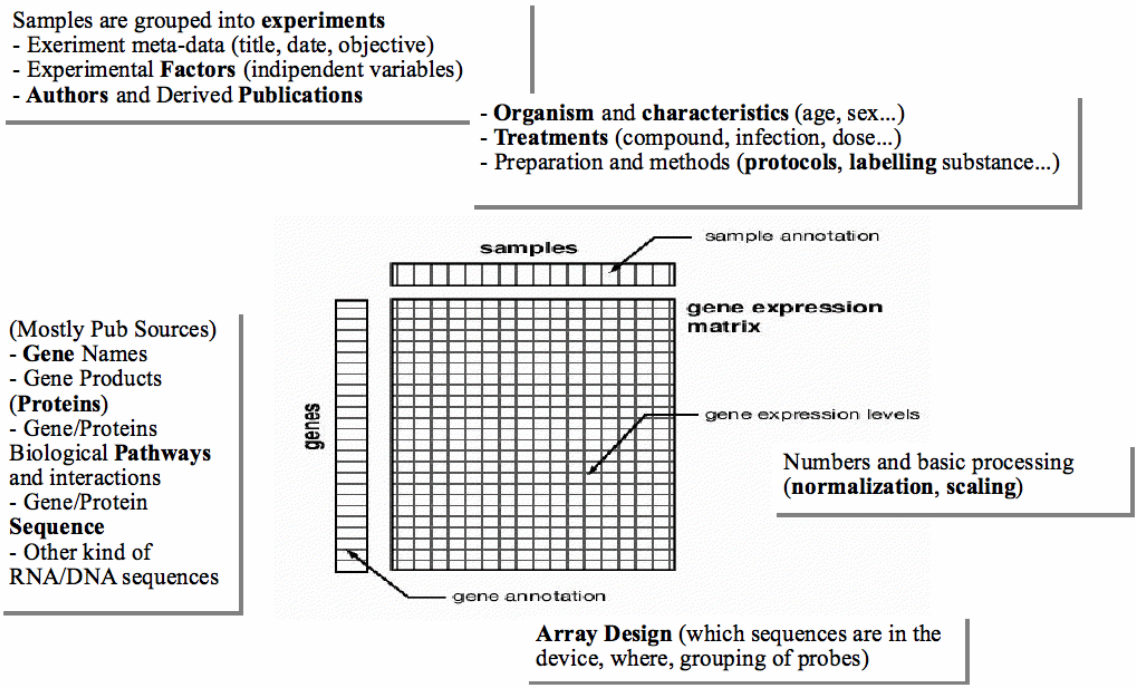


Figure 1.5: the kinds of information needed to report microarray experiments. (Source: [18])

1.5 Dealing with microarray data

A typical microarray chip has 10000 to 20000 probe sets, hence a single hybridisation produces the same number of expression levels. A set of hybridisations, that are logically related by the fact they have been performed with the aim of making a single scientific study, are often called a microarray experiment. Experiments are composed of a variable number of hybridisations, ranging from about 5 to several hundreds. All of this generates a high amount of data. That explains why data analysis methods are needed, primarily statistical analysis and data mining, to gather a biological meaning from the microarray data itself. It also explain why information technologies and software systems are so important for the management of microarray experiments and their results. In Figure 1.5 it is depicted how a microarray data set may be seen from a Computer Science point of view. The matrix at the centre of the figure represents the expression levels, the latter are related, on one side, to the different genes (or, more in general, sequences) that are probed by microarray chips. On the other side, the expression levels are associated to the biological samples that are being measured. In turn, the samples are grouped into experimental conditions and experiments. Concerning the former, there are many source of variability in a biological experiment, however they are often distinguished in uncontrolled (and mostly undesired) experimental variability, and those elements that are purposely changed over the samples of an experiment, with the aim of studying how the gene expression varies, depending on the varied experimental factor. In order to make a microarray experiment understandable and repeatable, two fundamental requirements in experiment-based science, not only have the gene and the samples involved in the experiment to be accurately described, but also several related aspects have to be reported too. These include:

- the treatments that the biological materials have undergone, including a description of experimental protocols that have been applied;
- a description of how the microarray devices used for extracting the expression levels are built, especially the standard names of the genes they probe;
- a description of the mathematical processing made to the data, to improve their quality or reliability, for instance which normalization method has been applied to reduce the effect of non biological variability.
- meta-data that describes an experiment, such as a title, a date, authors, associated publications.

1.6 Standard models and formats for microarray data

The need to provide standardised models for representing the elements described above is widely acknowledged in the Microarray community and in the "omics" based science. Microarray data needs to be publicly shared and understand. Comparison of data coming from different laboratories is also relevant for scientific research[19]. Comparing results from different microarray technologies is also important[20].

The main standard that has been defined for this field is the Minimal Information about Microarray Experiment (MIAME) and the Microarray and Gene Expression model (MAGE). The former is an informal guideline, recommending what has to be described when reporting a microarray experiment[18], while the MAGE model is an object model, published as a specification document, that includes UML class diagrams and XML representation[21]. The model provides a formal mean for storing and exchanging gene expression data. The MAGE model is often used in conjunction with other standards. Among these, we worth mention the Gene Symbol Database by the HUGO consortium[22], used to assign common identifiers to genes, the Gene Ontology standard for functional annotations [23], the MGED ontology[24] and the Ontology for Biomedical Investigations, or OBI[25], which are used for the annotation of the experimenatal design[3]. Other standards and public information, which are often linked to microarray data, include: BioPAX[26] and KEGG[27], for Biological pathways, SMBL for Systems Biology models [28].

3 The concept of ontology and the ontologies mentioned here, are aspects described in Chapter 5.

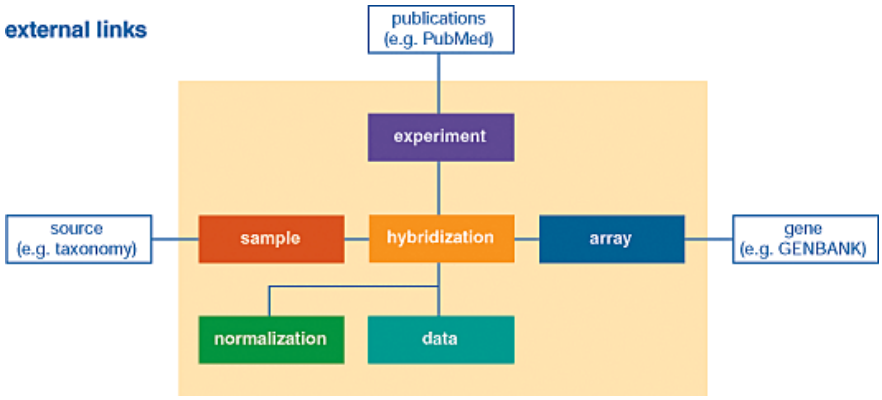


Figure 1.6: main elements in the MIAME guidelines and MAGE model. (Source: [18])

Given that MAGE trades simplicity with comprehensiveness, alternative, slightly less complete, tabular models has been developed. These are usable with Spreadsheet applications and easier to programmatically manage[29][30]. We mention in Chapter 6 the use we have made of one of these tabular formats.

1.7 How microarray data are analysed and which outcomes are produced

We have anticipated in Section 1.4 some of the ways the microarray technology is used. In this section we give some details about the microarray analysis methods.

Almost all microarray experiments are designed so that few independent biological or technical factors are explicitly varied, over different biological samples, and how the gene expression changes, depending on the examined factors, is studied. Typically, microarray measurements are performed on the "baseline" condition and compared with measurement taken from treated samples. This way, the response from samples, where the experimental factors are different than a reference state, may be evaluated. A special case of such an experimental setting is when one of the factors is the time, the so called time course experiments. For instance, some animals are first treated with a chemical compound of interest, and samples are taken at different times, so that the course of gene expression response over time may be analysed.

Once an experiment has been designed and data from hybridisations have been scanned, there are a number of analysis methods that may be applied. A typical analysis is made of the following steps.

- Normalisation. Expression levels are mathematically adjusted, in order to take into consideration noise and biological variability. Normalisation make the data comparable among different measurements. Describing the normalisation procedure used in an experiment is an important information, that may be used to evaluate final analysis results.

- Quality assessment. Several data quality analysis is possible, for instance one may check the correlation of gene expression levels coming from the biological replicas that are about the same experimental condition.

- Extraction of Differentially Expressed Genes (DEGs). A first interesting information, which may be considered in a data set, is the genes that have an higher or lower level of expression, with respect to the base condition. These genes may be the ones that are involved in the particular phenomenon under study. For instance, if yeast cells are irradiated with UV rays, proteins that are responsible for DNA repairing may be highly expressed. Gene sets is the typical formal result that comes from the DEGs finding analysis.

- Identification of expression patterns. Genes or experimental conditions, or both at the same time, are grouped together, according to patterns of similar expression levels. The typical approach used to achieve that is the use of clustering algorithms, which usually give hierarchical classifications of points in a vector space, according to some definition of distance. Clustering algorithms may be roughly divided into supervised algorithms, where final sets are some how a priori established, and unsupervised methods, where there is no a priory bias for the determination of resulting clusters. Which methods are more appropriate depends on the particular analysis being performed. Sets of genes, sets of conditions (and respective samples), combined sets of genes+conditions, are the structures that may represent the results from clustering operation.

- Gene enrichment and meta-data analysis. Interesting genes and clusters which are computed in previous steps may be related to existing public information. For example, supervised clustering algorithms may be guided by gene functional annotations, which are available through Gene Ontology model[31]. Set of expressed genes and corresponding conditions may be analysed in combination with existing models of biological pathways[32]. The formal results of this step may be reasonably characterised by sets of annotated genes, sets of annotated conditions, or the combination of the two. Set annotations may be formalised with structures, for example, as terms coming from ontologies.

As we will see in the next chapter, the main analysis methods, used in the microarray field, lead to results that are representable according to typical patterns and typical formal models. In turn, biological questions may be answered by means of these results and their formal representations.

2 Motivation of the thesis work

2.1 Microarray-related knowledge and possible models

As we have shown in the previous Section 1.7, studying microarray data is a complex process, composed of several mathematical and statistical steps, comparison with related information, and judgement of what the data are telling us, from the biological point of view. This analysis produces results that, in most cases, may be represented according to certain formal models. We now mention some examples.

The computation of differentially expressed genes in a data set may be reported by means of gene sets. Each set is characterised by the conditions under which the member genes are expressed, higher or lower with respect to a base condition. Sets and hierarchies of sets may also be used to represent the results of a clustering algorithm. Hierarchies of sets may be built on the basis of the "part of" relation.

Sets of genes may be related to hypotheses an conclusions about a biological phenomenon under study. Entities like hypotheses may be formalised at different levels of granularity. For instance, one may model the concept of "assertion" [33][34], by adopting basic building blocks, such as "subject, action, object" [35]. A high detailed, domain-specific formalisation could be attached to assertions viewed this way [36][37]. This may be useful both for retrieval purposes and for automatic inference.

Sets of interesting genes and assertions may be related to data which provide evidence for the sets. In turn, data may be linked to the experiment which have generated it, including the description of the experimental design.

Reporting such links, between the experimental activity and assertions that are made on the basis of the experimental evidence, is useful in order to justify the correctness and reliability of the assertions.

Assertions, data sets, experimental procedures, may be evaluated, for instance by a "user voting" system, provided by a knowledge management application or by a knowledge quality tool [38][39].

Finally, data, experiments and biological claims may be evaluated according to their authors, so that people, their professional role, and their expertise on a topic can be relevant aspects to represent and take into consideration. For instance, the fact that a professor, with many publications on viruses, claims the reliability of a data set about infected guinea pigs, may be more relevant than the same claim made by a less expert student.

2.1.1 Typical questions answered by means of microarray data

In the following, We give examples of questions that may be answered considering the concepts described in the previous section.

Gene and gene products. Given the standard names of certain genes, one may want to know the conditions (the experimental factors) under which the genes result expressed, according to some metric, which accounts for the expression level. Another interesting query may be knowing which experiments, hybridizations and biological samples, support a set of expressed genes. Another example is checking if the genes expressed under a given condition are involved in realising a certain biological function.

- *Conditions and experiments*. A typical query of this type is asking which experiments have been made to study a given cell type, or a given experimental factor. Examples include the reaction of the immune system to a known compound, or the study of gene expression in patients having a particular disease.

- *Data, protocols and methods*. A typical question could be if a given protocol performs particularly well, either from a general point of view, or with particular experiment types. The answer to such a query may be extracted from user evaluations, or comparing how many times a protocol is used in a particular experimental setting. Another aspect is that it is important to know whether the reliability of some data may be compromised by the wrong application of a protocol, or by problems that have been detected on a particular experimental device.

- *People, roles, knowledge authoritativeness*. When starting a new scientific investigation, it is often desired to check which publications exist about the object of the investigation, if there are experts in the organisation where one works, if there are experimental data produced by such experts.

2.2 Focus of the PhD project and organisation of the thesis

As it has been shown Section 1.6, the problem of standardising the way microarray data is produced has been widely addressed in literature. The MAGE standard has been proposed that comprehensively covers the representation of how the data are produced. Differently, for what concerns the representation of the results of the data analysis, i.e.: what it is understand from the data, some non microarray-specific work has been started only recently. In this PhD project we try to address this problem. We discuss, in the next two chapters previous work, where similar issues have been considered.

We divided the work into three parts. In a first part we defined a model for representing and managing the kind of knowledge that has been described in the previous Section 2.1. Such a model may be informally considered a kind of ontology. We have developed it by using OWL[40], one of the languages being proposed as part of the Semantic Web technologies. We explain, in Chapter 4, why the Semantic Web suites particularly well our purposes, while our model will be described in Chapter 5.

In a second part, we shows an example of how our model could be used in practical applications. In order to do that, we have developed a simple web application that allow to manage and browse results from microarray analysis, together with data and experiments they come from. As we illustrate in Chapter 6, such an application could be used to build a "gene expression atlas", typically focused on a particular biological topic, where biological information and experimental data are presented in a integrated and effective fashion. That allows the end user to navigate microarray scientific knowledge, according to the semantic links that connect different kinds of knowledge one each other. Moreover, in addition those semantic links which are explicitly provided during knowledge creation stages, additional knowledge connections are computed, by means of the automatic reasoning capabilities which the Semantic Web makes possible.

In the third part, we suggests that several semantics-based applications of our model and our demo application. These include the definition of knowledge queries, the use of a rule-based approach to make inferences that are specific of the studied domain, the use of ranking algorithms that allow to classify the knowledge according to relevance metrics. We will describe the latter point in Chapter 7.

Finally, some conclusive remarks are reported in Chapter 8.

3 Some Computational Solutions for Molecular Biology

In this chapter, we review some approaches and software tools that are used in the Molecular Biology field, with a focus on those systems that are related to microarray data management. We also pay attention to those features which are relevant for what concerns microarray analysis and the management of analysis results. In particular, we discuss those aspects which are present in single works, and which we have modelled in this this PhD project.

More examples, based on Semantic Web technologies, will be described in the next chapter.

3.1 Repositories and web-based solutions

A fundamental application type for microarrays is the data repository application. Often available as World Wide Web solutions, these tools allow to store and retrieve the data generated from microarray hybridisations. Most of them store information according to the MIAME guidelines, described in Chapter 1. Hence they group the hybridisation measurements into experiments, which are also containers for the biological materials the data come from. Moreover, in most cases, the probed genes or sequences, as well as the microarray devices, are described in detail. For example, genomic information from public repositories is reported.

3.1.1 ArrayExpress and Expression Profiler

ArrayExpress[41][42] is the public repository developed by the European Bioinformatics Institute, a well acknowledged EU-funded institute. It has been designed closely following the MAGE-OM model and is an important example of standard-based public repository. Among other features, experiments

stored in the system are coherently annotated with the factors that have been applied and studied. This, in combination with a proper statistical algorithm, make possible to search for experiments where genes of interest result significantly expressed. Search results of this type are visualised in the form of expression profiles, reporting the levels of expression corresponding to different values of the expressed genes. When searching a gene, proper ranking of profiles and experiments which are relevant for that gene is considered. Results are shown accordingly. Visualisation of expression profiles may be considered as a basic feature about the computation and management of biological results from microarray data.

Expression Profiler[43] is a project from the same team who has proposed ArrayExpress, and integrated to the latter. It is a web tool that may be used for performing basic microarray data analysis. Computation of differentially expressed genes and clustering are the basic functions provided. An option allows to save those lists of genes or samples, which have been output as an analysis step. Lists may be saved in a hierarchical fashion, according to the sequence of computations which are issued.

In our work we have applied concepts similar to the one of the gene list. However, while these lists may not shared between users in Expression Profiler, we focus on such lists as a sharable and collaboration item.

Several other repositories exists that publish microarray data, either large general data set, or small boutique of biologically specific data. Gene Expression Omnibus[44] and [45] are the most known.

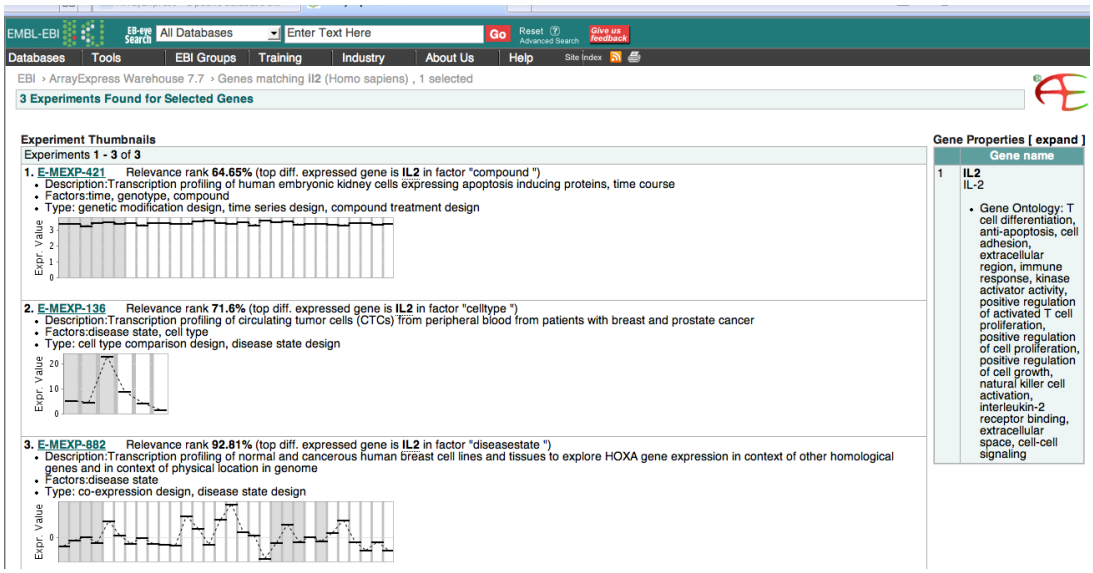


Figure 3.1: a screenshot about ArrayExpress.

3.1.2 The Genopolis Microarray Database

We have started the work described in this thesis when the author was working to the development of a repository software for the Genopolis Consortium[46]. Started as a thesis project, the project aims at being an intermediate solution between single laboratory systems (LIMS) and public repository. As such, it allows for a supervised annotation of microarray experiments which are based on the Affymetrix technology[47]. It is MIAME compliant and supports fine-grained access to the data, from multiple working groups. Gene set management is a feature designed for supporting the collaboration between users. Genes and samples may be searched by matching values from common annotations. Results may then be saved and shared with other users or research groups, attaching textual descriptions to the saved

data sets. The system has been used to store Immunology experiments made on Dendritic Cells. In this context we have defined lists of genes, which plays different roles in the Immunological processes. The lists may be shared and reused in a graphical browser, which is part of the application. This helps the user in performing a first data analysis over the whole set of experiments which are stored in the repository. Once possible interesting data have been selected this way, they may be exported and more accurately analysed with other available tools, such as Bioconductor[48] or GeneSpring[49]. Results from this more specific analysis may be imported and stored back to the repository. Although we have implemented the gene set management mechanism in a rather simple way, it has been proven to be useful for the Genopolis Database users, who are from several collaborative research projects, and who routinely access and share microarray experiments. The feature could be further extended, for instance by complementing the textual descriptions with formal annotations, like the ones described in this thesis.

BASE[50] and MaxD[51] are software tools similar to the Geneopolis Database.

3.2 Collaboration systems

"Groupware" systems are very popular in general. Many commercial[52] and open source solutions[53] of this type exist. Besides the great variety of features available in this applications, many of them are web-based solutions and include common features, such as message forums, calendars, document sharing, personal messaging. Moreover, most of them have a modular architecture, where new features may be added by means of plug-in mechanisms.

Specific collaboration systems for Life Sciences exists too. We now present two of them.

3.2.1 Synapsia

Synapsia[54] from Agilent has been one of the most feature-rich products of this type. It supports the collaboration of multi-disciplinary teams and through a "narrative", hypothesis-discovery paradigm. Comments, office-type documents, data sets and biological experiments may be organised as threads of documented evidence of experimental hypnotises and biological phenomena if interest. Furthermore, drug-industry oriented features are included, such as the management of research project milestones or the management of intellectual property. Synapsia is also integrated with data analysis tools, which makes it a comprehensive product.

With respect to our work, Synapsia is more oriented to the drug-discovery world. Furthermore, it has limited inference capabilities. For instance, it does not support ontology-based annotations, which are useful in Microarray field. Nonetheless Synapsia has inspired several concepts we defined in our project.

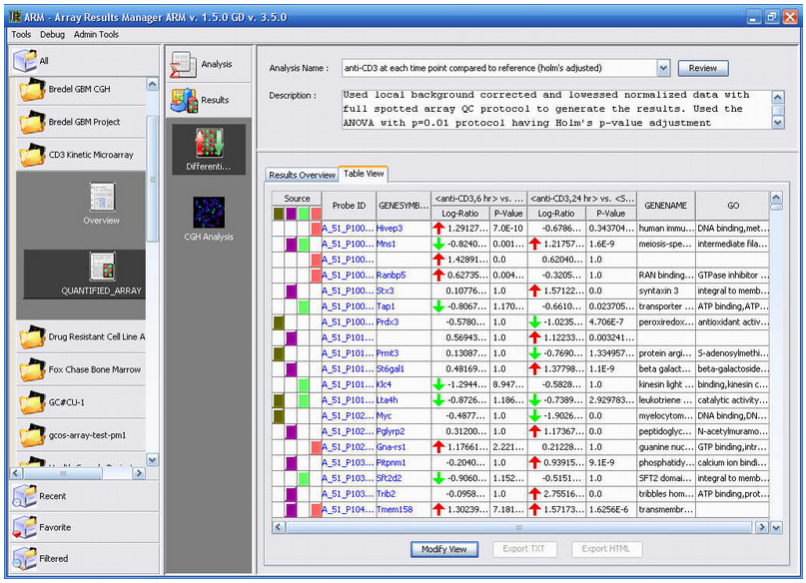


Figure 3.2: a screenshoot about Array Results Manager. (Source: [55])

3.2.2 Array Results Manager

Array Results Manager (ARM) from BioDiscovery[55] is a collaboration software which is microarray-oriented. On one side, ARM is similar to a microarray analysis tool, which allows the user to apply the so called analytic protocols and run an analysis work flow on a data set. In addition to these classical features, ARM allows to save and share results from analysis runs. Results from multiple experiments may be saved together and shared. Other collaboration and sharing features include: high scalability to large data sets, ability to run programmable analytic protocols, centralised and cross platform gene annotations.

Similarly to Synapsia, ARM does not appear to support features like annotation with ontology terms and inference.

3.3 Knowledge-based systems

Modelling of biological pathways, interaction networks, data source integration, are application domains where knowledge-based solutions have been proposed. Formal ontologies, First Order Logics and Logics-based Programming languages, frame-based systems are examples of paradigms used in this context. We show some examples in the following.

3.3.1 BioLingua

BioLingua[56] is both a pathway analysis tool and a data integration system. It is based on a frame system, which provides a unified language for accessing a varied of data sources and for querying genomic data and pathway information. It allows to integrate common analysis tools, such as

ClustalW[57] for molecular sequence search and alignment, with data banks, such as the KEGG[27] pathway repository. It has a collaboration feature, which, by using a wiki interface, allows to share programs and computation results. Highly expressive languages are used in BioLingua, because of the complexity of the application domain. The approach is valuable in integrating disparate pieces of information. OntoLingua is a stand-alone solution. It does not address the distributed nature of biological data, as it is done in other solutions.

3.3.2 HyBrow

HyBrow allows to evaluate hypotheses against a knowledge base of chemical reactions and biological pathways. It has successfully been used with yeast data and signal trasduction pathways. It is based on an "event-centered" ontology, which allows to describe processes like promoter molecules binding to its DNA site, or event's actors, such as mRNA or proteins. It also allows to define hypotheses as chain of events, that may then be verified or explained. It extensively uses inference. Recently the BioPAX format[58] has been proposed as standard, which is similar to the ontology used by HyBrow. Similar modelling of signal trasduction is proposed in [59].

As already mentioned, pathway knowledge is relevant to microarrays, and integrated analysis of pathways and gene expression data helps in understanding biological processes. However, none of the systems presented here have a tight integration with microarray data. The work presented in [60][61] considers microarray applications, although from a pathway analysis perspective. Differently, our project aims at supporting the need to share both data and informal pieces of information, such as comments, textual documents or papers.

The cases presented in this chapter show that in the microarray domain, as in general happens for the Life Sciences, there are a great variety of knowledge kinds, formats and data sources. Integrating such different sources is a step forward in Computational Biology. As we discuss in the next chapter, the Semantic Web is one of the most promising framework to make such an integration possible.

4 The Semantic Web and Life Sciences

It is not our intention to hereby provide a comprehensive essay on the Semantic Web subject. Much literature is available that extensively covers the topic[4][5][6]. Rather, in this chapter we focus to those aspects that are more relevant to the Life Sciences, particularly to our work.

The specific term “Semantic Web” has been originally proposed by Tim Berners-Lee, the inventor of the World Wide Web. Currently the term is mostly associated to the work made by the W3C consortium, particularly to a layered model, the “Semantic Web Cake”[62], which, to use the Berners-Lee words, “bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users”[4].

The World Wide Web, the concept of hypertext, and the HTML language have hugely contributed to the development of a worldwide-accessible network of information and services. The Semantic Web tries to make a step forward, by promoting standards to produce more machine-readable information. This kind of information may be better shared and exchanged by networking software, as well as re-elaborated, for instance, with techniques from the Artificial Intelligence.

The term Semantic Web is also associated to the languages and technologies which are proposed to realise the functions defined for each layer of its architecture, especially the ones which are proposed by the W3C as Semantic Web standards. These languages and technologies are in part still under development, and in several cases competing proposals are available. Furthermore, related work is being done, which have goals similar to the ones aimed at by the Semantic Web. Web 2.0[63], Social Web[64], folksnomies[65] are terms born from that work. Even these few considerations show how intense is the research in these areas of the Computer Science. An intensity which is having an important impact on Life Sciences too, as we argue in the next sections.

Most ideas and formalisms the Semantic Web is based on have been taken from years of Information Technologies developments, including results from Artificial Intelligence, information systems, distributed computing. However, there are novel aspects in the application of previously existing technologies to a context where worldwide access to a great amount of information and services is the main goal. We discuss about such peculiarities in this chapter.

4.1 Universal identifiers

Semantic Web aims at sharing information. One basic step toward such a goal consists in providing unique, worldwide-acknowledged, stable, unambiguous, identifiers, for identifying real entities or concepts. Mankind has defined universal identifiers since even before the advent of the WWW, or even before the Computer Science. ISBN codes used to identify books, or airport codes are just two examples of that. While the adjectives used above for characterising universal identifiers, should be self-explanatory, there are technical and social issues which arises in practice. For instance, the fact that a document identifier, like a web page URI, embeds information which links it to the system where to retrieve it from, may be or may not be desirable. For example, the usage of the Internet domain (DNS) of a particular company or department, as part of a universal identifier, often is not easily accepted, both for social reasons, and because of potential instability of such identifiers (i.e.: often DNS entries expire, are not renewed and hence their disappearance make impossible the associated targets).

Life Sciences has dealt with universal identifiers for long time too. For instance, identifying genes (intended here as DNA sequences) with internationally acknowledged names is of paramount importance. As for that, the Human Genome Organization (HUGO) has defined a public nomenclature for genes[22].

A URI which resolves to a document:

```
http://mged.sourceforge.net/ontologies/MGEDOntology.owl#CancerSite
```

A URI which does not resolve to a document:

```
http://www.brandiz.net/Entity
```

A URN, based on LSID standard:

```
urn:lsid:ncbi.nlm.nih.gov.lsid.biopathways.org:pubmed:12441807
```

Namespaces may be defined as shortcuts for URIs:

```
mged := http://mged.sourceforge.net/ontologies/MGEDOntology.owl#
mged:CancerSite
```

Figure 4.1: identifiers for the Semantic Web.

In the WWW and the Semantic Web contexts, the concept of Uniform Resource Identifier (URI) is defined as basic building block of Semantic Web languages[66]. A URI uniquely identifies a "resource". A resource is basically whatever may be formally described, such as a concept, a web page, or an existing physical object[4]. URIs may both be location-independent, in which case the term Uniform Resource Name (URN) is used, or they may embed information for retrieval, such as a network protocol (e.g.: http) and a server name. in this case, the term Uniform Resource Locator (URL) is often informally used[5].

4 We assume an intuitive definition of "existing physical object".

5 URI is a formally defined format, while URL is more ambiguous. Since every URL may be seen as a URI too, we would not need to talk about URL at all. The expression is still used because of historical reasons. See [URI] for details.

In the Bioinformatics field, Life Science Identifiers (LSID) are being proposed to identify digital entities related to the Life Sciences[67][68]. LSIDs are a particular class of URNs (with "urn:lsid" prefix) and a resolution protocol is defined to retrieve the piece of information these identifiers target. Although LSIDs are an elegant and effective solution, they either are affected by social and technical problems, as it happens in general for universal identifiers[69][70].

4.2 Information Integration with semantic networks

Another basic building block of the Semantic Web is the Resource Description Format (RDF), which is essentially a format for dealing with semantic networks of nodes and edges labelled with URIs[71].

The semantic network concept is well known in Computer Science[72]. A semantic network represents entities from the real world, or concepts, by means of the nodes in a graph, while conceptual relations, or other kind of relations, are represented by the labelled arcs that connect the nodes. In Figure 4.2 we give an example of such a network. One of the strong points in semantic networks is that the labels which mark the links between entities may be semantically and formally described. For instance, in the example in figure, the property "write" may be described as a sub-property of "produces", or as the inverse of "author".

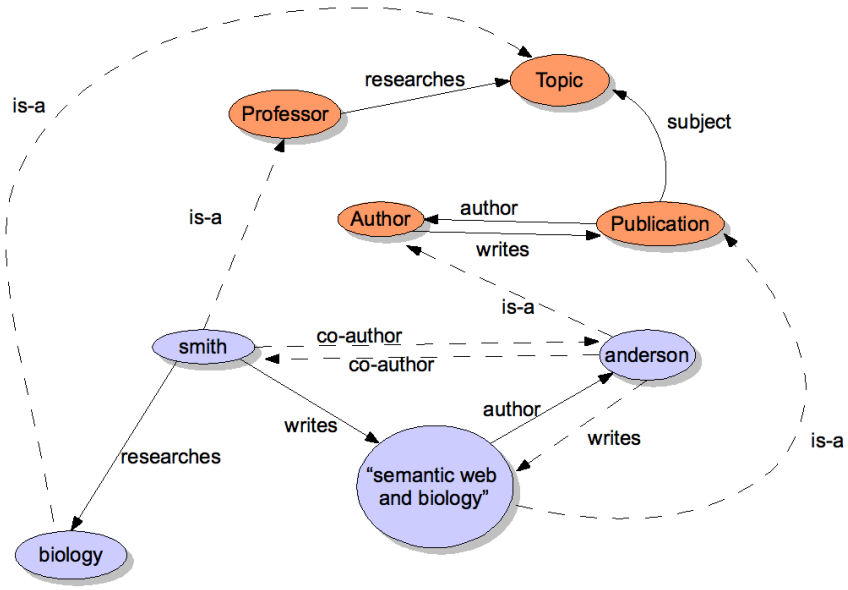


Figure 4.2: an example of semantic network. Inspired by [73].

Particular relations may be introduced to classify and characterise the nodes. For example, the original network in figure, about authors and publications, is extended with the relation "is-a", to describe how the original nodes may be classified according to a network of concepts. The relations between such concepts may be reported by means of semantic networks too. In the example, the domain and the range of "writes" relation are represented. Characterising edges and nodes of a Semantic network makes

possible to infer more knowledge from the knowledge which is explicitly stated. For instance, in our example, the relation “Smith is a Professor” is drawn with a dashed line, meaning that it was not explicitly provided, it was automatically computed from the fact that “Smith researches in biology”.

Similarly, the “co-author” pair of relations between Smith and Anderson (one per direction) may be inferred by the inverse of “author” and the fact that two writes occurs, pointing to the same publication.

Another feature of semantic networks, is that, contrary to object oriented modelling, or the relational theory, there is no need to have a priori defined schemes, with rigid structure, such as a fixed number of properties for a given class. This is another fundamental aspect of semantic networks: they are very suitable for representing heterogeneous information, especially when multiple information pieces are integrated from different data sources.

RDF may be considered a formalism to define semantic networks, by means of URIs, and to convert the resulting semantic network in a flavour of formats, such as RDF/XML[74] or N3[75]. In the RDF jargon, semantic networks are seen as a set of triple statements, where a triple of subject, property (or predicate, or relation) and object, defines the statement (like in the example above about the role of Smith). The use of URIs and the definition of standard syntax are the novel aspects, driven by the Semantic Web objectives, with respect to traditional semantic networks. The URIs make possible to work with entities identified in a standard way. The adoption of the semantic network paradigm, may be considered a natural evolution of “linked pages”, which is an essential concept in hypertexts. Indeed, the the simplicity and power of web links account for the great part of the enormous success of the WWW. RDF may be considered a way to define “typed links”, where links may be characterised with meaning, rather than being flat, as in the HTML language. Moreover, RDF makes available a formalism which is more flexible than, for example, XML.

Integration easiness and flexibility are two characteristics of great interest for Life Science applications. For instance, consider the example in Figure 4.3, inspired by[76], where spots on 2D electrophoresis gels are represented. The use of RDF makes easy to retrieve spots from a gel repository and to integrate their meta-data (such as ID or name), with information coming from analytical information repositories (e.g.: shape, shape size and location). The fact that an item is a spot makes possible to infer that it must have a shape. Configuration information could be provided to access repositories of shape data. As an example of flexibility, let us consider the case one needs to describe a new type of shape for spots. This information could be easily added to the graph in figure, by adding new RDF statements. A system able to deal with at least general properties of shapes (e.g.: the area) could automatically work to the newly added RDF.

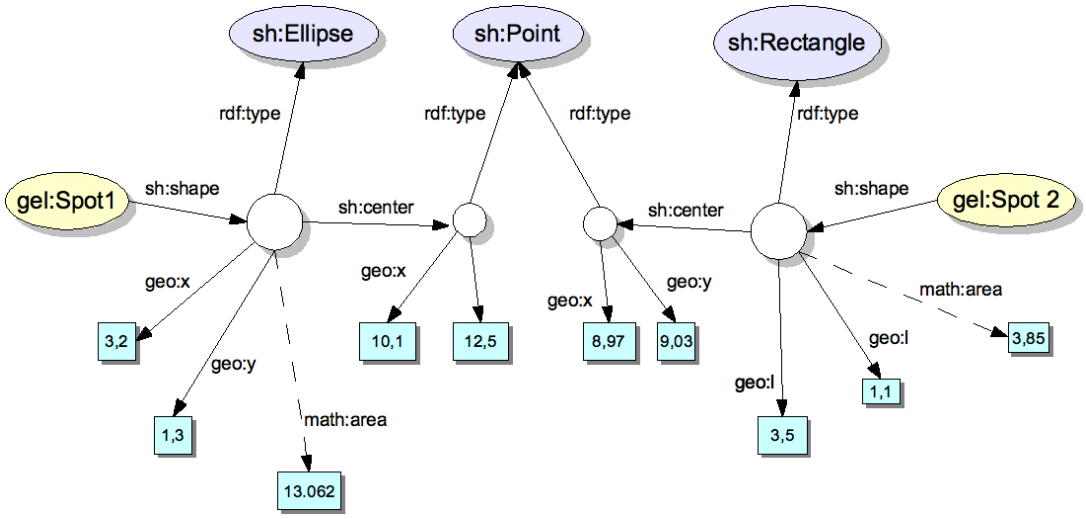
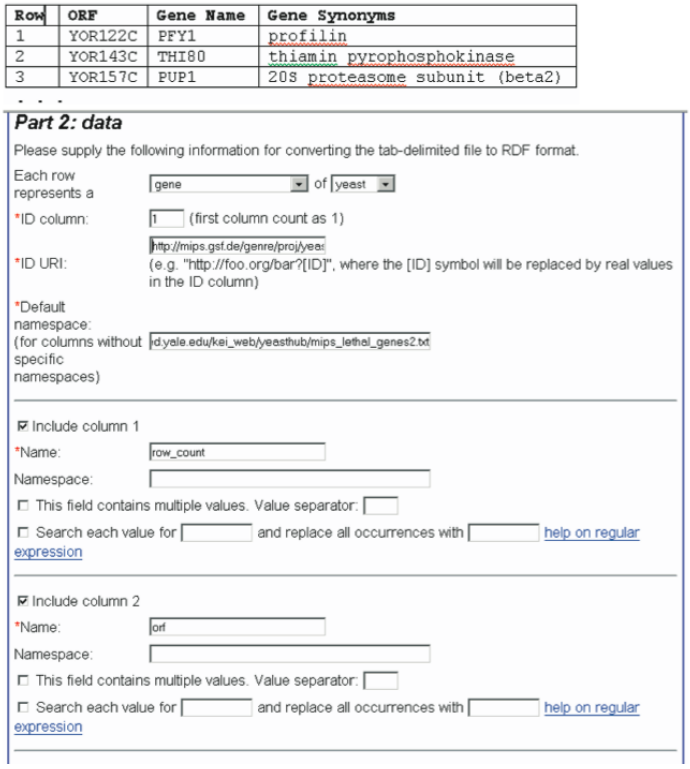


Figure 4.3: RDF used for 2D gel representation. Inspired by: [76].

There are many large sets of a variety of data sources which have been integrated into a large data warehouses, or which have been made accessible by means of federated systems[77].

As shown in the example above, RDF address particularly well such integration. YeastHub[78] and LinkHub[79] are two applications of this type. YeastHub is a data warehouse, where records from public databases are represented as RDF resources. A typical conversion that the system makes, is from tables in a relational database into RDF statements (Figure 4.4). Every record from a particular table corresponds to an instance of some concept, while the field values of the record are used as properties of the instance, assuming that a column name corresponds to a conceptual property. Similarly, relational foreign keys may be converted into relations between different RDF "object properties", which links two concept instances, rather than primitive values (e.g.: string, numbers, dates, etc.). This approach, of mapping relational schemes to RDF statements, is so common that tools are being developed for applying it. YeastHub uses D2RQ[80] as one of this tools. New data sources may be added to the system, by means of an administrative interface, where the data set may be described by attaching meta-data to it (the Dublin Core format is used[81]). Additionally, a new data source may be mapped into YeasHub, by defining the D2RQ mapping for the data, optionally using the visual interface. Queries may be built in YeastHub, that are able to search and retrieve data across multiple sources, mainly thanks to a unified schema, which is realised by the mapping above.



Row	ORF	Gene Name	Gene Synonyms
1	YOR122C	PFY1	profilin
2	YOR143C	THI80	thiamin pyrophosphokinase
3	YOR157C	PUP1	20S proteasome subunit (beta2)

. . .

Part 2: data

Please supply the following information for converting the tab-delimited file to RDF format.

Each row represents a: gene of yeast

*ID column: 1 (first column count as 1)

*ID URI: http://mips.gsf.de/genre/proj/yeas
(e.g. "http://foo.org/bar?[ID]", where the [ID] symbol will be replaced by real values in the ID column)

*Default namespace: (for columns without specific namespaces) id.yale.edu/kei_web/yeasthub/mips_lethal_genes2.txt

Include column 1

*Name: row_count

Namespace:

This field contains multiple values. Value separator:

Search each value for and replace all occurrences with help on regular expression

Include column 2

*Name: orf

Namespace:

This field contains multiple values. Value separator:

Search each value for and replace all occurrences with help on regular expression

Figure 4.4: an YeastHub form to map relational tables into RDF. (Source: [78])

LinkHub is a similar project, sharing some of the YeastHub's authors, which is focused on Proteomics. In LinkHub, "hubs" of data warehouses may be integrated and linked together, by exploiting RDF representations. This means it is possible to join, as in a federated approach, small warehouses of specific topics. The kind of heterogeneous information which may be integrated and cross-queried in both YeastHub and LinkHub clearly show the potential of RDF and Semantic Web technologies. As a further proof of that, examples of queries which span across both LinkHub and YeastHub have been shown by the authors.

Another example of information integration achieved by means of the Semantic Web, is the demo developed by W3C group named "Semantic Web Healthcare and Life Sciences Interest Group", or HCLS[82]. The demo, contributed by individuals from many research organisations and commercial companies, focuses on the Alzheimer's disease, for which many different source of information are available, including genome, molecular pathways, spatial localisation of gene expression. RDF conversion of such information has been produced for the demo. The execution of sample queries, written in the standard SPARQL language[83], has been shown. In computing query results, some inference mechanisms are exploited.

A different approach for information integration is followed in the BioDASH project[84], where the focus is on the merging of data at the user presentation level. Similar projects, non specific to the Biology fields, are described in [85][86].

We conclude this section by mentioning another type of computational integration, which is relevant to Bioinformatics: service integration and distributed services. The most relevant projects about this topic are BioMOBY[87] and myGRID[88], which mainly aim at distributed access of computational services

for Biology, and at standard composition of service work flows. For instance, the discovery of new transcription factors may be a combination of sequence search on public databases and microarray analysis. A complex analysis pipeline may be built with the Taverna tool[89], available from the myGRID project, whre visual composition of publicly available services is possible. In fact, services are represented in a standard way, similarly to what it is done the Web Service technologies[90]. Semantic Web is relevant in this kind of applications as well, particularly because RDF and ontology languages (described in the next section) are basic building blocks for data exchange between services. They are also a powerful formalisms to describe the service semantics (i.e.: what they provide and how). Semantic description of services is a basic step to enable their searching, either from human users, or from automated tools. Although these topics are outside the scope of this thesis, they are an important area of research, both in the specific Bioinformatics field and in general.

4.3 Conceptualisation and ontologies

Conceptual modelling is another area of much interest for both the Semantic Web, biological and medical applications. As it is well known, the term “ontology” has been borrowed by Phylosophy and the currently wide accepted definition of what ontologies are, from the Computer Science point of view, is the one given by Gruber[91]. According to the Gruber's paper about the Ontolingua ontology framework, ontologies are defined as formal specifications of the concepts and the conceptualisation which are used by mankind to refer the world they perceive and interact to. Such a conceptualisation activity includes: defining the concepts describing a particular domain; classifying real or conceptual entities, according to defined concepts; defining general relations among concepts, such as “sub class” or “being a part of”; defining domain-specific relations, such as “being transcription factor for”; formalising the nature of concepts and relations, for instance by considering the symmetry or transitivity of properties.

Guarino[92] has further clarified the relationship between the world of concepts, which is abstract and language-independent, and formal encoding of concepts, which is what usually is indeed meant as ontology in Knowledge Engineering. Such a distinction suggests that much attention must be paid to the design of formal ontologies, since the final result may significantly deviate from the real meanings which are intended in real world, when people apply the terms used by a particular ontological specification. This may lead to disastrous consequence in concrete applications of such badly modelled ontologies. This kind of problems are also a major concern in the field of biological ontologies[93][94].

Another issue in Life Sciences, is the fact that the word “ontology” is intended in many different ways. As it is described by Mc Guinness [95], from whose work the Figure 4.5 is taken, a range of increasing complex models may be meant as ontologies. These include: simple controlled vocabularies, taxonomies with informal classification, strictly defined classifications, complex formal models, made with First Order Logics.

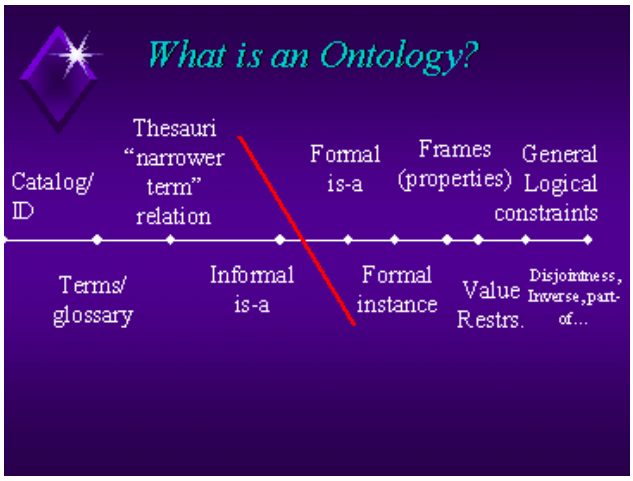


Figure 4.5: a range of different models are considered ontologies. (Source: [95])

In the Semantic Web stack, a simple way to make ontology-like models is the usage of RDF-Schema, or RDF-S[96]. This is a simple language which allows to use classes, the proprties subclass and instantiation, as basic conceptualisation tools, in addition to simple definition of properties and sub-properties. RDF-S itself is encoded in RDF and RDF-S semantic networks can easily extend networks about a particular piece of knowledge, similarly to what has been discussed above.

Logics is a popular tool for advanced ontology modelling[97]. In particular, Description Logics[98] is the base of the OWL standard, proposed for the Semantic Web. Description Logics(DL) is a subset of First-order Logics, specifically designed for knowledge bases where there is a strong focus on concepts and relations between concepts. DL has constructs to define sets of concept instances, including concepts defined by means of relation restrictions. Value restrictions, which axiomatically define the range of a relation, are an example. The success of DL is also due to extensive studies on its decidability and its computational complexity, which have produced different flavours of DLs, as well as algorithms for automatic reasoning tools, like Racer[99] or Pellet[100], which implement such algorithms. Finally, DL constructs may easily be translated into the RDF format, since its logics derives from the research on Semantic Networks.

A high number of ontologies have been defined so far in the Life Science field, an annual conference on bio-ontologies has been taking place for several years, and an "Open Biomedical Ontology Foundry" has been established[101].

An ontological approach in Bioinformatics is of great importance, because of the complexity and variety of concepts occurring in a science which study a large, multi-scale amount of complex phenomena, where multiple disciplines are on the scene, with different and often conflicting or redundant terminology. Common formal terminology, classification, finding instances of a given concept, consistency checking, are but a few examples of many possible uses of biological ontologies.

One of the most popular bio-ontologies is Gene Ontology, or GO[23], which is actually a set of three taxonomies, based on "is-a" and "part-of" relations, aimed at functionally characterising genes and gene products. While GO is not much "ontologically complex" it has given an invaluable contribute to the research in Functional Genomics. Functional annotation, term enrichment [31], text mining[102], are just few examples of the wealth of available applications.

OBI (formerly known as FUGO), the "Ontology for Biological Investigations" is another example of a large taxonomy-oriented ontology, which is being developed for supporting the description of biological experiments, including the support to experiment's objectives, materials and protocols used, type and characteristics of measurements. OBI wil eventually replace the Microarray Gene Expression Ontology (MGED Ontology) an early similar ontological effort, developed for the microarray field[25].

BioPAX[26] is an OWL model for describing molecular interactions and Biological pathways. Despite some confusion between object oriented approaches and ontological approaches, BioPAX may be considered an instance of biological ontology, modelled in OWL, contributed by a large community and proposed as a standard. BioPAX is layered on different levels, which cover metabolic pathways and other molecular interactions. BioPAX is being used in a number of public resources, such as Reactome[103] or BioCyc[104]. It is also used in software tools like Patika[105], or Pathway Tools [106].

Information integration and ontologies are often bounded. ONTOFUSION[107] is one example of such bound. ONTOFUSION is an Information integration system, similar to the previously described YeastHub. The relevant difference is that the mapping between relational schemes and RDF is built by means of domain ontologies, modelled in OWL, which are then unified into a general unified schema, modelled in OWL too. This makes possible complex queries against a common OWL knowledge base, which wraps an heterogeneous set of databases. Furthermore, as we discuss in the next section, OWL modelling is a starting point for the computation of new knowledge from existing one, by means of automatic reasoning.

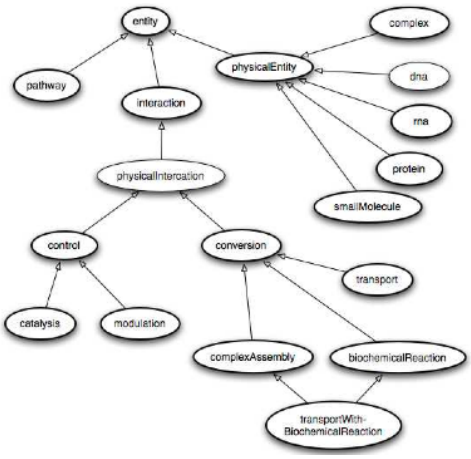


Figure 4.6: main top-level classes in the BioPAXformat. (Source: [60])

4.4 Inference and automatic reasoning

Part of ontologies usefulness lies on the entailments which are automatically computable from ontology-based knowledge. We have already mentioned some of the inferences which are possible with ontologies. To summarise, the most common cases of ontlogical inference are the following.

- Subsumption: infers if a class, or concept is a subclass of another one (the latter concept subsumes the former). In simple cases, the transitive closure of the "is-a" relation suffices to make this kind of inference, for instance the "transport" class in BioPAX may be easily identified as a subclass of "interaction" by considering the chain: "transport is-a conversion is-a physical-interaction is-a interaction". In other cases more advanced reasoning is necessary. For example, the fact that an uncle is a kind of person may be computed from:

```
uncle = ( has-brother or has-sister ) ( has-son )
range ( has-brother ) = Male
range ( has-sister ) = Female
Person = Male or Female
```

- Instantiation inference: a particular case of subsumption, consisting in establishing if an instance of a class belongs to other classes too and which are such classes.

- inference from the domain and range of properties: this is often misunderstand by people with object-oriented programming background. For instance if "myHouse has-sister nextHouse" is defined, due to the axiomatic nature of the OWL logics, nextHouse is classified as a female person, rather than yielding an inconsistency. This is because of the classificatory purposes of DLs and ontologies.

- Inference based on cardinality constraints: for instance from "hand part-of only one arm" and "john-hand-1 part-of john-left-arm", "john-hand-2 part-of john-left-arm", we have the conclusion that hand 1 and hand-2 are the same resource. This is another tricky case: no inconsistency is produced, because OWL is based on the so called "open world assumption". Essentially: nothing is true or false until explicitly stated. Another OWL principle is the non univocity of identifiers: while a single URI identifies a single resource, the same resource may have more than one URI. These principles are all due to the open domain the Semantic Web technologies are supposed to be used on: a worldwide web of information, which is only partially visible to single computational agents.

Apart from the inference with the DL semantics, more automatic reasoning is possible, by considering the properties and the classes defined for a particular domain. A common way to make other type of inference with RDF is by using inference rules. A standard for rules is being proposed to the W3C as part of the Semantic Web. Rules allows, in the form of "if certain premises are true, deduce certain conclusions", to extract more knowledge from existing one. They are easy to define and expressive. However, they may easily lead to excessive long time computations, or even to infinite loops.

Automatic reasoning and inferencing with rules, would be of great utility in Life Sciences. Although their usage is still limited, mainly because of performance and scalability problems, we may expect more research in this field in the future.

A work that relies on OWL-based reasoning is presented in [108], where a system for dealing with the phosphatase protein family is described. The characteristics of protein domains, i.e. the parts of a protein having a functional role, are modelled in OWL. New domains in phosphatase proteins have been identified by classification activity, which shows the validity of this modelling approach.

Another work that puts together data integration, ontological-based inference, and rule-based inference, is described in [61], where a method is presented for analysing pathways modelled in BioPAX and other ontologies, like GO. Pathways may be queried with predefined SPARQL queries and useful results

are returned by applying reasoning over OWL semantics. Additional entailments are possible by defining custom rules, for instance, some rules could define general relations such as “interacts”, which may include several type of BioPAX interactions. The system allows also to make an integrated analysis of pathways and microarray data, which is useful to gather insights about molecular interactions.

A even more technologically comprehensive example of Semantic Web application is the one proposed in [109], where a system for Translational Medicine is presented. Translational Medicine aims at better integration between base research and clinical application. The mentioned system is similar to a traditional decision support system, with a focus on the application of Semantic Web technologies. In particular: several clinical and genomic data are imported from existing laboratory repositories (LIMS) and integrated together by means of RDF and OWL; a “translational medicine ontology” is defined in OWL that models the relationship between diseases and their genetic causes. A decision support system is designed and implemented, which exploits the OWL knowledge base and defines inference rules for its purposes. As in other similar projects, the system takes into account the way real people make clinical decisions on the basis of clinical evidence and available medical knowledge. Systems like this are among the most interesting examples of Semantic Web applications, where the experience gained in the field of expert systems may be extended to distributed, large-scale biomedical software. They are also a stimulating use case for improving performance and scalability of Semantic Web tools.

A final kind of possible computations with RDF we worth mention are the ones based on graphs. For instance in Social Network Analysis[110][111][112], network of social relations are considered and typical patterns are considered, for instance: the connectivity of a node, or the clustering of a network into sub-networks of social groups. Another type of graph based analysis is that which aims at ranking knowledge, namely, resources and RDF statements. We will present some work on this topic in Chapter 7.

4.5 Limits of the Semantic Web

Semantic Web is a pioneering technology and it is still unclear the extent of success it will have in the Life Sciences. We conclude this chapter by mentioning some critical aspects of the Semantic Web technologies.

One difficulty is given that the design and usage of sound ontologies is not easy. Both knowledge engineering expertise and application domain expertise is often necessary to design ontologies. Another problem is that people from Biology and clinical side have no immediate interest in pay attention to formal description of their research activities.

The main approach that allow to deal with these issues seems to be the development of new tools and new computational methods for automatic ontology building and automatic ontology alignment. Extracting formal knowledge from informal piece of information help as well. This is the case of many text mining applications.

Similarly, standardisation in Life Sciences is not easy: many aspects of the same piece of reality are often considered by different groups and merging them together in a uniform standard is a complicated task, which requires the participation of many members from the domain being standardised, as well as an adequate financial support.

A different problem lays on the performance of Semantic Web tools, especially the performance of reasoners and rule engines. At the moment, most of inference systems keep all the RDF knowledge base they work with in memory, no distributed solution is still available. This fact, for technologies that are focused on the "web", is a significant deficiency.

Nonetheless the issues above, the Semantic Web appears to be the right way to produce better knowledge integration and representation in Life Sciences.

5 MannOnto: an OWL model for the representation of microarray knowledge

The MannOnto Ontology (Microarray Annotation Ontology) is an OWL model, developed for the representation and the annotation of Microarray knowledge. As already anticipated in the previous chapters, its main purpose is to allow to describe the outcomes resulting from microarray analysis. Such outcomes may be linked to the data providing evidence for them. Furthermore, analysis results and data may also be associated to the people working with them, who, in turn, may be described in terms of their roles and their professional relationships. Although the ontology shares some concepts and names with the MAGE-OM object model, it has different goals. In fact, the main purpose of MAGE-OM is describing the experimental design and its data data at an high level of detail, so that one may exactly understand all the steps performed to realise the microarray experiment and possibly reproduce it. MannOnto has different aims, listed in the following.

- abstract representation of most important data analysis results, so that they may be browsed quickly and related to biological knowledge that is achieved from analysing the experimental data.

- data quality issues: we are interested in representing possible issues with data that allows to derive results. For instance, let's assume an expert realises a certain data set has reliability problems (e.g.: house keeping sequences are not present or statistical quality controls reveal problems). MannOnto allows to link the issues about the data set to the biological conclusions that are stated analysing the data.

- promoting the collaboration: for instance, a data set may be commented with considerations about its biological meaning, or an experimental protocol may be recommended, for its good performance. Moreover, the ontology allows to represent the people who are studying a given set of genes, or find the colleagues of a paper's author, or who is expert on a given biological topic.

- ranking the knowledge: for instance, if an assertion about gene expressions has been derived from many experiments, the assertion should be considered more relevant than another one, which is linked to one experiment only. Another more complex example, is about biological assertions whose support

consists of data produced by people who are relevant for that kind of data. For instance, a data set about dendritic cells, produced by a laboratory which is internationally acknowledged in studies about dendritic cells, should be highlighted during data searches. MannOnto also allows the users to directly assign evaluations to knowledge, as explained in this chapter.

We conclude this introduction by briefly mentioning the fact that so far we have not used the word "ontology" to define our model. As discussed in the previous chapter, this word is associated to many different kind of knowledge models, especially in Life Sciences field. We have worked on the definition of an "OWL schema model", a set of classes and properties, mainly using the Protegé tool[113]. We have not followed any particular school of ontological development for the Computer Science, mainly because we care about application aspects and we mostly integrate already existing models and ontologies. According to the definitions in [97], our OWL model may be considered an "application ontology". In the next sections we will informally use the term "ontology" to refer MannOnto.

5.1 Entities

Every class in MannOnto is a subclass of the `Entity` (and therefore every class instance). Although this is not strictly rigourous, from the Ontological Engineering point of view, it allows a certain degree of control at the application level. For instance, one may easily use properties, such as `entityName`, that are common to all class instances dealt by the ontology. The following properties are defined that have `Entity` as domain.

- `entityName`, `entityTitle`, `entityDescription`: allow to define a short name, a title, a long description. These are functional properties, capturing the fact that, for instance, an entity may have only one name.

- `entityCreationDate`, `entityDate`: allow to associate a creation and last change date. They are functional properties.

- `evaluation`: allows to evaluate an entity and to use evaluations to rank knowledge according to relevance or quality criteria. More details about this aspect are provided in the next sections.

- `entityTermAnnotation`: allow to annotate an entity with a term from a taxonomy or a controlled vocabulary.

5.2 Main concepts

In this paragraph we give a short overview of the main concepts defined in MannOnto (i.e.: entity types, as described above). More details will be presented in the next sections.

- `GeneExpressionEntity`: this is what MannOnto aims at manage. It refers to conceptual or concrete objects belonging to the Gene Expression domain. Most subclasses of `GeneExpressionEntity` comes from the MAGE-OM model.

- `CommunityEntity`: this has subclasses like: `Person`, `Conference`, `DocumentEntity`. It targets the representation of the scientific communities working in the microarray domain, their relations with microarray entities (such as the authors of experiments) and their activities, especially those about cooperations, events like conferences or artifacts like papers.

- `Term`: instances of this class may be used to represent taxonomies of terms, such as Gene Ontology (all the three sub-ontologies), NCBI Taxonomy or MESH. These are examples of ontologies without strict hierarchical relations, which are much used in Life Sciences.

- `Assertion`, `AndAssertion`, `OrAssertion`, `IsExpressedAssertion`: these classes of MannOnto may be considered a way to reify statements which have context information related. Thanks to contextualised assertions, and to the use of the "`evaluation`" relation, it is possible to rank the knowledge according to metrics like reliability or significance. Details will be described in this chapter.

- `Measurment`: subclasses of this class may be used to provide a measurement, including a numeric or enumerated value and a unit, about a measurable quality. `FactorValue` is an example of Measure, representing a particular experimental condition, which is investigated through one or more microarray hybridizations.

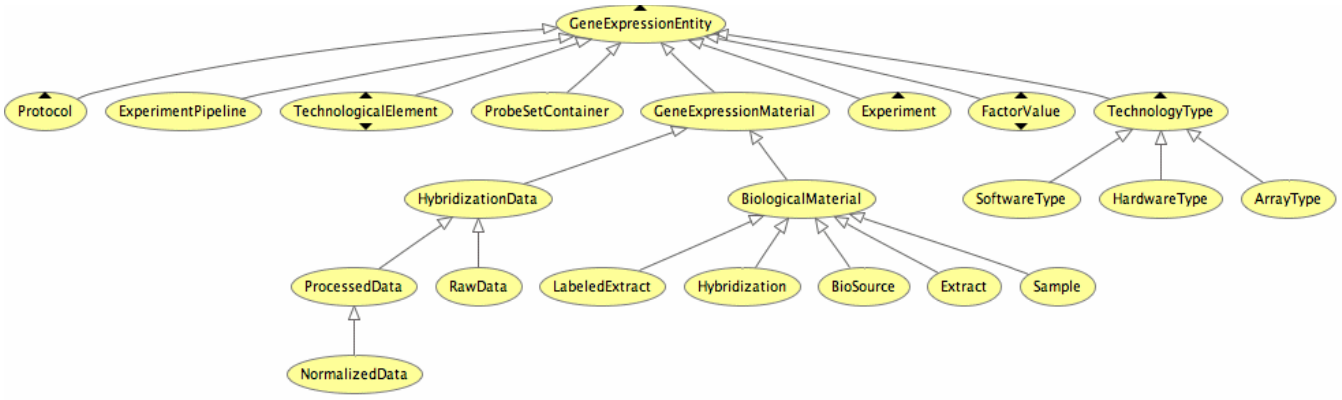


Figure 5.1: entities about microarray experiments.

5.3 Details about entity types

5.3.1 GeneExpressionEntity

The concepts defined in this sub-tree of the ontology mainly correspond to the ones defined in the MAGE-OM standard (and partly to terms defined in MGED Ontology). Our aim is to report those aspects of microarray data and experiments which may be used to provide evidence to biological assertions. According to this goal, the following classes have been defined.

- `Experiment`, `FactorValue`, `BiologicalMaterial`, `Hybridization`, `HybridizationData`: allow to describe the data achieved from experimental activity. Typically `HybridizationData` will be used to report the list of genes which have a relevant expression profile in an experiment (e.g.: they are DEGs). These data may be linked to the factor values they refer to, or to the biological materials they come from.

- `ArrayType`, `Array`, `Protocol`, `ProtocolApplication`: allow to trace technologies, physical devices and procedures which are used to produce experiments and data sets. Even in this case, we are interested in modelling possible problems, or other aspects about a device.

- `ProbeSetContainer`: allows a lightweight representation of an array design element. This corresponds to the composition of molecular entities, to which a unique expression level is eventually associated (e.g.: several fragments of DNA, located in different spots of the device, aiming at probing the expression level of a single gene). We consider the description of array manufacturing details out of the scope of MannOnto. Also, we use instances of `ProbeSetContainer` to report annotations like UniGene accession or Gene Ontology term. We do not aim at rigourous formal modelling of concepts like genes or proteins, for two reasons: first, because this is already the scope of other standardisation efforts, like BioPAX, which could be integrated in MannOnto in future and second, because concepts like "gene" are still controversial in literature, while "pointers" like probe sets are much more clear. This approach has been inspired by the GandrKB project[114].

- `ExperimentPipeline`: this is a reified relation. Microarray experiments may be described by tree-structured graphs, which report the usage dependencies between starting biological materials (the biological source), intermediate materials, their treatments, and the final data (Figure 5.2). A single pipeline is a single path, from an initial source material to a final data set. Pipeline has members, which are the nodes of this path. We introduce this class in order to simplify the applications like the one described in the next chapter.

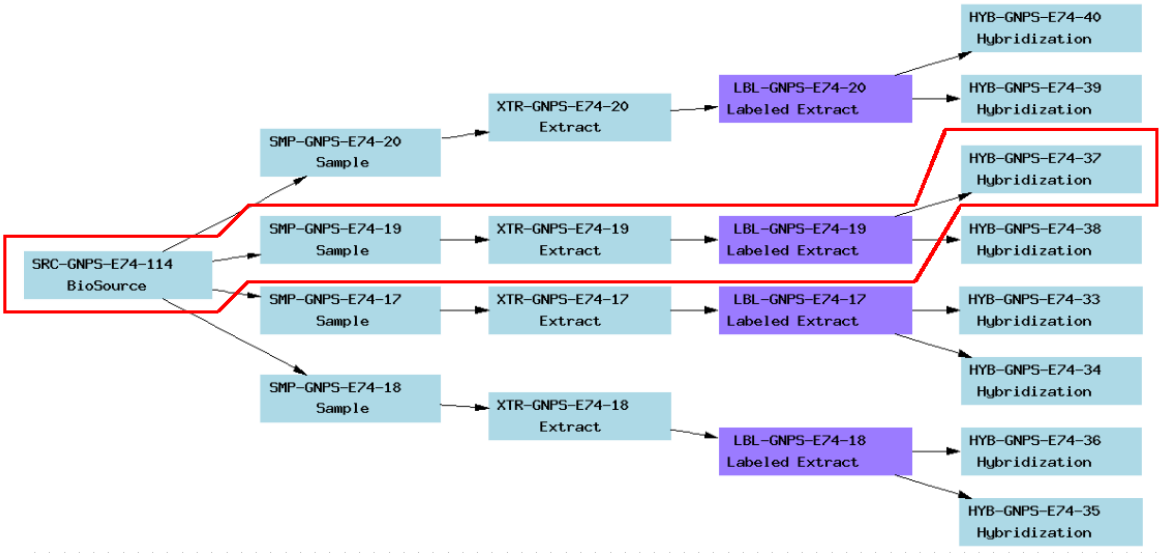


Figure 5.2: typical material graph in a microarray experiment. The `ExperimentPipeline` *concept refers to a single source-to-data path, like the one highlighted.*

We have defined a set of OWL properties, which may be used to relate gene expression entities. `geRelatedGEntity` is a root property that generically allows to state such relations. Several sub-properties of are defined for it. These include the following.

- `usesGEntity`, which in turn has the sub-properties: `arrayProbes`, `usesArrayType`, `pipelineProtocol`, `pipelineMaterial`. These allow to report that a device or procedure has been used to generate some data. As mentioned above, such usage dependencies helps in relating results from gene expression analysis to the data analysed.

- `geMaterialTypeAnnotation`, `geExperimentalFactorAnnotation`, `geMaterialCharacteristicAnnotation`, `experimentQualityControlAnnotation`: these may be used to annotate gene expression entities with taxonomy terms, such as the ones imported from MGED Ontology.

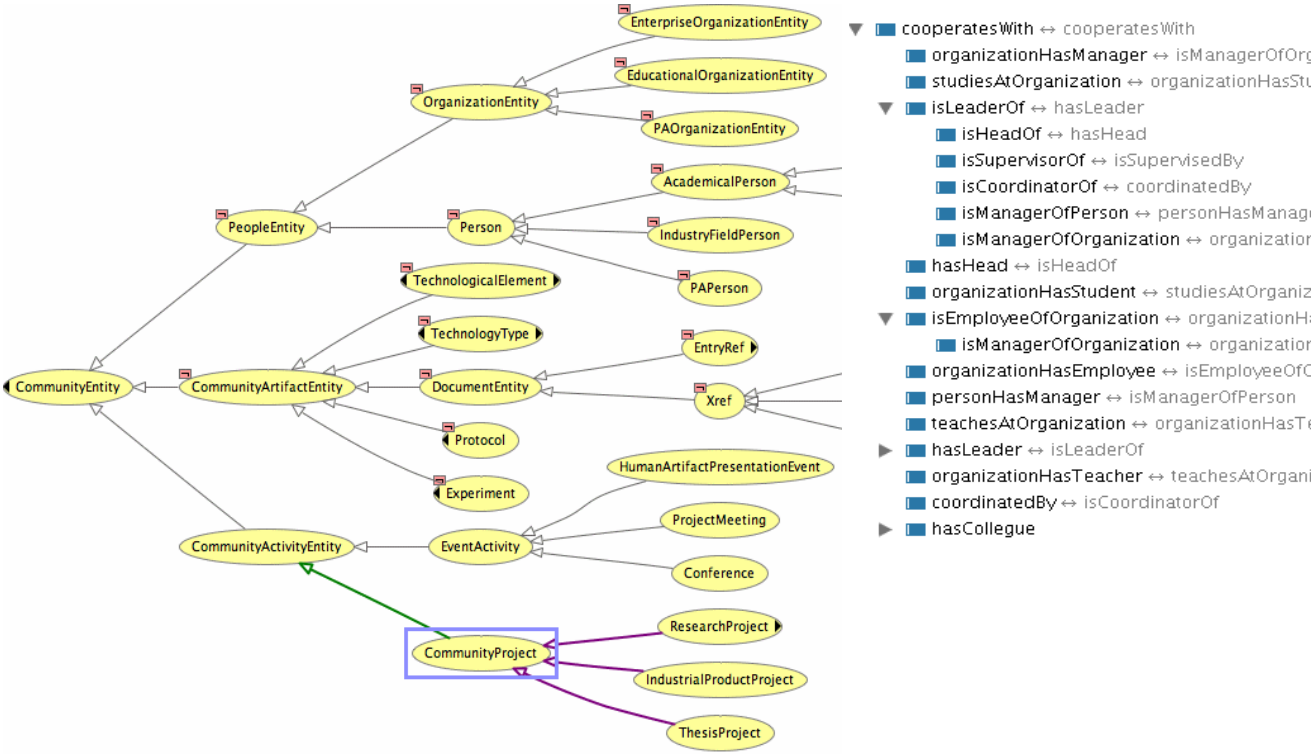


Figure 5.3: the modelling of people, their role and the artefacts they produces.

5.3.2 CommunityEntity

This part of the ontology models people, their roles and the main artefacts which people use to collaborate. We have three subclasses at the first level of this ontology branch.

- `PeopleEntity`, which has `Person` and `Organization` as subclasses. It models the persons and the organizations (companies, Universities, etc.) people work for or they have other relations with.

- `CommunityActivityEntity`, which allows to represent activities like research projects, or conferences and workshops.

- `CommunityArtifactEntity`, which include concepts like `DocumentEntity` or `SoftwareType` and which models artefacts that are used in the scientific activity and in the Microarray domain.

The main purpose of this part of the ontology is to trace people who produce microarray data and conclusions from the experimental investigation. This is useful for several reasons. For instance, one may be interested in knowing which university has many publications on a given topic (e.g.: `peopleProducesArtifact` property could be used), possibly related to a data set or a list of genes. These connections among people (e.g.: `cooperatesWith` or `peopleActivityParticipant` properties),

the artefacts they produce and gene expression entities may be useful in promoting collaboration. Another important aspect is about reputation and knowledge ranking. For instance, a data set or a biological claim about a topic should be weighted more if stated by several people who have given talks on international conferences about the same topic (e.g.: `activitySpeaker`). This part of the ontology has been inspired by the SWRC ontology[115].

5.3.3 Term

This part of the ontology is inspired by the SKOS ontology[116] and we are planning to use directly SKOS in the future. The idea is that each instance of `Term` is not the formal representation of a concept, but a lexical term, such as a word or a title, which is related to an existing concept and may be related to other MannOnto Entity instances (by means of the property `entityTermAnnotation`). This is motivated by the fact that, from the point of view of the Ontology theory, statements like: `Organism subClassOf (OntologyEntry union BioMaterialCharacteristics)` (from MGED Ontology) imply a strong commitment and lead to non senses like "`Organism has_database some OrganismDatabase`" (again, from MGED Ontology, due to the mixing of different interpretations of organism as a type of entry in a software system and the usual intended meaning of organism as a type of living being). We have defined relations like `termHasBroaderTerm`, `termHasNarrowerTerm`, `termHasPartTerm`, `termHasSynonimTerm`, which allow to relate terms one each other. Although this kinds of links between terms are not strictly formalised, they are useful in many cases. Every possible link from an entity to a term is stated via the `entityTermAnnotation` property, or one of its sub-properties. This makes `entityTermAnnotation` a bridge between formally defined OWL individuals and less formal terms. The idea has been inspired by the bio-zen ontology[117].

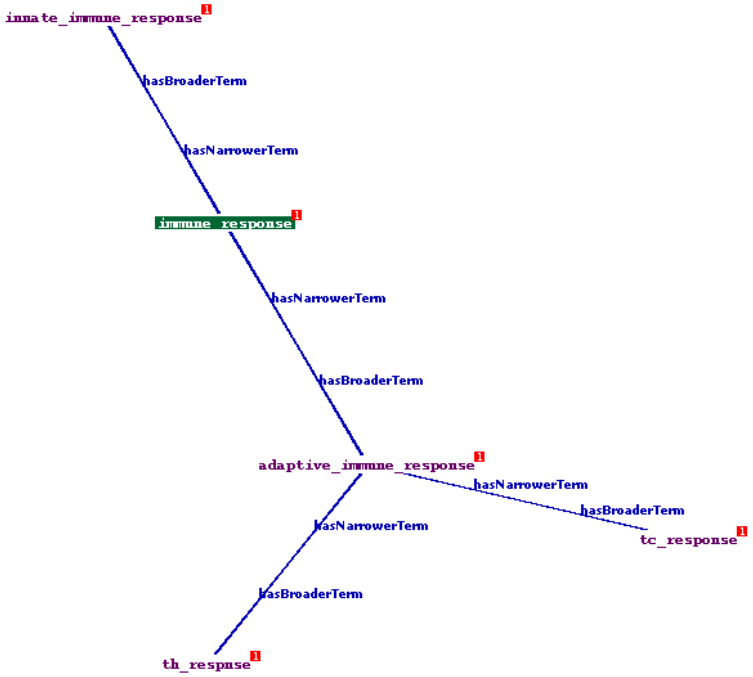


Figure 5.4: use of MannOnto to define terms with lightweight semantic links.

5.3.4 Assertion

An assertion is a claim about a subject entity (`assertionSubject` property) and which refers to a certain context (`assertionContext` property). A `TargetedAssertion` is an extension of `Assertion` that has a target as well (`assertionTarget`). Statements like "these genes enable this process" may be represented by this type of assertion. Assertions are used to define statements that are not absolute truths. There are several ways to represent such a relativity. First, one may consider as valid only those assertions which have certain properties, for instance that are associated to certain authors. Second, since an assertion is an `Entity`, one or more evaluations may be attached to it, using the values from the `evaluation` property. This way, entities may be ranked, according to qualities, such as correctness or reliability. Finally, the context may be used to specify when the assertion is considered valid or meaningful. In particular, terms may be attached to context information, so that one may easily say, for instance, that he/she is referring a particular organism or a given biological process. We have different kind of assertions in the ontology: `AndAssertion` and `OrAssertion` are used to mean that the assertion subjects have a "conjunctive" or "disjunctive" meaning. For example, one may define an instance of a `CausalAssertion`, that is used to state that some subjects, such as genes, are the cause of a certain target, such as a disease. If the assertion is declared as instance of `AndAssertion` too, this will mean all the genes are expressed at the same time, and they causes the disease when all of them are activated. If, instead, an assertion is instance of `CausalAssertion` and `OrAssertion`, then this means some genes in the set, not necessary all, may cause the disease. We can see here imprecise and ambiguous statements are possible. Similarly assertions may be composed by formal parts (for example, the subjects or the context) and informal parts (e.g.: `entityDescription`, which point to a text). This "semi formal" approach allows to deal with the complex application domain of the Life Sciences. hierarchy of properties is also available, to relate assertions one each other (Figure 5.5).

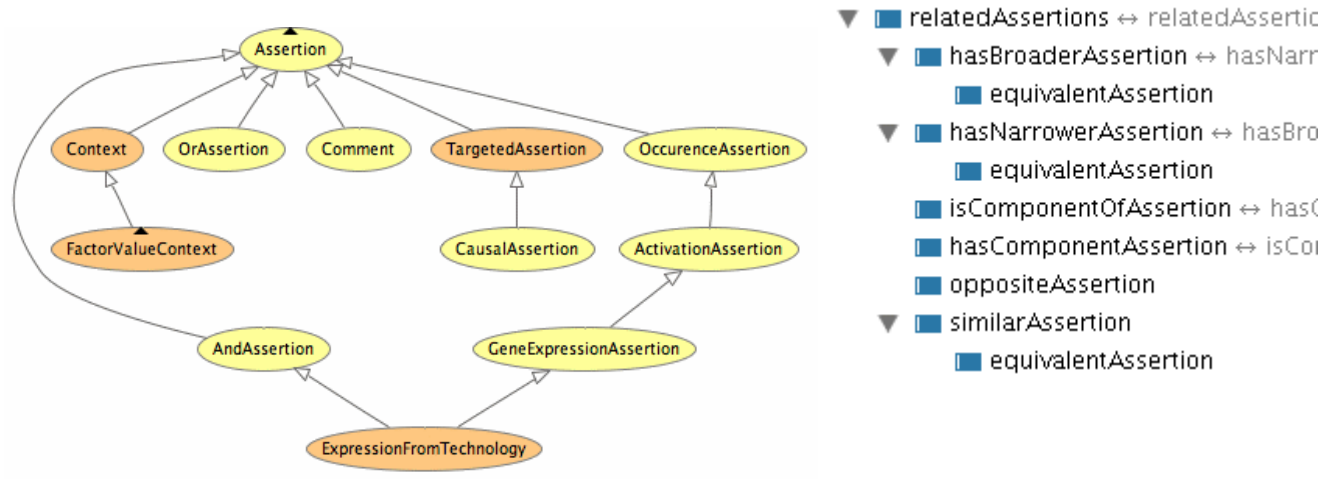


Figure 5.5: concepts used to make claims, experimental hypotheses and conclusions.

5.3.5 Collections

Assertions may be used to define collections of genes (sets). When a particular set of genes has a specific meaning, for instance because the genes are co-expressed under a particular condition, then a specific type of assertion, such as `GeneExpressionAssertion`, may be used to represent the set. When

a set has a more generic meaning, which is difficult to formalise, a class like `Comment` (a type of `Assertion`) may be used. The genes will be attached the genes to the comment, by means of `assertionSubject`. Multiple inheritance may be used to characterise the assertion types.

5.4 Examples of use

In this section we show examples of how the MannOnto ontology may be used. Will use a pseudo-code format, which is similar to the N3 notation[75].

5.4.1 A set of DEGs

The following is an example of how to define a set of genes, which result expressed from a data set analysis.

```
expSet0 type ExpressionFromTechnology
  expressionData
    (chip0:il2 type ProbeSetContainer)
    (chip0:nfkb type ProbeSetContainer)
    (chip0:ifnb type ProbeSetContainer)
  assertionContext
    ( _ type FactorValue
        geExperimentalFactorAnnotation timeTerm
        measurmentValue 4.2
        unitAnnotation mo:hours
    )
    ( _ type FactorValue
        entityDescription "genes expressed under X, maybe innate answer to X"
        geExperimentalFactorAnnotation mesh:diseaseX
    )
  intensity 1.73
  realness 0.8
  interestingness 1
  isSupportedBy
    hybData0, hybData1
```

We report the result as a particular assertion (`ExpressionFromTechnology` is a subclass of `OccurenceAssertion`). `expressionData` is a sub-property of `assertonSubject`. This reflects the fact that a result like “the gene is expressed”, produced by analysing gene expression data, is not an absolute truth, rather it is a claim, which is made according to some experimental evidence. As such, the assertion has attached properties like `realness`, which account for how much the expression statement is believed true, or `interestingness`, which is a vote on the scientific relevance of this result. As previously described, the assertions may refer to contexts. In this case the context is defined as the experimental factor values, that is the condition under which the genes are expressed, which corresponds to the particular conditions under which are the biological materials the measured and elaborated data come from. In this particular case the disease state of the biological material and the time elapsed after an initial event (such as the administration of a compound) are the experimental factors considered.

5.4.2 Gene annotations

We show an example of how a microarray probe set may be annotated with information about the biological entity associated to the probe set.

```
98088_at type ProbeSetContainer
  entityTitle "CD14 antigen"
  probedBy 74av2
  entityTermAnnotation
    (go_0006620 type GoBiologicalProcessTerm
                entityTitle "post-translational membrane targeting")
    (go_0005886 type GoBiologicalProcessTerm
                entityTitle "plasma membrane")
  entitySeeAlso cd14Paper

  isSubjectOfAssertion
    (com0 type Comment entityDescription "PCR temperature for this sequence: 74°C")
```

Here, we can see how it is flexibly possible to describe a probe set, attaching to it a variety of target entities, which are related to the probe set by a number of properties. In the example above, we report two annotating terms from Gene Ontology. We also link the probe set to a publication, which presumably is about the probed gene. Finally, the probe set is the subject of a comment.

5.4.3 An experimental pipeline

We support a simplified syntax to describe the chain of materials and treatments that an experiment is composed of. The same syntax may be used to relate biological materials to the data that have been extracted by the them. For instance, a single route, from source to final data, of a path like the one in Figure 5.2, may be represented as follows.

```
pipe0 type ExperimentPipeline
  pipelineMaterial (hybData1 type RawData)
  pipelineProtocol scanProto0
  pipelineMaterial (hyb1 type Hybridization)
  pipelineProtocol hybProto0
  pipelineMaterial (sample type Sample)
  pipelineProtocol extrProto0
  pipelineMaterial (source0 type BioSource)
```

We can see how the single path is formalised as an instance of `ExperimentPipeline`, a reified relation that accounts for the nodes and steps in the path. The materials and protocols defined above are sorted according to an implicit order and an implicit dependency structure. For instance, it is known that the source is the starting biological material. We will show, in the next chapter, how these implicit relations may be defined and exploited by means of inference rules.

5.4.4 Assertions

We have already shown how sets of differentially expressed genes may be defined in MannOnto, by means of assertions. What follows is a more general example of assertion.

```
ass0 type CausalAssertion, AndAssertion
  assertionSubject
    (chip1:tlr2 type ProbeSetContainer)
    (chip0:tlr6 type ProbeSetContainer)
  assertionTarget
    (unigene:myd88 type ProteinTerm)
  assertionContext
    ( _ type Comment
       entityTermAnnotation term:dc_cell
    )
  realness 0.95
  isSupportedBy
    exp1
  entityOwner ( john type PhDStudent )
  similarAssertion ass1

ass1 type CausalAssertion, AndAssertion
  assertionSubject
    chip1:tlr3, chip1:tlr7, chip1:tlr9
    (chip0:tlr6 type ProbeSetContainer)
  assertionTarget
    (unigene:myd88 type ProteinTerm)
  ...
```

The assertion is claiming that the simultaneous activation of the receptors TLR2 and TLR6 enables the MYD88 molecule. The simultaneity is modelled by making the assertion an instance of `AndAssertion`. We are also contextualising the assertion, telling it makes sense for dendritic cells (DC). The open nature of RDF and OWL allows to add different conditions for the context, for instance a term such as "macrophage" could be added as context. This openness may be an advantage, when pieces of knowledge have to be integrated. However it also make difficult to define closures like "in DC cells only". We are currently leaving this as an open issue.

We also note other attributes that may be attached to an assertion, such as the "realness", the support, the person who has created and is maintaining this assertion.

The last part of the code above is an example of how assertions may be related one each other, by means of proper properties, such as `similarAssertion`.

5.4.5 People

In the previous section, we have an assertion which is connected to an author, by means of the `entityOwner` property. Now we give an example of relations between people and their intellectual productions.

```
ass0
  entityOwner
    john
      type PhDStudent
      isSupervisedBy
        ( profLee type Professor entityTermAnnotation term:immunoInformatics )
      studiesAtOrganization cambridgeUniversity
      speakerAt ( ismb07 type Conference )
```

We plan to use this community relations for ranking the knowledge, as it is further explained in 7.

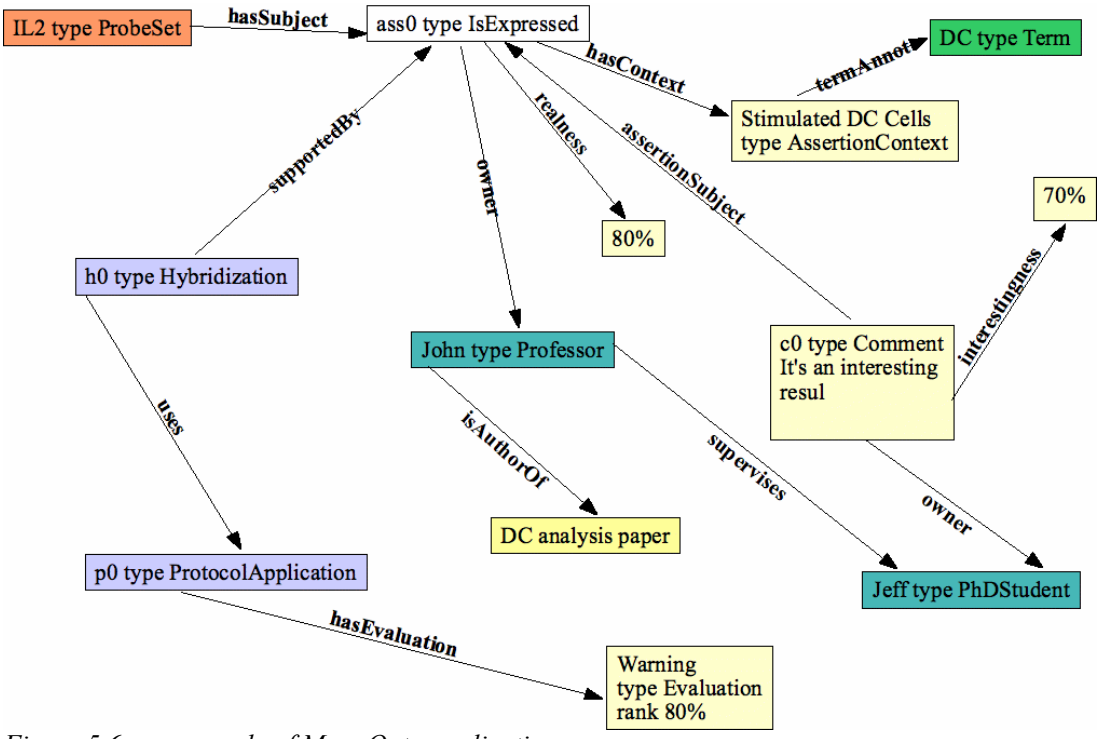


Figure 5.6: an example of MannOnto application.

5.4.6 Knowledge evaluations

Entities in MannOnto may receive different kind of evaluation. We support two structure types for evaluating entities. The first one consists in the use of the data property `evaluation`, which has sub-properties like `interestingness`, `realness`, `precision`. While all this properties have a numerical range, any type of range could be used, provided that something useful is done with the evaluations. The second structure type one may a `Comment` instance attached to an entity. This second evaluation method has to be used when an evaluation has to be characterised with more properties than a single value. For instance, a comment is needed if the entity is evaluated by a user different than its owner, so that the evaluation's author may be attached to the comment and later taken into consideration. We show here an example of evaluations usage.

```
myExtractionProtocol
  precision 0.8
  hasComment
  ( comm0
      reliability 0.4
      entityDescription
        "We have observed many problems with this protocol in our lab"
      entityOwner micheal
  )
```

Evaluations are mainly useful for knowledge ranking. For instance, the protocol above may be scored according to the various quality scores that are provided by the users. Complex ranking combinations are also possible, for instance an evaluation provided by an expert could be weighted more than the one provided by a student (i.e.: we may combine the people roles and their evaluations). Another reason why the evaluations are useful is in the possibility to link entities, like gene expression data, to

evaluation of the materials and the methods used to generate the data. For instance, a problem in a protocol could be shown in an application, while the user is looking at data that has been generated using that protocol.

5.4.7 A comprehensive example

We report in Figure 5.6 a modelling example that include the most relevant aspects of the MannOnto. Here we can see how a biological result, the expression of the IL2 gene, may be linked to supporting data, may receive different kind of evaluations, and every entity may be related to its author, which in turn is characterised by a role, relations with other people (the supervision relation beween John and Jeff) and with produced artefacts (the paper about dendritic cells). We will see in the next chapters how semantic networks like the one in the figure are useful in providing the user the ability to navigate microarray knowledge, as well as in being the basis for knowledge ranking.

6 MannWiki: a Semantic Web based demo application for collaborative sharing of Microarray information

In this chapter we present MannWiki, an application that makes extensive use of the MannOnto ontology, described in the previous chapter. MannWiki is a semantic wiki application. Semantic wikis have recently been proposed as a simple and intuitive mean to create Semantic web content. Wikis, in general, are gaining popularity in the Life Sciences too, as it already has happened in other business fields. Although Semantic wikis are not a perfect application type for the Life Science domain, they may be quickly adapted to specific needs and have allowed us to quickly develop a practical application, which shows the benefits of using Semantic Web formalisms for modelling microarray knowledge. Moreover, the usage of wikis and semantic wikis for the Biology fields could be a base to develop more effective, hybrid interfaces, which combine the collaborative nature of wikis with more structured presentation of web-form based applications. Concerning this point, the ongoing project briefly mentioned in[118] has a conceptual approach which is very similar to the one we propose in this thesis.

6.1 RDF frameworks and Jena

Several tools exists, called Semantic Web frameworks or RDF stores[119], which play for the Semantic Web languages a role which is similar to the role that Database Management Systems (DBMS) have for SQL and DDL languages. They aim at supporting the management of "RDF stores" (also called models), which, in practice, are semantic networks, based on the RDF syntax. They come with an API for one or more common programming languages. Such an API includes interfaces like models, nodes,

RDF resources and RDF statements. Additionally they usually have several automatic reasoning capabilities. Among the many existing tools of this type, we have chosen Jena, distributed by Hewlett Packard. We summarise the reasons that has motivated such a choice.

- It is an Open Source project, promoted by a big Information Technology player like HP and actively kept alive by a large community of users and developers.

- It has a Java API, that maps the several languages proposed by the W3C for the Semantic Web layer cake.

- It has a clear interface for store management. RDF stores may be transparently persisted, either on files, or on a traditional relational databases. Store support includes the ability to read and write files formatted according to standard RDF syntaxes, such as RDF/XML[74] or N3[75]. This is useful for reading ontology definition files, like the ones made with the Protégé tool[113].

- It supports reasoning in different ways. It has two generic rule-based reasoners. One is similar to rule production systems, such as OPS5 or CLIPS[120]. The other one is instead inspired by the Prolog language[121]. The two reasoners may be combined together, resulting in a sophisticated rule system. The inference that may be computed from the semantics behind OWL or RDFS is provided by means of these rule systems. Additionally, Jena is able to transparently use external reasoners, provided that they are compliant with the DIG format[122]. Pellet[100] is often used in Jena this way.

- It allows to make queries over RDF stores, by means of the standard SPARQL query language[83]. Jena has been one of the first RDF frameworks which has started the support for this W3C standard. SPARQL, like other similar languages (e.g.: RDQL), resembles SQL in the syntax, while it is a graph pattern language, for what concerns its semantics. SPARQL queries may be run over regular RDF models (stores), or, more interestingly, over “inferred models”, that is: graphs of explicitly asserted RDF statements, which are augmented with further inferred statements.

Another aspect is that Jena is integrated in Makna, the semantic wiki we describe in the next paragraph. In evaluating the tools to be used for this project, the combination of Makna and Jena has been considered particularly effective.

6.2 Semantic Wikis and Makna

The wiki approach for producing and sharing web contents is well known, mainly because of the success of the WikiPedia project[123], the large encyclopaedia, which is revolutionary, in that one may be both its reader, and freely access a large base of web documents, and writer, and contributing to its growth and improvement. The encyclopaedia contents are organised in web pages, which may be browsed and edited by means of a web application, called wiki. Every page has an identifier and the page may be edited, by accessing a web form and using a simple declarative wiki syntax (which is simpler than HTML). The syntax includes formatting rules, layout templates image embedding. Like in the case of the HTML, a page may have links to other pages (as well as to external web sites and external web pages), which allow for the navigation over the wiki content. This also means that links establish a graph of connections between pages, as it happens for any other hyper text. However the wiki links, like HTML links, are flat: they establish a relation between two pages, but they do not define the meaning of that relation, or why the pages are connected. For instance, a page about a book may

have a reference to the book's author, but the authorship nature of such a link is by no means formally represented, and hence that meaning cannot be computationally exploited. Semantic wikis attempt a step forward, by making available *semantic links*, which are links from the page that is being edited to other pages, and which have attached the property which characterise the nature of the link. An example is shown in Figure 6.1: a semantic network of pages about London is built by editing the page contents and by means of semantic links. From the outcomes of this approach it is straightforward to associate pages to RDF resources (for instance using the page's URL as resource URI), and RDF-S or OWL properties to the semantic links. The resulting RDF translation may be used for knowledge integration (for instance contents from many wikis may be integrated), or for enriching the semantic connections in the wikis, by means of inference. The semantic wiki paradigm straightforward to use for producing RDF-annotated contents. Since many semantic wiki software has been developed recently, it is also relatively easy to implement a specific Semantic Web based application for managing knowledge about a specific topic.

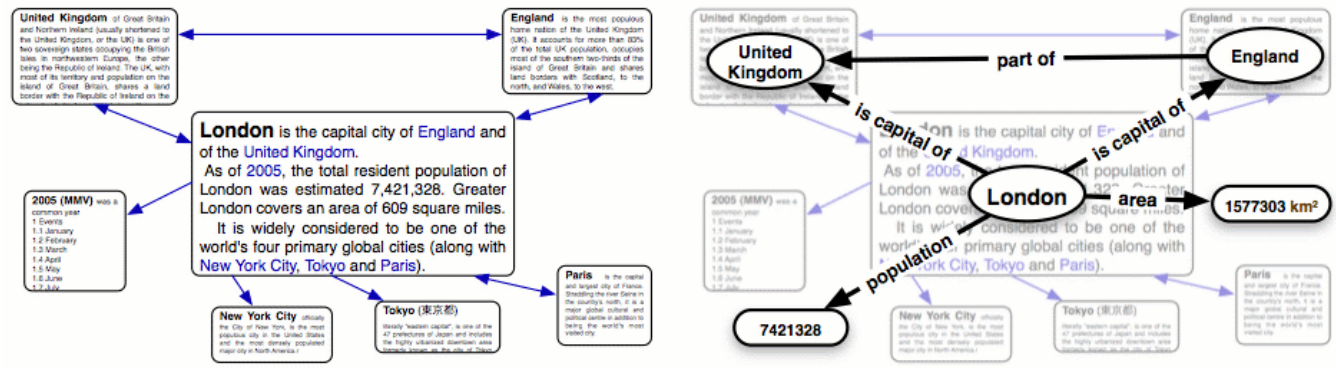


Figure 6.1: how the semantic wikis work. (Source: [124])

We worth mention some disadvantages that semantic wikis have. One is that wiki syntax is not easy to be learn by users with poorest computer skills. Another problem is that they are likely not scalable, at the least, given how they are designed in most cases. For instance, in Molecular biology applications a page about a gene may exist, which may have hundreds or even thousands of links to other pages/entities the gene is related to. We may expect that semantic wikis will be improved in the future and that these problems will be addressed. For instance, the AJAX approach could improve their interface[125]. Moreover, the statements about a page, that are usually presented next to the page itself, could be ranked according to some criterion, and only the most important ones could be presented. We show examples of such ranking in the next chapter.

Several semantic wiki solutions exist. Semantic MediaWiki[124] is based on the MediaWiki, the wiki system developed for Wikipedia. As such, it may count on a mature and feature-rich wiki product. It also offers extensions to the well known MediaWiki syntax, to use semantic links. Semantic links in Semantic MediaWiki give the user a simplified view on OWL-DL, in the sense that it is possible to predefine a list of properties that can be used for semantic links, together with simplified names. Both predefined properties and simplified names are mapped to OWL resources. Likewise, the existing availability of page categories is exploited to map pages onto OWL classes. A weak point of Semantic MediaWiki is the limited support for automatic reasoning, in particular with OWL entailments. Moreover, the fact that it uses PHP for managing the RDF triple store may be a critical aspect for what concerns tboth he performance and the scalability.

IkeWiki[126] is a semantic wiki built from scratch, rather than being based on an existing wiki. With respect to Semantic MediaWiki, it allows a more direct access to the underline RDF knowledge base, where pages and their relations are represented. For instance, direct use of URIs is possible. Semantic links to a page can be added by interactive interfaces. Since these interfaces are separated by the page content, it is possible to add semantic content and semantic links to the pages as page metadata. This may be an advantage or not, depending on the particular content domain. Concerning inference capabilities, it is not clear if and how IkeWiki uses some reasoning facility.

SweetWiki[127] has several differences with respect to the typical semantic wiki. Its focus is the use of tags, a typical so called "Web 2.0" paradigm[63]. User may tag a page with keywords. Tags may be semantically characterised, by administrator users, by using RDF/OWL ontologies. The authors refer to so called "folksonomies": semantic structures which are built, manually or automatically, by observing the typical usage of tag terms. Although this approach requires much less formalisation effort, it is not very appropriate in fields like the Life Sciences, where instead, formalising a wide and often unstructured knowledge domain is a key challenge. SweetWiki proposes also a WYSWIG editor and AJAX components. For instance, when the user starts typing a tag, the AJAX code behind the scenes automatically searches those terms that best match the one being inserted.

In this project we have chosen the Makna semantic wiki[128], a semantic wiki that has been built on the existing JspWiki[129]. Makna allows for usage of semantic links directly into the wiki syntax and the edited pages. The administrator may configure the ontologies to be imported and used in the wiki (by specifying the location of RDF/XML documents to be loaded). Once ontologies are loaded, they may be used by specifying URIs of classes and properties. Namespaces may be applied when providing a URI. When rendering a page, Makna formats it according to its wiki syntax. Additionally, it show a list of statements that are related to the page, because they have it as statement's subject or as its object. Furthermore, it is possible to invoke an "inferred statements" view, which, in addition to the statements defined via wiki syntax computes and shows the statements which may be inferred by the underlining ontologies. The inference is based on a Jena inference model, to which a reasoner is attached.

It is easy to change the code, so that a custom Jena model is used by Makna. For example, a model making inference from a custom rule set may be defined and passed to the wiki.

We have chosen Makna mainly because it is fairly stable, it has a clear semantic-web oriented interface. it is relatively easy to program, it is integrated with Jena. However, when coming to a biological domain, like the one of microarrays, it has the limits mentioned above. The author of this thesis, together with the main developer of Makna Wiki, is planning enhancements to the system, which would benefit both the specific microarray application and, in general, other content types.

We worth mention Platypus[130], which is similar to MannMakna in both the interface and the use of Jena. We have preferred Makna to Platypus, because is not clear if the latter is still an alive project and how much is stable and mature.

6.2.1 Wikis and Life Sciences

Use of wiki solutions in Life Sciences is relatively recent and still largely unexplored. The usefulness of wikis for promoting collaboration and collective contribution to the building of public Biological knowledge has been pointed out recently. Applications have been proposed for the management of gene annotations. As for that, the project mentioned in [131] is the most close to the work we present hereby. A similar approach is proposed by the BOWiki project[132]. mybio.net[133] is an example of traditional wiki tools used for the Biological field.

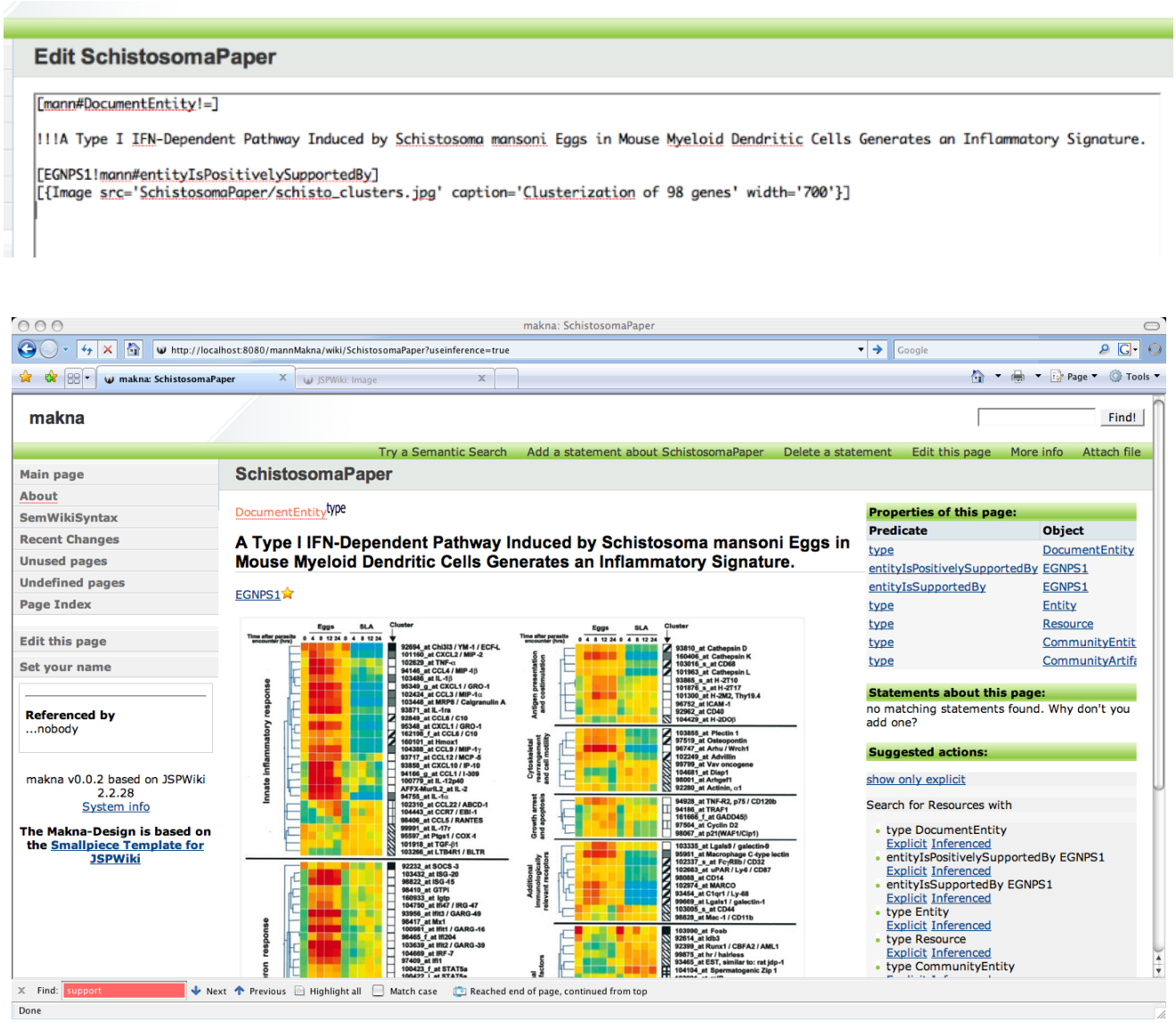


Figure 6.2: an example from MannWiki. Top: the page described in wiki syntax. Bottom: The corresponding rendered result, with semantic links on the right side.

6.3 The MannWiki application

MannWiki is a simple application, based on MannOnto ontology and Makna, which allows to collaboratively publish results and other information that comes from microarray studies. We intend the application mainly for demonstration purposes. It shows the potential of using collaboration systems and Semantic Web technologies, namely our Microarray-dedicated ontology, for producing and sharing knowledge about the Gene Expression field. Notwithstanding the proof-of-concept nature of our application, we believe it may already be used by small groups of collaborating scientists. In fact, we are using it to implement a form of "Gene Expression atlas" about the field of Dendritic Cells, in the context of a project funded by the European Framework Program[134].

In Figure 6.2 we report the editing form for a wiki page, that is associated to the MannOnto class `DocumentEntity`. We can see the special syntax that is used to format the page content, and the extensions that are provided, for the definition of semantic links. The same figure also shows how this syntax is actually rendered. Common formatting elements are supported, as it is usual for wiki tools. Image embedding is also possible. Semantic links, which relate the current page to other pages or data values, are rendered on the right side of the page. In the example in figure, the "inferred statements" view is reported. For instance, the fact the paper is based on the experiment "EGNPS1" is explicitly defined in the page content, while the fact the page is a `ComunityArtefact` is inferred by the fact it was originally defined as `Document`, a subclass of the former.

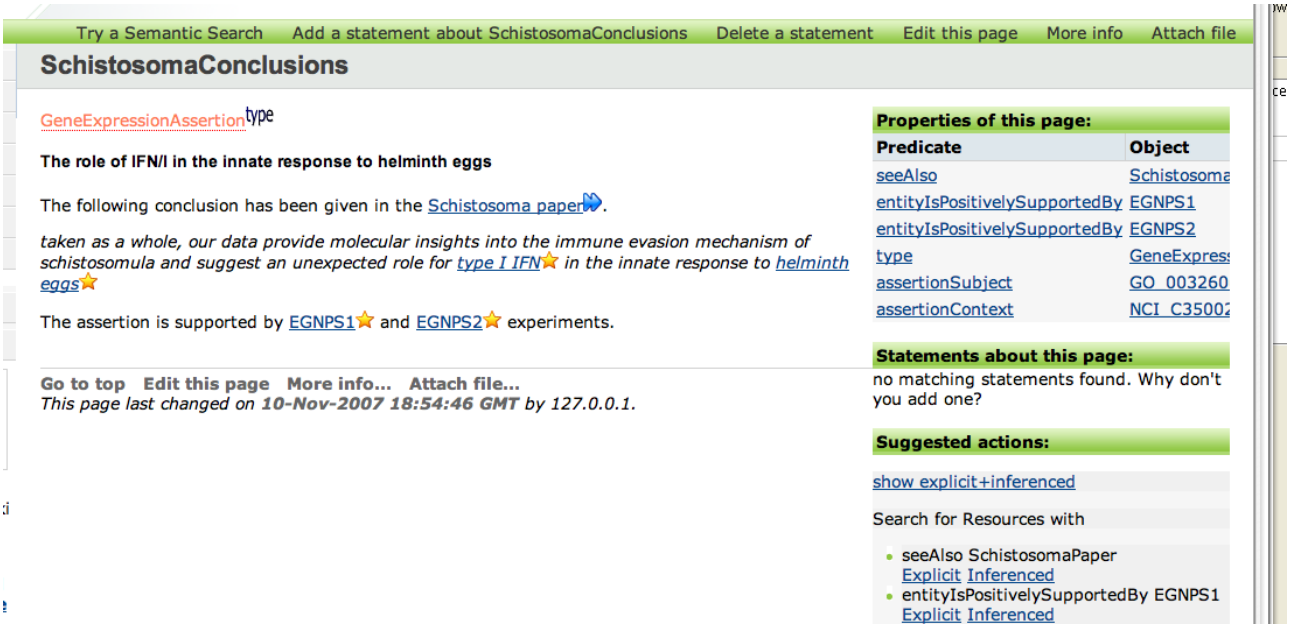


Figure 6.3: conclusions made from the analysis of two experiments.

6.3.1 A case study

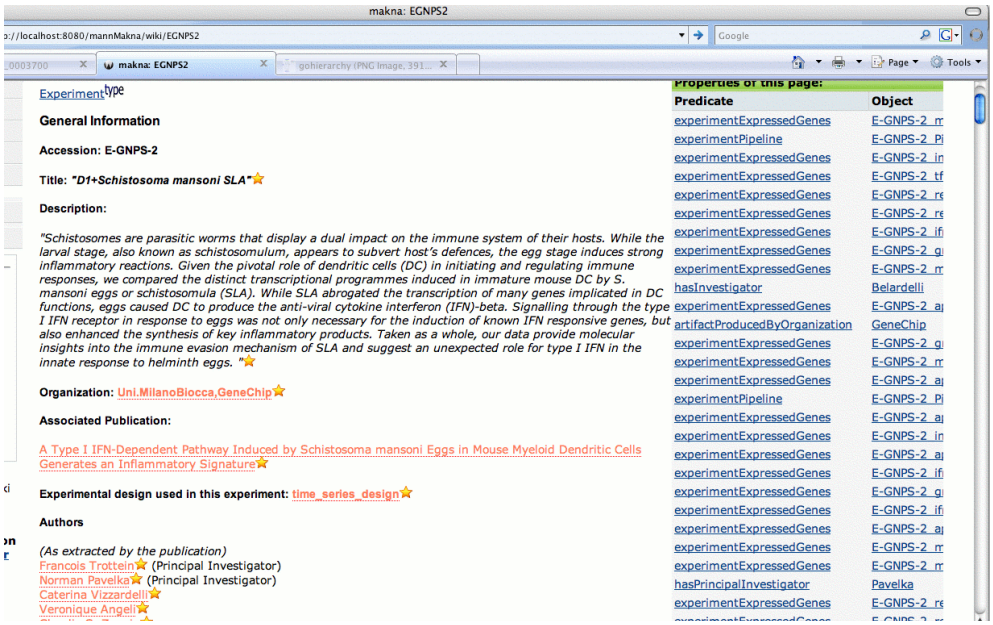
The paper reported in Figure 6.2 is a case study we have considered to populate MannWiki with an example from reality. It is about two experiments made with mouse dendritic cells. Dendritic cells (DCs) play a bridge role in the induction and regulation of the immune response to pathogens[135]. They are plastic cells which, after first contact with external pathogens, are able to emit a variety of signalling molecules, interact with different cell types of the immune system and determine different immunological responses, on the basis of the initial pathogen type. Since DCs exhibit a complex and flexible behaviour, by means of the gene expression reprogramming, many studies have been done through the microarray technology[136]. In the work we have chosen as a case study, they have stimulated DCs with two variants of the schistosoma helminth parasites, which is a pathogen of relevant interest for the Immunology, given its responsibility in causing the schistosomiasis, a several common tropical disease. Microarray hybridisations have been made over several time instants after intitial DCs infection. This time course experiment has allowed to study the dynamics of the gene expression which regulates the DCs answer to the initial stimuli. As for the results, the data analysis has lead to a set of 283 differentially expressed genes, 98 of which have been clustered and classified according to functional categories in Gene ontology, plus few Immunological categories (e.g.: inflammation response, interferon response, antigen presentation). We will show later how we have represented these

results.

The paper also mentions some speculative conclusions. For instance, the paper abstract reports:

"*taken as a whole, our data provide molecular insights into the immune evasion mechanism of schistosomula and suggest an unexpected role for type* ***I IFN*** *in the innate response to* ***helminth eggs****"*.

We have created a page that corresponds to a `GeneExpressionAssertion` (Figure 6.3). Both the text above and a MannOnto characterisation of it have been saved in the MannWiki. NCI Thesaurus and GeneOntology have been used for the expressions "helminth eggs" (associated to NCI "Schistosoma Mansoni Infection", ID C35002) and "interferon I" ("interferon type I production", ID 0032606). Even this simple term-based annotation is useful for knowledge browsing purposes. For instance, a search for the assertions that are related to the GO term "cytokine production" (term no. 0001816) could give back the assertion above as relevant result, given that interferon is a specific cytokine.



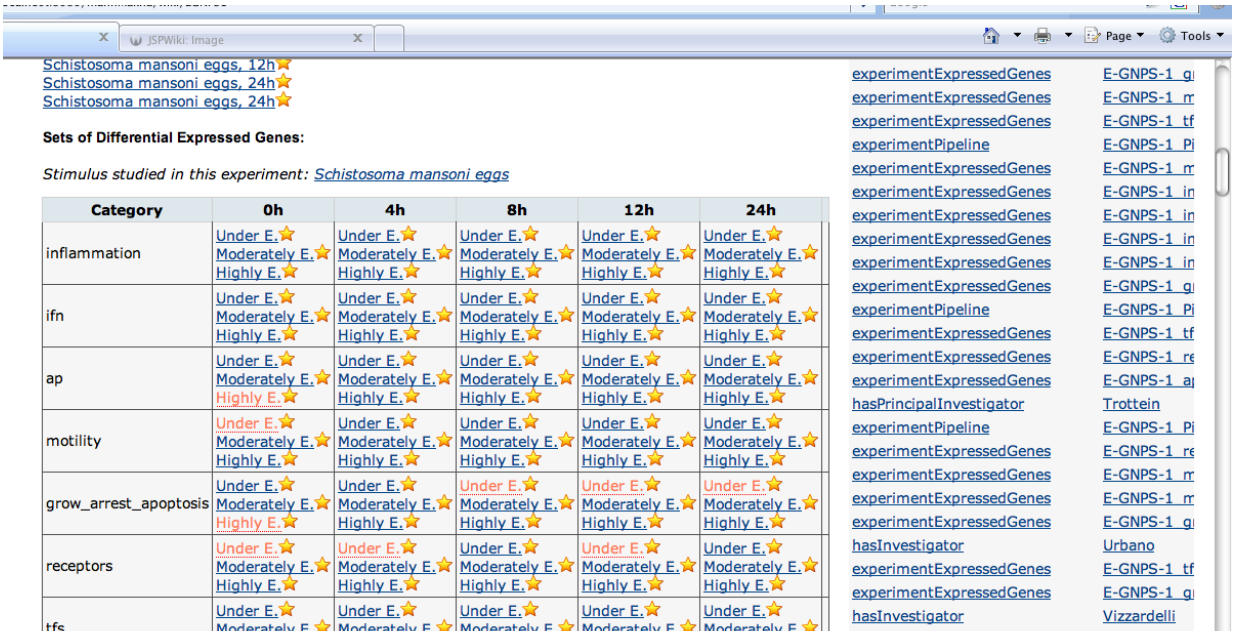


Figure 6.4: a screen-shoot about an experiment. Top: general data. Bottom: the list of interesting DEGs, classified according to experimental factors and functional category.

6.3.2 Representation of experiments

In Figure 6.4 one of the two schistosoma experiments are reported. General information about the experiment have been put in a wiki page, using the MannOnto general properties, such as title, abstract, publishing date. The experiment is linked to the paper described in the previous section, which is also used to infer the experiment authors[6].

The same page also reports links to the experimental processing which produced for the generation of the experiment data. `experimentPipeline` property is used to link the experiment to one path of material and data processing, as explained in Section [X, 5.3.1]. In Figure 6.5 a screen-shoot about one of these pages is shown. The materials and the protocols which have lead to a single final transcriptome are described. On the right side, terms and other annotation types (e.g. comments), associated to materials or protocol reported in the pipeline, are automatically inferred as being associated to the pipeline itself and to upstream elements, such as the final data. Such an inference is computed by means of production rules, as it is described in Section 6.4.4.

6 The authors of a publication about an experiment are not necessarily all the people involved in laboratory activities and data analysis. However they are a significant subset, so this heuristics can reasonably be adopted.

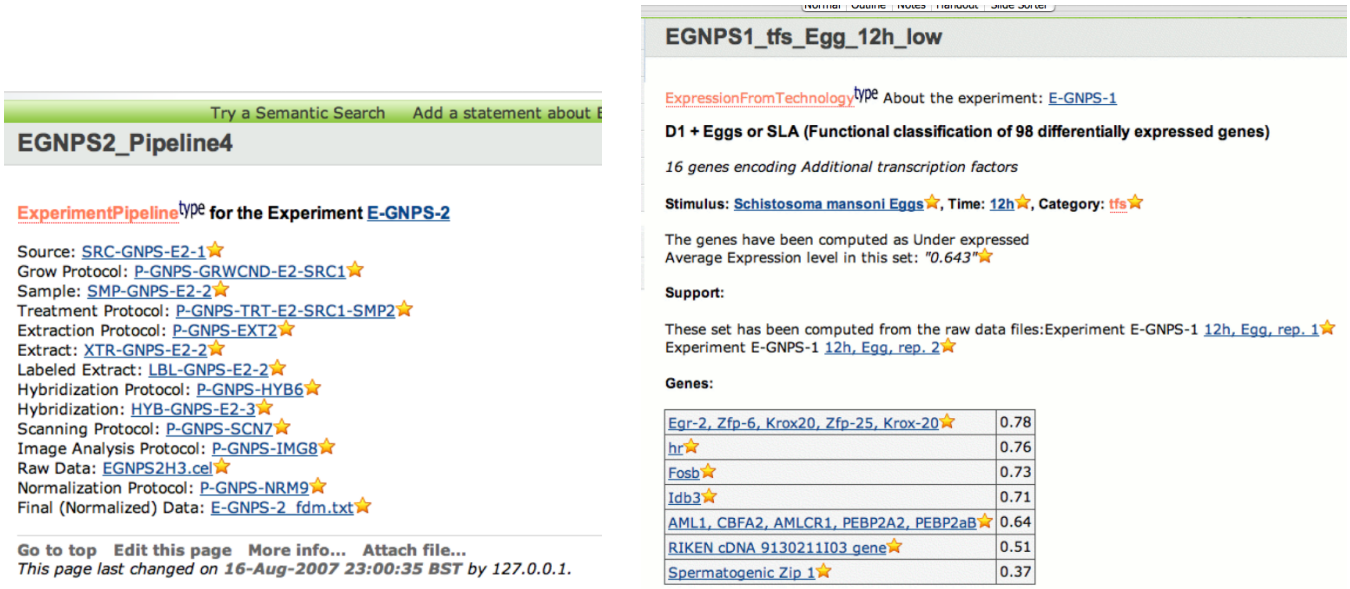


Figure 6.5: a processing pipeline, describing part of the materials and methods used for an experiment (left). A stored set of differentially expressed genes (right). In both cases, wiki syntax and MannOnto elements have been used to formally define the information rendered in the figures.

6.3.3 Analysis results

Results about clustering and classification of genes according to the expression levels are reported in the last part of the experiment page (Figure 6.4). Here, links to pages which correspond to instances of `ExpressionFromTechnology` class are defined. The links use the `experimentExpressedGenes` property. In Figure 6.5 one of such expression assertions is shown. It reports the biological conditions the set refers to (which, in the specific case, are about the schistosoma type and the 12 hours time point). The gene set is also annotated with a functional category, "transcription factors (TFs)". Both the biological conditions and the functional category are formally modelled as the assertion context, i.e. the context under which the assertion is to be considered valid. While the expression intensities of each gene are reported in figure, formally only one value may be associated to the expression set, which has been computed by averaging all the values of the genes in the set. This is done mainly because of performance reasons. In fact, if all the levels were formalised in MannOnto, we would need a very high number of assertions and the system would likely become very slow and memory demanding. We admit this is a limit that only permits qualitative formalisation of gene expression levels. Another related problem, that we chosen not to address in this work, is the heterogeneity of expression levels across a variety of different conditions, e.g.: different organisms or different microarray platforms. In general, such different data sets are comparable only among hybridizations which are part of the same experiment. In MannOnto, we assume that some mathematical operation (such as scaling) has been done on the intensities, to deal with this problem. Another option we are currently considering, is using discrete values only for the expression intensity, i.e.: -1, 0, 1 or words like "low", "high". This would account for results which are relative to the respective experiment and the corresponding experimental conditions.

The assertion described in Section 6.3.1 is also linked to the experiment. This link is automatic. In fact, we define a production rule which says that, whatever conclusion is derived from a paper (like the hereby discussed) and the paper is supported by a given experiment or data set, then the conclusion is supported by the same experiment or data set.

Figure 6.6: a page about a microarray probing element.

6.3.4 Gene annotations

In Figure 6.6 we can see an example of a page that is associated to a probe set. This kind of pages may flexibly report many kinds of information, available for a probe set and associated biological elements. In particular, we maps Gene Ontology annotations, which are one of the most used ontology-like models for describing genes. In Fig. Figure 6.7 it is shown an example of a Gene Ontology term, which has been imported in the system, converted into our SKOS-like representation of terms, and described in a wiki page. Since the wiki approach is a not very suitable for browsing taxonomies, especially the largest ones, we plan to develop a better interface for terms, for instance using the approach in [137][138].

6.4 Exploiting the Semantic Web in MannWiki

In this section we mention some of the features available in Makna for searching knowledge. We also present extensions we have made to Makna, in order to better support the specific microarray knowledge we deal with.

6.4.1 Traditional text queries

Being based on the pre-existing wiki JspWiki, Makna makes available the wiki features of the former. This include a text box-based search functionality, that retrieves and rank pages matching input keywords. Both search and ranking are based on the popular Lucene search engine[139], which

internally uses a TF-IDF ranking algorithm[140]. The fact that both semantic queries and traditional free text queries are possible in MannWiki is a confirmation of the benefits of mixing formal and informal knowledge.

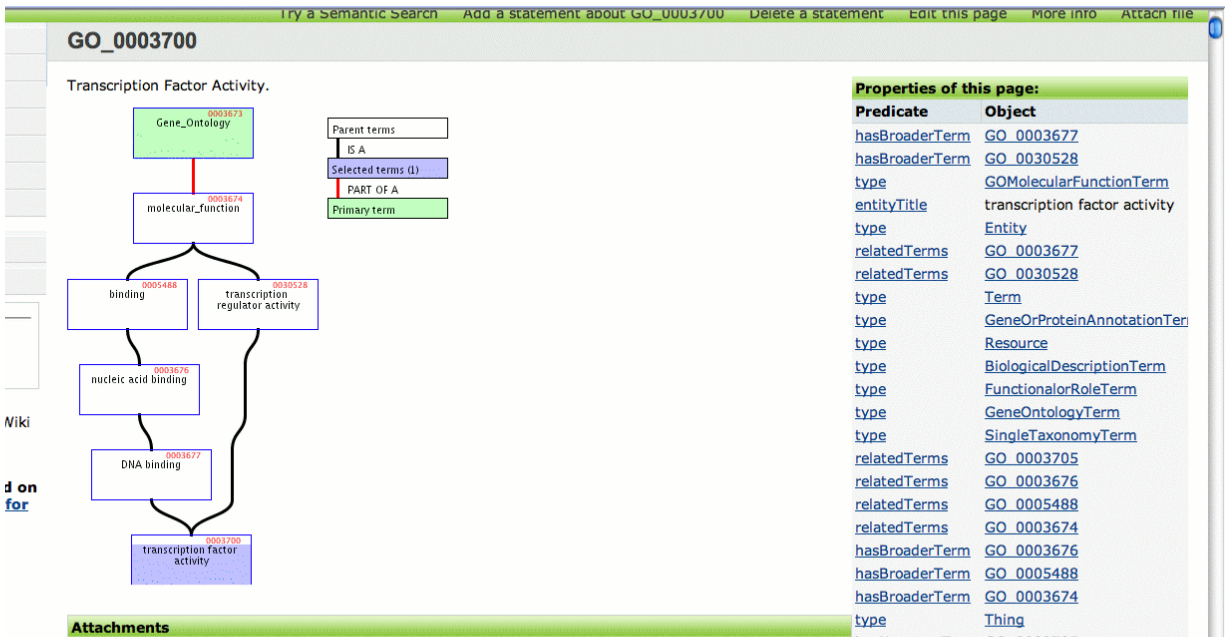


Figure 6.7: use of MannWiki to show information about a gene ontology term.

6.4.2 Semantic Searches

Makna has a "Try a Semantic Search" feature, that allows to compose RDF queries, by means of interactive query composition. For instance the query of type named "Search for instances of a class" finds all the instances of a class, which may be selected from pre-configured ontologies. Although these type of queries are not intuitive for the end user, they may be useful for advanced searching.

```
SELECT
  DISTINCT ?pbset ?pbsTitle ?geAss ?ctx ?ctxTerm ?ctxTermTitle ?expLevel
WHERE
{
  ?geAss rdf:type mann:GeneExpressionAssertion;
         mann:assertionSubject ?pbset . ?pbset rdf:type mann:ProbeSetContainer

  . OPTIONAL { ?pbset mann:entityTitle ?pbsTitle }

  . OPTIONAL {
    ?geAss  mann:intensity ?expLevel
    . FILTER ( xsd:float ( ?expLevel ) > $level )
  }

  .
  {
    { # ___ Match the URI ___
      ?geAss mann:assertionContext ?ctx
      . ?ctx mann:geExperimentalFactorAnnotation ?ctxTerm
      . FILTER regex ( str ( ?ctxTerm ) , "$keywords", "i" )
      . OPTIONAL { ?ctxTerm mann:entityTitle ?ctxTermTitle }
    }

    UNION
    { __ Or match the Title __
      ?geAss mann:assertionContext ?ctx
      . ?ctx mann:geExperimentalFactorAnnotation ?ctxTerm
      . ?ctxTerm mann:entityTitle ?ctxTermTitle . FILTER regex( ?ctxTermTitle, "$keywords", "i" )
    }
  }

}

ORDER BY DESC(xsd:float(?expLevel))
```

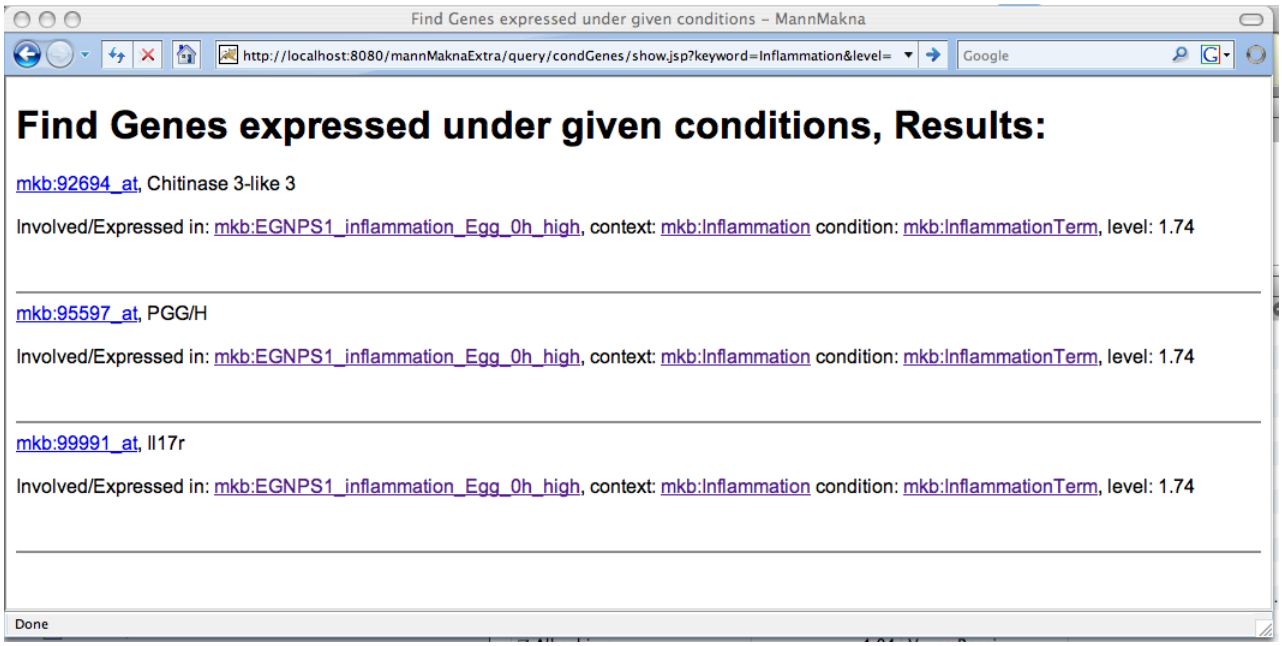


Figure 6.8: a SPARQL query that finds genes expressed under a given condition. Bottom: a sample result.

6.4.3 Queries and parametric queries

Several, similar, RDF query languages exists. The W3C has leveraged on existing dialects and has proposed SPARQL (SPARQL Protocol And Query Language), which is becoming the standard query language of the Semantic Web[83]. As already mentioned, SPARQL is substantially a graph pattern language. It makes possible to describe a graph template, by means of variables that are unified with matching graphs. We have used SPARQL to extend the search functionalities available in Makna. We

have defined parametric SPARQL queries, which allow to perform relevant searches. For instance, we show, in Figure 6.8, the case of conditions/genes query, which finds the conditions that match a given keyword (provided as parameter), plus the genes that are expressed at a level greater than another input parameter.

The "`$keyword`" and "`$level`" tokens are replaced by the user input, which in turn is taken from a web search form. The query string is passed to the Jena query engine, after having replaced the parameters.

All the queries built this way are executed against the inferred graph and therefore not only does it return explicitly asserted statements, but also those statements that are inferred by the OWL semantics and the use of custom rules (described below).

6.4.4 Rules

Ontology languages such as OWL have expressiveness limits. The Description Logics they are based on works well for tasks like classification or automatic reasoning over terminological knowledge. Whereas this features are relevant for the Biology domain, there are additional entailments that cannot be supported by such a formalism. For example it is difficult in OWL to define that two individuals belongs to a relation which is the composition of two properties (e.g.: `genex produces x iff genex transcripts rnax and rnax translates x`). The limits of OWL and DL are one of the reasons why the W3C defines a rule layer for the Semantic Web. Rule language standards, such as SWRL[141], are being proposed.

We use the Jena Rule engine to define and apply useful rules that are based on the MicroAnnOnto ontology. We preferred the Jena rules to the use of some external SWRL-enabled reasoner, such as Pellet, mainly because the former is a simpler solution, from the deployment point of view. Furthermore, we are able to mix the backward and forward reasoners, and take advantage of the way they are coupled together. For instance let us consider a simple definition of semantic distance between term individuals, which are related by the `broaderTerm` relation:

```
broaderTerm(x,y) => broaderTerm0(x,y)
broaderTerm0(x,y), broaderTerm0(y,z) => broaderTerm1(x,z)
 broaderTerm1(x,y), broaderTerm(y,z) => broaderTerm2(x,z)
 broaderTerm(x,y),  broaderTerm1(y,z) => broaderTerm2(x,z)
...
```

where $\texttt{broaderTerm}_i$ is a sub-property of `broaderTerm` The bigger is *i,* the bigger is the chain that links two terms, i.e.: their semantic distance. Rules like the ones above, make sense if they are applied before considering the transitivity of the `broaderTerm` property. This is what happens by defining them as forward-chaining rules, since the Jena's forward reasoner is run before the backward one, and the transitivity is implemented by by backward-chaining rules. Relying on the order the rules are applied with, makes a rule system more complicated than the case this is not done, and likely this approach will not be part of the standards for the Semantic Web. However there are cases, like the one above, where this gives valuable advantages.

We now describe the type of rules we have implemented for the MannWiki application.

Usage-cascading rules. These are applied when the usage of some device or technology may have an impact on final results, which depends on that usage. For instance, the following rule encodes the fact that, in case a biological material is used to produce another biological material, and the former uses a given microarray, then the second material uses the array as well:

```
[matArray:
       (?matx mann:usesArrayType ?arry)
       <-
              (?maty mann:usesArrayType ?arry),
              (?matx rdf:type mann:GeneExpressionMaterial),
              (?maty rdf:type mann:GeneExpressionMaterial),
              (?matx mann:usesGEntity ?maty)
]
```

Rules of this type are important in propagating the evaluation of items being used to produce data and final results.

Annotation cascading. When an entity is annotated with a term, it may happen that other entities, related to the annotated one, are related to the same term. For example, if a given biological material is part of a given experimental pipeline, the latter shares the term annotations about biological characteristics that the material has. This is captured by the following rule:

```
[pipelineCharact:
    (?ep mann:geMaterialCharacteristicAnnotation ?term)
    <-
       (?ep mann:pipelineMaterial ?mat),
       (?mat mann:geMaterialCharacteristicAnnotation ?term)
]
```

Pipeline reification. The `ExperimentPipeline` class is indeed the reification of a relation. In structures like the one in Figure 5.2, we have several nodes belonging to an instance of this class, that are indeed connected by usage relations. We use the `ExperimentPipelineClass`, rather than an explicit representation of tree graphs, like the one in figure, in order to keep simple the editing and visualisation of experimental pipelines by means of the wiki interface. This may be formalised by means of proper rules. For instance, the following rule propagates the fact that certain type of materials are being used by certain other types, if they are in the same pipeline:

```
#
# Inference about Material usage:
#  If x type Class1, y type Class2, Class1 usesMaterialInPipelines Class2,
#    x,y in the same pipeline
#  THEN x uses y
#
# For instance: a sample uses a source that is in the same pipeline
#
[materialPipelineUse:
 (?matx mann:usesGEntity ?maty)
 <-
  (?ep mann:pipelineMaterial ?matx),
       (?ep mann:pipelineMaterial ?maty),
       (?matx rdf:type ?MatTypeX),
       (?maty rdf:type ?MatTypeY),
       (?MatTypeX mann:usesMaterialInPipelines ?MatTypeY)
]
```

where `usesMaterialInPipelines` is a relation which predefined class pairs belong to, for example:

```
-> (mann:NormalizedData mann:usesMaterialInPipelines mann:RawData).
-> (mann:HybridizationData mann:usesMaterialInPipelines mann:BiologicalMaterial).
-> (mann:Hybridization mann:usesMaterialInPipelines mann:LabeledExtract).
-> (mann:LabeledExtract mann:usesMaterialInPipelines mann:Extract).
...
```

The above rules are applied together with the predefined Jena's rules which accounts for reasoning by means of OWL semantics. The predefined base set of rules to be used may be configured. We have also tried RDF-S plus our custom rules. The computed inferences are available at the level of the wiki interface, on the page's statements visualisation. As already stated, they are available with the "Semantic Searches" function, as well as with SPARQL-based searches.

We find the definition of practical rules ,like the above ones, a powerful way to extract and show the user useful knowledge about microarrays. However, we have experienced that combining OWL inference and custom rules leads to a rather slow system, especially when updates occurs. We are evaluating the use of asynchronous updates and mechanisms for caching Jena models.

6.5 Implementation notes

We provides, in this section, some details about the implementation of the MannWiki application. It is mainly based on the Makna wiki application. We have changed the Makna code to add the possibility of using our custom reasoner, which is the one using Jena's rule engine, configured with our custom rules (in addition to the ones for OWL or RDF-S).

For the SPARQL-based queries, we extended the part of the application that is concerned with the web interface (i.e.: some JSP pages and related code). We read the SPARQL queries, described in Section 6.4.3, from plain text files and run them, after having replaced the query parameters, through the Jena query engine (and against the inference model being used).

We have written a small library of utilities for Jena, that we have used for this project. For instance, the library makes easy to issue a SPARQL query, by passing just a string to a proper interface, which instantiates the components necessary for running the query against a RDF model.

Other relevant functions we have developed are the code that imports microarray experiments coming from the GCA microarray repository, and the code that imports the Gene Ontology OWL file.

6.5.1 Importing from TAB2MAGE

We have already described the GCA repository software in Section 3.1.2. In order to build the Schistosoma use case, we imported data stored and annotated in this repository. Tabular, spreadsheet-based formats are becoming popular in the "omics" field[29][30]. In fact, we have previously developed a tool for the exporting of a GCA experiment to the Tab2MAGE format, with the ultimate goal of building a data transfer pipeline, from GCA to the well known public repository ArrayExpress. We created the experiment-related pages in MannMakna, by writing PHP code that read the Tab2MAGE files produced by the GCA export tool. The code output is in the wiki syntax required by Makna. Semantic links are properly defined in this output, by means of MicroAnnOnto elements. The conversion has been eased by the fact that MicroAnnOnto covers concepts defined in the MAGE model.

We have also written some more specific code which imports in the wiki the results presented in the Schistosoma paper. As for this task, we have used the spreadsheet files provided as supplemental materials. We are aware that gathering detailed results from microarray analysis, which are described by the papers, is currently difficult. Relying on supplemental materials is one of the few solutions which are currently possible. We plan to develop plug-ins for analysis tools, such as GeneSpring or BioConductor, which allow to export results from microarray analyisis in MannOnto format.

6.5.2 Importing Gene Ontology

We are writing code for importing in the MannWiki knowledge relevant existing biological ontologies. Currently we have completed the import of the Gene Ontology OWL files. The import code converts the classes defined in these files into instances of MannOnto's `Term` class. As we have shown in Chapter 5, we encode this way those ontologies which are essentially taxonomies of terms. In dealing with the imported GO terms, we have experienced performance problems, especially with reasoning and rules. Since we are interested in some categories only, especially the ones which are relevant for the Immunology, we have chosen to get round the performance concerns, by adopting a particular strategy. We have selected a set relevant GO classes for importing. We then have imported al the classes that are upstream to the relevant ones, according to the "subclass" or "part-of" relations. In addition, we have added the subclasses of selected ones up to a predefined depth level. This simple strategy allows us to select a significant part of a taxonomy and use it in our application, without compromising its performance.

7 A proposal for ranking OWL-based information

The most interesting uses of the Semantic Web are browsing and searching. The application we presented in the previous chapter mainly provides browsing capabilities, which are based on a "resource centric" approach. An important aspect of both browsing and searching features, is ranking the knowledge, so that it can be presented in significance order to the user.

There are several examples in literature of knowledge ranking, mostly proposed for the development of keyword-based search engines[140]. Moreover, there is a wide literature on the topic of searching and ranking semantic networks[142][73][143].

In this chapter, we present a ranking approach called Spreading Activation, or SA[144]. So far, this method has been used for searching purposes, precisely, for finding keyword-relevant nodes in a semantic network. We propose to adapt the SA approach to the specific characteristics of the Semantic Web technologies. Namely, we propose to combine the expressivity of SPARQL with the SA ranking approach. We further propose to use the technique for browsing knowledge, not only for searching from initial keywords. Such browsing with the SA algorithm is possible by exploiting the knowledge ranking which can be computed from quality metrics and other evaluations.

7.1 Spreading Activation

Spreading Activation algorithms have originally been designed for the semantic networks. They can be used with whatever type of graphs and of course with RDF graphs, either inferred graphs or not. The algorithm starts with a set of initial nodes, which typically come from the application of other searching

methods (e.g.: keyword-based text search). The initial nodes are provided with an initial score. The basic idea is to find additional nodes by exploring the network they are part of, and by "spreading" the initial scores over such network. Figure 7.2 reports an example.

The core of the SA algorithm is the loop that propagates the rankings (Figure 7.1). In the simplest version, such a propagation is achieved by selecting a node (which has been included in a to-be-processed queue) and considering the semantic links it holds. The node's score is transferred to the neighbouring nodes, after it has been weighted, according to the labels of the edges that connect the initial node to its adjacent ones, i.e.: according to the semantics of the link.

The rank spreading may be described by the formula:

$$I_j = \sum_i w_{ij} \cdot O_i \cdot \beta$$

j is the node about which we are computing the rank, which has the i inbound nodes. O_i is the rank of node i, also called output or activation. I_j is called total input of node j. In general, this is a function of the node's output O_j, which properly adjusts how to propagate the nodes rank. Such an adjustment is typically made by considering threshold or saturation functions (e.g.: step function or sigmoid function). In the simplest cases, I_j is set equal to O_j. As mentioned above, we have the weights w_{ij}, which, in semantic networks, vary according to the semantic label of the edge *(i,j)*. The algorithm shown in Figure 7.1 applies the formula above considering one of the i nodes at every iteration, and increasing the input of all outbound j nodes. A decay factor β is usually applied, so that the intensity of the activation decreases with the topological distance. This allows to avoid that all the network is meaninglessly scored. It also implicitly accounts for transitive relations.

```
queue = initialQueue(); stopFlag = false;
while ( !queue.empty() && !stopFlag )
  i = queue.pull()
  if ( checkPreRestrictions(i) )
    for each j in (i,j)
      j.in += w(i,j) * i.out * beta
    j.out = f(j.in)
      if ( !j.visited )
        queue.push ( j )
      }
    }
  }
  stopFlag = checkPostRestrictions()
}
```

Figure 7.1: the Spreading Activation algorithm.

In summary: an initial set of ranked nodes is extended by means of the "diffusion" of their rank, through the "pipelines" of the network edges. The "pipeline capacities" are mapped by the edge semantic labels. In the context of the Semantic Web, such labels corresponds to RDFS properties, or OWL properties. We refer the interested reader to [73][143] for examples of practical applications of such an approach to the Semantic Web domain.

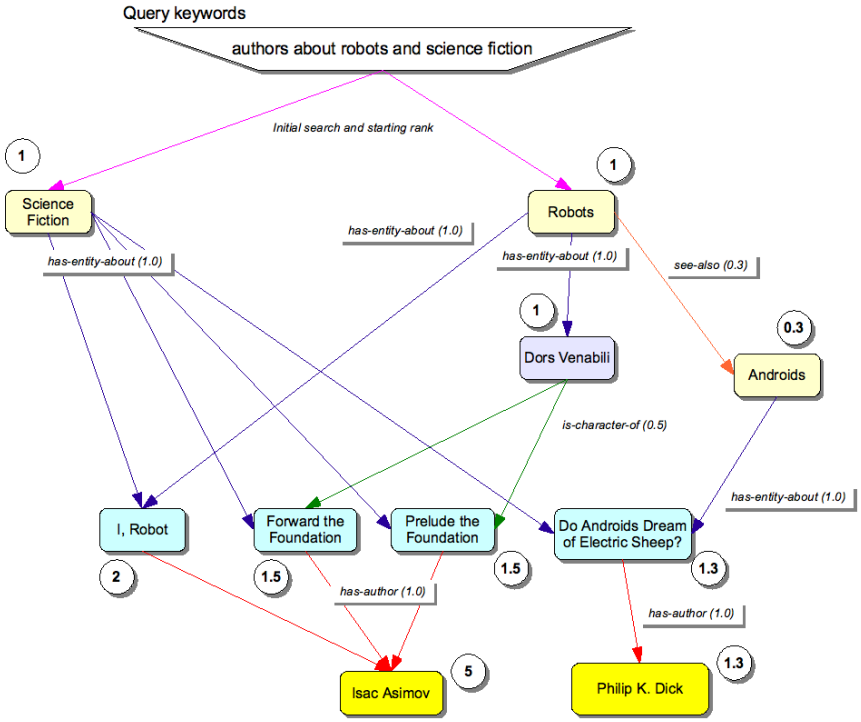


Figure 7.2: the Spreading Activaction algorhitm in practice. The topics "Science Fiction" and "Robots" are initially ranked by the keyword-based search. The score is propagated through nodes which are semantically related. The propagation is weighted according to link semantics. Each node receives ranking contributes from multiple paths.

7.2 Spreading Activation with graph query languages

The spreading formula introduced in the previous section is simple and effective in many cases. However, there are situations where it has limited expressiveness. For instance, let us consider the example in Figure 7.3, taken from [61]. It graphically shows a query on a RDF knowledge base which is based on BIOPAX, the ontology for describing biological pathways.

Suppose we give importance to interactions which have as one of the participants, a physical entity, which is in turn annotated with "P53" name. In practice, an interaction would receive some scoring when it is indirectly connected with a node that has certain properties, namely the properties of being physical entity and being related to P53. Scoring a node with such a criterion is not easy with the formula above, and consequently it is not easy, in the main step of the SA algorithm, shown in Figure 7.1, to select nodes which are meaningfully related to the one being processed.

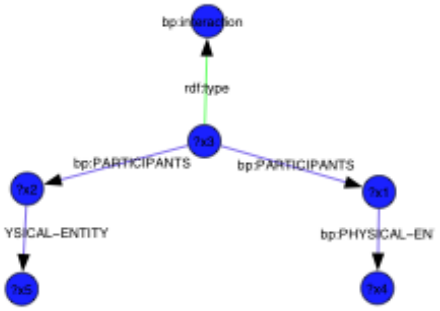


Figure 7.3: Applying Spreading Activaction to Semantic Web technologies. (Source: [61])

We propose a different version of the SA approach, consisting of a generalisation of the method the nodes are selected with. In the original formulation of the SA algorithm, all the outgoing nodes from the current one are selected, and the activation value is propagated to them. We instead propose to select, as "outgoing nodes", all the nodes that match a SPARQL query, which has the current node as parameter. Let us define a set of propagation selectors PS, where as "propagation selectors" we use parametric graph queries:

$$PS := \{ Query_k: i, v \rightarrow \{ j \} \}$$

Where *i* is the node parameter passed to the query, while *{j}* are a (possibly empty) set of nodes that the query returns back. More precisely, *i* and *j* are node identifiers and therefore, in our RDF context, they will correspond to URIs. All the queries are defined according to some graph query formalism, we will assume SPARQL. The definition above intends that the query must treat *i* as a URI and must return at least one variable that matches URIs (and not literals). Such variable is identified by the *v* parameter.

In the propagation step, shown in Figure 7.1, we may select "outgoing" nodes by using the formula:

$$\forall\, j \in Query_k(i, v): I'_j := I_j + w_k \cdot O_i \cdot \beta$$

that is: we run every query in SQ and, for every resulting node, we increase its input value, the same way it was done in the original version of the algorithm. w_k is a weight that represents the importance given to the query *k* and the results it produces.

This alternative selection approach allows more flexibility in deciding how the activation values may be spread over the network. We define this version of the algorithm "Semantic Spreading Activation" (SSA).

7.3 Application to Microarray knowledge

Let us consider some examples of how our method may be applied to knowledge bases described by means of MicroAnnOnto.

7.3.1 Initial scoring by means of evaluations

While SA is commonly used for searching, we may extend it to the general task of ranking an existing knowledge base. The key to do that is having some criterion, different than a search result, which provides an initial set of nodes, plus an initial scoring of the nodes.

We may use SPARQL queries to define such nodes and their scores. As an example, let us consider the MicroAnnOnto's property "evaluation". We may use the following algorithm to rank those entities which has received one or more evaluations:

```
For ech x in:
  SELECT ?x WHERE ?x mann:evaluation ?v
do
  x.out += v
```

We could specialise the scoring above by considering specific sub-properties of `mann:evaluation`. For instance `mann:relaness` could be a weighted more than `mann:precision`. Having a set of initial RDF resources that are, so to speak, "explicitly ranked" by the explicitly provided evaluations, we can "infer" the ranking of other nodes. By using the generic evaluation property, rather than the more specific ones, these go in the loop above anyway, since, as in previous examples, the queries are executed against the inferred graph.

More in general, any combination of semantic relations may be used to either provide the initial score, or the propagation selectors. For instance, in the the following example we give some importance to instances of `Assertion` concept:

```
for each x in:
  SELECT ?x WHERE ?x rdf:type mann:Assertion
do
  x.out += 1
```

Again, thanks to the inference, all possible specific assertions are scored by the code above.

7.3.2 Propagation of evaluations given by means of assertion

Backing to the definition of a proper set of selectors for MicroAnnOnto, we score those entities which are evaluated by means of `Comment`, a subclass of `Assertion`:

```
CommentSelector(c, x) :=
  SELECT ?x WHERE
    $c rdf:type mann:Comment
    $c mann:assertionSubject ?x

weight(CommentSelector) := 0.8
```

Note that we attenuate the weight of evaluations made by means of comments, due to the fact that the ones that are asserted by means of the `evaluation` property are usually made by the creator of the entity being evaluated.

7.3.3 Propagation of support

Another interesting case, is about assertion ranking, done by considering the kind of evidence attached to the assertions. We can use the following selector:

```
PositiveSupportSelector(s, x) :=
  SELECT ?x WHERE $s mann:entityPositivelySupports ?x

weight (PositiveSupportSelector) := 0.8
```

A similar rule may be defined for the property `entityNegativelySupports`. A negative weight is assigned to this property in this case. We can push in the selector above entities which are inferred both from the OWL semantics and from the application of inference rules.

7.3.4 Author-based ranking

Although the modern, Galilean science relies on the experimental evidence, people and their role are often considered in assigning a relevance to what they assert. The idea is that, when one has to screen a big amount of information, it is preferable to examine first that knowledge which has been confirmed by well acknowledged authorities in the field the knowledge is about. This will not ensure that correct information is extracted. However, even resources retrieved this way are usually based on experimental activity and it is reasonable to assume that something deserves more attention, when it has been concluded by many relevant people. The wide use, in the editorial field, of the impact factor index is an application of such concept.

The following selector transfers the rank of a person to the experiments he or she is responsible of:

```
ExperimentsWithAuthorSelector(a, x) :=
  SELECT ?x WHERE
    $a rdf:type mann:Person
    ?x mann:Experiment
    ?x mann:hasPrincipalInvestigator $a
```

We implicitly mean that the weight of this selector is 1 (by default when not specified). The author's ranking may be built by exploiting the part of MicroAnnOnto ontology that models the author's roles, their participation to conferences, their track record of publications and the terms the publications are annotated with. Complex ranking are brought on by, for example, transferring the publication ranks to the authors. In turn, this would spread to the people's experimental data and to their assertions, or the assertions supported by their data.

We plan to further study the relationships between networks of experimental data and networks of people, by investigating the literature about social network analysis[110][111][112].

7.3.5 Ranking assertions about gene expression

We conclude this review of query selectors by illustrating an example about gene expression assertions, which combines inference rules and Semantic Spreading Activation. We define an inference rule that creates summarising assertions about gene expression. Such assertions aim at counting how many times the expression of a gene and a condition type (identified by the annotating term) is reported.

```
[(?ass mann:assertionSubject ?pbset)
 (?ass mann:assertionContext ?ctx)
 (?ctx mann:entityTermAnnotation ?term)
```

```
=>
 addSupportToGExpression ( ?pbset, ?term, ?ass )
```

The expression `addSupportToGExpression ( ?pbset, ?term, ?ass )` is a custom built-in. The Jena rule reasoner allows to add this code hooks, so that custom tasks, which are hard to be expressed with the rule grammar, may be defined by means of the Java language. The job of the built-in above is to create or update the following statements:

```
[<id> rdf:type InferredGExpression
      mann:assertionSubject ?pbset
      mann:assertionContext [ <ctx:id> mann:entityTermAnnotation ?term]
      mann:supportedBy ?ass]
```

Where `id` and `ctx:id` are auto-generated identifiers. In practice: for a single pair of a probe set and a condition term, being stated as an expression combination, an assertion is created that receives support from the original assertions. This operation groups together relevant gene/term combinations about expressed genes. Once such summarising assertions are created, they are automatically considered for ranking, by selectors like the one defined in Section 7.3.3. At this stage, the summarising assertions receive the ranks that come from the original expression claims the summaries are based on. This also means that all the aspects considered for ranking the original assertions (e.g.: rank of evidence data, rank of experiment's authors) are automatically propagated to the summaries above.

The method of considering expressed pairs of gene/condition is a rather simple, other algorithms could be used to create entities which are similar to the summaries hereby described[145]. While we plan to introduce such algorithms in future, we are hereby interested in scoring gene expression statements, by exploiting their semantic representation and the variety of relations that link together results from expression analysis, experimental data and people working with data.

8 Closing remarks

We have started the project described in this thesis, aware of the benefits provided by distributed computing in managing scientific knowledge. In particular, we have focused on the activity of analysing and interpreting scientific experiments and data they generate. The outcomes of such activity are mostly made publicly available by means of everyday natural language, in the form of scientific papers and other publications. This is undoubtedly the most natural, easiest and most expressive way to disseminate and communicate knowledge. However, the natural language is ambiguous and in general far too much complex to be computationally exploited at full power, in the same way humans do. This is the more critical, the more the amount of scientific publications increases, even considering a specific field, like gene expression analysis.

Even a limited degree of formalisation may be helpful with such issues.

We have chosen to model the knowledge related to gene expression, by means of the languages and technologies which the W3C consortium is proposing for a new, Semantic Web.

The World Wide Web technologies have been used for long time in the field of Life Sciences as well. Such technologies make available a huge quantity of experimental data and related information. They allow to share information and promote collaboration. The growing interest that Life Sciences field is giving to the Semantic Web, is a natural consequence of the fact that the latter may be viewed as an evolution of the traditional web.

The RDF formalisms is a way to standardise the use of semantic networks for representing web resources, which, in general, are linked together without predefined structures. The RDF paradigm suits particularly well with much existing biological and medical information. Such information is already widely available through web applications and is encoded in a great variety of formats and structures. In other words, it is highly heterogeneous and highly interconnected.

The strong need for formal ontologies that there is in Life Sciences is another reason for the increasing interest in the Semantic Web. Even simple taxonomies helps in many situations, where complex searches and computations hardly would be possible without at least basic annotations in place. In addition, several, more complex and expressive ontologies are already being used in various kind of projects, while other ones are under development.

Inference and automatic reasoning are aspects related to ontologies and semantic networks. Although reasoners and the use of paradigms like production rules, are still limitedly used in the Semantic Web context, and in particular for the Life Sciences, interesting applications are being proposed [37][61].

In this thesis we have shown an example of Semantic Web technologies usefully applied to the specific Biological field of the microarray data analysis.

On one side, we have considered the need to face with the heterogeneous information which describes microarray experiments and the biological systems the experiments aim at investigating.

On another side, we have tried to provide a semi-formal representation of the outcomes of gene expression studies, by providing a model for representing biological assertions, together with a reference context and together with supporting by data sets.

We also have modelled the people who work with microarrays, including their role and related Research-artefacts, such as papers or conferences. We plan to further investigate the use of this kind of information, especially for what concerns its acquisition from public web sites.

Among the basic elements we have provided in our model, there are evaluations and several properties which allow to attach judgements about quality or similar aspects to microarray knowledge.

We propose, in Chapter 7, to introduce a ranking algorithm that would help in browsing and searching relevant microarray knowledge. Evaluations and quality representations are an important starting point in applying such an algorithm. The approach we propose may be considered as an additional type of inference, which complements (and is used in combination with) two other kinds of inference we apply to our model.

One type is the one that exploits the expressiveness of OWL and Description Logics. Even simplest semantic representations, such as the symmetry or transitivity of properties, are useful in the application domain we have considered.

Furthermore, more specific automatic reasoning is possible by means of the definition of domain-specific inference rules. In this case too, even simple rules are effective in computing useful knowledge. One example is about the rules which allow to propagate the terms attached to biological materials onto the data that are derived from such materials.

We have made practical experiments of our microarray knowledge modelling, by creating a semantic wiki application and populating it with the results coming from few real microarray studies. Although the application is still in a developmental stage, it effectively allows to access gene expression information and collaboratively work with it.

8.1 Discussion and future developments

There are several aspects of our work which will be interesting to investigate more in depth.

We have not addressed the problem of cross-comparing gene expression intensities coming from different microarray platforms or different experiments. In general, the microarray measurements return values that are proportional to quantities of DNA or RNA fragments which belong to target genes.

Therefore they give an indirect measurement of the final expressed proteins. Comparing results from different experiments, different technologies or different species and experimental conditions, is no way a trivial task. A typical approach used to overcome this problem, is considering specific data sets and redoing statistical normalisation on the whole data set. Another approach is considering, in the context of the same experiment, the differential expression under some altered conditions, with respect to a so called baseline, so that results which are only partially comparable are produced. We have chosen to fully rely on externally-provided values of expression intensities, which we assume to be comparable in a single knowledge base, modelled with our MicroAnnOnto ontology. This is a simple approach, which has allowed us to focus on semantic aspects of microarray data and analysis. In future, we aim at investigating more on this issue.

As for the wiki interface we have described in Chapter 6, we mainly address the Bioinformatics community, rather than the one of pure biologists. We plan to improve the user interface aspects of our work. One way to do that would be the introduction, in the wiki, of visual components and the use of the AJAX technology. A further improvement would be the review of the data import procedures, for accepting more input formats and better integrating existing tools. Finally, we plan to study an integration of our wiki interface with existing, more traditional microarray tools. One of such tools is the repository software BASE, which we intend consider for the future, and integrate with both the work presented in this thesis and what we have done for the Genopolis Database, described in Chapter 3. The main idea is to complement BASE with the graphical browsing of expression profiles, to allow the user to manage interesting subsets of genes and experimental conditions. We also plan to develop functions for exporting data sets discovered via the graphical browsing, using RDF and our MannOnto ontology as export format.

Similarly, tools for microarray analysis, such as Bioconductor or GeneSpring could export analysis results in a format compatible with MannOnto. This would allow to quickly share and compare such results.

We are aware that performance is an issue in applications of our MicroAnnOnto model. That is a particularly critical issue when facing with reasoning. We are studying possible solutions for improving speed in computing inference. One way to achieve that, is considering alternative reasoners to the ones embedded in Jena, such as Pellet or InstanceStore[146]. Another possible improvement is to work on the caching of Jena triple stores. A third option could be reviewing the Jena rules that realise OWL reasoning. Speed increase could be achieved from keeping only those rules that are most important for our MicroAnnonto model. For instance, in most cases we are not interested in cardinality restrictions. This would allow us to have a more agile and faster reasoning engine, tailored to our specific needs.

Several existing ontologies and OWL models could be usefully integrated in our Microarray knowledge model. One of them is the EXPO ontology[36], which is designed for the representation of experiments and their objectives, experimental hypotheses and conclusions. Although EXPO is not microarray-

specific, we believe a possible integration between our MicroAnnOnto model and EXPO would be interesting, especially considering the recently started ART project[147], which aims at developing tools for the formal representation of the content of scientific papers.

BioPAX is another interesting model. Its ability to represent, with semantically rich OWL constructs, complex networks of bio-molecular interactions could be used to provide more detailed assertions about biological discoveries made with microarray analysis.

Concerning taxonomy-like ontologies, the ambitious OBI project is currently finalising a first version of the Ontology for Biological Investigations, which should be a useful starting point for the whole Functional Genomics field, and of course we are interested in using such an ontology in MicroAnnoOnto, as a source of terms.

Similarly, we are interested in dealing with the management of multi-omics data, using an approach similar to the one we have presented in this thesis.

Another project which is relevant to us is the demo which has been developed by the W3C group named "Semantic Web Healthcare and Life Sciences Interest Group", or HCLS[82]. It would be interesting to export our microarray knowledge in an RDF format which can be integrated with other kind of neurology-related information in the HCLS demo. In fact, by doing that, we could prove the benefits of RDF formats in achieving knowledge integration. Microarray knowledge would be put in a network of gene functional annotations, medical subject annotations, pathway information.

We have drafted, in the previous chapter, a method for ranking the MicroAnnOnto-based knowledge. This is based on an extension of the Spreading Activation algorithm, what we call Semantic Spreading Activation (SSA). Our approach takes advantage of semantic network query languages. We have shown examples based on SPARQL. We plan to provide a full implementation of our method. Our SSA approach may be used in combination with traditional inference over OWL knowledge.

We believe that interesting applications of this approach could be realised in our microarray domain. On one side, genes and terms about experimental conditions could be usefully scored, according to their involvement in interesting experimental findings. In turn, experimental results could be ranked by means of data quality criteria and other evaluations. On another side, experiments and analysis results could be ranked according to people working with them.

We are aware that it is currently difficult to find interesting structured information about people working in the Biology field. One available tool is the article annotations provided by public web sites, such as PUBMED. For instance in [102][148] the integration with PUBMED and GeneOntology is described. These annotations could be used with co-citation analysis techniques [149], to gather useful information. In addition, the so called social networking is becoming popular in Life Sciences too. For instance, Nature Network[150] could be a possible source of people-related data, as it is described in [151]. Having a knowledge base of people and their links with experimental activity, could be an interesting starting point for worthily applying Social Network Analysis techniques[110][111].

Finally, among the ongoing and future developments of this project, we worth mention the DC-THERA European project project[134], for which we are developing a form of "gene expression atlas", that would allow people to share microarray-related knowledge, including experiment descriptions and

relevant analysis results. We plan to develop an application which is substantially like the semantic wiki we have presented in Chapter 6, and to integrate it with the results produced by tools developed by other partners from the DC-THERA project.

Doing that will be a more concrete application example of the work presented in this thesis. It would also be another proof of the benefits of using the Semantic Web in microarray data management and in the Life Sciences.

References

[1] Edited by Achinstein P, Hannaway O. *Observation, Experiment, and Hypothesis in Modern Physical Science*. Bradford Books March 1985.

[2] Jones AL. *Logic, Inductive and Deductive: An Introduction to Scientific Method.* Henry Holt, 1909.

[3] Falciani F. *Microarray technology through applications.* Taylor & Francis. 2007.

[4] Berners-Lee T, Hendler J and Lassila O. *The Semantic Web.* Scientific America 2001.

[5] Passin TB. *Explorer's Guide to the Semantic Web.* Manning July 2004.

[6] Antoniou G, van Harmelen. *A Semantic Web Primer.* The MIT Press, 2004.

[7] Edited by Albert B. *Molecular Biology of the Cell*. Garland Publishing Inc, US. Mar 2002.

[8] Crick F. *Central dogma of molecular biology*. Nature. 1970 Aug 8;227(5258):561-3.

[9] Werner E. *Meeting report: the future and limits of systems biology.* Sci STKE. 2005 Apr 5;2005(278):pe16.

[10] Stransky B, Barrera J, Ohno-Machado L, De Souza SJ. *Modeling cancer: integration of "omics" information in dynamic systems*. J Bioinform Comput Biol. 2007 Aug;5(4):977-86.

[11] Trevino V, Falciani F, Barrera-Saldaña HA.*DNA microarrays: a powerful genomic tool for biomedical and clinical research.* Mol Med. 2007 Sep-Oct;13(9-10):527-41.

[12] Butte A. *The use and analysis of microarray data*. Nat Rev Drug Discov. 2002 Dec;1(12):951-60.

[13] Stoughton RB. *Applications of DNA microarrays in biology.* Annu Rev Biochem. 2005;74:53-82.

[14] Joyce AR, Palsson BO. *The model organism as a system: integrating 'omics' data sets.* Nat Rev Mol Cell Biol 2006 Mar;7(3):198-210.

[15] *BioMap, an infrastructure for storing and integrating biological investigations, employing transcriptomics, proteomics and metabolomics technologies*. http://www.ebi.ac.uk/net-project/projects.html

[16] Kim S, Misra A. *SNP Genotyping: Technologies and Biomedical Applications.* Annual Review of Biomedical Engineering Vol. 9: 289-320 (Volume publication date August 2007) .

[17] Mockler T C, Chan S, Sundaresan A, Chen H, Jacobsen S E, Ecker J R. *Applications of DNA tiling arrays for whole-genome analysis*. Genomics, Volume 85, Issue 5, May 2005, Page 655.

[18] Brazma A, Hingamp P, Quackenbush J, Sherlock G, Spellman P, Stoeckert C, Aach J, Ansorge W, Ball CA, Causton HC, Gaasterland T, Glenisson P, Holstege FC, Kim IF, Markowitz V, Matese JC, Parkinson H, Robinson A, Sarkans U, Schulze-Kremer S, Stewart J, Taylor R, Vilo J, Vingron M. *Minimum information about a microarray experiment (MIAME)-toward standards for microarray data*. Nat Genet. 2001 Dec;29(4):365-71.

[19] Yauk CL, Berndt ML, Williams A, Douglas GR. *Comprehensive comparison of six microarray technologies*. Nucleic Acids Res. 2004 Aug 27;32(15):e124.

[20] Wang H, He X, Band M, Wilson C, Liu L. *A study of inter-lab and inter-platform agreement of DNA microarray data*. BMC Genomics. 2005 May 11;6(1):71.

[21] Spellman PT, Miller M, Stewart J, Troup C, Sarkans U, Chervitz S, Bernhart D, Sherlock G, Ball C, Lepage M, Swiatek M, Marks WL, Goncalves J, Markel S, Iordan D, Shojatalab M, Pizarro A, White J, Hubley R, Deutsch E, Senger M, Aronow BJ, Robinson A, Bassett D, Stoeckert CJ Jr, Brazma A. *Design and implementation of microarray gene expression markup language (MAGE-ML)*. Genome Biol. 2002 Aug 23;3(9):RESEARCH0046. Epub 2002 Aug 23.

[22] Eyre TA, Ducluzeau F, Sneddon TP, Povey S, Bruford EA, Lush MJ. *The HUGO Gene Nomenclature Database, 2006 updates*. Nucleic Acids Res. 2006 Jan 1;34(Database issue):D319-21.

[23] The Gene Ontology Consortium. *GO Gene Ontology: tool for the unification of biology*. Nature Genet. 25: 25-29, 2000.

[24] Whetzel PL, Parkinson H, Causton HC, Fan L, Fostel J, Fragoso G, Game L, Heiskanen M, Morrison N, Rocca-Serra P, Sansone SA, Taylor C, White J, Stoeckert CJ Jr. *The MGED Ontology: a resource for semantics-based description of microarray experiments*. Bioinformatics. 2006 Apr 1;22(7):866-73. Epub 2006 Jan 21.

[25] Whetzel PL, Brinkman RR, Causton HC, Fan L, Field D, Fostel J, Fragoso G, Gray T, Heiskanen M, Hernandez-Boussard T, Morrison N, Parkinson H, Rocca-Serra P, Sansone SA, Schober D, Smith B, Stevens R, Stoeckert CJ Jr, Taylor C, White J, Wood A; FuGO Working Group. *Development of FuGO: an ontology for functional genomics investigations*. OMICS. 2006 Summer;10(2):199-204.

[26] Luciano JS, Stevens RD. *e-Science and biological pathway semantics*. BMC Bioinformatics. 2007 May 9;8 Suppl 3:S3.

[27] Kanehisa M, Goto S, Hattori M, Aoki-Kinoshita KF, Itoh M, Kawashima S, Katayama T, Araki M, Hirakawa M. *From genomics to chemical genomics: new developments in KEGG*. Nucleic Acids Res. 2006 Jan 1;34(Database issue):D354-7.

[28] Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC, Kitano H, Arkin AP, Bornstein BJ, Bray D, Cornish-Bowden A, Cuellar AA, Dronov S, Gilles ED, Ginkel M, Gor V, Goryanin II, Hedley WJ, Hodgman TC, Hofmeyr JH, Hunter PJ, Juty NS, Kasberger JL, Kremling A, Kummer U, Le Novere N, Loew LM, Lucio D, Mendes P, Minch E, Mjolsness ED, Nakayama Y, Nelson MR, Nielsen PF, Sakurada T, Schaff JC, Shapiro BE, Shimizu TS, Spence HD, Stelling J, Takahashi K, Tomita M, Wagner J, Wang J; SBML Forum. *The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models*. Bioinformatics. 2003 Mar 1;19(4):524-31.

[29] Rayner TF, Rocca-Serra P, Spellman PT, Causton HC, Farne A, Holloway E, Irizarry RA, Liu J, Maier DS, Miller M, Petersen K, Quackenbush J, Sherlock G, Stoeckert CJ Jr, White J, Whetzel PL, Wymore F, Parkinson H, Sarkans U, Ball CA, Brazma A. *A simple spreadsheet-based, MIAME-supportive format for microarray data: MAGE-TAB*. BMC Bioinformatics. 2006 Nov 6;7:489.

[30] *MAGE-TAB and Tab2MAGE Home*. http://tab2mage.sourceforge.net/

[31] Joslyn CA, Mniszewski SM, Fulmer A, Heaton G. *The gene ontology categorizer*. Bioinformatics. 2004 Aug 4;20 Suppl 1:i169-77.

[32] Grosu P, Townsend JP, Hartl DL, Cavalieri D. *Pathway Processor: a tool for integrating whole-genome expression results into metabolic networks*. Genome Res. 2002 Jul;12(7):1121-6.

[33] Marcondes CH. *From scientific communication to public knowledge : the scientific article web published as a knowledge base*. In: Edited by Dubrova M, Engelen J. Proceedings International Conference on Electronic Publishing, 9th, ICCC ElPub, pp. 119-127, 2005.

[34] Chklovski T, Ratnakar V, Gi Yl. *User Interfaces with Semi-Formal Representations: a Study of Designing Argumentation Structures*. Proceedings of Conference on Intelligent User Interfaces (IUI05)., San Diego. 2005.

[35] Buckingham Shum SJ, Uren VS, Li G, Sereno B, Mancini C. *Modelling Naturalistic Argumentation in Research Literatures: Representation and Interaction Design Issues*. International Journal of Intelligent Systems, (Specia lIssue on Computational Models of Natural Argumentation), 22, (1), pp.17–47 2007.

[36] Soldatova LN, King RD. *An ontology of scientific experiments*. J R Soc Interface. 2006 Dec 22;3(11):795-803.

[37] Soldatova LN, Clare A, Sparkes A, King RD. *An ontology for a Robot Scientist*. Bioinformatics. 2006 Jul 15;22(14):e464-71.

[38] Missier P, Embury S, Greenwood M, Preece A, Jin B. *An Ontology-Based Approach to Handling Information Quality in e-Science*. Proc 4th e-Science All Hands Meeting (AHM2005), 2005.

[39] Missier P, Embury S, Greenwood M, Preece A, Jin B. *Quality views: Capturing and Exploiting the User Perspective on Data Quality*. Proc 32nd International Conference on Very Large Data Bases (VLDB 2006), ACM Press, pages 977-988, 2006.

[40] Horrocks I, Patel-Schneider PF, van Harmelen F. *From SHIQ and RDF to OWL: The Making of a Web Ontology Language*. J. of Web Semantics, 1(1):7-26, 2003.

[41] Parkinson H, Kapushesky M, Shojatalab M, Abeygunawardena N, Coulson R, Farne A, Holloway E, Kolesnykov N, Lilja P, Lukk M, Mani R, Rayner T, Sharma A, William E, Sarkans U, Brazma A.*ArrayExpress--a public database of microarray experiments and gene expression profiles*. Nucleic Acids Res. 2007 Jan;35(Database issue):D747-50.

[42] Brazma A, Kapushesky M, Parkinson H, Sarkans U, Shojatalab M. *Data storage and analysis in ArrayExpress*. Methods Enzymol. 2006;411:370-86.

[43] Kapushesky M, Kemmeren P, Culhane AC, Durinck S, Ihmels J, Korner C, Kull M, Torrente A, Sarkans U, Vilo J, Brazma A. *Expression Profiler: next generation--an online platform for analysis of microarray data*. Nucleic Acids Res. 2004 Jul 1;32(Web Server issue):W465-70.

[44] Edgar R, Domrachev M, Lash AE. *Gene Expression Omnibus: NCBI gene expression and hybridization array data repository*. Nucleic Acids Res. 2002;30:207–210. doi: 10.1093/nar/30.1.207.

[45] Ikeo K, Ishi-i J, Tamura T, Gojobori T, Tateno YCR.*CIBEX: center for information biology gene expression database*. C R Biol. 2003;326:1079–1082. doi: 10.1016/j.crvi.2003.09.034.

[46] Splendiani A, Brandizi M, Even G, Beretta O, Pavelka N, Pelizzola M, Mayhaus M, Foti M, Mauri G, Ricciardi-Castagnoli P. *The genopolis microarray database*. BMC Bioinformatics. 2007 Mar 8;8 Suppl 1:S21.

[47] *Affymetrix Inc*. http://www.affymetrix.com

[48] Gentleman R, Carey V, Huber W, Irizarry R, Dudoit S.*Bioinformatics and Computational Biology Solutions Using R and Bioconductor*. Released 1 Sept 2005. Springer Online.

[49] *GeneSpring Analysis Platform*. http://www.genespring.com

[50] Saal L, Troein C, Vallon-Christersson J, Gruvberger S, Borg Ă, Peterson C.*BioArray Software Environment: A Platform for Comprehensive Management and Analysis of Microarray Data*. Genome Biol. 2002;3:software0003. doi: 10.1186/gb-2002-3-8-software0003.

[51] Hancock D, Wilson M, Velarde G, Morrison N, Hayes A, Hulme H, Wood AJ, Nashar K, Kell DB, Brass A.*maxdLoad2 and maxdBrowse: standards-compliant tools for microarray experimental annotation, data management and dissemination*. BMC Bioinformatics. 6:264. 2005 Nov 3.

[52] *Novell GroupWise*. http://www.novell.com/products/groupwise

[53] *eGroupWare*. http://www.egroupware.org/

[54] *Fostering Collaboration to Accelerate Discovery Research*. http://www.chem.agilent.com/cag/feature/09-03/sep03_synapsia.htm

[55] *Array Results Manager*. http://www.biodiscovery.com/index/arm

[56] Massar JP, Travers M, Elhai J, Shrager J. *BioLingua: a programmable knowledge environment for biologists*. Bioinformatics. 2005 Jan 15;21(2):199-207.

[57] Clustal W. *improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice*. Nucleic Acids Res. 1994 Nov 11;22(22):4673-80.

[58] Luciano JS. *PAX of mind for pathway researchers*. Drug Discov Today. 2005 Jul 1;10(13):937-42.

[59] Baral C, Chancellor K, Tran N, Tran NL, Joy A, Berens M. *A knowledge based approach for representing and reasoning about signaling networks*. Bioinformatics. 2004 Aug 4;20 Suppl 1:i15-22.

[60] Splendiani A. *Integration of ontologies and high-throughput data in Bioinformatics*. PhD Thesis, University of Milano-Bicocca. May 2006.

[61] Splendiani A. *Semantic browsing of pathway ontologies and biological networks with RDFScape*. Working Paper. http://drops.dagstuhl.de/opus/volltexte/2006/474/.

[62] *Semantic Web*. http://en.wikipedia.org/wiki/Semantic_Web

[63] *Tim O'Reilly. What Is Web 2.0*. http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

[64] *The Social Web: Creating An Open Social Network with XDI*. http://journal.planetwork.net/article.php?lab=reed0704&page=1

[65] *Folksonomy Coinage and Definition*. http://www.vanderwal.net/folksonomy.html

[66] *Uniform Resource Identifier*. http://en.wikipedia.org/wiki/Uniform_Resource_Identifier

[67] Clark T, Martin S, Liefeld T. *Globally distributed object identification for biological knowledgebases*. Brief Bioinform. 2004 Mar;5(1):59-70.

[68] Martin S, Hohman MM, Liefeld T. *The impact of Life Science Identifier on informatics data*. Drug Discov Today. 2005 Nov 15;10(22):1566-72.

[69] *URL +1, LSID -1, or why I don't care about the sematic web*. http://www.nodalpoint.org/2007/07/22/url_1_lsid_1_or_why_i_dont_care_about_the_sematic_web

[70] *LSID URN/URI Notes*. http://esw.w3.org/topic/HCLSIG_BioRDF_Subgroup/LSID_URN_URI?highlight=%28lsid%29

[71] *RDF Primer*. http://www.w3.org/TR/REC-rdf-syntax/

[72] Sowa JF. *Semantic Networks*. Encyclopedia of Artificial Intelligence, edited by Stuart C. Shapiro, Wiley, 1987, second edition, 1992.

[73] Rocha C, Schwabe D, de Aragao MP. *A hybrid approach for searching in the semantic web*. In Proceedings of the 13th International World Wide Web Conference, 2004.

[74] *RDF/XML Syntax Specification*. http://www.w3.org/TR/rdf-syntax-grammar/

[75] *Primer: Getting into RDF & Semantic Web using N3*. http://www.w3.org/2000/10/swap/Primer

[76] Wang X, Gorlitsky R, Almeida JS. *From XML to RDF: how semantic web technologies will change the design of 'omic' standards*. Nat Biotechnol. 2005 Sep;23(9):1099-103.

[77] Edited by Lacroix Z, Critchlow T. *Bioinformatics, managing scientific data*. Morgan Kaufmann Publisher. 2003.

[78] Cheung KH, Yip KY, Smith A, Deknikker R, Masiar A, Gerstein M. *YeastHub: a semantic web use case for integrating data in the life sciences domain*. Bioinformatics. 2005 Jun;21 Suppl 1:i85-96.

[79] Smith AK, Cheung KH, Yip KY, Schultz M, Gerstein MK. *LinkHub: a Semantic Web system that facilitates cross-database queries and information retrieval in proteomics*. BMC Bioinformatics. 2007 May 9;8 Suppl 3:S5.

[80] *The D2RQ Plattform - Treating Non-RDF Databases as Virtual RDF Graphs*. http://sites.wiwiss.fu-berlin.de/suhl/bizer/d2rq/index.htm

[81] *Dublin Core Metadata Initiative*. http://dublincore.org/

[82] *HCLS Banff2007Demo*. http://esw.w3.org/topic/HCLS/Banff2007Demo

[83] *SPARQL Query Language for RDF*. http://www.w3.org/TR/rdf-sparql-query/

[84] Neumann EK, Quan D. *BioDash: a Semantic Web dashboard for drug development*. Pac Symp Biocomput. 2006;:176-87.

[85] Huynh D, Mazzocchi S, Karger D. *Piggy Bank: Experience the Semantic Web Inside Your Web Browser* International Semantic Web Conference (ISWC) 2005.

[86] Tummarello G, Morbidoni C, Nucci M. *Enabling Semantic Web communities with DBin: an overview*. Proceedings of the Fifth International Semantic Web Conference ISWC 2006, November 2006.

[87] Wilkinson M, Schoof H, Ernst R, Haase D. *BioMOBY successfully integrates distributed heterogeneous bioinformatics Web Services. The PlaNet exemplar case*. Plant Physiol. 2005 May;138(1):5-17.

[88] Stevens RD, Robinson AJ, Goble CA. *myGrid: personalised bioinformatics on the information grid*. Bioinformatics. 2003;19 Suppl 1:i302-4.

[89] Oinn T, Addis M, Ferris J, Marvin D, Senger M, Greenwood M, Carver T, Glover K, Pocock MR, Wipat A, Li P. *Taverna: a tool for the composition and enactment of bioinformatics workflows*. Bioinformatics. 2004 Nov 22;20(17):3045-54. Epub 2004 Jun 16.

[90] Newcomer E, Lomow L. *Understanding SOA with Web Services*. Addison Wesley. 2005.

[91] Gruber T. *A Translation Approach to Portable Ontology Specifications*. Knowledge Systems Laboratory. Technical Report KSL 92-71. April 1993.

[92] Guarino N. *Formal Ontology in Information Systems*. Proceedings of FOIS'98, Trento, Italy, 6-8 June 1998 .

[93] Soldatova LN, King RD. *Are the current ontologies in biology good ontologies?*. Nat Biotechnol. 2005 Sep;23(9):1095-8.

[94] Stoeckert C, Ball C, Brazma A, Brinkman R, Causton H, Fan L, Fostel J, Fragoso G, Heiskanen M, Holstege F, Morrison N, Parkinson H, Quackenbush J, Rocca-Serra P, Sansone SA, Sarkans U, Sherlock G, Stevens R, Taylor C, Taylor R, Whetzel P, White J. *Wrestling with SUMO and bio-ontologies*. Nat Biotechnol. 2006 Jan;24(1):21-2; author reply 23.

[95] McGuinness DL. *Ontologies Come of Age*. In: Ed Fensel D, Hendler J, Lieberman H, Wahlster W. *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential* MIT Press, 2003.

[96] *Vocabulary Description Language 1.0: RDF Schema*. http://www.w3.org/TR/rdf-schema/

[97] Guarino N, Carrara M, Giaretta P. *An Ontology of Meta-Level Categories*. {KR}'94: Principles of Knowledge Representation and Reasoning 270--280 (1994) .

[98] Baader F, Calvanese D, McGuinness D, Nardi D, Patel-Schneider P. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[99] *Racer Systems GmbH*. http://www.racer-systems.com/

[100] Sirin E, Parsia B, Cuenca Grau B, Kalyanpur A Katz Y. *Pellet: A practical OWL-DL reasoner*. Journal of Web Semantics, 5(2), 2007.

[101] *OBO Foundry Ontologies*. http://www.obofoundry.org/

[102] Doms A, Schroeder M. *GoPubMed: exploring PubMed with the Gene Ontology*. Nucleic Acids Res. 2005 Jul 1;33(Web Server issue):W783-6.

[103] Vastrik I, D'Eustachio P, Schmidt E, Joshi-Tope G, Gopinath G, Croft D, de Bono B, Gillespie M, Jassal B, Lewis S, Matthews L, Wu G, Birney E, Stein L. *Reactome: a knowledge base of biologic pathways and processes*. Genome Biol. 2007;8(3):R39.

[104] Caspi R, Foerster H, Fulcher CA, Kaipa P, Krummenacker M, Latendresse M, Paley S, Rhee SY, Shearer AG, Tissier C, Walk TC, Zhang P, Karp PD. *The MetaCyc Database of metabolic pathways and enzymes and the BioCyc collection of Pathway/Genome Databases*. Nucleic Acids Res. 2007 Oct 27.

[105] Demir E, Babur O, Dogrusoz U, Gursoy A, Nisanci G, Cetin-Atalay R, Ozturk M. *PATIKA: an integrated visual environment for collaborative construction and analysis of cellular pathways*. Bioinformatics. 2002 Jul;18(7):996-1003.

[106] Karp P, Paley S, Romero P. *The Pathway Tools Software*. Bioinformatics 18:S225-32 2002.

[107] Perez-Rey D, Maojo V, Garcia-Remesal M, Alonso-Calvo R, Billhardt H, Martin-Sanchez F, Sousa A. *ONTOFUSION: ontology-based integration of genomic and clinical databases*. Comput Biol Med. 2006 Jul-Aug;36(7-8):712-30. Epub 2005 Sep 6.

[108] Wolstencroft K, Stevens R, Haarslev V. *Applying OWL Reasoning to Genomic Data*. In: Edited by Baker CJO, Cheung KH. *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*. Springer Verlag, 2006.

[109] Kashyap V, Hongsermeier T. *Can Semantic Web Technologies enable Translational Medicine?*. In: Ed Baker CJO, Cheung KH. *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*. Springer Verlag, 2006.

[110] Carrington PJ, Scott J, Wasserman S. *Models and methods in social network analysis*. Cambridge University Press, 2005.

[111] de Nooy W, Mrvar A, Batagelj V. *Exploratory social network analysis with Pajek*. Cambridge University Press, 2005.

[112] Mika P. *Social Networks and the Semantic Web*. Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI'04), IEEE, Computer Society, 2004.

[113] *Home of the Protégé platform.* http://protege.stanford.edu

[114] Schober D, Leser U, Zenke M, Reich J. *GandrKB--ontological microarray annotation and visualization.* Bioinformatics. 2005 Jun 1;21(11):2785-6. Epub 2005 Mar 31.

[115] Sure Y, Bloehdorn S, Haase P, Hartmann J, Oberle D.*The SWRC Ontology - Semantic Web for Research Communities.* In Proceedings of the 12th Portuguese Conference on Artificial Intelligence (EPIA 2005). Springer, December 2005.

[116] Miles A, Matthews B, Wilson M. *SKOS Core: Simple Knowledge Organisation for the Web.* 2005-09-12. http://isegserv.itd.rl.ac.uk/public/skos/press/dc2005/dc2005skospaper.pdf.

[117] *The bio-zen Ontology Framework*. http://neuroscientific.net/index.php?id=43

[118] Giles J. *Key biology databases go wiki*. Nature. 2007 Feb 15;445(7129):691.

[119] *SWAD-Europe Deliverable 10.1: Scalability and Storage: Survey of Free Software/Open Source RDF storage systems. Technical report. 2002.* http://www.w3.org/2001/sw/Europe/reports/rdf_scalable_storage_report/

[120] Giarratano JC, Riley GD. *Expert Systems: Principles and Programming*. Course Technology; 4Rev Ed edition (15 Oct 2004).

[121] Sterling L, Shapiro EY. *The Art of PROLOG: Advanced Programming Techniques (Logic Programming).* The MIT Press; 2Rev Ed edition (21 April 1994).

[122] *DIG 2.0: The DIG Description Logic Interface*. http://dig.cs.manchester.ac.uk/

[123] Waldman S.. *Who knows?*. The Guardian. 26 Oct 2004.

[124] Völkel M, Krötzsch M, Vrandecic D, Haller H, Studer R. *Semantic Wikipedia.* In Proceedings of the 15th international conference on World Wide Web, WWW 2006, Edinburgh, Scotland, May 23-26, 2006.

[125] *Wikipedia:How to edit a page.* http://en.wikipedia.org/wiki/Wikipedia:How_to_edit_a_page

[126] Schaffert S, Westenthaler R, Gruber A. *IkeWiki: A User-Friendly Semantic Wiki.* 3rd European Semantic Web Conference (ESWC06). June 2006.

[127] Buffa M, Gandon F. *SweetWiki : Semantic Web Enabled Technologies in Wiki* conference ACM Wikisym 2006. Agost 2006.

[128] Dello K, Paslaru Bontas Simperl E, Tolksdorf R. *Creating and using Semantic Web information with Makna.* Proceedings of the First Workshop on Semantic Wikis, ESWC 2006.

[129] Youngman N. *An Introduction to JSPWiki.* Linux Gazette, Issue 108. November 2004.

[130] Campanini SE, Castagna P, and Tazzoli R.*Platypus Wiki: a Semantic Wiki Wiki Web* First Italian Semantic Web Workshop Semantic Web Applications and Perspectives (SWAP). 2004.

[131] Wang K. *Gene-function wiki would let biologists pool worldwide resources.* Nature. 2006 Feb 2;439(7076):534.

[132] Backhaus M, Hoehndorf R, Bacher J, Loebe F, Visagie J,Herre H, Kelso J. *BOWiki - a collaborative gene annotation and biomedical ontology curation framework*. Poster, Bio-Ontologies Special Interest Group Workshop, ISMB 2007. July 2007.

[133] *MyBio is the biologist's wiki workbench*. http://www.mybio.net

[134] *About the DC-THERA European Network.*

[135] Kelsall BL, Biron CA, Sharma O, Kaye PM. *Dendritic cells at the host-pathogen interface*. Nat. Immunol. 3:699. 2002.

[136] Tang Z, Saltzmann A. *Understanding human dendritic cell biology through gene profiling.* Inflamm Res 2004, 53:424-441.

[137] *Home of OntoSpere3D tool*. http://ontosphere3d.sourceforge.net

[138] Cote RG, Jones P, Apweiler R, Hermjakob H. *The Ontology Lookup Service, a lightweight cross-platform tool for controlled vocabulary queries.* BMC Bioinformatics. 2006 Feb 28;7:97.

[139] *Home of Lucene*. http://lucene.apache.org/

[140] Dhyani D, Keong Ng W, Bhowmick SS. *A survey of Web metrics.* ACM Comput. Surv. 34(4): 469-503 (2002).

[141] *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*. http://www.w3.org/Submission/SWRL/

[142] Esuli A, Sebastiani F. *PageRanking WordNet Synsets: An Application to Opinion Mining*. n Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL), pp.424--431, 2007.

[143] Ding L, Finin T, Joshi A, Pan R, Scott Cost R, Sachs J, Doshi V, Reddivari P, Peng Y*Swoogle, a Search and Metadata Engine for the Semantic Web*. Proc. 13th ACM Conference of Information and Knowledge Management (CIKM '04), Nov. 2004.

[144] Crestani F. *Application of Spreading Activation Techniques in Information Retrieval*. Artificial Intelligence Review, 1997.

[145] Kim KY, Ki DH, Jeong HJ, Jeung HC, Chung HC, Rha SY.*Novel and simple transformation algorithm for combining microarray data sets*. BMC Bioinformatics. 2007 Jun 25;8:218.

[146] Horrocks I, Li L, Turi D, Bechhofer S. *The instance store: Description logic reasoning with large numbers of individuals*. IJCAR2004, Jan. 2004.

[147] Soldatova LN, Batchelor CR, Liakata M, Fielding HH, Lewis S, King RD. *ART: An ontology based tool for the translation of papers into Semantic Web format*. Bio-Ontologies Special Interest Group Workshop, ISMB 2007. July 2007.

[148] Plikus MV, Zhang Z, Chuong CM. *PubFocus: semantic MEDLINE/PubMed citations analytics through integration of controlled biomedical dictionaries and ranking algorithm*. BMC Bioinformatics. 2006 Oct 2;7:424.

[149] Synnestvedt MB, Chen C, Holmes JH. *CiteSpace II: visualization and knowledge discovery in bibliographic databases*. AMIA Annu Symp Proc. 2005;:724-8.

[150] *Home of Nature Network*. http://network.nature.com/

[151] *Exploring the 'Nature Network': SVG javascript foaf JSON XHTML*. http://plindenbaum.blogspot.com/2007/05/exploring-nature-network-svg-javascript.html