TECHNISCHE UNIVERSITÄT MÜNCHEN

Lehrstuhl für Genomorientierte Bioinformatik

Towards
Effective Biomedical Knowledge Discovery through Subject-Centric Semantic Integration of the Life-Science Information Space

Karamfilka Krasimirova Nenova

Vollständiger Abdruck der von der Fakultät Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

genehmigten Dissertation.

Vorsitzender: Prof. Dr. M. Hrabé de Angelis

Prüfer der Dissertation:

1. Univ.-Prof. Dr. H.-W. Mewes
2. Univ.-Prof. Dr. R. Zimmer
   (Ludwig-Maximilians-Universität München)

Die Dissertation wurde am 20.10.2008 bei der Technischen Universität München eingereicht und durch die Fakultät Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt am 19.03.2009 angenommen.

Acknowledgements

This is a great opportunity to express my deep gratitude to many people who supported me during my Ph.D. research and in the writing process of this thesis. Without their help, guidance and encouragement, this study would not have been completed.

Above all, I would like to thank Dr. Volker Stümpflen, who inspired me for the undertaken research. His professional competence, valuable advices and commitment were indispensable for my work. As a mentor and leader of the Biological Information Systems group at MIPS, he has been always open-minded for constructive discussions and core contributor to the very friendly atmosphere in the group. I am also very grateful to Prof. Dr. Hans-Werner Mewes for giving me the opportunity to acquire doctor's degree at the Institute for Bioinformatics and Systems Biology. I highly appreciate his trust in me and my work and I am thankful for his guidance in the key moments of my research, the unreserved support and the patience during the writing process. I would like to extend these thanks to Prof. Dr. Inge Schestag, a former study advisor at the University for Applied Sciences Darmstadt and a good friend, for encouraging me to dare my Ph.D. study.

At this place, I would like to thank also past and present colleagues at MIPS for listening, giving advices and helping me in the diverse topics regarding my work, in particular Dr. Matthias Oesterheld, Richard Gregory, Thorsten Barnickel, Roland Arnold, Mara Hartsperger, Florian Büttner, Octave Noubibou, Christoph Oberhauser, Thorsten Schmidt, Sebastian Toepel, Dr. Thomas Rattei and Dr. Louise Gregory. Moreover, I have to thank Dr. Ulrich Güldener, Dr. Martin Münsterkötter, Dr. Corinna Montrone, Dr. Irmtraud Dunger, and Dr. Andreas Ruepp for sharing their biological expertise with me. Christian Chors and Dr. Matthias Klaften from the Institute for Experimental Genetics I thank for the interest in my work and the constructive feedbacks. Additionally, I want to thank Cornelia Canady, Elisabeth Noheimer, Petra Fuhrmann, Gabi Kastenmueller, Wanseon Lee, Yu Wang and all other institute members for the nice working atmosphere.

Last but not least, I would like to thank my family in Bulgaria for their concerns and continuous encouragements, Annett Würfel, Markus and Julius Döbereiner for cheering me up in the writing process, Tommaso Nuccio for helping me with the thesis revision, and a very special thank goes out to Kuo-Sek Lee for his trust in me.

Karamfilka K. Nenova

Abstract

Key premises for successful life-science research are the access, combination, and interpretation of already acquired knowledge. Within the last two decades, considerable data regarding various biological research aspects has been generated and collected in a huge number of diverse life-science information resources. Although most of the resources provide public access on the Web to share the already gained knowledge, a vast knowledge gap has emerged between the generated volume of information and discovered novel knowledge in life-science. Not only because related biological information is commonly spread over several distributed resources, but also because essential problems exist regarding context dependency and differentiation of biological concepts and entities. Accordingly to bridge this crucial gap, biological information has to be integrated and provided not only in a homogenous way, but also in the right context for the more effective exploration and interpretation, which represents still a demanding knowledge management task.

The objective of this thesis was the development of an integrative approach also applicable for the life-science information space that allows a more effective knowledge discovery. The generated solution follows a new paradigm for subject-centric knowledge representation, which reflects the human way of associative thinking in terms of subjects and associations between them. The novel integrative approach is realized by applying both state-of-the-art technologies for dynamic information request and retrieval and also the semantic technology *Topic Maps*. The designed approach was implemented within the software framework *GeKnowME* (Generic Knowledge Modeling Environment), which supports scientists with powerful tools for exploration and navigation through correlated biological entities to accelerate the discovery process in a specific knowledge domain. The framework is generic enough to be applicable for a broad range of use cases. To illustrate the potential of the GeKnowME system, a sample use case called *“Human Genetic Diseases”* is introduced by integrating distributed resources containing relevant information. The emerged coherent information space is explored for novel insights.

Zusammenfassung

Die Forschung in den unterschiedlichen Biowissenschaften führte in den letzten zwei Jahrzehnten zur Erhebung großer Datenmengen, die in einer Vielzahl von biologischen Informationsressourcen gesammelt und gepflegt werden. Obwohl die meisten Ressourcen öffentlich zugänglich sind und somit den Zugriff auf bereits erworbene Erkenntnisse ermöglichen, nimmt die Kluft zwischen diesen und dem daraus neu gewonnenen Wissen in den Biowissenschaften stetig zu. Ursache hierfür ist einerseits, dass zusammenhängende biologische Entitäten häufig über mehrere Ressourcen verteilt sind, und andererseits, das Fehlen einer einheitlichen Repräsentation kontextabhängiger biologischer Konzepte und Entitäten. Neben einer homogenen Integration der biologischen Information ist die Einordnung dieser in den relevanten Kontext erforderlich um eine effektive Exploration und Interpretation zu ermöglichen. Gerade in den Biowissenschaften stellt sich dies als eine besondere Herausforderung für das Wissensmanagement dar.

Ziel dieser Arbeit war die Entwicklung und Umsetzung eines Konzepts für ein Verfahren, welches für die Integration biologischer Wissensdomänen anwendbar ist und auf diese Weise eine effektivere Erkenntnisgewinnung ermöglicht. Das entwickelte Konzept basiert auf einem neuen *subject-centric* Ansatz zur Wissensrepräsentation, welcher das menschliche assoziative Denken im Bezug auf Entitäten und deren Verbindungen abbildet. Sowohl aktuelle Technologien zur dynamischen Informationsgewinnung als auch die semantische Technologie *Topic Maps* wurden eingesetzt um diesen innovativen Integrationsansatz zu realisieren. Das Software-Framework *GeKnowME* (Generic Knowledge Modeling Environment), welches Wissenschaftlern leistungsfähige Werkzeuge für die Erforschung und Navigation durch zusammenhängende biologische Entitäten in spezifischen Wissensdomänen bereithält, stellt die Implementierung dieses Ansatzes dar. Das generische System eignet sich für ein breites Spektrum an Anwendungsfällen. Die Leistungsfähigkeit von GeKnowME wird exemplarisch am Anwendungsfall „*Genetische Erkrankungen des Menschen*“ aufgezeigt. Hierzu wurden erforderliche verteilte Ressourcen integriert und das daraus entstandene Informationsnetzwerk umfassend analysiert. Die Resultate erlauben neue Einblicke basierend auf bereits bekannter Information.

Contents

1 Introduction

> *"Where is the wisdom*
> *we have lost in knowledge?*
> *Where is the knowledge*
> *we have lost in information?"*
>
> ***Thomas Stearns Eliot (1888 - 1965)***

Over the past two decades, the research in the area of life-science, for instance in the fields of molecular biology, biochemistry, or genetics, has lead to deep insights in the mechanisms of life. Due to advanced experimental and computational methods such as high-throughput genomic sequencing, mass spectrometry, or prediction of protein structures, huge amount of biological research data has been generated and collected in thousands of databases focusing on particular research aspects. The popularity of the World Wide Web (WWW) and the adoption of the WWW-related technologies in the biomedical domain have encouraged scientific communities to provide public access to the information available in the curated databases through the internet and consequently to share their already gained knowledge. Therefore, not just the number of available information resources on the Web has increased tremendously, but also their relevance to a successful biological research[1].

Although significant biomedical insights can be drawn by exploring the current life-science information space, the process of knowledge discovery represents a tedious task for scientists. Since life-science continues its growth in complexity and scope, comprehensive research requires the assembly of knowledge from various sub-disciplines and thus the access to numerous distributed and autonomous information resources containing highly heterogeneous information. To support scientists in their

[1] The term *information resource* refers throughout the thesis to any kind of structured and organized data collection such as database or set of delimiter-separated files. Additionally, the term *information space* regards as a set of particular information resources.

research endeavors, the life-science information space has to be represented consistently for the more efficient information exploration and interpretation. Therefore, the integration of biological information has been long recognized as an essential part in the life-science knowledge management process. However, current integrative methods still struggle to cope with the diverse technical and conceptional difficulties regarding the interconnection of biological entities available in the various distributed information resources.

In this thesis, a novel integration approach is presented that addresses the challenges needed to be overcome to provide a coherent information space for more effective knowledge discovery. The approach adopts not only established techniques from state-of-the-art integration technologies for dynamic information request and retrieval, but also follows a new paradigm for knowledge representation. It reflects the human way of associative thinking, in terms of subjects and associations between them and it is realizable by applying semantic technologies. The designed integrative approach is embedded in a generic software framework called GeKnowME (Generic Knowledge Modeling Environment), which allows researchers to find relevant biological entities and to investigate how they are interrelated. Hence, the developed system improves the life-science research by accelerating the knowledge discovery process.

To assist the understanding for the designed subject-centric semantic integration approach, several important topics are introduced in *section 2*. The first part of the section provides main insights in the field of knowledge management and particularly its significance for the life-science domain. An overview of key biological information resources and the reasons for their emergence are given in the second part. The challenges and evolved approaches to integrate these information resources are discussed in the third subsection. The last background part describes the evolution and realization of important knowledge representation methodologies.

Section 3 provides a detailed description of the developed GeKnowME framework implementing the subject-centric semantic integration approach. The system is represented precisely from five different perspectives to show the entire functionality. In the subsequent *section 4*, the utilization process of the framework is introduced and sample applications in the area of *“Human Genetic Diseases”* with the drawn results are illustrated to demonstrate the developed integrative approach. Its strengths and limitations are discussed in *section 5* with the directions of possible future extensions of the GeKnowME system and further potential applications. The last *section 6* contains a short summary of the represented work.

2 Knowledge Management in Life-Science

> *“Science is organized knowledge.”*
>
> ***Herbert Spencer (1820 - 1903)***

As an interdisciplinary field *Bioinformatics* supports a broad spectrum of research areas like analysis of biological sequence data and genome content, structural and functional prediction of macromolecules, study of comparative genomics, and many others. One of the key tasks of bioinformatics is as well the development of comprehensive computational tools and methods to organize and manage biological data. Over the past two decades, not only the number of biological databases has grown tremendously, but also their essentiality, since they are used daily by life-scientists around the world. However, with the ambitious goals of the field *Systems Biology* further requirements are demanded from the biological resources. Even a small vertical slice of cell biology crosses many disciplines of knowledge, therefore information of distributed and heterogeneous data resources have to be combined.

Since knowledge is a prime prerequisite for successful research in life-science, powerful techniques for its modeling, organization, and management are a necessity to cope with the flood of biological information. The concepts *data*, *information*, *knowledge*, and *wisdom* are in particular defined in the following section to provide adequate understanding to the corresponding computer science technologies and their reference to life-science research.

2.1 Knowledge Hierarchy and Life Complexity

Biology is a knowledge-based discipline and success in life-science research is based on identification, creation, representation, and distribution of knowledge. The discipline *Knowledge Management* aims to address such challenges and can be broadly defined as the tools, techniques, and processes for the most effective and efficient management of knowledge available in an organization or a community in order to maximize performance (Davies, et al., 2003). Although there is no universal definition of what constitutes knowledge, in the context of knowledge management it is generally agreed that there is a continuum of *data*, *information*, and *knowledge* (Waltz, 2003). This continuum is also known as *knowledge hierarchy*, *taxonomy of knowledge*, or *knowledge pyramid* and represented in *Figure 2-1*.

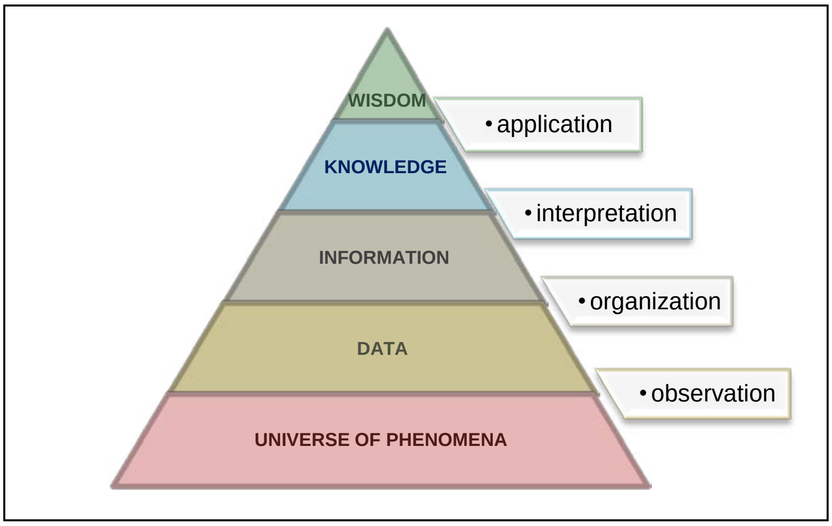


Figure 2-1: Knowledge pyramid representing the continuum of data, information, and knowledge, which is based on the observations made in the universe of phenomena and completed by wisdom as applied knowledge.

In general, this cognitive hierarchy consists of the three levels of abstraction data, information, and knowledge, which can be extended by a level above knowledge: *wisdom*. Furthermore in the context of natural sciences, the foundational level of the knowledge taxonomy is the *universe of phenomena*.

- *Universe of Phenomena* represents any occurrence in nature that is observable. For the purpose of this thesis, the universe of phenomena is considered to any fact or event that can be measured in relation to a biological system.

- *Data* are numerical quantities or other attributes like images, videos, etc. collected not only by experimental methods and observation of the universe of phenomena, e.g. microarray data, but also by applying advanced computational analysis methods, e.g. on genome or proteome data. Data on its own has no meaning.

- *Information* refers to organized sets of data, which are created by analyzing relationships and connections between data. The organizational process may include various steps like alignment, transformation, sorting, indexing, and linking data in order to place the data elements in a relational context for subsequent querying and analyzing. Biological data is commonly structured, stored, and managed in either flat-file systems or relational database management systems.

- *Knowledge* emerges from information after putting it into a context and interpreting it. Once information is analyzed, understood and explained, it becomes knowledge. The process of interpretation may contain comprehension of static and dynamic relationships between sets of information and generation of models to explain those relationships. In biology, the combination of distinct pieces of information like known enzyme inhibitors and protein-protein interactions may lead to the understanding of complex mechanism like the machinery of a particular metabolic pathway.

- *Wisdom*, in the context of the knowledge pyramid and as its last abstraction level, regards as a uniquely human cognitive capability – the ability to correctly apply knowledge based on experience and even intuitive understanding to perform an action effectively to achieve a desired objective. For instance, the diagnosing of a particular disease and the administering of an appropriate treatment by a physician can be considered as wisdom in this context. Since such actions require the appliance of composite knowledge, they are too complex for execution by computer systems.

In addition, the representation of the knowledge continuum as a pyramid gives another significant aspect to this subject matter: the universe of phenomena is almost infinite, for which large amount of data are collected and distilled to a smaller quantity of information. In turn, this information is aggregated to create yet more distilled knowledge, which may possibly be applied to achieve an objective.

Another perspective of the knowledge continuum is its representation as a linear chain in relation to context, understanding and time, as shown in *Figure 2-2*. The main idea of this depiction is that knowledge can be gained through both context and previous understanding. On the one hand, when the context is familiar, one can identify and infer various relationships based on previous experiences. The broader the context is, the greater the variety of experiences is that one can rely on. On the other hand, the better one understands the subject matter, the more one is able to weave past experiences into new knowledge by absorbing, doing, interacting, and reflecting (Shedroff, 2001).

Furthermore, one can consider the knowledge continuum in relation to time. Since data and information are based on gathering facts and adding relational context to them, they concern the past. Contrarily, knowledge deals with the present in terms of deducing novel, not trivial findings such as rules, lows, and principles explaining the behavior of complex systems. Additionally, the collected information can be used to predict some new features in the universe of phenomena, which have never been observed. For instance, the probability of a protein sequence can be estimated by considering similar multiple alignments. Furthermore, the gained knowledge gives the opportunity to perform further actions effectively in the future to achieve new desired objectives.

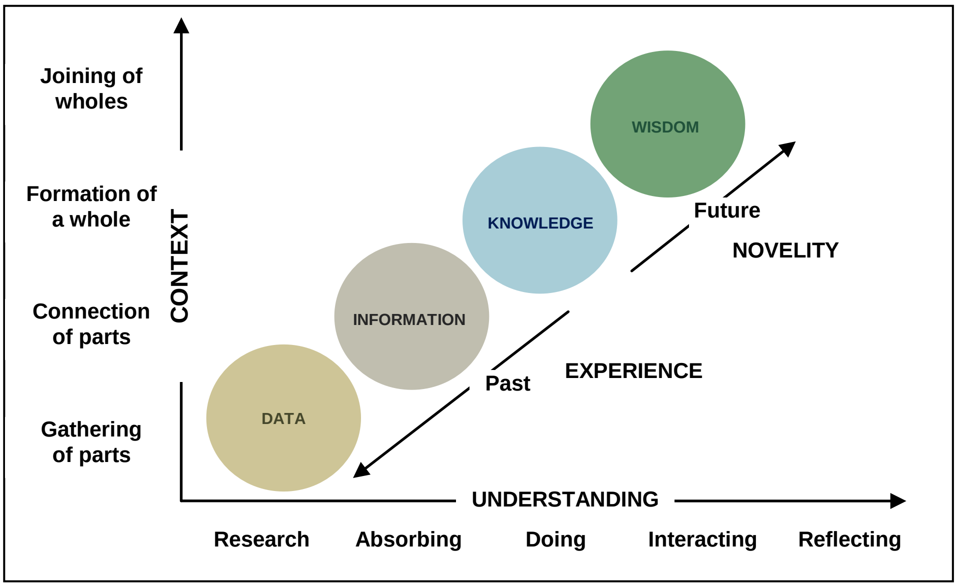


Figure 2-2: Continuum of knowledge in relation to context, understanding, and time (Shedroff, 2001).

In the context of life-science, the natural world may refer to different kinds of biological systems as stated previously in the definition of the universe of phenomena. To understand the challenges for the modeling, organization, and management of biological knowledge, one has to understand the broader context in which the single entities of the knowledge continuum exist. The most interesting circumstance about knowledge organization in the context of nature is that all living things are already organized. There are various levels of biological system organization as illustrated in *Figure 2-3*. Each level of organization is more complex than the level preceding it and has properties beyond those of the former level. Each new level of biological organization has emergent properties that are due to interactions between the parts making up the whole, which increases the complexity (Mader, 2004). Fortunately, all properties, even the emerged ones, are controlled by the laws of physics and chemistry.

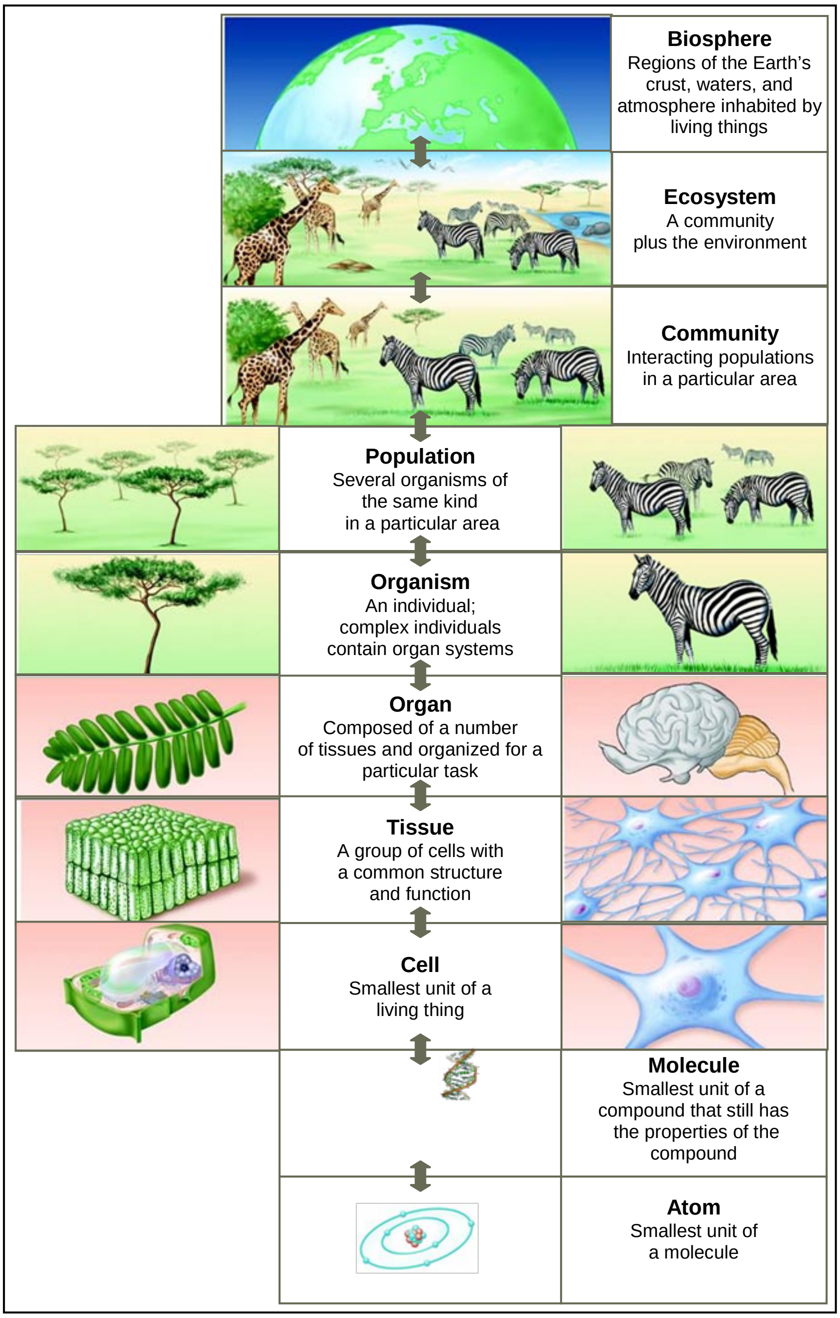

Biosphere
Regions of the Earth's crust, waters, and atmosphere inhabited by living things
Ecosystem
A community plus the environment
Community
Interacting populations in a particular area
Population
Several organisms of the same kind in a particular area
Organism
An individual; complex individuals contain organ systems
Organ
Composed of a number of tissues and organized for a particular task
Tissue
A group of cells with a common structure and function
Cell
Smallest unit of a living thing
Molecule
Smallest unit of a compound that still has the properties of the compound
Atom
Smallest unit of a molecule

Figure 2-3: Levels of biological system organization (Mader, 2004).

The complex organization of living things begins with the *cell*, the smallest, most basic unit of life. It is composed of different compounds called *organelles*, which are built out of nonliving chemicals arranged in *molecules*. The *DNA* macromolecule, as part of the cell nucleus, contains the genetic instructions for the construction of other components of cells, such as *RNA molecules* and *proteins*, which serve as building blocks for biological functions. Proteins physically aggregate to create more complex units of biological function known as *protein complexes*, which interact with other proteins or complexes in *pathways* or networks to carry out higher level biological processes such as the neuronal signaling pathway. In multi-cellular organisms, the pathways in turn may be assembled into more complex systems of multiple interacting pathways, which usually control the cell’s functions. The cells (such as neurons) in turn interact with one another and form *tissues*, having particular structure and function, to build higher order structures termed *organs*. Organs work together in systems, for example, the brain works with the spinal cord and a network of nerves to form the nervous system. Organ systems are joined within an *organism*. There are levels of biological organization that extend beyond the individual organism. All members of one species in a particular area belong to a *population*. Several populations make up a *community*, which interacts with the physical environment and forms an *ecosystem* in turn resulting in the Earth’s *biosphere*.

This hierarchical progression points out the fact that in order to be able to understand the complexity of life and in particular the mechanisms of diseases, one has to consider a broad diversity of biological entities, environmental factors, and the interrelations between them at the different organizational levels. However, since the discovery of the molecular structure of the DNA by Watson and Crick in 1953 and its crucial role in the mechanisms of replication, one can think of the *genome* as the machine code for creation and operation of biological organisms. The emanating study of the genomes of many model and non-model organisms has led to the collection of considerable data regarding diverse research aspects. The following section describes in brief the emerged life-science information resources and their importance for the discovery of novel biological insights.

2.2 Life-Science Information Resources

> *“Knowledge is of two kinds.*
> *We know a subject ourselves,*
> *or we know where we can find*
> *information on it.”*
>
> ***Samuel Johnson (1709 - 1784)***

Over the past two decades, the amount of data in the area of life-science has grown exponentially, as the annual report *“The Molecular Biology Database Collection”* of the journal Nucleic Acid Research (NAR) shows (Galperin, 2008). The current issue includes almost 1.100 databases, where these large biological data sets are collected and organized. These databases represent actually a small portion of all biological databases in existence today. The flood of biological information in the form of diverse databases available nowadays on the Web can be traced directly to the coordinated international investment of large amounts of funding to sequence the human genome and understand the basis of the human health and morbidity. In 1990, the US National Institute of Health (NIH) in cooperation with international partners established the Human Genome Project (HGP), which grand vision, as stated in the *First Five-Years Plan* (NHGRI, 1990), was:

> *“The information generated by the human genome project is expected to be the source book for biomedical science in the 21st century and will be of immense benefit to the field of medicine. It will help us to understand and eventually treat many of the more than 4000 genetic diseases that afflict mankind, as well as the many multifactorial diseases in which genetic predisposition plays an important role.”*
>
> ***HGP First Five-Years Plan (1990)***

Elaborate investigations into the fields of molecular biology, biochemistry, and genetics of different model and non-model organisms have become indispensible in these efforts (Collins, et al., 2001). The success of the HGP together with the popularity of the WWW has made the resulting data of these researches available to the scientific community through the internet. The emerged databases differ from each other quite a lot by focusing on specific aspects of life-science such as nucleotide or protein sequences (e.g. GenBank (Benson, et al., 2008), and SWISS-Prot (Boeckmann, et al., 2003)), molecular structures (e.g. PDB (Berman, et al., 2007)), functional annotation (e.g. FunCat (Ruepp, et al., 2004), and GO (Gene Ontology Consortium, 2006)), metabolic pathways (e.g. HPRD (Mishra, et al., 2006), and BIND (Bader, et al., 2003)), specific organisms (e.g. MGD (Bult, et al., 2008)), or diseases (e.g. OMIM (Hamosh, et al., 2005)). In addition, the bioinformatics community has been developing and applying numerous tools and methods on these prime data to generate new information about further biological features. Protein-protein-interaction networks, gene predictions, or pathways deducted out of gene expression arrays can be pointed out as few examples for such secondary information available in hundred of additional databases. Furthermore, the amount of biomedical knowledge recorded in texts has been also growing tremendously over the last years and the speed of this development is still accelerating (Jensen, et al., 2006). The *PubMed* database, developed and maintained by the NIH department National Center for Biotechnology Information (NCBI), provides a searchable compendium of over 17 million citations from diverse life-science journals for biomedical articles back to the 1950s (Wheeler, et al., 2008).

Not only has the number of available information resources changed over the past decades, but also their relevance in the daily workflow of life-scientists. At the beginning, biological data collections were set up to ensure long term availability and accessibility of experimental results (see *Figure 2-4.A*). But the new visions and goals of modern life-science, especially those in the field of systems biology, are to move ahead from the single and isolated studies of interests to the construction of biological models describing the complexity and dynamics of entire biological systems in the different organizational levels. Although, each database can answer questions in its domain, it cannot help with questions that span domain boundaries, since it contains a different subset of biological knowledge. The recent research approaches to identify biological knowledge represent more a cycle process as shown in *Figure 2-4.B*, where the information in the scientific databases has to be systematically exploited to generate hypotheses for in-silico discovery, which, after experimental verification, can be used to populate other databases (Philippi, et al., 2006). For instance, systems biologists have to deal with many heterogeneous data resources to model complex biological system like the TLR signaling pathway (Oda, et al., 2006).

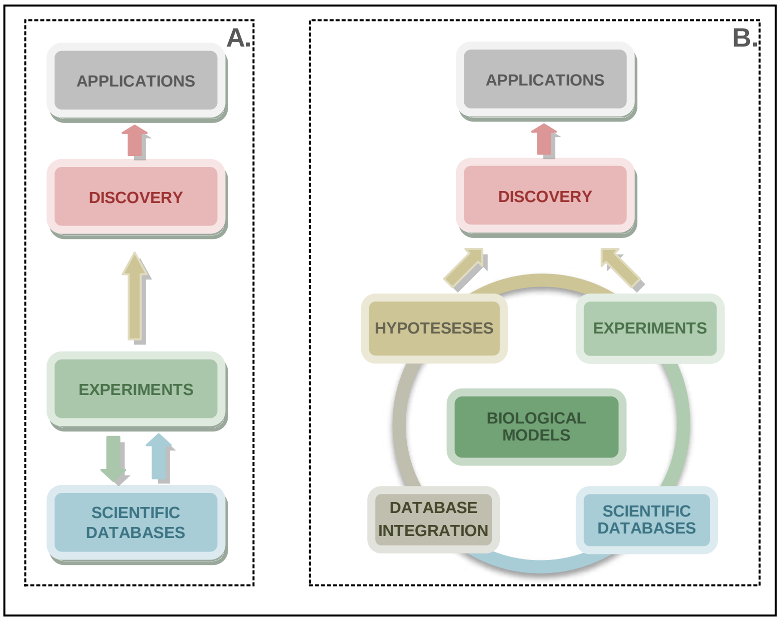


Figure 2-4: Traditional and recent role of life-science databases in the scientific process in biology (Philippi, et al., 2006).

At a conceptual level, all biological information is consistent and interconnected, given the fact that all players in any biological system are well organized. However, life-scientists encounter several problems in the process of dealing with biological information, since it increases not only in volume, but also in both complexity and diversity. More difficulties can be observed in the process of trying to find the right biological entities and how they are interconnected to create new hypotheses, or to identify unknown entities. On the one hand, there is *lack of information*, because the researcher might know that some information objects and relations already exist, but they cannot be found through the complexity of hundreds of independent, overlapping, and heterogeneous data resources. On the other hand, there is an *overload of information*, since too many information objects and relations could be found with no or minor relevance. For that reason, one can say that in the past decade a vast *knowledge gap* has emerged between the generated volume of information and discovered knowledge in life-science.

To close this crucial gap biological information has to be integrated and provided not only in a homogenous way, but also in the right context for the more effective exploitation and interpretation in the various biological domains. The integration of the diverse information resources has been long recognized as a very essential knowledge management process and become one of the most important fields in bioinformatics (Philippi, et al., 2006), (Stein, 2003). In the following sections, the integration challenges and developed approaches to solve them are discussed by poining out their advantages and disadvantages.

2.3 Integration of Life-Science Information Resources

> *"Integrity without knowledge is weak and useless. Knowledge without integrity is dangerous and dreadful."*
>
> ***Samuel Johnson (1709 - 1784)***

Apparently, it would be much easier to bridge the knowledge gap in life-science, if all available data would have been collected just within a single database. But this approach would have caused a loss of information by imposing restrictions, since the diverse databases reflect the expertise and interests of the specific research communities. The optimal approach would be to keep the scientific and political independent information resources, but provide methods to access and combine the information contained in such a way that cross-database exploration and queries are still possible (Stein, 2003). Though, the process of information integration is not just the process of combining information residing at different resources and providing the user with a unified view over it. The process of information integration is more the methodology how to consolidate the information *correctly*, *completely*, and *efficiently* from *distributed*, *autonomous*, and *heterogeneous* resources and represent it in a *consistent* and *structured* way, so that more *effective* usage of the information is assured (Leser, et al., 2006). Another related field named *Enterprise Application Integration* (EAI) deals with similar tasks but it lays emphasis more on speed and simplicity. An EIA system provides methods mainly to exchange messages between distributed information systems; on the contrary an *Information Integration System* (IIS) combines information and represents it as a whole as shown in *Figure 2-5*. Since numerous integration challenges regarding biological data have to be considered, the development of an elaborate IIS in the field of life-science is not trivial.

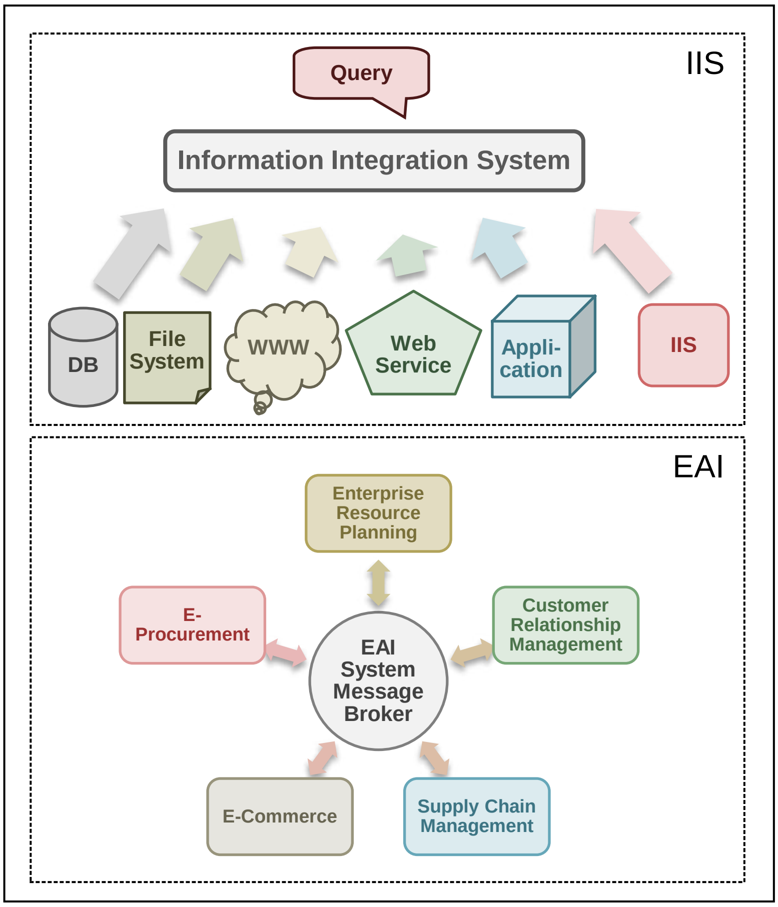

IIS
Query
Information Integration System
DB
File System
WWW
Web Service
Appli-cation
IIS
EAI
Enterprise Resource Planning
E-Procurement
EAI System Message Broker
Customer Relationship Management
E-Commerce
Supply Chain Management

Figure 2-5: Information Integration approach versus Enterprise Application Integration (Leser, et al., 2006).

2.3.1 Challenges

The obstacles that have caused the expansion of the knowledge gap in life-science can be grouped in two main categories – technical and conceptual. The *technical* difficulties consider more data exchange formats and accession techniques; whereas the *conceptual* ones regard its content and meaning.

Technical Challenges

One of the major integration problems to overcome is the diversity of the provided data access techniques. Some databases provide access to the data by using public programming interfaces of the different *database management systems* (DBMS). For instance, the DBMS *Oracle*, *PostgreSQL*, and *mySQL* support mature standard interfaces such as ODBC (Open Database Connectivity) and JDBC (Java Database Connectivity) to enable remote access and querying mechanisms. Other database providers offer access to the data via *Web Services* (WS), which are based on accepted internet standards. By supporting web services, not just access to underlying data is allowed, for example gene sequence entries, but also additional data transformation or pre-processing is possible like calculations of similarity hit records. This established technology for the web wide distributed systems has become more and more popular in the last few years in the bioinformatics community as indicated by the growing number of institutes providing web services. A prominent example is the NCBI web service, which enables developers to access the data outside of the regular web query interface (NCBI, 2008).

Despite the awareness of the necessity for automated access mechanisms, many communities provide their data collections just as large *flat-files* for download, which still somehow allows sub sequential large-scale data integration. However, regular downloads and updates are needed for such type of accession, because biological entities, their properties, and interrelations may change very often after taking awareness of new experiments or findings (e.g. changing names of genes or their relations to diseases). These permanent adjusts have an additional effect, because they lead not only to changes of the data content, but also to changes in the data structure. Therefore, biological database schemas may have to be expanded or ever totally rearranged sometimes. Nevertheless, if a database is not available even for download, its web pages represent the primary mode for access, which is actually the worst case, since a special data-extraction software is needed for each such resource that has to be updated for every change in the web interface.

Not only is the diversity of the provided data access techniques a key integration challenge, but also the variety of the formats in which the queried information is retrieved. Unfortunately, flat-files are still quite often used for data exchange. Given the fact, that there is no standardized format for flat-files and they are not generic in their structure, there are actually many exchange formats for the great part of the biological information resources. Suitable parsing steps are needed for their further processing and integration. For some of the more popular databases, software *parsers* have been developed by open source projects such as *BioJava* (Holland, et al., 2008). However, for the lager part of the databases there is no free parser available. Thus, the development of parsers is an indispensible part of almost all integrative projects in life-science.

With the adoption of the web service technology in the bioinformatics community, the importance of the representation of the life-science data in the form of XML (Extensible Markup Language) documents has become more popular. XML is a descriptive language, which primary purpose is to facilitate the sharing of structured data across different information systems, particularly via the internet. The self-describing XML files are much easier to parse, because generic XML parsers are available for almost every platform and programming language. For instance, by using a SAX-parser a JAVA programmer can break up straightforwardly an XML document into its comprising elements and then use them without knowing the structure of the document. Meanwhile, many of the popular databases provide their data in an XML format. As the power of XML has been recognized as a powerful tool within the process of information integration, several initiatives have been launched to work on the standardization of XML-based data exchange formats. One of the already established standards developed by the *Proteomics Standards Initiative* is the PSI-MI format, which main focus is on the representation and annotation of *Molecular Interaction* data (Orchard, et al., 2008). The representation of queried information in an XML form provides many advantages for the further processing, but nevertheless there is still a huge heterogeneity in the structure of the exchanged XML documents, which makes the universal interoperability difficult.

Conceptual Challenges

Although the mastering of the above described technical obstacles is quite challenging, the main integration problems are actually related to the conceptual structure of the life-science information resources. Since biology is a quite broad knowledge based science, problems with concept differentiation exist. On the one hand, there are problems with *synonymous* concepts. For example, if two distributed databases store data about gene sequences, then perhaps they are called “genes” in the first one and “coding sequences” in the second one. On the syntactical level of database structure, these two terms are totally different, but semantically both of them refer to the same concept “gene” (Gerstein, et al., 2007). On the other hand, the problem of *homonyms* is another type of biological concept clash. Sometimes, same terms (e.g. phenotype) can represent different concepts and therefore have different meanings (e.g. an observed characteristic of an organism or a disease). Biological ontologies can be used in such cases as semantic references, since they define commonly agreed definitions of real-world concepts and relationships between them. However, some of the current ontologies in the domain of life-science do not follow strict concept specifications and thus cannot help overcoming the homonym problems[2].

These conceptual obstacles can be also transferred to the biological entity level. Usually, there are many synonyms for the same underlying biological entity as a consequence of researchers independently naming entities for use in their own datasets or because of legacy common names arbitrarily given to biological entities. Some of such names are still commonly used and thus cause many synonymous obstacles; for example in the area of gene and protein identifiers. Additionally, there can be also lexical variants of the same underlying identifier (e.g. RefSeq gene identifiers gi|202472 vs. gi_202472 vs. gi:202472). Moreover, there is still the problem of

[2] General concepts regarding ontologies and a detailed overview of ontologies essential for the life-science domain is given in *section 2.4.2*.

homonyms. Many difficulties occur with the assignment and maintenance of the correct names and correspondingly meanings of biological objects across multiple databases. One of the most prominent examples is the ambiguity with gene identifier *Rad24* in *S.cerevisiae* and its mapping to other model organisms like *S.pombe* and *C.elegansas*, as pointed out by Stein (Stein, 2003). The necessity of public accepted subjected identifiers has been already recognized and initiatives like the *HUGO Gene Nomenclature Committee* (HGNC, 2007) focus on the agreement of unique and meaningful names to biological entities, in this case on genetic elements submitted from the HGP.

Another key problem during integration of information from distributed resources is the identification of relationship types between biological entities. As already stated biological entities are usually highly interconnected and novel knowledge can be easily gained as transitive closures from graphs representing the coherence of the biological entities. However, inferences risk to be incorrectly derived without the exact definition of the concrete association types and their meanings.

Most of these conceptual challenges arise from the fact that computers cannot always interpret the information in the right way. Semantic technologies, dealing with the meaning of terms and precisely introduced in *section 2.4.1*, are challenged to solve these problems by providing meaningful data descriptions called meta-data that define unambiguously what the underlying data is about and in which scope it is valid. These meta-data enable also computer applications and not just humans to understand the context in which a piece of information is placed. In the process of modeling knowledge about biological systems the determination of the context plays a crucial role, since it throws light on the meaning of the involved biological players. For example, information about the human brain can be modeled at the different biological organizational levels and each granular level represents a different view of the information and needs specific interpretation.

2.3.2 Integrative Approaches

Over the past decade, a broad range of approaches have been pursued to bridge the gap between the often unconnected islands of biological knowledge. Each of them tries to cope with one or more of the above described technical or conceptual challenges.

Hypertext Linking

The hypertext link “integration” of life-science information has been one of the most widespread approaches, because it follows the nature of the WWW. The main idea of this technique is the availability of hypertext links between related documents. Usually, scientists start their research from a single database portal and are forwarded by the links to further internal or external web pages, so they are supported in browsing and exploring the content of different information resources. Prerequisite for the successful linking is the regular maintenance of the URLs, since they change quite often. In general, web links provide just an explorative content browsing; they do not allow complex querying across several databases.

Full-Text Indexing

An extension on the link integration is embodied in the full-text indexing, which is actually based on similar technologies like the current search engines. Such systems locally mirror the content of preselected databases and generate full-text indices usually over certain fields in the replicated data. This approach slightly differs from the one used in established web search engines, because these systems recognize the existence of structured fields in the data collections and a field from one database can be explicitly related to a differently named field in another. Thus, full-text-indexing systems provide the possibility to address several databases with a single query, but there is still no real information integration beyond the shared full-text index and it is difficult to find the right biological entities, because the queries are still semantically weak. In fact, the integration and interpretation must be still done by the researcher. An interesting example for such system represents the search engine *Bioinformatic-Harvester*; its search index is based on a protein information collection and it provides cross-links over 28 popular bioinformatics resources (Liebel, et al., 2005).

Data Warehousing

One of the advanced, but technically demanding, integrative approaches is *data warehousing* (DW). The main concept is to bring all specified data into a single database with a generalized, global schema as illustrated in *Figure 2-6*. For the set up of such a warehouse several steps have to be performed. The first step, actually significant also for any other advanced information integration approach, is the identification of suitable data resources for the required application. The next step is the development of a unified data model, which can represent all the information that is available in the chosen information resources. Afterwards software programs have to be developed to carry out some preprocessing procedures – extraction of the necessary data, its transformation into more accessible formats, data cleansing and filtering, mapping to the generalized schema, and execution of the data imports into the warehouse. Usually these steps are called ETL (Extract, Transform, and Load) and the out-coming data is temporary loaded into a DW staging area. Once the data resources have been integrated, access to the resulting DW can be provided through different types of interfaces.

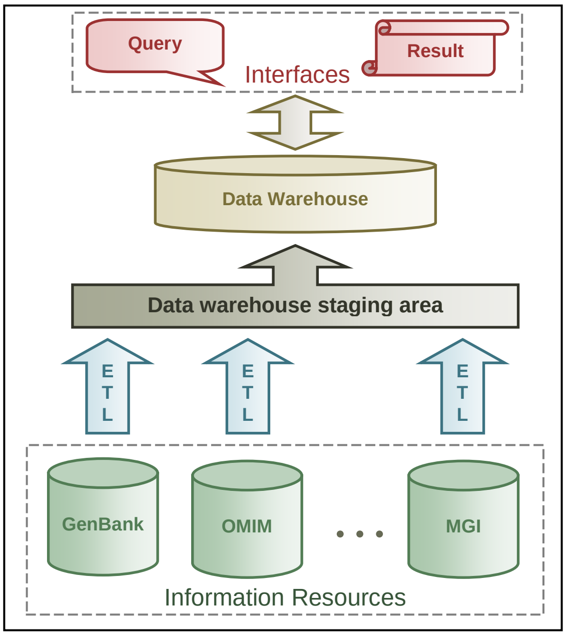


Figure 2-6: Data warehousing information integration. The information from the multiple databases is extracted, transformed, and loaded into the data warehouse and organized within a global schema.

This integrative approach allows researchers to ask complex questions, which the system can handle, as well as those that require integrative knowledge that the individual resources do not have. Additionally, one of the key requirements of biologists regarding the system performance is also satisfied, since DWs are considered to be reliable and provide fast access and excellent response time to user queries. However, this approach has some very crucial drawbacks in the context of biological information integration. The consolidation of all chosen data into a single large database is not the only issue, its maintenance and keeping the data up-to-date are other problematic points. Since not just the content of the biological data changes frequently, but also the data models by adding new concepts and new relationships among them, e.g. new field names and nomenclature. The updating of a warehouse represents a serious maintenance issue, since the results of the queries are only as relevant as the latest updates. Another important aspect regarding knowledge management is the fact that it is very difficult to design a global schema that captures all nuances of the diverse information resources. On the one hand, the precision of the individual resources can be lost if one assigns just the common elements to the general schema. On the other hand, the complexity of a global schema representing all details of the underlying resources can become very bulky (Louie, et al., 2007).

Considering these limitations, the DW approach may be best suited for integrative applications that focus on a specific and narrow area of research. One prominent example for the collapsing of a huge DW is the *Integrative Genome Database* (IGD) project, which ambitious attempts were to combine human sequencing data with the multiple genetic and physical maps from over a dozen primary databases (Stein, 2003). Nevertheless, there are also well-known examples, which apply this approach, but they reflect really small knowledge domains and the integration is as relevant as the updates of the data. For example the integrative information resource *PhenomicDB* focuses on the relationship phenotype-genotype in multiple organisms (Groth, et al., 2007).

Federated Database Management Systems

In contrast to the DW approach, in federated database management systems (FDBMS) the data remains at the source and is accessed via a computer network, thus the distributed databases stay autonomous. Similar to warehouses, in FDBMS a global schema has to be designed that specifies the integrative conceptualization over the remote databases and relies on schema mapping for the integration of the disparate resources. The schema mapping is based on rules that define in which way the entries of a certain resource have to be matched to the common data model. Therefore, a federated system is able to decompose a query into sub-queries and submit it to the relevant underlying databases and afterwards to compose the result sets of the sub-queries as illustrated in *Figure 2-7*. Since heterogeneous database management systems employ various query mechanisms, software programs called *wrappers* implement functionalities to translate the sub-queries into the appropriate query languages (Brayner, et al., 2006). Through this transparent integration, federated database systems provide a uniform front-end user interface.

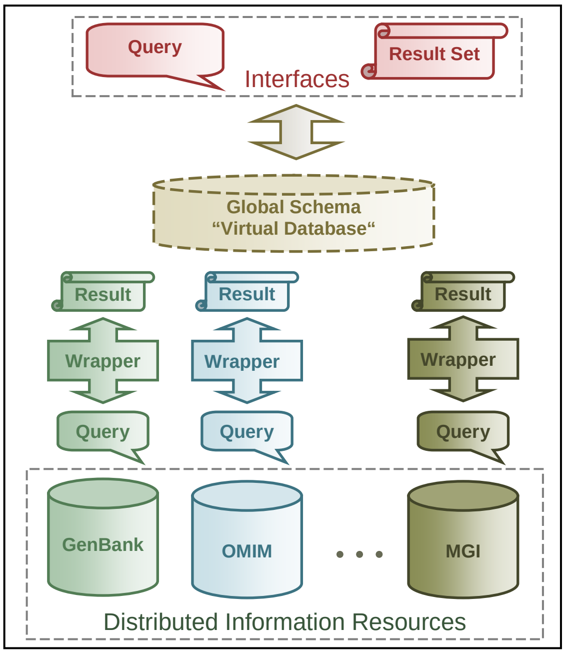


Figure 2-7: Federated data integration. The user poses queries to a “virtual database” implementing a global integrative schema. The source databases are still autonomous and interfaced with a wrapper code.

The most significant advantage of the federated approach is that the returned data entries are always up-to-date. Additionally, if the underlying data models of single databases change, just the corresponding wrappers and the schema mapping rules have to be updated. One does not have to re-import the whole data coming from the distributed resources. Unfortunately, there are several drawbacks of this paradigm. The response time for complex queries can take long time, because the performance depends on the query load capacities of all members of the federation. Another considerable issue is the data cleansing, since no data is stored locally; such procedures must be done on-the-fly. Comparable to the DW approach, FDBMS use a global schema thus they face the same difficulties to represent diverse data types and data granularity. Given this constraint, an improved approach has been introduced where instead of a global schema, mediated schemata are used. A mediated schema covers just the domain of interest, allowing the development of a comprehensive data model for a particular subset of data without considering all possible queries or domains of interests to all potential users. The advantage of the usage of mediated schemata is that a group of them can be created and depending on the exploration needs the current schema can be exchanged with another one (Louie, et al., 2007). However, this improved data modeling does not address all problems regarding the semantic challenges dealing with biological data. There are still integrative difficulties in the stage where the resulting sets have to be assembled together, since they are not semantically described.

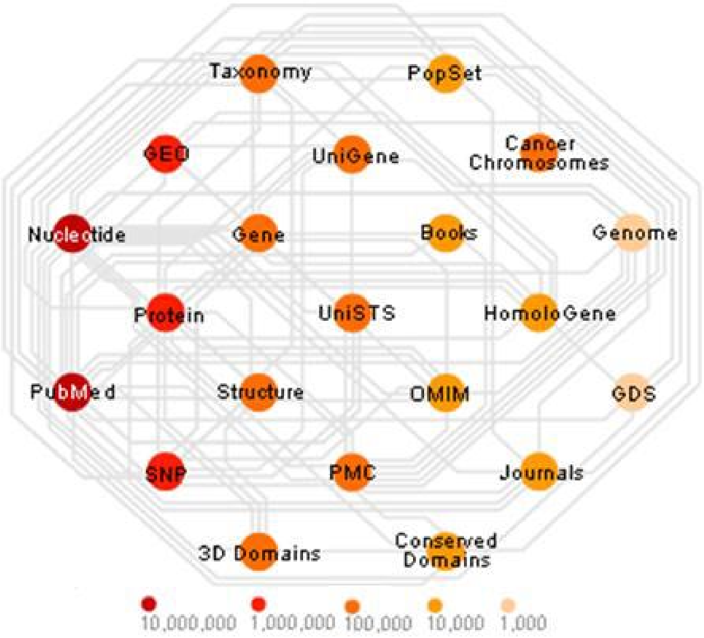


Figure 2-8: Rough overview of the *Entrez* databases and the connections between them. Each database is represented by a colored circle, where the color indicates the approximate number of records (NCBI, 2007).

In general, the federation approach provides advanced techniques to solve many of the technical and conceptual challenges related to information available in life-science. It is best suited for situations where up-to-date information is required, or where heterogeneous information resources are available within an institution. The *Entrez Global Query Cross-Database Search System* (NCBI, 2007), developed and maintained by NCBI, is the most well-known and successful example for FDBMS in the area of life-science. It covers all available databases within NCBI (see *Figure 2-8*) and provides a powerful cross-database search and retrieval mechanisms for an explorative research.

Peer Data Management Systems

Peer data management systems (PDMS) are a natural extension of the FDBMS and an evolution of the P2P (Peer To Peer) systems (Gribble, et al., 2001). They represent highly dynamic, completely decentralized infrastructures for large-scale information integration. They provide an approach how to cope with the key limitation of FDBMS using mediated schemas. The development of a mediated schema for small sets of data resources is easy, but similar to the development of a global schema, if the domain of interest that the schema covers increases, then design, scaling, and maintenance issues occur. The PDMS paradigm addresses this problem by the development of multiple specialized schemas. PDMS are built of multiple autonomous peers and each of them implements such schema and accepts queries against it (see *Figure 2-9*). Additionally, each peer offers a semantic mapping to either one or a set of other peers. The peers in a PDMS are inter-connected by these schema mappings forming a semantic network, which the PDMS can traverse to answer complex questions. Responses to queries submitted to one peer are composed from data residing at that peer and data reached by repeated query reformulation along the paths of mapping (Louie, et al., 2007).

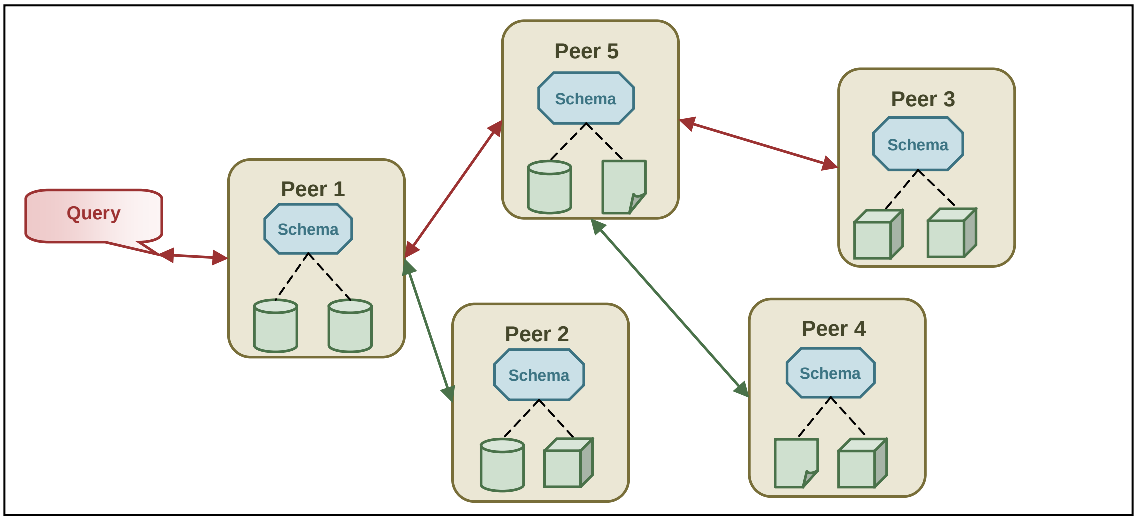


Figure 2-9: Peer data management system consists of multiple peers, each representing an integrating component of information resources with a particular schema. Each peer knows its neighbors by schema mappings represented as arrows. The purple arrows show the traversed path to answer the particular query, the green ones show other possible schema interconnections (Heese, et al., 2005).

The main advantage of a PDMS over a FDBMS is the usage of semantic mapping between single schemas representing different domains of interests. This approach is also quite flexible, since a new peer only needs to generate a semantic mapping to the schema of some similar peer and thus to be immediately part of the system, which can be quite helpful dealing with biological data. However, there are also some crucial drawbacks in this paradigm. The response time increases tremendously, if a large number of peers are involved in answering a query, because the result data is redundantly transported through the network of peers on different traverse paths. If the overall semantic schema is too modular, finding the relevant peers can become also problematic. Currently, in the field of life-science there is no well-known information integration project implementing the concepts of PDMS.

All above described approaches have something in common; they try to interconnect the existing information in a way that a broader context is available for the user to interpret the information correctly. Navigation mechanisms and/or complex queries aim to recognize and extract relationships between single entities in order to understand how the parts of a system are organized and work together. Novel knowledge can be easily acquired as transitive closures from networks representing such found coherences. Therefore, knowledge management depends tremendously on the process of joining autonomous parts. The technical difficulties how to structure the software architecture of the integration are still challenging but solvable with the information technologies evolved in the past two decades. However, a successful information integration solution needs paradigms how to define clearly abstract models representing the concepts within different knowledge domains. If there are any connecting points between them, the models can be combined to provide a broader context like in the PDMS approach. The development of unambiguous models representing the concepts and their relationships, which can be interpreted by computers, is a fundamental task. Not only the models, but also the contents have to be defined clearly to avoid incorrect association assignments. As already stated, semantic technologies provide approaches how to cope with such challenges. In the following section, the main ideas and principles of these technologies are presented to show why they are relevant in the processes of knowledge representation and knowledge modeling.

2.4 Knowledge Representation

> *“The real voyage of discovery consists*
> *not in seeking new landscapes*
> *but in having new eyes.”*
>
> ***Marcel Proust (1871 - 1922)***

In general, *knowledge representation* is the study of how knowledge about the real world can be represented and what kinds of reasoning can be done with that knowledge. It developed in the 1950s as a branch of artificial intelligence – the science of designing machines to perform tasks that would normally require human intelligence (Sowa, 1999). The main goal of the field is to encode human knowledge – in all its various forms – in a manner that the knowledge can be used also by computer systems to achieve intelligent behavior by facilitating inferences; for example by drawing conclusions (Croasdell, et al., 2006). Making hidden knowledge accessible to support the research discovery is another significant aspect in the field of knowledge representation. The key issues faced by designers of knowledge representation technologies are miscellaneous, e.g. nature of the knowledge, purposes of the representation whether it deals with a particular or general domain, expressiveness of knowledge representation models, mechanisms by which knowledge from disparate resources can be combined, reasoning methods, etc.. Hence, knowledge representation is a multidisciplinary subject that applies theories and techniques from other fields like *semantics*, *logic*, or *ontology design* (Zarri, 2006). With the vast expansion of the WWW and the way how information is organized and accessed via the internet, new challenges evolved in the field of knowledge representation and it still represents an active area of research. In the following subsections the most relevant techniques and approaches regarding knowledge in the domain of life-science are discussed.

2.4.1 Semantics – The Meaning of Meaning

In life-science data is a major corporate resource. Descriptions of data are essential for their proper understanding and used by researchers inside or outside a certain community. Such descriptions are called *metadata* and they include the meaning, or semantics, of the data. Metadata on its own is data, too. It enriches the available data with machine processible semantics (Fortier, et al., 2006). Therefore, semantics is crucial for information compatibility and interoperability. Generally, semantics is the study of meaning in communication. The issue whether two terms or statements are the same or different is fundamental to semantics. It is often contrasted with syntax. The syntax of a language defines what statements can be expressed in the language; it is about the grammar of the language. Contrarily, semantics is concerned with extracting a single abstract *concept* from the many ways that the concept can be represented, such as words, abbreviations, and pictures known also as syntactic variations (Baclawski, et al., 2006). In order to be able to describe correctly how to define a meaning or the semantic, some terms need to be introduced:

- *Object*: Something imaginable or noticeable, also known as referent, for instance the BRCA1 gene.
- *Property*: Attribute used to describe or distinguish an object.
- *Characteristic*: Abstraction of a property of a set of objects (e.g., “BRCA1 is located at 17q21.” means “17q21” is the property of the gene BRCA1 associated with the characteristic “chromosome location”).
- *Concept*: Mental constructs, units of thought, or units of knowledge created by a unique combination of characteristics (e.g. “Gene”).
- *Definition*: Expression of a concept through natural language.
- *Designation*: Representation of a concept by a sign, which denotes it (e.g. term or symbol).
- *Concept system*: Set of concepts structured according to the relations among them. (Gillman, 2006).

The ancient Greek philosophers studied the formation of concepts in language and discovered a useful relationship between designation, concept, object, and definition (Wedberg, 1982), which is illustrated in *Figure 2-10*. Concepts, terms (more generally designations), definitions, and referents (objects) are related but separate constructs. Each of them plays a role in our understanding.

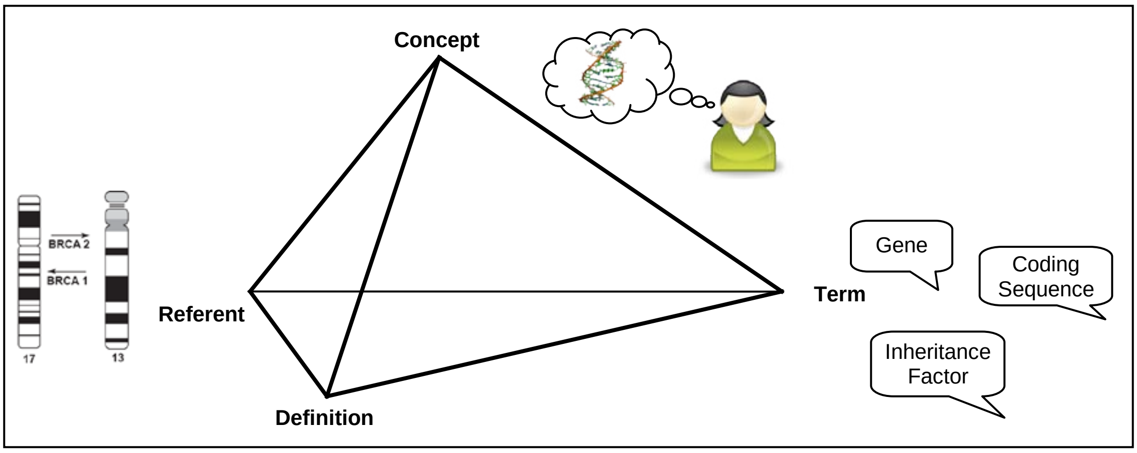


Figure 2-10: Relationships and differences between the constructs concepts, terms (more generally a designation), definitions, and referents (objects) (Gillman, 2006).

An important observation is that concept systems or mental models are human constructions. We as humans have the semantics of the world or part of it in our minds (Daconta, 2003). For instance, when we view a textual document, we can interpret the symbols on the page (designations) with respect to what they mean in our mental models and to which objects they refer, i.e. we supply the semantics. In addition, meaning is always relative to a context. The context influences the way we understand the designations and it has to be considered during the interpretation, which means it is part of the semantics. There is no knowledge in documents or collections of data without someone or something interpreting their semantics. Semantic interpretation makes knowledge out of otherwise meaningless designations (e.g. symbols on a page).

However, in order to make computers to assist researchers in the utilization of the knowledge embedded in the diverse information resources, the semantic interpretation process needs to be at least partially automated. A portion of the mental models about specific domains needs to be described and represented in a computer-usable way. Ontologies provide such capabilities and are one of the most widespread techniques in the field of knowledge representation and rather established in life-science.

2.4.2 Ontologies

Originally, the term ontology was used in philosophy and referred as a branch of thoughts concerned with the nature of existence and what kinds of entities comprise it. In the era of artificial intelligence and knowledge representation, the term ontology acquired a new meaning and refers currently as an explicit representation of a shared understanding in some domain of interest by describing the important concepts and relationships and by formalizing the common terminology (Buchholz, 2006). They have the ability to express knowledge in a machine-readable form. Therefore, one of the fundamental purposes of ontologies is the representation of knowledge and their usage in computer systems. Well defined ontologies provide the basis for interoperability between systems and can be used also as a query model for information resources. Overall, ontologies lead to a better understanding of a field and to more effective and efficient handling of the information in that field.

Characteristics

Ontologies can represent different kind of information. Beside *concepts*, they can describe also different types of *relations*. For example, one type is the *specialization* relationship *is-a* (e.g. “increased bone mass” is an “abnormal bone structure” phenotype), another type of relation is the *part-of* (e.g. “limbs” are part-of “skeleton”). Sets of *characteristics* describing components can be also part of an ontology. Additionally, ontologies can define *axioms* that represent facts that are always true in the topic area. These can be domain, cardinality, or disjointness restrictions and they can be used in logical operations (Lambrix, et al., 2007).

Depending on what kind of components and the information they contain, ontologies can be classified in several types:

- *Controlled vocabulary* is a simple type of ontology and represents actually lists of concepts. Each concept has a definition and is described by predefined, authorized designations or terms that have been preselected by the designer of the controlled vocabulary, as contrast to natural language where there is no restriction on the vocabulary that can be used.

- *Taxonomy* is an extension of a controlled vocabulary. In taxonomies the concepts are organized in an *is-a* hierarchy. One of the most common ways that people cope with complexity is to classify concepts into categories and then organize them hierarchically. This is a powerful technique and taxonomies make us of it.

- *Thesaurus* is slightly more complex type of ontology, since the concepts are organized as graphs. The edges of the graph represent a predetermined set of relations, such as *synonym*, *narrower term*, *broader term*, *similar term*, or *anonym*. Some comprehensive thesauri allow definition of a concept hierarchy, sets of characteristics and relations, and also a limited form of axioms.

- *Knowledge base* can contain all types of components even instances of concepts representing the actual objects. It is based on logic and its main purpose is an automated deductive reasoning, which can be used also as checking the consistency of the ontology.

Ontologies and their components can be represented in a spectrum of formalization languages ranging from very informal to strictly formal ones. In general, the more formal the used representation language is, the less ambiguity there is in the ontology. Formal ontologies are more useful, because informal and implicit assumptions often result in misunderstandings (Baclawski, et al., 2006). However, building formal ontologies is not an easy task. Currently, there is no established way how to define ontologies and no universal ontology language. A broad diversity of approaches exists and some of them are discussed later.

Ontologies in Life-Science

In practice, ontology coverage of biological content emerges primarily from pioneering efforts of biologists to provide controlled vocabularies of scientific terminology to assist the annotation of experimental data. Usually, the designers of these ontologies are domain experts and not experts in knowledge representation. They concentrate on the gathering of concepts and the agreement upon definitions. However, many of the ontologies available in the life-science domain have reached a high level of maturity and stability regarding the knowledge representation process (Chute, 2005). The diversity of biological ontologies is very high. They differ in the type of biological knowledge they describe, their intended use, the level of abstraction, and the knowledge representation language. Some of the most established ones are pointed out, because they are relevant for the further understanding required in the thesis.

The *Unified Medical Language System* (UMLS), supported by the US National Library of Medicine (NLM) (NLM, 2008), is a collection of many controlled vocabularies in the biomedical sciences for facilitating software to process and manage biomedical documents. The UMLS offers three major resources:

- *Metathesaurus* forms the base of the UMLS and collects the concepts and terms from over 100 incorporated controlled vocabularies and their relations. It includes over 1 million biomedical concepts and 5 million terms. Some of the more prominent incorporated ontologies are *MeSH* (Medical Subject Headings) (NLM, 2008) classifying concepts used for indexing, cataloging, and searching for biomedical and health-related information in documents and the ICD-10 (International Classification of Diseases Version 10) published by the WHO (WHO, 2006).

- *Semantic Network* is the ontology over the Metathesaurus and provides the categorization of the used concepts and relationships. Currently there are about 135 semantic types and 54 relationships.

- *Specialist Lexicon* is a lexicon containing syntactic definitions for both biomedical terms and general English terms for use in natural language processing.

UMLS provides also several supporting software tools used in the project *Semantic Knowledge Representation* (SKP); the main goal is to provide usable semantic representation of biomedical free text (NLM, 2007). These three resources and tools provide a framework and ontology that can be used to facilitate the communication between different systems, or to develop systems that parse biomedical literature. The NLM itself uses UMLS for processing the documents available in PubMed.

Besides biomedical documents, it is also important not only for researchers but also for computers to understand the different terminologies for genes and proteins. The *Gene Ontology* (GO) project provides a comprehensive controlled vocabulary describing the role of genes and gene products in any organism (Gene Ontology Consortium, 2006). Actually GO can be split in two parts; the first is the ontology itself and the second part represent the instances annotated with the terms from the ontology. Besides, the ontology itself consists of three public available controlled vocabularies: *biological process, molecular function,* and *cellular component*. The concepts in GO are arranged as nodes in a direct acyclic graph, where multiple inheritance is allowed. A similar ontology to GO is the *Functional Catalogue* (FunCat) developed at MIPS. It represents a taxonomy containing 28 main protein functional categories that cover general fields like cellular transport, metabolism and cellular communication/signal transduction (Ruepp, et al., 2004). Since it has a tree structure with a depth of up to six levels of increasing specificity, it is much easier to apply than GO during manual or automated annotation of diverse genomes.

An area where many ontologies have been developed is *anatomy*. There are specific anatomy ontologies for different organisms (*Homo sapiens, Caenorhabditis elegans, Drosophila melanogaster, Saccharomyces cerevisiae, Danio rerio, Mus musculus, etc.),* cell types, and enzyme sources. Another ontology is the *Mammalian Phenotype Ontology* (MP) that covers standard terms for annotating mammalian phenotypic data. This controlled vocabulary has been developed by MGI (Mouse Genome Informatics) community and has a tree structure built with is-a relations (Bult, et al., 2008).

Many of the above described ontologies are available via the *OBO Foundry* (Open Biomedical Ontologies, formerly Open Biological Ontologies). The main goal of this collaborative project is the establishment of a set of principles for ontology development to create a suite of orthogonal interoperable reference ontologies in the biomedical domain (Smith, et al., 2007). The most common format for representation of ontologies in OBO is the OBO flat-file syntax. It aims to achieve human readability, ease of parsing, extensibility and minimal redundancy. Additionally, mappings between ontologies are provided to bridge concepts existing in separate ontologies but having logical relations. For instance, a GO-FunCat mapping has been developed and is available for the interested communities.

Biological ontologies are still mainly used for annotation of experimental data and various software tools (e.g. BLAST2GO) exist to support or predict the annotations for data entries using biological ontologies like GO, FunCat, or MP. Ontologies are also used in different steps of ontology-based searches in many information resources like genome specific databases. For instance, a user can search in the CORUM (Comprehensive Resource of Mammalian protein complexes) database (Ruepp, et al., 2008) by using terms from the FunCat ontology as query terms to retrieve protein complexes annotated with a particular biological function. Ontologies act as community references and can be used for information integration and information exchange across different biological and medical domains, since they allow both researchers and computer systems to share information in a meaningful way.

Although ontologies have been around for a while, it is only during the last decade that the development and use of biological ontologies have emerged as important topic. The efforts on designing ontologies have been accepted as essential in some of the grand challenges in biomedical research. Ontologies have been recognized as fundamental tools in the efforts to partially understand and to semantically interpret the information buried in diverse resources available on the Web. Several paradigms like the Semantic Web try to cope with the challenges in the process of knowledge discovery. However in the world of life-science several aspects should be considered regarding the concepts of knowledge representation, thus it is important to point out some essential historical facts and novel paradigms.

2.4.3 The World Wide Web

The World Wide Web, or the Web, has changed tremendously the work of life-scientists, since it provides access to already gained knowledge in order to successfully complete a task, to create a new hypothesis, to identify an unknown entity, or to classify already known ones. As already pointed out in *section 2.2* and represented in *Figure 2-4*, the Web plays a crucial role in the process of knowledge discovery. However, success in life-science research depends on the ability to indentify, navigate, integrate, and query information resources and the tools for this purpose continue to be the limiting factors. Most significantly, the Web has revolutionized the way information is organized and accessed via the internet and the technical achievements of the Web have evolved far beyond its original conceptualization (CERN, 2008). Nevertheless, along with the success of its paradigms, there is awareness of its limitations. Generic Web search engines or specific information resources allow the users to find documents, but do not link them directly to the subject they are interested in and their connections to related entities to provide conclusive support for decision making. Some of the causes for these limitations can be followed in the evolution of the Web.

A bit of History

The Web is actually a system of interlinked *hypertext* documents, called *Web pages*, accessed via the internet. Beside text, Web pages may also contain images, videos, and other multimedia. The term hypertext refers to Web page text that contains links and connections called *hyperlinks*. They lead the user to related information available on the same or another Web page (Landow, 1997).

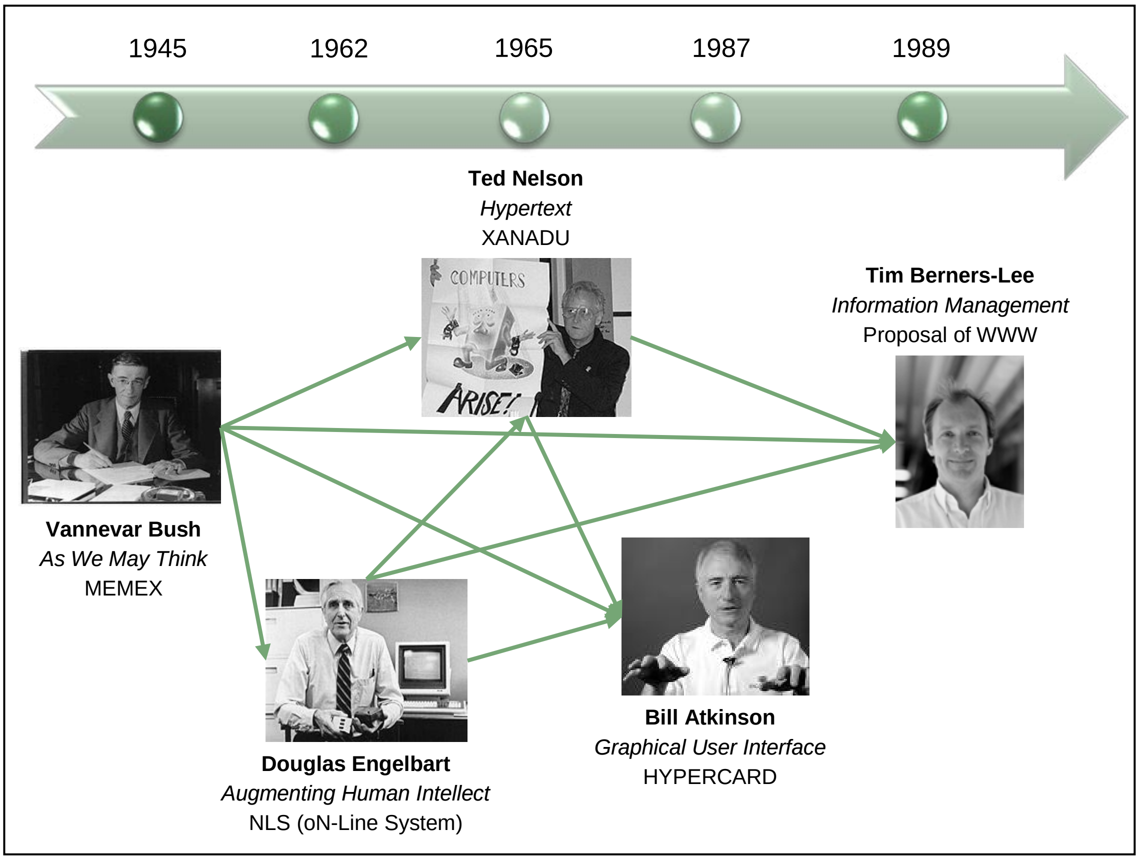


Figure 2-11: Summarized timeline of the hypertext technologies with some of the most significant researchers in the field, starting with the inspirer Vannevar Bush (Pepper, 2008).

The term hypertext was firstly introduced by Ted Nelson in 1965 (see *Figure 2-11*). His work and the work of Douglas Engelbart, who developed the first hypertext interface called *oN-Line System* (NLS) introduced in 1968 (Nyce, et al., 1991), were inspired with the thoughts of Vannevar Bush stated in the essay *“As We May Think”* in 1945

(Bush, 1945). In the article Vannevar Bush, as an engineer and a science advisor, was concerned about finding information with the increasing amount of research results:

> *“Mendel's concept of the laws of genetics was lost to the world for a generation because his publication did not reach the few who were capable of grasping and extending it; and this sort of catastrophe is undoubtedly being repeated all about us, as truly significant attainments become lost in the mass of the inconsequential.”*

The key answer to the problem was that each record useful to science had to be continuously extended, stored, and above all, consulted. However, the existing technologies were unable to cope with these challenges. Therefore, new approaches and mechanisms were demanded. The main idea of the solution was to get away from hierarchical systems of organization and adopt new techniques that reflect how the human brain works. He introduced the paradigm of *associative thinking*:

> *“The human … operates by association. With one item in its grasp, it snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails carried by the cells of the brain… The speed of action, the intricacy of trails, the detail of mental pictures, is awe-inspiring beyond all else in nature... Selection by association, rather than indexing, may yet be mechanized.”*

The proposal of Bush was the development of the *MEMory EXtender* device (MEMEX) considered as *“a sort of mechanized private file and library”* represented in *Figure 2-12*. It consists of a desk containing:

- a very large set of *documents* stored on microfilm,
- screens on which those *documents* are projected,
- a device for photographing new *documents*,
- a mechanism for retrieving *documents* at the push of a button,
- the ability to create links between *documents*, and
- the ability to build trails through *documents*, add comments to *documents*, insert new *documents*, etc..

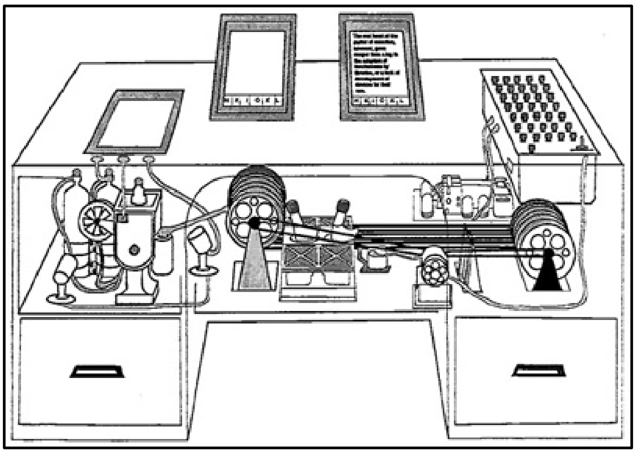


Figure 2-12: Draft of the futuristic device MEMEX in the article *“As We May Think”* (Bush, 1945).

Surprisingly, everything revolves around documents, in this context digital artifacts representing information. However, people do not think in terms of hyperlinked documents as represented in *Figure 2-13*, but in terms of concepts and associations between concepts as emphasized in *section 2.4.2*. In general, documents are about subjects that exist as concepts in our brains. The way we store knowledge is by building mental models, where the concepts are connected into a network of associations. Documents are just a representation of some part of that knowledge. Nevertheless, the basic idea of Vannevar Bush was brilliant to organize the information associatively, as the way we think, in order to make it easier to find (Pepper, 2008). The inventor of the present Web Tim Berners-Lee and his forerunners adopted this idea and applied however the document-centric approach for its implementation (Berners-Lee, et al., 1990).

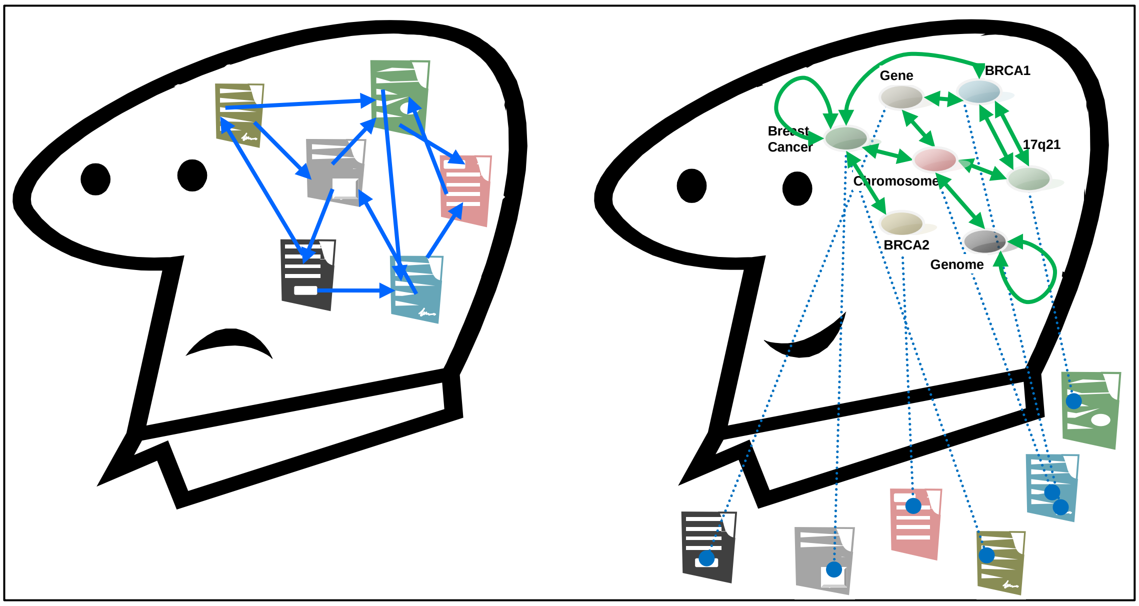


Figure 2-13: Document- and subject-centric navigation approaches through the information space.

The progression of online encyclopedias such as Wikipedia shows how useful the document-centric approach can be and how helpful computers can be for users to discover and navigate through related documents. However, major problems remain for the users to keep up with the rapid expansion of the Web information space and computers still cannot assist them in this issue. It is still quite surprisingly how little some technologies have actually advanced since the publication of *"As We May Think"* in 1945. Technologies dealing with querying based on natural languages and associative ways of connecting information represented not just in documents are still underdeveloped and not at all established. Most Web users are interacting with computers in non-natural ways, adapting to the existing technology instead of having technologies adapted to the users. Perhaps one of the reasons for that is the way information is currently represented and structured through the Web. The *Semantic Web* tries to address some of these issues by establishing a new information infrastructure that should enable computers to tackle the information needs of the users.

The Semantic Web

In 1998, the inventor of the Web Tim Berners-Lee proposed his vision of the Semantic Web (SW):

> *"The Web was designed as an information space, with the goal that it should be useful not only for human-human communication, but also that machines would be able to participate and help. One of the major obstacles to this has been the fact that most information on the Web is designed for human consumption, and … that the structure of the data is not evident to a robot browsing the web. Leaving aside the artificial intelligence problem of training machines to behave like people, the Semantic Web approach instead develops languages for expressing information in a machine processable form."*
>
> ***Tim Berners-Lee, 1998***

To realize the vision of the SW (Berners-Lee, 1998), several research communities, dealing with knowledge representation, information retrieval, multi-agent systems, and other topics, have concentrated their research efforts on the development of a number of standard technologies. The *World Wide Web Consortium* (W3C) has been facilitating, developing, and promoting such Web-based standards called also "W3C Recommendations" since 1994 (W3C, 2008). One of the first established standards was the *Uniform Resource Identifiers* (URI) standard for identifying objects in the Web space. The *Uniform Resource Locators* (URL) and *Uniform Resource Names* (URN) are special cases of URI. A URL specifies the location of a web resource and a URN defines something's identity (Jacobs, et al., 2004). In general, URI are unique identifiers for Web resources and do not need to correspond to downloadable resources, although they often do. However, to fulfill the vision of the SW, further technologies were needed. These include the development of the *Resource Description Framework* (RDF) and the *Web Ontology Language* (OWL) for encoding knowledge in the form of standard machine-readable ontologies. The goal of these standards is to migrate from syntactic Web of documents to the semantic Web using ontologies, since hypertext links by themselves do not convey any semantic meaning and do not explicitly specify the relationship between the two linked resources.

RDF is the fundamental component of the SW technology and as its name suggests, it is a language for representing information about *resources* in the WWW by providing metadata models. In the RDF metadata model everything imaginable or noticeable is represented by a particular resource and resources are connected via predicates. A resource, according to the RDF primer (Manola, et al., 2004), *“is anything that is identifiable by a uniform resource identifier”* reference. Thus one could use URI to represent diseases, proteins, and genes even though none of these are Web resources in the original sense. The basic information unit in RDF is an RDF statement in the form of *subject-predicate-object* expression, called also a *triple*. Each RDF statement can be modeled as a graph comprising two nodes connected by a directed arc as illustrated in *Figure 2-14*. The subject denotes the resource and the predicate denotes traits or aspects of the resource and expresses a relationship between the subject and the object (e.g. “BRCA1 is located at 17q21” means “BRCA1” is the subject, “is located at” represents the predicate and the object is denoted as “17q21”). A set of such RDF statements can jointly form a large directed labeled graph representing different knowledge domains. The semantics of a RDF model is obtained via references to RDF Schema (RDFS) or OWL ontology. Both languages RDFS and OWL are layered on top of RDF to offer support for inferences. Additionally, in the SW queries can be defined via the SPARQL querying language.

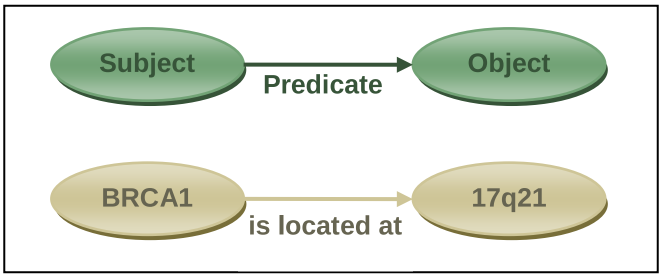
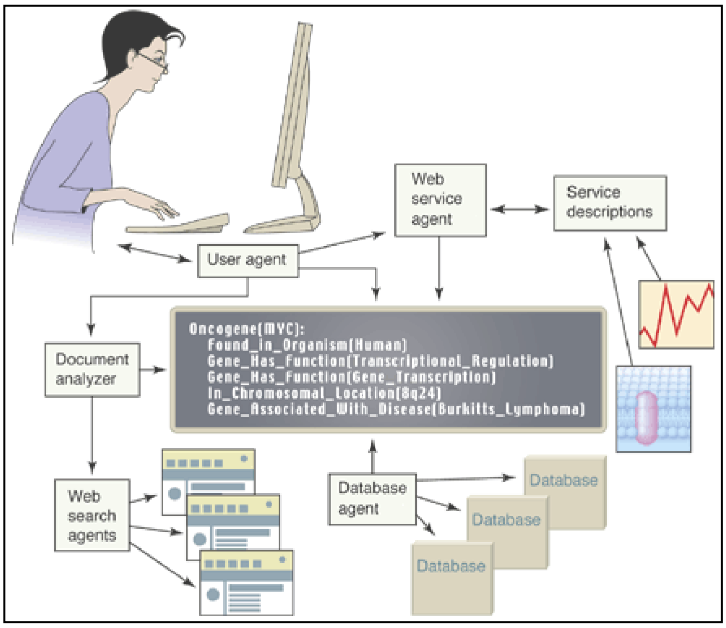


Figure 2-14: Graph model for an RDF statement. An RDF statement can be modeled as direct labeled graph with resources (subjects and objects) as nodes and predicates as the directed edges connecting from subjects to objects.

In general, all these technologies supporting the implementation of the SW vision try to put the information in a formal way that machines are able to semantically interpret it. The so called *agents*, or *intelligent agents*, or *software agents* are actually the computer systems that should be able to use the information in the supplied metadata to perform tasks for users of the SW such as answering queries as shown in *Figure 2-15*. Usually, agents do not act in isolation, but interact with each other to achieve their objectives, resulting in what is typically called Multi-Agent Systems (MAS). In MAS different types of agents have different responsibilities: interaction with the user, planning of objectives, scheduling of tasks, or interaction with external resources such as databases. Since agents have to be able to work together, Agent Communication Languages (ACL) are required (Burger, 2007).



```
Oncogene(MYC):
  Found_in_Organism(Human)
  Gene_Has_Function(Transcriptional_Regulation)
  Gene_Has_Function(Gene_Transcription)
  In_Chromosomal_Location(8q24)
  Gene_Associated_With_Disease(Burkitts_Lymphoma)
```

Figure 2-15: Multi-Agent System working in the Semantic Web environment (Keele, et al., 2005).

Over the past several years, large research efforts have been invested in the development of these SW technologies and it has been important to provide test-beds for their application. Many believe that the life-science domain can serve as an excellent test bench for the SW technologies and the so-called *Life-Science Semantic Web* (LSSW) was founded. This belief can be substantiated with not only high publicity through the many keynotes, workshops, and special sessions at major international Semantic Web conferences (e.g. “International Semantic Web Conference 2007”, “NETTAB 2007 A Semantic Web for Bioinformatics: Goals, Tools, Systems, Applications”), but also through the support of the SW community (e.g. “W3C Semantic Web for Health Care and Life Science Interest Group” has been founded in 2005). Additionally, a large number of papers have been published in prestigious journals (Hendler, 2003), (Wang, et al., 2005) and special issues on SW (Clark, 2007) as well as textbooks (Baker, et al., 2007), (Daconta, et al., 2003).

Diverse tools have been represented in conferences and described in these publications; some of them provided by commercial vendors, others developed by academic institutions as open source software. One can classify them into three categories. Some of these tools provide functionalities to adapt existing biological ontologies to the OWL standard. The "*bio-zen OWL ontology framework*" is such a system that generates OWL ontologies like *bio-zen-MESH.owl*, or *bio-zen-GO.owl*. This framework is actually part of the *Semantic Synapse Project*. Its goal is to develop SW ontologies for use in neuroscientific and biomedical research. A prototype version of a web portal called *Entrez Neuron*, which makes use of integrated neuroscientific information in SW formats, is also available for public testing (Neuroscientific Net, 2008). The second category of software tools are the ones providing RDF annotation for existing biological entities. The web application *YeastHub* is a typical example for information integration using the SW approach in this case in the yeast research community (Cheung, et al., 2005). There are also other tools like *Uniprot-RDF*, *LinkHub*, *Boca* (Feigenbaum, et al., 2007), *SWAN* (Clark, et al., 2007), and *SenseLab* (Crasto, et al., 2007) providing RDF repositories containing the RDF metadata of the information integrated from databases important for research groups such as the *NeuronDB* and *ModelDB* databases relevant for neuroscientists. These tools follow the data warehousing approach for the generation of annotation repositories, thus the RDF files containing the triples have to be updated frequently (for more details see *Data Warehousing* in *section 2.3.2*). The last group of developed applications represents tools providing mechanisms for browsing through RDF graphs (e.g. SIMILE (Mazzocchi, et al., 2005), Haystack (MIT, 2008), RAP (Westphal, et al., 2008)). However, just the SW browser *BioDash* has specifically targeted the life-science community (Quan, 2007).

After an extensive evaluation of even more than the above mentioned methodologies and representing tools, several significant conclusions have to be pointed out. Although, LSSW is a very active research field, unexpectedly there is still no demonstrative application that shows the benefits of using the SW. There are plenty of biological ontologies formalized in the OWL standard, but most of them are in fact not used. The few existing RDF metadata repositories are often out of date or incomplete. The visualization applications are much too complicated and laborious to use; some of them are only executable in external software development environments and not within the standard web browsers, thus not suitable for life-science researchers. In the few working applications such as BioDash or LinkHub, there is no intuitive workflow for information exploration. Last but not least, there are almost no software agents in a widespread use (Keele, et al., 2005). The key conclusion regarding SW application can be briefly expressed by answering the question posed in an article about the LSSW *"Are We There Yet?"* (Neumann, 2005) – "*Not at all! We even haven't started!* " and the situation depicted in *Figure 2-15* remains just a vision for scientists at the moment.

Semantic Web Clashes Life-Science

To a similar conclusion came the authors of a review article with the provocative title *"The Life Sciences Semantic Web is Full of Creeps"* (Good, et al., 2006). They address several aspects why the life-science community has appeared reluctant to fully adopt the standards and technologies of the SW. For them the one of the most important factors is the unwillingness of the acceptance of the Life Science Identification System (LSID) for all entities within the LSSW instead of the usage of the common URI. Therefore, the key reason according to the authors and in general to the SW community is more social than technical that the leading players in the life-science community refuse to participate and therefore provide their own biological data according to the SW standards. However, I believe that the key reason is much more fundamental.

The Semantic Web is about teaching computers to collect information from resources all over the Web and interpret it in a correct way. It is much more about the machine-machine rather than the human-machine interaction.

> *"I have a dream for the Web [in which computers] become capable of analyzing all the data on the Web – the content, links, and transactions between people and computers. A 'Semantic Web', which should make this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The 'intelligent agents' people have touted for ages will finally materialize."*
>
> ***Tim Berners-Lee, 1999***

The SW approach may be quite helpful for diverse areas of life such as communication but it is not very applicable for science. The driving force in science is always the researcher with his ideas, inspirations, and visions. Scientists seek for explanations of diverse phenomena in nature by applying systematic approaches. Based on different types of observations or coincidences, hypotheses can be built, and lately validated to acquire novel knowledge. The duty of computers is just to assist and not to substitute a scientist.

Unfortunately, nowadays scientists have to adapt to the existing Web technologies instead having Web technologies adapted to their needs. And the reason for that is essential: the information available in the Web space is organized in a document-centric way – as machines may think and not as we may think. Even though in the SW approach, the OWL ontologies provide an adequate knowledge representation they are separate from the RDF metadata and therefore all already existing resources have to be annotated with metadata according to the RDF standard. This process results in a so-called bottom-up approach. One can ask the question, if there is enough funding for RDF annotation of the tremendous amount of data available in the life-science domain and who is able to perform these enormous efforts to make the SW successful. Most of these problems can be addressed, but this will represent a fundamental change in the way information is organized and represented on the Web. Particularly in the domain of life-science, we have to shift the paradigms from the document-centric to subject-centric computing, as we may really think (see *Figure 2-13*), and keep in mind that researchers are the driving force for discovery in science. We should have open eyes to discover new landscapes.

Subject-Centric Computing

Quite long time ago Plato (428 – 347 BC), a classical Greek philosopher, posited that there is a separate plane of existence, accessible only by our minds, containing the ideal "forms" for every object and concept known to man (Roberts, 1905). In the real world, everything and anything can be a *subject* of discussion, and every subject of discussion can be a hub around which data can orbit. Currently, computers reside at the center of the universe of information, since information is organized the way machines may think (see *Figure 2-16.A*). Diverse information resources, described by metadata, revolve around them. Subjects are hardly to be seen or at least hardly to be found, since they are situated at the periphery. The subject-centric view reflects the way humans think (in terms of subjects, concepts, ideas) and therefore subjects are located right next to the middle of the information universe. When a subject happens to be data, then metadata, and diverse information resources, from which the data comes, can spin around the subject (compare *Figure 2-16.B*). In few words, all existing data in the diverse information resources is data about subjects, but only some of the existing subjects are themselves data; however most subjects reside not in information resources but just in our minds. The essence of the subject-centric computing is to organize the information in subjects, because that's what human beings are really interested in. Consequently, the solution to the problem of global and generic knowledge interchange can become much easier and simpler (Newcomb, 2003).

Nevertheless, there is still one problem, because computers cannot access subjects unless those subjects happen to be information resources themselves. Therefore, in the subject-centric computing the semantic still plays a major role, since it is important to represent the information in a meaningful way to provide semantic interoperability, which becomes even more significant if we are interested in the knowledge structures rather than just their carriers – the data resources. The separation of the knowledge structure level or conceptual level from the resource level aids the semantic interoperability. Different view models or interpretation contexts built over the same resources can represent different subject matters. Interoperability is achieved by flexibly keeping such interpretation models apart, or by combining them as needed (Sigel, 2003).

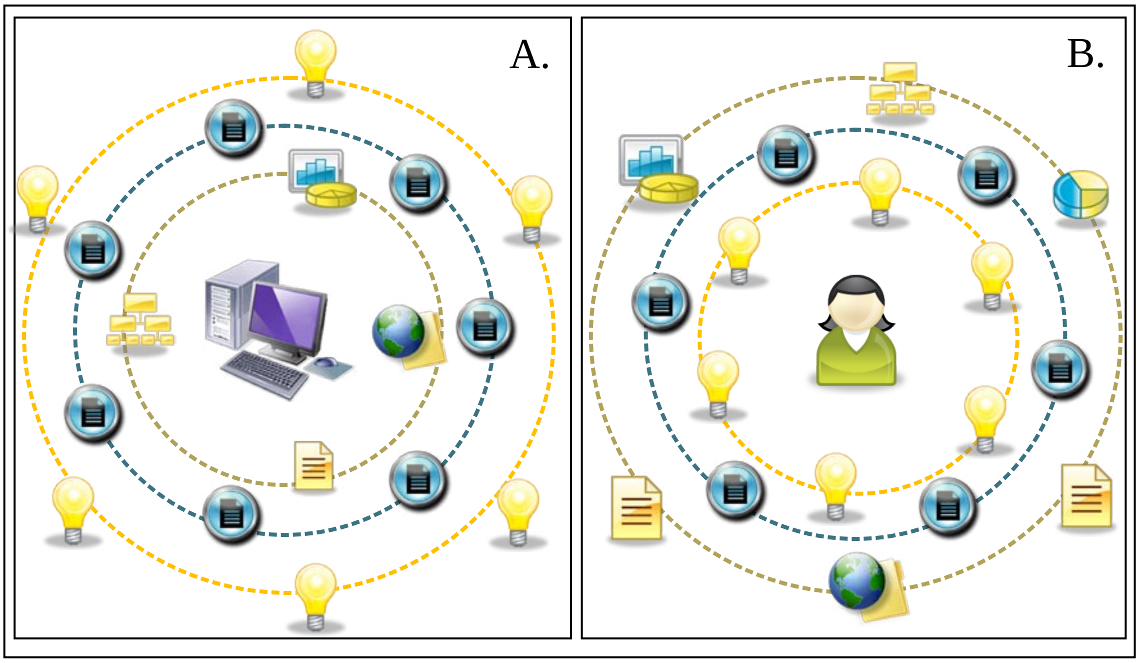


Figure 2-16: Organization of the information universe **A.** Document-centric computing or organization in the way machines may think **B.** Subject-centric computing or organization in the way humans think (Pepper, 2008).

Parallel to the development of the Semantic Web paradigm, another semantic technology called *Topic Maps* (TM) has evolved to address the issues of knowledge representation and organization of the Web information space. The initial ideas behind TM, which date back to the early 90's, arose from the need to model intelligent electronic indexes of glossaries, tables of contents, thesauri, or cross references. The goal of the TM paradigm was to semantically characterize and categorize documents and sections of documents on the Web with respect to their content – in other words, what *topics* or *subjects* those documents actually address. After several years of discussion and evolutionary development cycles, the TM model has developed into something much more powerful that is no longer restricted to simply modeling indexes. The established in 2000 and refined in 2003 standard ISO/IEC 13250 Topic Maps (ISO, 2007), provides a reference model for the generic semantic structuring and organization of any knowledge domain. The TM paradigm addresses the knowledge representation aspects from the human perspective and focuses mainly on the subjects (the things humans want to know more about) and consequently on orbiting data around them or resources. Since the TM technology follows the subject-centric approach, the main concepts, characteristics, and some comparisons to the Semantic Web technologies are discussed in the following section.

Topic Maps

The power of Topic Maps has been described as being the GPS of the information universe (Pepper, 2000). TM provides a mechanism to overlay semantics and structure onto existing, possibly dispersed and heterogeneous, information resources as illustrated in *Figure 2-17*. Thus, this mechanism corresponds to a top-down approach. Like street maps enable pedestrians to find their way from A to B, topic maps[3] enable Web users to navigate within the scope of the mapped information, thus they build associative semantic networks. They can be created and stored independently from format, structure and location of the underlying resources. Importantly, they let users navigate through the information space without having to be aware of the data structures or the internal relationships between independent resources. The way the data is organized in different databases, or other information resources, is hidden from the user. Additionally, in the TM paradigm, one can have multiple topic maps representing the referenced subjects in different ways built over the same information resources or provide different views to different users (similar to a book having multiple indexes, such as a name index, a subject index, etc.).

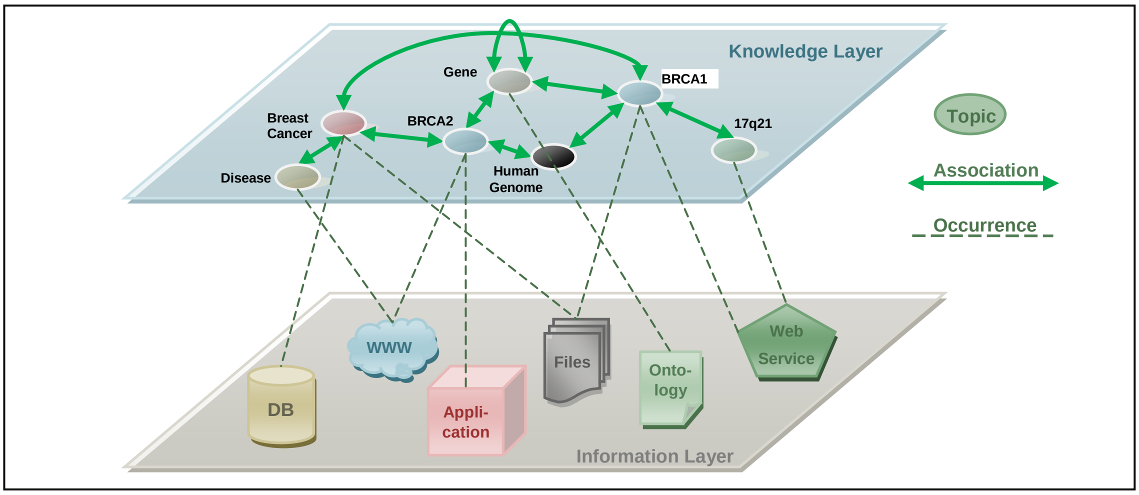


Figure 2-17: A topic map as an external overlay onto existing information resources can represent any knowledge domain by describing the semantics of the comprising information subjects (topics) and their relationships (associations). Occurrences provide the binding between the two separated knowledge and information layers.

The ISO standard provides a data format for interchanges based on XML syntax that is called XML Topic Maps (XTM). The basic constructs of topic maps are *topics*, *associations* and *occurrences* also known as TAO model (Pepper, 2000). Additionally

[3] The ISO committee advocates that it should be carefully distinguished between "Topic Maps" (a singular noun that refers to the ISO standard of that name, or the technology itself), and "topic maps" (the plural of topic map, the artifact around which an application is built). The former should be capitalized; the latter should be lower cased. This recommendation is followed throughout the text.

there are some further extended concepts, which are shortly discussed and illustrated in *Figure 2-18*:

- *Topics* are the fundamental building blocks of topic maps. They represent real world subjects. Since subjects can be anything, topics can be anything. They act as binding points for all the information related to these subjects. In general, topics representing not only general concepts like “disease” or “gene” can be defined, but also their referents like “Breast Cancer” or “BRCA1”. This mechanism provides an important feature for semantic systems, because so it is possible to identify the *type* of thing being described. For example, “BRCA1” and “gene” are two different subjects, thus two different topics, but the topic “gene” is also the typing topic of “BRCA1”, thus “BRCA1” is “gene” (compare *Figure 2-18*).

- *Associations* represent the relationships between the subjects of specific topics, with each topic involved in the association being a *member* playing a specific *role* in it. Associations are the key to develop independent knowledge layers on perhaps same information resources, i.e. building interpretation contexts. As with topics, associations can (and should) have a defined *type*. In general, associations are completely independent from the information resources and therefore they represent the essential additional content of the topic map. One association is able to illustrate the way from A to B and from B to A. They are bi- or multidirectional (not restricted to two members); thus, the association has meaning when viewed from the perspectives of all the constituting members. As depicted in *Figure 2-18* , “BRCA1 causes Breast Cancer” is an association from type “Gene-Disease Effect”, in which “BRCA1” is a gene and “Breast Cancer” is a disease topic, but exactly the same association can be viewed from the opposite perspective “Breast Cancer is caused by BRCA1”. There is no need to make the decision whether to make the relationship a property of the disease or the gene, as the topic map will always link both. This is contrary to RDF, where two distinct statements would have to be defined to express the relationship from the two points of view.

- *Occurrences* of a topic add information about the subject the topic represents; they express properties of the referents. They can be any information resource which the author deems relevant to the topics (documents, image files, etc.). They may be internal textual resources, such as a simple text description (e.g. a text representation of a DNA sequence). More frequently, they are references to external resources expressed in the form of URL. As with topics, the type of the occurrence (the characteristic of the topic) can be defined by a reference to the topic representing the notion of an appropriate subject, such as “description”, “web page”, or “DNA sequence” (see *Figure 2-18*). By using occurrences it is not only possible to bind diverse information to the subject, but also to powerfully manage the link information.

- *Scope* expresses the context in which an assertion is valid and thus provides support for contextual knowledge and the ability to represent multiple, even contradictory “Weltanschauungen”. Within this definition it can be applied to a wide range of uses, for instance to constrain information spaces to the context of “*Homo sapiens*” related information.

- *Subject Identifiers* enable topic map processing applications to uniquely identify the subject of a topic and importantly, to know whether two topics represent the same or different subjects and cope with the problem of homonyms. Although topic map authors are free to define their topics, they benefit when a recognized set of identifiers is used to denote subjects in a given domain. In such cases references to existing ontologies can be beneficial. Such identifiers are called *published subject indicators* (PSI).

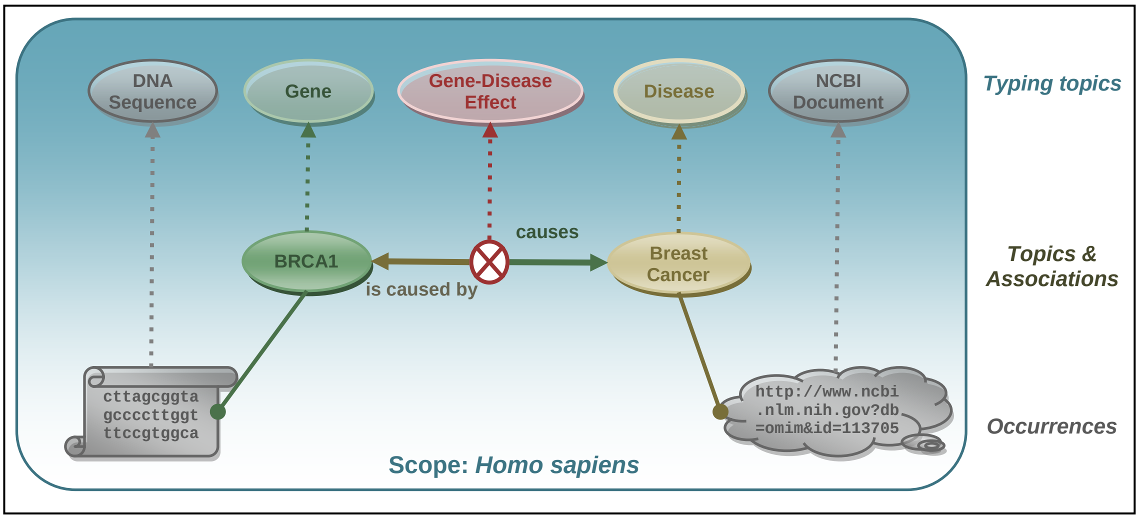


Figure 2-18: Key concepts of Topic Maps. Typing topics describe concepts and thus the semantics for referent topics. Associations connect topics and can be viewed from the perspectives of each playing member. Properties are anchored to topics and called occurrences.

All these Topic Maps concepts support the subject-centric approach for knowledge representation and enable the exchange and integration of knowledge spread over different sometimes partially overlapping domains. With TM different view models can be described to represent different subject matters. Knowledge interoperability can be achieved by flexibly keeping such models apart, or by combining them as needed as illustrated in *Figure 2-19*. Additionally, one can use this approach to partition large areas of knowledge into manageable sub-areas. A mechanism of *merging* provides possibilities to join different topic maps by applying certain rules to offer new knowledge perspectives and discover novel insights. The paradigm of inference is one of the most powerful and useful paradigms for information exploration in the context of science. This approach is possible, since the information is semantically described and organized in a subject-centric way. Further characteristic of topic maps is that they are well suited to represent ontologies and thus to facilitate a way of describing a shared common understanding. So the usage of common biological ontologies can support the knowledge exchange to finally overcome integration problems persisting since the introduction of the very first sequence databases.

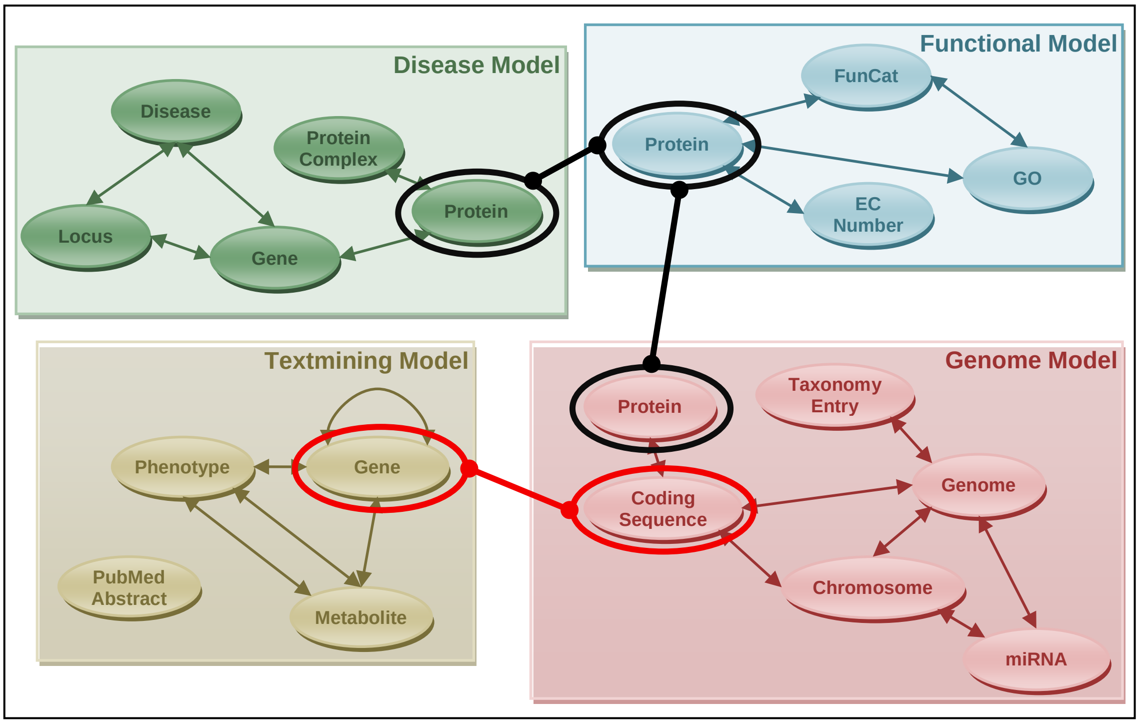


Figure 2-19: Merging of separated TM models representing different subject matters. When needed models can be combined, if there are topics describing the same subjects even though they may have different names.

Both paradigms Semantic Web and Topic Maps, having the same goals, attempt to represent the information available on the Web space in more powerful way in order to improve access, provide clearer overview, and supply more effective finding aids. Both approaches are defined as open, recognized standards and provide generic structure, which can be applied to any knowledge domain. Although, both of them are capable of addressing the issues of semantically connecting distributed information resources, they do it in two completely different ways (bottom-up and respectively top-down approach). In the area of science, especially of life-science, the paradigm of Topic Maps is more applicable and straightforward, since the topic/association layer mirrors the associative way humans think and so it can be applied as a navigation interface for the occurrence layer, which contains the information spread over the Web. In contrast to Semantic Web, topic maps are not separated from the describing ontologies like RDF is from OWL/RDFS, thus the Topic Maps paradigm provides a “higher-level” of knowledge management (Smith, 2003).

In summary, one can consider a topic map as a *unified knowledge model* that constitutes a map of some subject domain and associates information originating from any kind of information system related to those subjects. Such model represents a rich network of connections between related subjects and thus provides user-friendly navigation paths. While the model might start out as a simple layer, providing improved access to a set of information resources, it can evolve smoothly into a knowledge hub. Furthermore, depending on the user's needs related models can be connected to form even more comprehensive models. Such united models overcome the obstacles concerning disconnected information (meaning consequentially disconnected knowledge), because knowledge can be then shared, new relationships can be identified, and novel insights can be drawn.

For the successful accomplishment of knowledge management in the area of life-science, not only the knowledge representation techniques have to be considered very carefully, but also proper integrative methods have to be applied. During my PhD research, I developed a comprehensive software framework that combines essential technical and conceptual integrative approaches. The main goal of the framework is to assist life-scientists in their research efforts by giving them the possibilities to explore the WWW information space and build models of related biological entities to explain the complexity of life, in particular in the field they are interested in. The framework represents a *Generic Knowledge Modeling Environment* shortly called GeKnowME. The main features and their realizations are illustrated in the following section.

3 GeKnowME

Generic Knowledge Modeling Environment

> *"Failure is only the opportunity to begin again more intelligently."*
>
> ***Henry Ford (1863 - 1947)***

Without any doubt, the Web has become the most important medium for many scientific communities to share their knowledge. It supports tremendously life-scientists in their research endeavors, since it provides almost instant access to information offered by many other communities. Consequently the Web has changed the way research is performed nowadays. However, as modern life-science continues its exponential growth in complexity and scope, the need for assembly of knowledge coming from related scientific disciplines is becoming more and more important. Current knowledge based integrative approaches and technologies, as pointed out in the preceding chapter, are still insufficient to satisfy those needs. The motivation for the development of the GeKnowME framework was to provide a novel system, which should improve the effectiveness of life-science research by accelerating the knowledge discovery process and supporting scientists with powerful tools for analysis and navigation through correlated biological entities. This goal is achieved by offering the user an environment where he can define straightforward semantically rich and sufficiently correct models to ensure meaningful and reasonable use of the knowledge extracted from distributed domain-specific information resources. The researcher can decide which biological entities are relevant for his exploration and consider them during the model generation. The framework is as well generic enough to be applicable for a broad range of use cases.

During the design and development of the GeKnowME system, established integrative technologies with their beneficial concepts and methods have been taken into consideration based on the experiences made in the field of integrative bioinformatics. However, novel approaches combining these recognized techniques with new methods like the paradigm shift from document- to subject-centric knowledge representation have been applied to achieve the ambitious objectives of the framework.

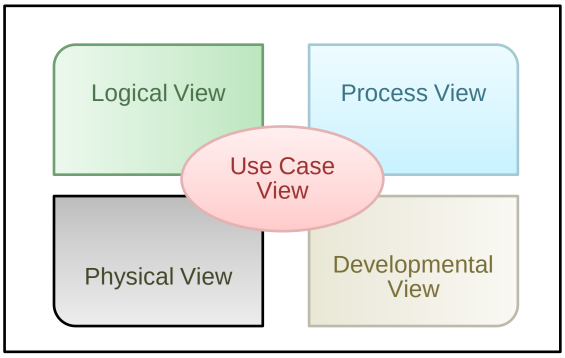


Figure 3-1: Kruchten's 4+1 View Model for describing software architecture from different perspectives (Kruchten, 1995).

Since the structure of the GeKnowME framework is rather complex, its system architecture is presented by different types of UML (Unified Modeling Language) diagrams. In general, a system architecture provides the conceptual understanding of the system's design and functionality in the form of its major components and how they interact. For the precise description of the GeKnowME system architecture, *the Kruchten's 4+1 View Model* (shown in *Figure 3-1*) is used, since each of the five concurrent views addresses specific facets of the software system.

The *use case view* plays a central role in the 4+1 view model, because use cases, as situations capturing pieces of functionality provided by the system to fulfill one or more user's requirements, affect all other steps within the system design and behavior. After the definition of the specified use cases, in the system *logical view* the major concepts to realize the required functionalities are described. The technologies necessary for the implementation of the introduced concepts are explained in the following *physical view*. The precise description of the developed software components implementing these concepts is given in the *developmental view*. At the end, in the *process view* the interactions within the GeKnowME system are captured.

3.1 Use Case View

The GeKnowME modeling environment fits mainly to the needs and wants of life-scientists to guarantee applicability and effectiveness. Essential system's requirements from the user's perspective have been collected and afterwards considered during the design and implementation of the framework. One can group the functional requirements in three main use cases as illustrated in *Figure 3-2*.

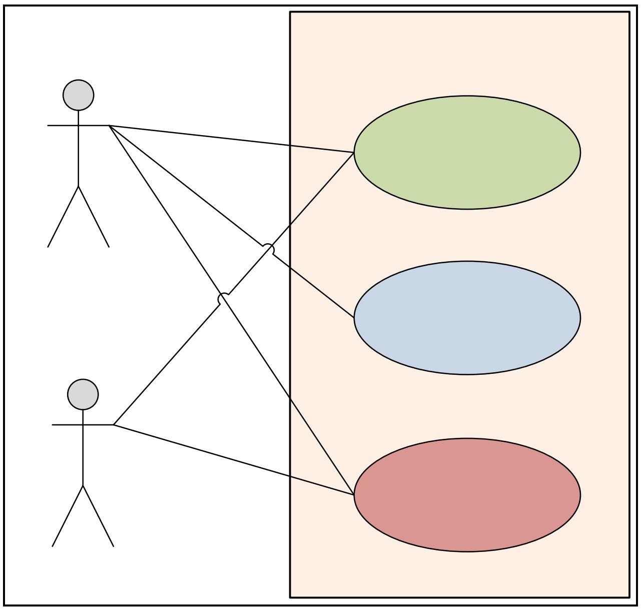


Figure 3-2: Use case diagram showing the actors and main use cases of the GeKnowME system.

Two types of actors interact with the GeKnowME system. The first ones are the end-users of the system in this case the *scientists*. The second types of actors are *developers* that are responsible for the configuration, maintenance, and extension of the system. The three major use cases embody the most important solutions to close the knowledge gap existing in the life-science domain (refer to *section 2.2*):

- *Definition of Knowledge Domain Models* use case addresses the problems regarding *lack of information*. By defining view models reflecting the concepts and their relationships only for the subject matter of interest and then by mapping these ontological models to only relevant information resources, the scientist defines where the correct information objects and their correlations can be found though the complexity of hundreds of independent, overlapping, and heterogonous resources. In this use case, developers take care of the implementation or adjustment of the required software components to perform these tasks.

- *Information Exploration* use case deals with difficulties concerning *overload of information*. Since the information complexity is reduced (not just its volume but also its heterogeneity), the user does not have to cope with too many information objects and relations with no or minor relevance and is able to navigate though coherent information space.
- *Building of Semantic Networks* use case provides tools to manage the acquired knowledge by representing and modeling it in the form of meaningful networks. Not only scientists can use them, but also developers can adapt these tools to provide automated generation of biological networks.

More precise descriptions of the supposed behavior and features of the system have been gathered from life-scientists and discussed with software developers. The most important ones are listed here to show more specifically the requirements detected before the development of the framework:

- The information space provided for exploration shall be consistent and contain up-to-date information.
- Since the existing biological information is quite heterogeneous not only in the format but also in the meaning, semantic information integration shall be ensured.
- The information space shall be concentrated to information resources only relevant for the research process to reduce the information complexity.
- Integration of well-established biological ontologies shall be allowed to provide concept mappings.
- The exploration information space shall include information extractable from free texts from articles in biomedical journals or established biomedical descriptions.
- View models representing the biological concepts and their correlations shall be easily adjustable and extendable to keep up with the changing understanding of the complexity in biological systems.
- Possibilities for combination of models representing different knowledge domains shall be provided to achieve a broader knowledge overview on demand.
- The system shall be kept generic to provide applicability for a broad range of life-science research areas.

- The user interface shall be geared to current internet based representation technologies and standards to ensure a broad acceptance.
- The user interface shall provide a basic infrastructure for system's customization and personalization.
- The user interface shall provide clear mechanisms for navigation through related biological entities to guarantee usability.
- The user interface shall ensure possibility to model networks representing the interactions within biological systems.
- In the user interface, the biological networks shall be based on graph representation.
- Biological networks shall be formalized in an XML format to ensure further automated analysis.
- State-of-the-art technologies for information integration shall be used.
- The system development shall follow a software component based approach to separate conceptual principles, increase reusability, and reduce maintenance efforts.

The GeKnowME system is designed to meet all above described functional requirements. However, the specific requirement concerning extraction of information from biomedical free texts turned out to be a very complex and challenging task. Therefore, a separate textmining engine called EXCERBT (EXtraction of Classified Entities and Relations from Biomedical Text) has been developed by Thorsten Barnickel in our group BIS at MIPS (Barnickel, et al., 2008). EXCERBT major objective is to extract information from natural language texts and structure it semantically. To achieve this goal, the system combines common textmining approaches such as generation of synonym lists and indexes over biomedical texts, entity recognition, information extraction, and modeling of semantic relations between found entities. EXCERBT is adapted to the subject-centric approach and can be easily plugged in to the GeKnowME framework to ensure seamless integration with other information resources. The developed concepts and system architecture to fulfill all other requirements to the GeKnowME framework are described in the system's logical view.

3.2 Logical View

The basic idea behind the logic of the GeKnowME framework is that different scientific communities can represent abstractly their area of research in the form of concepts and relationships. These associative models of knowledge domains can be then easily mapped to topic map models representing the topic types and the corresponding association types, since the Topic Maps approach follows the human associative way of thinking. The topic map models, representing the structure of the scientific knowledge domain of interest, can be then overlaid on the top of any arbitrary information resources as illustrated in *Figure 3-3*. Since topic map models contain the semantics of the included concepts, a model can be simply combined with other models sharing the same concepts and thus a scientific community can obtain a much broader overview of the subject matter if needed.

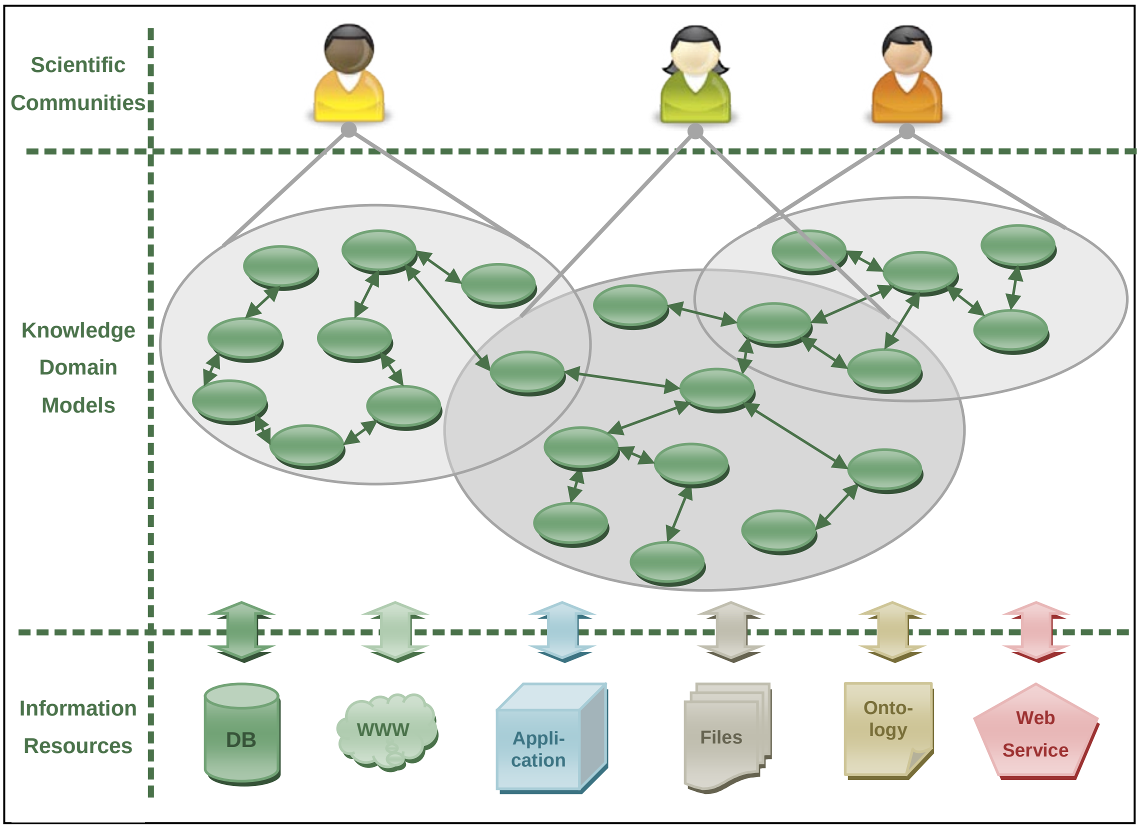


Figure 3-3: Knowledge domain models, reflecting the areas of research by describing concepts and interrelations, are placed over demanded information resources. The models, represented in the form of topic maps, can be easily merged together.

The mapping of the defined knowledge domain models to the existing information resources is not a trivial task and is decomposed into three separate functional steps, or *layers*. They are referred as *integrative logic* and include *Integration Layer*, *Syntax*

Layer, and *Semantic Layer*. Actually, the whole GeKnowME system consists of altogether five segregated layers, additionally *Information Resource Layer* and *Presentation Layer*, whereas each single layer, or tier, encapsulates distinct functionalities and is weakly interconnected with its lower and higher layers. The organization of the layers is represented in the GeKnowME system architecture in *Figure 3-4*.

In general, breaking down a system into layers represents one of the most powerful software architectural patterns, which has a number of benefits. The most important aspect is that one can separate the system logic in distinct tiers to reduce the functional complexity. Once a layer has been built, one can use it for many different higher-level services. Additionally, in an n-tier system one minimizes the dependency between the composing software units, or *components*, and thus the maintenance efforts (Fowler, 2002). In principle, a software component can hide any arbitrary functional complexity. Components are able to communicate with each other over well-defined interfaces. Usually, several specific software components reside within a separate layer and allow great flexibility and reusability.

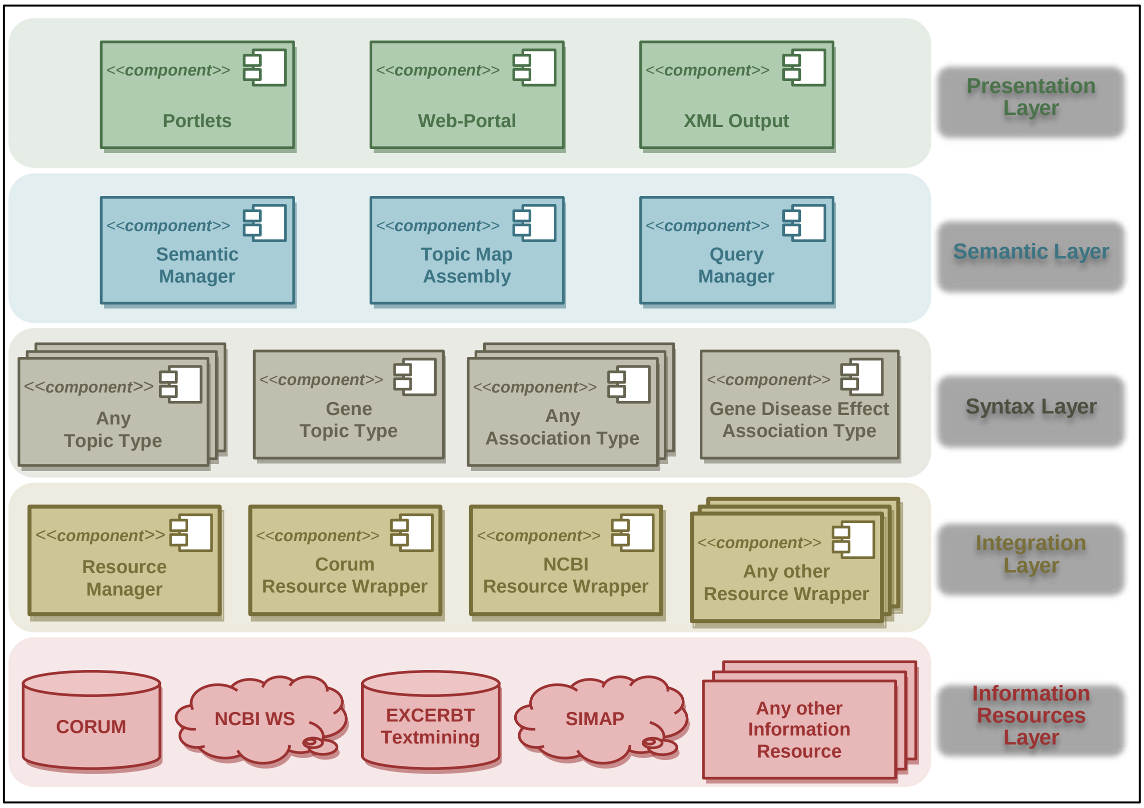


Figure 3-4: GeKnowME five tier system architecture.

The component oriented multi-layer architecture allows any of the tiers or just single components to be upgraded or replaced independently in case some of the requirements or underlying technologies changes. Since a higher level in an n-tier architecture uses the services defined by the lower levels, the description of all logical layers within the GeKnowME system architecture starts with the lowest one. In the logical view, each layer is described briefly to introduce only the main ideas behind the developed integrative approach and refers to *Figure 3-4*. In *section 3.4* discussing the developmental view, more precise explanations of each tier components with their interfaces and exact functionalities are provided.

3.2.1 Information Resources Layer

The lowest layer in the system architecture represents information resources that contain valuable data, which can be referenced within the defined topic map models. Only carefully selected and reliable resources are plugged in to the framework to ensure high quality. Any kind of information resource can be found within this layer: rational databases, applications, Web Services, ontologies, etc.. The main purpose of this tier is to provide the occurrence space for topic map models, through which the user can navigate. No modifications have to be done to the referenced resources, thus the integrative logic is independent and in most cases does not have to take care of all updating and maintaining issues concerning the data.

3.2.2 Integration Layer

Since the life-scientists require up-to-date information and, in addition, the information is highly distributed, the integration is based not on data replication but on dynamical information retrieval. In this case, the integrative methodology of resource wrapping followed in the federated database systems is adopted (compare *Federated Database Management Systems* in *section 2.3.2*). The integration layer, as the name states, is responsible for the integration of all resources available in the information resource layer. In order to provide information integration for each referenced resource a single component, called in general, *Resource Wrapper* has to be developed. Each *Resource Wrapper* is in charge of not only establishing and maintaining a connection to the underlying resource but also of executing syntax specific queries and forwarding results to the upper layer or further components within the same layer. All *Resource Wrapper* components implement the same interface to ensure a seamless communication across the components in the integration layer[4].

The federated integrative approach has been followed, since most of the available information resources provide already fast querying mechanisms. Thus, the predefined queries can be executed quickly by the corresponding resource wrappers during

[4] The precise functionalities of the components within the integration layer are described in *section 3.4.1* and represented in *Figure 3-11*.

runtime. For instance, by invoking the NCBI Web Service within the developed *NCBI Resource Wrapper*, databases such as the Entrez Gene database can be searched fast, since the web service uses the already indexed data by the internal supporting DBMS. For information resources, which are available as flat-files or do not provide such fast querying mechanisms, additional processing steps are required. For example, small ontologies such as the FunCat catalogue can be loaded directly into the corresponding resource wrapper components.

In the information resource layer there is another component called *Resource Manager* that conducts the communication between the components within the upper *Syntax Layer* and the different *Resource Wrappers*. It receives requests and distributes them among the proper *Resource Wrapper* components (see *Figure 3-5*). In case that a new information resource has to be integrated into the system, a corresponding new *Resource Wrapper* component has to be developed and registered at the *Resource Manager* by just modifying configuration files. This integration approach allows a very flexible and highly extendable way of information integration, whereby each *Resource Wrapper* component takes care of all resource specific access procedures. Furthermore, the integration components can be used also outside the GeKnowME system for other purposes.

3.2.3 Syntax Layer

Basically every topic map model is composed of topic types and association types. The main idea behind the syntax tier is that these topic types and association types can be mapped to single software components as shown in *Figure 3-5*. Thus, the syntax tier consists of two general component types: *Topic Types* and *Association Types*, providing defined interfaces for overall functionality. Each component in the syntax layer is aware where information about the subject or correspondingly the relation between the subjects can be found. This is achieved by describing the mapping information in configuration files in the syntax tier. For instance, each instance of the topic type *Gene* has an occurrence *DNA sequence* (compare *Figure 2-18*). The implementing component of this topic type is called *Gene Topic Type* and is configured in such way that it is aware that the information regarding this occurrence is retrievable from the Entrez Gene database by using the *NCBI Resource Wrapper*. Respectively, the component implementing the *Gene-Disease Effect Association Type* is set up to gather the gene-disease associations by searching the OMIM database through the *NCBI Resource Wrapper.*

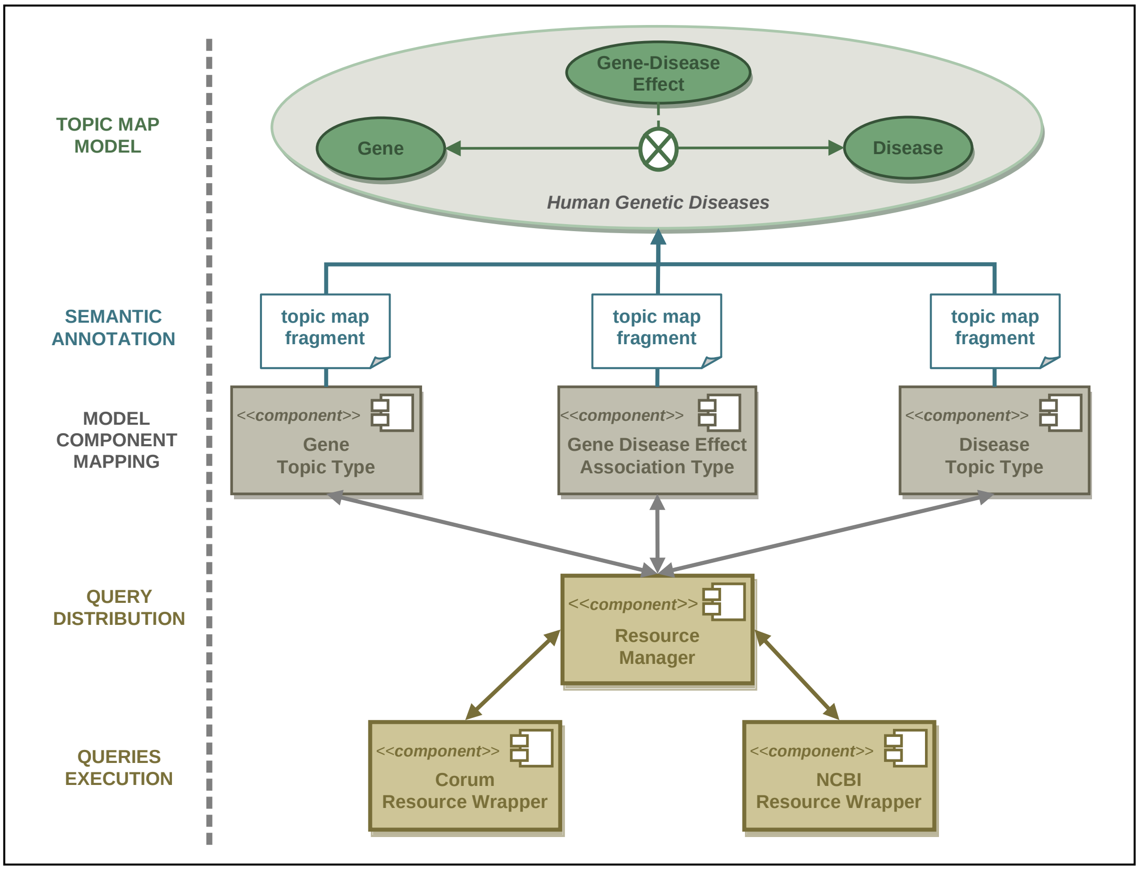

TOPIC MAP MODEL
SEMANTIC ANNOTATION
MODEL COMPONENT MAPPING
QUERY DISTRIBUTION
QUERIES EXECUTION
Gene-Disease Effect
Gene
Disease
Human Genetic Diseases
topic map fragment
topic map fragment
topic map fragment
<<component>> Gene Topic Type
<<component>> Gene Disease Effect Association Type
<<component>> Disease Topic Type
<<component>> Resource Manager
<<component>> Corum Resource Wrapper
<<component>> NCBI Resource Wrapper

Figure 3-5: Integrative logic including components involved in the integration, syntax, and semantic layers. Defined topic map models are mapped to software components of two types *Topic Type* and *Association Type*. They communicate with a *Resource Manager* that is responsible for the query distribution among the particular *Resource Wrappers*. They execute the specific resource queries.

When a request reaches a component in the syntax layer, it is forwarded to the Resource Manager with the notification which resources have to be queried, since each syntax component is aware of the related resources to the subject it represents. Then each *Topic Type* component (e.g. *Gene Topic Type*) integrates the retrieved information, such as existing names, identifiers, internal or external occurrences, and transforms it into a single topic map fragment (syntax conversion), with the advantage that this information is semantically described. Correspondingly, *Association Type* components provide the necessary semantic information about relationships between subjects (e.g. *Gene Disease Effect Association Type*). This approach allows on-the-fly semantic annotation for the desired entities and their relations. Depending on the defined topic map models, as many as needed *Topic Type* and *Association Type* components can be developed and flexibly reused.

3.2.4 Semantic Layer

The main purpose of the *Semantic Layer* is to organize the knowledge domain models defined by the different scientific communities and describes their semantics. A component called *Semantic Manager* is responsible for the mapping between a topic map model and the corresponding *Topic Type* and *Association Type* components of the *Syntax Lay*er. This mapping is again modifiable in configuration files. For instance, the *Semantic Manager* is aware of the simple topic map model *"Human Genetic Diseases"* represented in *Figure 3-5* and manages the implementing syntax components, in this case two topic types and one association type. It has a "dispatcher" role during the composition of a semantic network. In addition, the semantic manager is able to identify whether different topic map models share same concepts. This feature is beneficial, if a user needs to expand the knowledge domain of research by including further topic map models (compare *Figure 3-3*).

Within this tier, a further component called *Query Manager* is responsible for the generation of subject-centric queries, which are forwarded to the proper syntax components. An extended functionality of this component is the definition and execution of inference rules. Additionally, the component *Topic Map Assembly* is responsible for the final assembly of the topic map fragments delivered by the invoked syntax components. It applies strict predefined merging rules for the construction of a valid topic map containing the union of all available topics and associations connected directly to the explored subject. It generates the resulting semantic networks[5].

The integrative logic represented within the system architecture of the GeKnowME framework, including the semantic, syntax, and integration layers, follows the associative human way of thinking by reflecting knowledge domains into topic map models. These models are composed of subject-oriented components and apply the subject-centric computing, since multiple resources may be referenced to a single subject. Additionally, the dynamic semantic annotation allows the seamless combination of partially overlapping life-science domains to acquire a broader unified overview.

[5] More detailed description of these components is given in the developmental view of the GeKnowME system in *section 3.2.4* and their interactions are represented in *Figure 3-14.*

3.2.5 Presentation Layer

The presentation logic of the GeKnowME framework is adapted to a web-based graphical user interface (GUI) viewed within a web-browser program. A *Web-Portal* technology is chosen to fulfill the user requirements for personalization (i.e. adjustment of the GUI based on user attributes such as community, functional area, or role) and customization (i.e. modification of the GUI by specifying what content should be displayed). The GUI within the portal is composed of web components called *portlets* that process requests and generate dynamic content (see *Figure 3-4*). Scientists can interact with the system by querying for subjects within the defined knowledge domains and intuitively navigating through the related information within the generated semantic networks[6]. In general, the GeKnowME software components deliver XML documents by default, whereas the components within the syntax and semantic tier work with valid XTM documents. Therefore, within the presentation layer an XML output is also provided.

[6] A precise overview of all available portlets in the presentation tier is given in *section 3.2.5*.

3.3 Physical View

The physical view shows how the design of the system architecture, as defined in the preceding logical view, is brought to life as a set of real-world entities. Its main purpose is to describe how the abstract parts map into the running system. Carefully selected technologies have been used for the implementation of the designed concepts. The most important ones are briefly introduced to show which role they play in the physical overview of the framework.

3.3.1 Technologies Used

Mainly *open-source* technologies have been chosen for the realization of the system architecture. In the area of information integration and web representation, they conform to the state-of-the-art approaches and provide efficient and fast development solutions for production of reliable and qualitative software. Usually, open-source technologies outstand with rather innovative methodologies, since they are the product of collaboration among a large number of different software development communities. Moreover, open-source technologies have been broadly established and successfully applied in both industry and academia for the past decade.

Additionally, most of the applied technologies are defined as *standards* or are informally considered to be open standards by representing recognized specifications. In the context of software development, standards often arise for the reason that universally agreed sets of guidelines for interoperability are needed for better interaction among different participants and for more efficient and qualitative information exchange and representation.

Middleware technologies benefit from such software standards. In general, middleware represents a software system that allows the communication between distributed software components across a network. Therefore, middleware technologies use clearly defined *interfaces* to hide the complexity of the involved application. Software *components* implement the declared interfaces and thus represent a higher level of abstraction than common classes and objects. Established examples for application of *component-oriented* middleware technologies are the *Java Platform Enterprise Edition* (Java EE, formerly J2EE or Java 2nd Platform Enterprise Edition) (Sun Microsystems, Inc., 2006) and Microsoft's *.NET Framework* (Microsoft Corporation, 2008). An essential requirement of the component-oriented approach is that components have to be written in the same programming language. Systems designed in *Service Oriented Architecture* (SOA) style overcome this obstacle, since they are based on Web Services and WS are considered as platform and program language independent

(Channabasavaiah, et al., 2003). The WS technologies have evolved tremendously in the past five years. Currently, there are straightforward mechanisms for extending software components to WS, which is very recommendable.

Java Platform Enterprise Edition

The GeKnowME framework is built on the platform Java EE in the current version 5.0. The Java EE platform provides a set of open-source technologies that support software programmers to develop, deploy, and manage multi-tier Java software, based largely on modular components running on an application server. Therefore, it suits the concepts designed for the implementation of the GeKnowME system.

In general, an *application server* is a software engine that delivers applications to client computers. For instance a Java EE application server delivers Java EE applications to the client as illustrated in *Figure 3-6*, whereas *Java EE Web Application* runs within a web browser program. A Java EE application server can handle diverse infrastructure tasks such as transaction processing, scalability, concurrency control, security, performance, or life-cycle management of the deployed components to the server. Thus, software developers can concentrate on the implementation of the required functionality of the components and not on the application infrastructure. A broad range of Java EE application servers are available; both commercial products such as *BEA WebLogic Server* and *Oracle Application Server*, and non-commercial ones such as *Red Hat JBoss* and *Apache Geronimo*. The GeKnowME components run under the open-source Java EE application server *GlassFish v2*, which is based on the commercial *Sun Java System Application Server*.

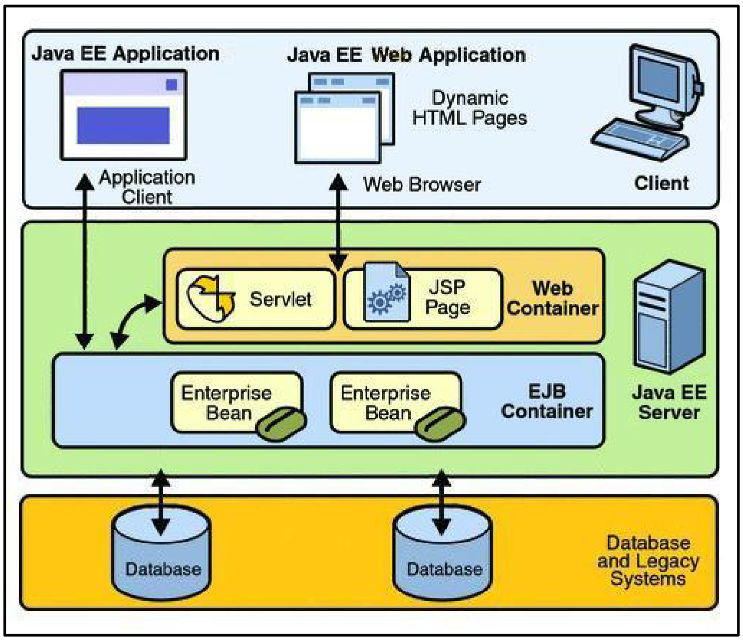


Figure 3-6 : Java Platform Enterprise Edition multi-tier architecture (Sun Microsystems, Inc., 2007).

Generally, Java platforms provide technological specifications called *Java APIs* (Application Programming Interfaces) to define interfaces and support interoperability. The Java EE platform comprises several APIs including those following the *EJB 3.0* specification. An *EJB* (Enterprise Java Bean) is a managed server-side component for modular construction of enterprise applications that encapsulates certain business logic. EJBs are deployed and run in a surrounding environment within a Java EE server called *EJB Container* (see *Figure 3-6*). An EJB container manages the execution of EJBs for Java EE applications at runtime. An EJB container holds two major types of beans: stateless or stateful *Session Beans* as well as *Message Driven Beans*. The components within the GeKnowME framework are mainly stateless session beans. In principle, stateless session beans are distributed objects that do not have state associated with them throughout the session and are less hardware-resource intensive.

The EJB 3.0 specification defines also mechanisms how EJBs are deployed to the EJB container. These mechanisms are defined by *Java annotations* or described in XML configuration files called *deployment descriptors*. Once EJBs are deployed on an application server, they can be accessed by local and remote client applications over the protocol *IIOP* (Internet Inter Object request broker Protocol), which is provided by the *Java RMI* (Remote Method Invocation) API. Usually the accession of components residing within a Java EE application server is accomplished through lookup services named *JNDI* (Java Naming and Directory Interface). Additionally, EJBs can expose easily their business methods as Web Services by using *the Java API for XML Web Services* (JAX-WS).

A Java EE application server can also provide another runtime environment called *Web Container* that manages the execution of web components (e.g. *JSP pages* or *Servlets*) for web applications (see *Figure 3-6*). Web applications are accessed by a request-response programming model. Web components, running within a web container, generate dynamic web pages formalized in various types of markup languages (e.g. HTML, or XML), when a request is received from a web client application. The generated content is then responded to the client via the *Hypertext Transfer Protocol* (HTTP), where a web browser renders the pages received from the server. A web application running under a Java EE application server can be very simple, or it can hide rather complex functionalities implemented by EJB components.

Both EJBs and web components are able to access legacy information systems or database systems via services provided by the Java EE application server. The *Java Database Connectivity* (JDBC) API allows Java EE application components to access and interact with the underlying resource managers of enterprise information systems via specific resource adapters. All common vendors of database systems in the field of life-science, such as *MySQL*, *PostgreSQL*, *Oracle*, *DB2*, *Microsoft SQL Server*, etc., provide such adapters that can be plugged in to any Java EE application server and used by application components. Additionally, the Java EE platform offers the *Java Persistence API* (JPA) as a Java standards-based solution for data persistence. JPA uses an object-relational mapping approach to bridge the gap between an object-oriented model and a relational database. It consists of three parts: interfaces, a query language, and object/relational mapping metadata.

Within the Java EE platform, additional APIs are specified to provide further services like security management, or message handling. However, since they are not relevant to the implementation of the GeKnowME framework, their description is not considered in this thesis. All above mentioned technologies as part of the Java EE platform have been applied as foundation for the implementation of the requested functionalities of the framework. For the more specific demands, further Java API specifications not included in the Java EE platform have been used.

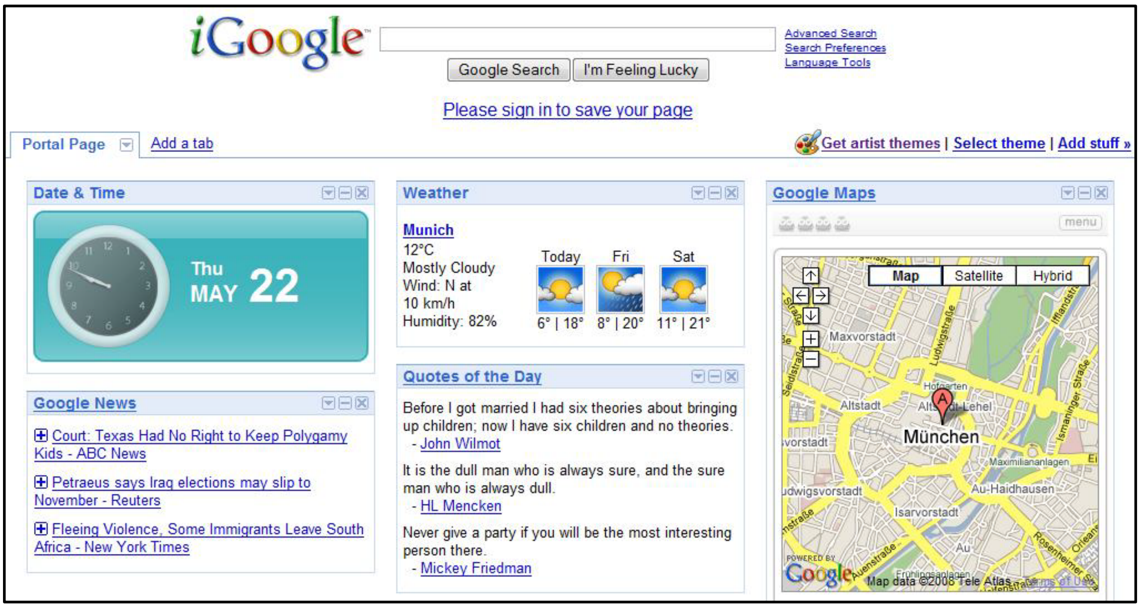
Topic Maps

TMAPI (Topic Map Application Programming Interface) is an open-source set of core and supplementary interfaces for accessing and manipulating data held in a topic map. The TMAPI specification is implemented by several programming communities. The *tinyTIM* implementation represents a small and lightweight in-memory Topic Maps engine providing methods for working and modifying topic maps with Java. The methods enable developers, for instance, to create topics, associations or occurrences, merge maps, modify identifiers, etc. Most of the components within the syntax layer of GeKnowME system have been built with the help of the tinyTIM implementation. Another TMAPI implementation *Topic Maps For Java* (TM4J) has been mainly used for the implementation of complex merging mechanisms necessary for the implementation of components within the semantic layer.

Portal and Portlets

An important feature of the GeKnowME framework is that different research communities are able to define their knowledge domain of interest and map it to carefully preselected information resources. Therefore, it is important to provide a user interface infrastructure that is capable of commonly managing these diverse communities and consistently offering adapted views for the defined topic map models. As already pointed out in the description of the GeKnowME presentation layer, the user interface of the GeKnowME system is built on the *Web-Portal* technology. In general, a portal is a web-based gateway for users to locate relevant content and use the applications they commonly need to be productive, in this case the application provided by the GeKnowME framework.

From a technical point of view, a web-portal is a web-based application running in a web container that resides within an application- or web-server. In addition, a portal manages and displays pluggable user interface components called *portlets*. It provides a runtime environment for portlets called a *portlet container*. Portlets produce fragments of markup code that are aggregated into a portal page. Typically, following the desktop representation, a portal page is displayed as a collection of non-overlapping windows, where each portlet window displays a portlet as illustrated in *Figure 3-7* showing a sample portal page of the *iGoogle* portal.



Munich
12°C
Mostly Cloudy
Wind: N at 10 km/h
Humidity: 82%

Court: Texas Had No Right to Keep Polygamy Kids - ABC News

Petraeus says Iraq elections may slip to November - Reuters

Fleeing Violence, Some Immigrants Leave South Africa - New York Times

Before I got married I had six theories about bringing up children; now I have six children and no theories.
- John Wilmot

It is the dull man who is always sure, and the sure man who is always dull.
- HL Mencken

Never give a party if you will be the most interesting person there.
- Mickey Friedman

Figure 3-7: *iGoogle* portal solution as example for providing adjustable pages containing multiple portlets.

Historically, different vendors created their own proprietary APIs for developing portlets, and runtime environments for executing them. The existence of different and incompatible APIs became a problem and a *Java Standardization Request-168* (JSR-168) was established as a standard for development and execution of portlets. With JSR-168, developers can implement portlets that can be deployed to any JSR-168 compliant container. Currently, several portal environments are designed to deploy portlets that adhere to the JSR-168 API. The most advanced open-source portal framework *Liferay Portal* is used for the GeKnowME presentation logic. Other frameworks include *JBoss Enterprise Portal Platform*, *Apache Pluto*, and the commercial *Oracle Portal* and *BEA's AquaLogic User Interaction*. Many useful portlets are bundled with the Liferay portal (document library, calendar, and message boards, to name a few). Furthermore, the portal provides elaborate techniques for personalization, customization, and workflow management of portal pages.

Rich Internet Applications

Portals facilitate the aggregation of content in an integrated user interface. However, portlets are typically rendered in HTML and thus inherit the restrictions of HTML for building applications. HTML-based applications have been limited by their static page-orientation, where the processing is performed on the server and a client browser is only used to display static content. Each step requires a round-trip to the server to advance the application state. This synchronized communication keeps the browser operating in lockstep with the server. *Rich Internet Applications* (RIA) advance this design by adding a data cache to the browser, allowing it to maintain its own sense of state and operate as independent client. RIAs introduce an intermediate layer code, called *client engine* or *RIA platform*, between the user and the server. The client engine acts as a browser extension, which takes over responsibility for rendering the application's user interface and for server communication. The enrichment of a browser with such a RIA platform does not force an application to depart from the normal synchronous pattern of interaction with the server; in most cases it just performs an additional asynchronous communication with the server.

Basically, RIAs are web applications that have the features and functionality of traditional desktop applications, but the benefits of web applications. They offer a *richer* interface to users, since they include client services such as advanced windowing components, drag-and-drop services, vector based graphics, audio-video playback, etc. Additionally, RIAs are more *responsive* than HTML-based web applications, because there is no need to communicate constantly in synchronous way to the server. These features provide the needed functionality to meet the demands for exploring and modeling semantic networks within a web application. To improve the usefulness and usability of the GeKnowME framework, the user interface of portlets are rendered using the RIA approach.

Several vendors provide platforms to run RIAs; the most established RIA client engine is *Adobe Flash* that is available in almost all common browsers. Adobe offers also a framework called *Adobe Flex* for development of RIAs. Another common platform is *Dynamic HTML* (DHTML) for which both open-source and commercial frameworks have been developed, also known as *Ajax Frameworks*. Generally, *Ajax* refers to the combination of techniques such as JavaScript and XHTML that can be applied to develop RIAs. The Ajax-based framework *Google Web Toolkit* has been successfully used for projects such as *Gmail* and *Google Maps*. *Microsoft Silverlight* is a further platform for execution of RIA developed with *.NET* framework, but still not very wide-spread.

OpenLaszlo is one of the very few open-source frameworks for development of RIAs that is capable of compiling from the same source code into the two most common runtime platforms Flash and DHTML. OpenLaszlo applications are written in *LZX* source code and run under *OpenLaszlo Compiler* executed within a Java EE web container. Therefore, OpenLaszlo is easily adaptable to the preselected GlassFish Java EE application server in combination with Liferay portal and used for the construction of the GeKnowME rich web application. A precise illustration of how all above described technologies are applied in the system physical design is given in the following section.

3.3.2 System Physical Overview

The GeKnowME framework is built on the Java EE platform following the component-oriented approach. The comprising components run under a GlassFish v2 application server as shown in *Figure 3-8*. The components representing the integrative logic (integration, syntax, and semantic layers) are implemented as EJBs and executable under the GlassFish EJB 3.0 container. In contrast to the UML diagram depicted in *Figure 3-8*, it is not necessary that all EJBs run under the same physical server. The integrative logic can be distributed among several Java EE application servers, since EJBs implement predefined interfaces and encapsulate particular business logic. Therefore, high flexibility and load balancing is achievable in the physical architecture. All available EJBs within the GeKnowME system are easily extendable to Web Services by applying Java annotation and thus accessible also by other programming languages besides Java. Additionally, the EJBs are adjustable to different database resources by using further configuration files such as *resource.xml* and *persistence.xml* describing specific configuration parameters to establish connection to databases or to map Java objects to rational schemata.

The GeKnowME EJBs communicate with other EJBs running under different Java EE application servers over the protocol RMI-IIOP. For example, a *Resource Wrapper* can access the SIMAP (Similarity Matrix of Proteins) database, which contains pre-computed homologies of over than 6 million protein sequences, over public accessible EJB interfaces. The GeKnowME EJB components are also able to communicate with Web Services available on the Web, such as the NCBI Utils, via HTTP. Different kinds of database management systems are directly accessible over JDBC (e.g. the textmining database EXCERBT running under a PostgreSQL DB server).

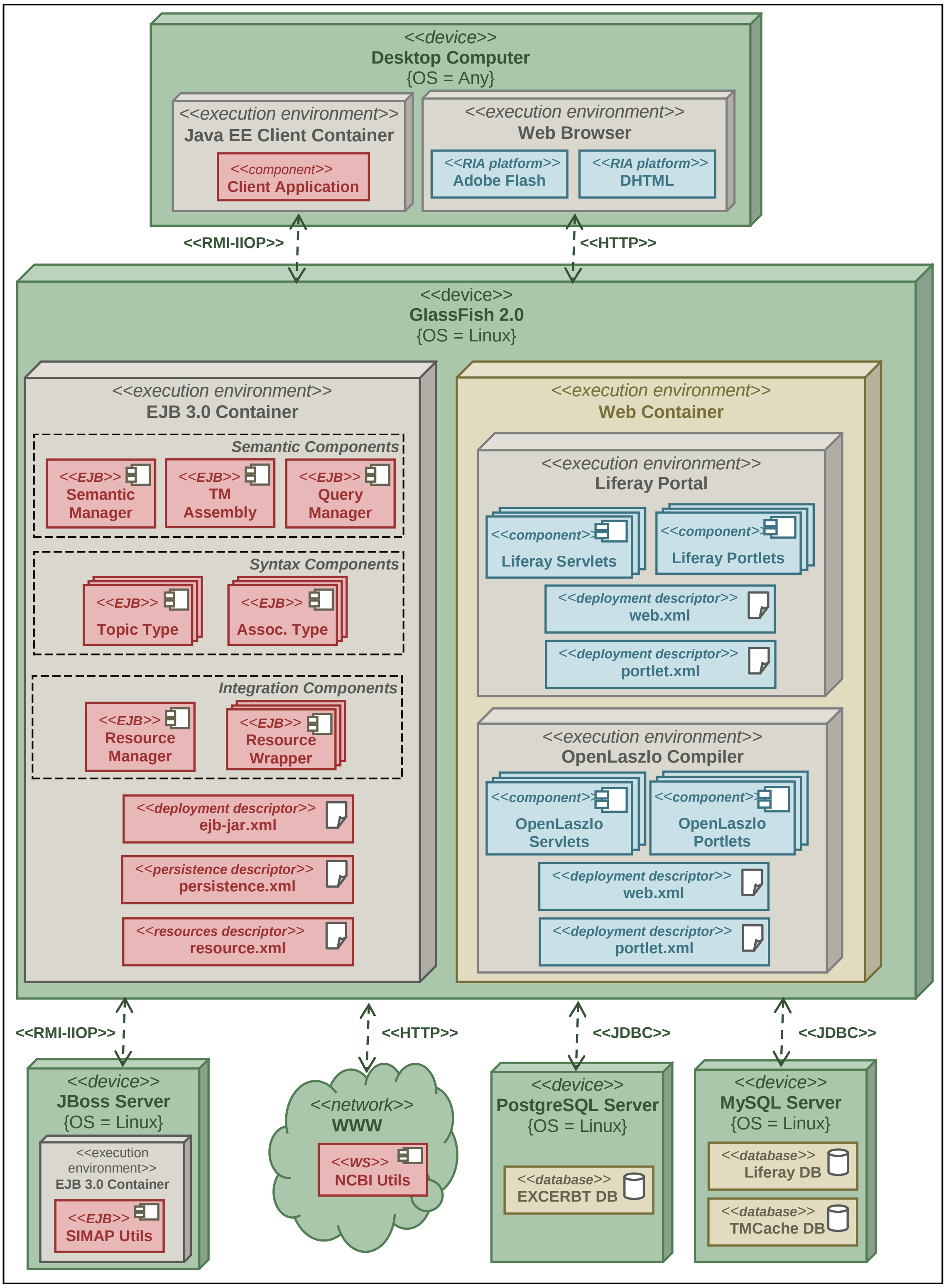


Figure 3-8: Detailed overview of the GeKnowME physical design.

As already mentioned, the presentation logic is also based on components, whereas the components implementing the user interface are called portlets. Both the bundled Liferay portlets and the developed GeKnowME portlets are managed by servlets provided by the Liferay portal, which runs within the Web Container of the GlassFish server. Specific functionalities and features of the portlets and generally of the portal are configured by parameters defined in description files such as *portlet.xml* and *web.xml*. The Liferay portal persists some of the configured parameters and further user adjustments such as layout or portlet arrangement data into a rational database. The Liferay database runs under *MySQL Server v5.1* and is accessed directly from the Liferay servlets.

The content of the GeKnowME portlets is rendered in a RIA style and the RIA web components are managed by OpenLaszlo servlets. They are executed in the OpenLaszlo Compiler environment running also within the web container of the GlassFish server. The RIA web components are embedded into portlets in order to be adaptable to the portal solution and registered at the Liferay portal during the deployment process. The generated biological models within the RIA application can be persisted into a MySQL rational database called *TMCache* to assure faster access to the results delivered during the exploration and navigation of the researched knowledge domain.

The accession of GeKnowME components residing within the same physical GlassFish server is accomplished through dependency injection. Since the components run under the same application server, the Java EE container handles automatically the complexities of component instantiation and initialization when this is required. In case that the components are distributed among several servers, JNDI lookup services are used to generate an instance of the desired component.

The JNDI service allows also client applications running within a Java EE Client Container to discover and lookup GeKnowME components via their declared names. In this case, the client applications are mainly focused on generation of XML representation of semantic networks describing interrelations between biological entities. In this type of clients, the communications to the server components is realized over the protocol RMI-IIOP. In contrast, when the client application runs within a web browser, the application is accessed over a common URL and the HTTP transfer protocol is used for the client-server interactions. Since the GeKnowME web application is RIA-based, either an Adobe Flash or a DHTML RIA platform is required for the rendering of the RIA portlets.

Next section illustrates the developed GeKnowME components, both EJBs and portlets, running within the GlassFish server. Primarily, the components functionalities are discussed with respect to the framework design.

3.4 Developmental View

Generally, in the process of application development, specific programming problems occur over and over again. For the past 15 years, diverse design patterns have evolved as simple and elegant solutions to such problems (Gamma, 1995). Patterns capture these solutions in a succinct and easily applied form independently from the chosen programming language. By using design patterns, software engineers can be sure to use solutions that have been applied in a large number of test cases and have been verified to meet broad range of demands. During the development of the GeKnowME framework, several design patterns have been applied to provide system's flexibility and reusability. As already mentioned in the description of the system's logic view, the layering pattern has been adopted to the GeKnowME architectural design to reduce the functional complexity.

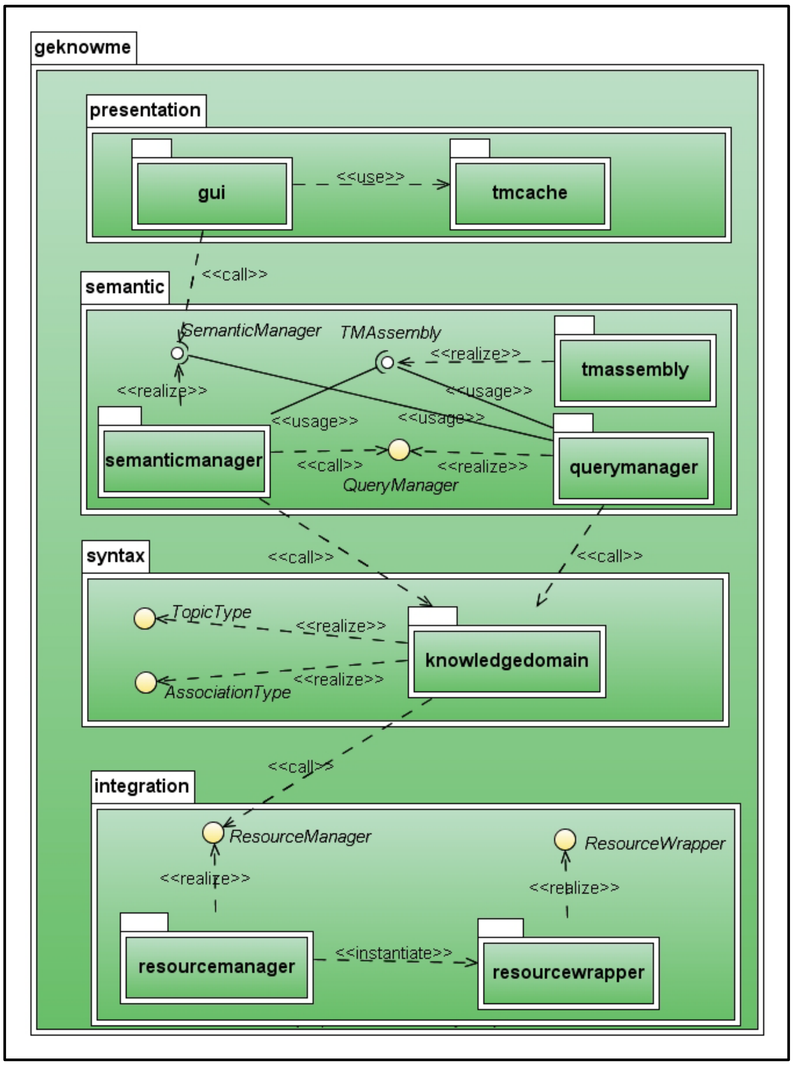


Figure 3-9: Package diagram of the GeKnowME framework showing the dependencies among the composing packages, which structure the developed classes and interfaces. Circles represent interfaces and dashed arrows show dependencies with the annotated roles.

In the development process of the GeKnowME system, the logical layers have been mapped to programming packages. They impose structure into the developed classes implementing the predefined requirements. The package diagram shown in *Figure 3-9* illustrates this mapping and gives additionally an overview of the dependencies among the packages by representing the corresponding interfaces. It is important to present these dependencies; since the overall functionality and stability of the system relies on them (e.g. a package can lose its functionality, if another package on which it depends changes). Similar to the explanation of the system's logical view, the four major `geknowme` packages (`geknowme.integration`, `geknowme.syntax`, `geknowme.semantic`, and `geknowme.presentation`) and correspondingly their interactions are discussed in such order that the packages offering operations for classes in the upper layers come first.

3.4.1 Integration Package

The `geknowme.integration` package encapsulates the procedures needed for the dynamic retrieval of information from distributed resources. The package offers its functionality over the `ResourceManager` interface (compare the <<*call*>> dependency between the `geknowme.integration` and `geknowme.syntax` packages represented in *Figure 3-9*). However, the integration functionalities are actually performed by components representing resource wrappers. All classes responsible for the integration of one information resource are organized in a single package. Each such `geknowme.integration.resourcewrapper` package realizes the same `ResourceWrapper` interface (see *Figure 3-9*). This approach allows a fluent communication with the diverse information resources by applying the same procedural mechanisms and can be utilized by the implementation of the `ResourceManager` interface.

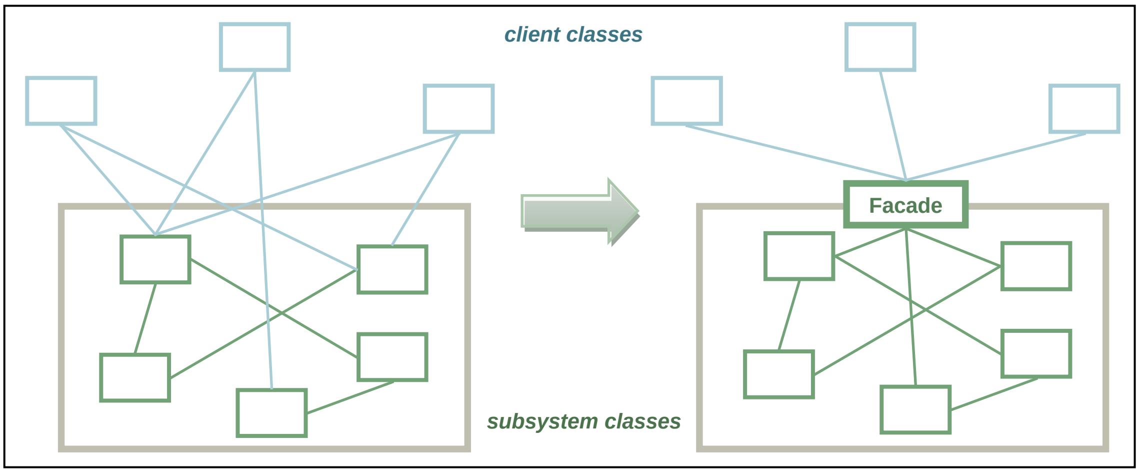


Figure 3-10: Overview of the structural design pattern *Facade* that provides a single simplified interface to the more general facilities of a subsystem to reduce communication overload and dependencies (Gamma, 1995).

The intention of the ResourceManager interface is to provide a unified interface for the different resource wrappers implemented within the integration package. The structuring of the GeKnowME system into layers or subsystems helps to reduce the functional complexity. However, a common development recommendation is to minimize the communication and dependencies between subsystems by introducing a *facade* object (see *Figure 3-10*). In general, a facade defines a higher-level interface that makes the subsystem easier to use. In this case, the ResourceManager interface realizes the facade design pattern and provides a single, simplified interface to the high number of resource wrappers available in the integration layer. Thus, the operations defined in the ResourceManager interface are quite similar to the operations described in the ResourceWrapper interface as represented in the class diagram of the integration package shown in *Figure 3-11*.

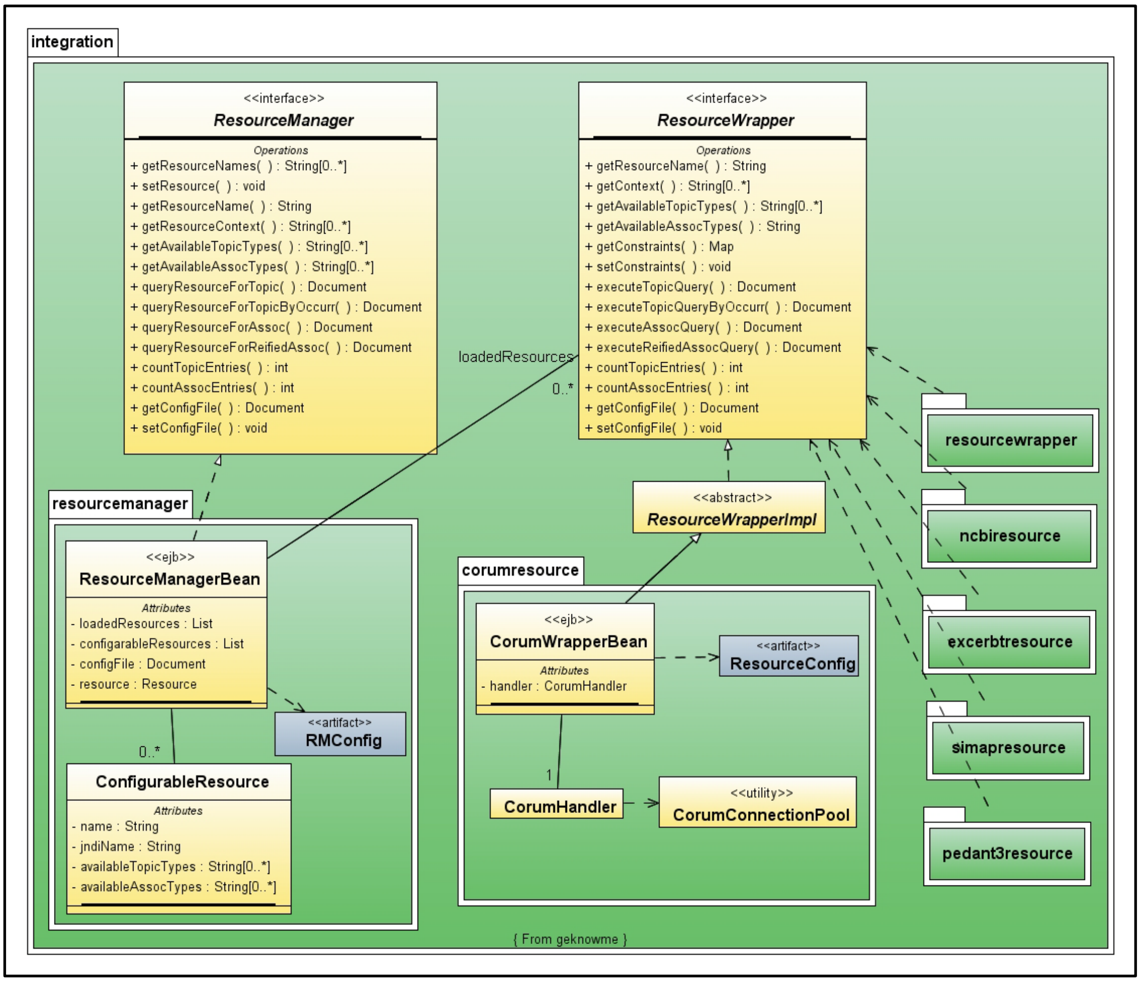

integration
<<interface>>
ResourceManager
Operations
+ getResourceNames( ) : String[0..*]
+ setResource( ) : void
+ getResourceName( ) : String
+ getResourceContext( ) : String[0..*]
+ getAvailableTopicTypes( ) : String[0..*]
+ getAvailableAssocTypes( ) : String[0..*]
+ queryResourceForTopic( ) : Document
+ queryResourceForTopicByOccurr( ) : Document
+ queryResourceForAssoc( ) : Document
+ queryResourceForReifiedAssoc( ) : Document
+ countTopicEntries( ) : int
+ countAssocEntries( ) : int
+ getConfigFile( ) : Document
+ setConfigFile( ) : void
<<interface>>
ResourceWrapper
Operations
+ getResourceName( ) : String
+ getContext( ) : String[0..*]
+ getAvailableTopicTypes( ) : String[0..*]
+ getAvailableAssocTypes( ) : String
+ getConstraints( ) : Map
+ setConstraints( ) : void
+ executeTopicQuery( ) : Document
+ executeTopicQueryByOccurr( ) : Document
+ executeAssocQuery( ) : Document
+ executeReifiedAssocQuery( ) : Document
+ countTopicEntries( ) : int
+ countAssocEntries( ) : int
+ getConfigFile( ) : Document
+ setConfigFile( ) : void
loadedResources
0..*
resourcewrapper
ncbiresource
excerbtresource
simapresource
pedant3resource
<<abstract>>
ResourceWrapperImpl
resourcemanager
<<ejb>>
ResourceManagerBean
Attributes
- loadedResources : List
- configarableResources : List
- configFile : Document
- resource : Resource
<<artifact>>
RMConfig
0..*
ConfigurableResource
Attributes
- name : String
- jndiName : String
- availableTopicTypes : String[0..*]
- availableAssocTypes : String[0..*]
corumresource
<<ejb>>
CorumWrapperBean
Attributes
- handler : CorumHandler
<<artifact>>
ResourceConfig
1
CorumHandler
<<utility>>
CorumConnectionPool
{ From geknowme }

Figure 3-11: Class diagram representing the involved participants within the `geknowme.integration` package. Classes are represented in different notions (omitting attributes or operations) to provide a clearer overview.

The `ResourceManagerBean` implements all operations defined in the ResourceManager interface to conduct the communication between the syntax components and all available resources. It contains a list of all loaded resource wrappers, which are implemented to provide the occurrence space for the defined knowledge domain models and is executable as an EJB. Additionally, a mechanism for the fast and easy registration and usage of new resource wrappers is implemented within the `geknowme.integration.resourcemanager` package. The description data needed to instantiate a `ConfigurabeResource` object, which can be used by the ResourceManager to load the corresponding new resource wrapper, can be appended to `RMConfig.xml` file. The approach allows simplified switching-on or switching-off of the available resource wrapper components.

As already mentioned, all classes responsible for the integration of one information resource are structured in a single package. For instance, the `geknowme.integration.corumresource` package encapsulates the procedures necessary for the dynamic retrieval of information available from the CORUM database as shown in *Figure 3-11*. All resource wrapper components have to realize the same `ResourceWrapper` interface to ensure the easy extendibility of the system. For reusability reasons, an abstract class `ResourceWrapperImpl` has been introduced to implement same recurring operations in the diverse resource wrappers. These operations are adjustable by defining specific parameters in a configuration file called `ResourceConfig.xml` for each integrated information resource. However, there are still several abstract operations, which have to be implemented by the concrete resource wrapper EJBs, e.g. `CorumWrapperBean`, depending on the more specific wrapper behavior.

For each resource wrapper, it is recommendable to provide a single global point of access to the real information resource that it integrates to guarantee high resource performance. The implementation of such a global accession point represents the usage of a further design pattern called *singleton*. For example, the class `CorumHandler` is implemented as a singleton and ensures that only one instance of the class is accessible at runtime. The handler class, as the name states, handles the queries to the resource and accesses resource utilities. A sample resource utility is the JDBC connection pool `CorumConnectionPool`, managed by the GlassFish Java EE container. It is used as a cache of CORUM database connections so that the connections can be reused when the requests for data are received and thus to enhance the performance of executing commands on the database.

In a similar manner, further resource wrappers can be implemented within the GeKnowME system. The packages depicted on the right in *Figure 3-11* represent some further information resources such as the PEDANT3, NCBI, EXCERBT, and SIMAP systems, which are available within GeKnowME framework. Depending on how the diverse information resources provide access to the underlying data, different types of utility classes are required. For example, in order to retrieve information from NCBI, a specific utility class has been developed to initialize and invoke the corresponding NCBI web services. Each separate resource wrapper is implemented as a single EJB component and thus can be run on a separate application server. To make a resource wrapper accessible within the GeKnowME system, it has to be just registered to a running ResourceManager.

Eventually, the application of the mentioned design patterns in combination with the configuration methods in the development of the resource wrapper components within the `geknowme.integration` package allows the possibility to reuse resource wrappers in many knowledge domain models. The GeKnowME integration layer is easily extendable and the maintenance efforts are kept low, since the separate resource wrappers are encapsulated and loosely coupled. The developed resource integration approach provides a solid foundation for the enhanced semantic information integration.

3.4.2 Syntax Package

The classes implementing the functionalities needed for the semantic information integration are split-up into two logical layers and respectively organized in two single java packages `geknowme.syntax` and `geknowme.semantic` (compare *Figure 3-9*). As already introduced and illustrated in *Figure 3-5*, the defined knowledge domains are mapped to composing topic type and association type components, structured within the syntax package. They all realize either the `TopicType` or the `AssociationType` interface, which operations are represented in the class diagram of the syntax package in *Figure 3-12*.

Since different scientific communities define their own knowledge domain of interest, it is recommendable that all components mapped to a single model are organized in a separate package. For instance, the implementation of the models represented in *Figure 2-19* can be structured to corresponding packages such as `geknowme.syntax` `.diseasemodel`, `.functionalmodel`, `.geknomemodel`, and `.textminingmodel`. This type of encapsulation allows the deployment of separate knowledge domains to different GlassFish application servers and increases the system flexibility and simplifies the software maintenance. This type of distribution follows somehow the Peer Data Management integration approach (compare *section 2.3.2*). Each knowledge domain package can be also arranged into two sub-packages `topictypes` and `associationtypes`, containing correspondingly the implementation of the topic type and association type components.

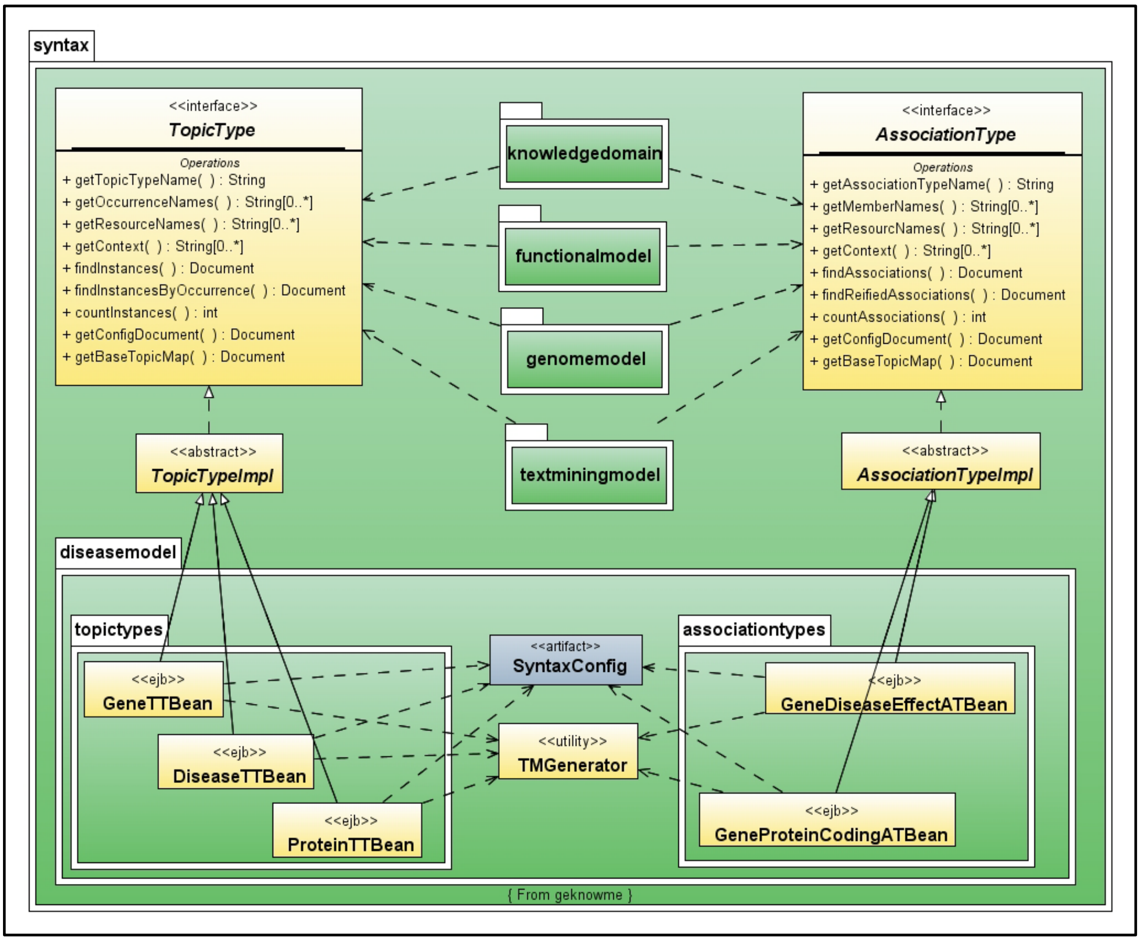

syntax
<<interface>>
TopicType
Operations
+ getTopicTypeName( ) : String
+ getOccurrenceNames( ) : String[0..*]
+ getResourceNames( ) : String[0..*]
+ getContext( ) : String[0..*]
+ findInstances( ) : Document
+ findInstancesByOccurrence( ) : Document
+ countInstances( ) : int
+ getConfigDocument( ) : Document
+ getBaseTopicMap( ) : Document
knowledgedomain
functionalmodel
genomemodel
textminingmodel
<<interface>>
AssociationType
Operations
+ getAssociationTypeName( ) : String
+ getMemberNames( ) : String[0..*]
+ getResourcNames( ) : String[0..*]
+ getContext( ) : String[0..*]
+ findAssociations( ) : Document
+ findReifiedAssociations( ) : Document
+ countAssociations( ) : int
+ getConfigDocument( ) : Document
+ getBaseTopicMap( ) : Document
<<abstract>>
TopicTypeImpl
<<abstract>>
AssociationTypeImpl
diseasemodel
topictypes
<<ejb>>
GeneTTBean
<<ejb>>
DiseaseTTBean
<<ejb>>
ProteinTTBean
<<artifact>>
SyntaxConfig
<<utility>>
TMGenerator
associationtypes
<<ejb>>
GeneDiseaseEffectATBean
<<ejb>>
GeneProteinCodingATBean
{ From geknowme }

Figure 3-12: Class diagram representing the involved participants within the `geknowme.syntax` package.

The main procedures performed by the syntax components are to parse the resulting XML documents coming from the diverse resource wrappers and to add the proper semantic annotation to the information (transform the syntax). Since there is a particular structure in the parsing mechanism, a behavioral design pattern called *template method* has been applied to define a skeleton of parsing steps, which are implemented in the two abstract classes `TopicTypeImpl` and `AssociationTypeImpl`. Each concrete topic type or association type EJB implementation such as `GeneTTBean` and `GeneDiseaseEffectATBean` redefine certain steps of the parsing procedure according to the specific needs without changing its structure. In order to be able to generalize such kind of parsing procedural structure, a configuration mechanism has been introduced in any `geknowme.syntax.knoweledgedomain` package including a utility class called `TMGenerator` and an XML file `SyntaxConfig`, which schema is depicted in *Figure 3-13*.

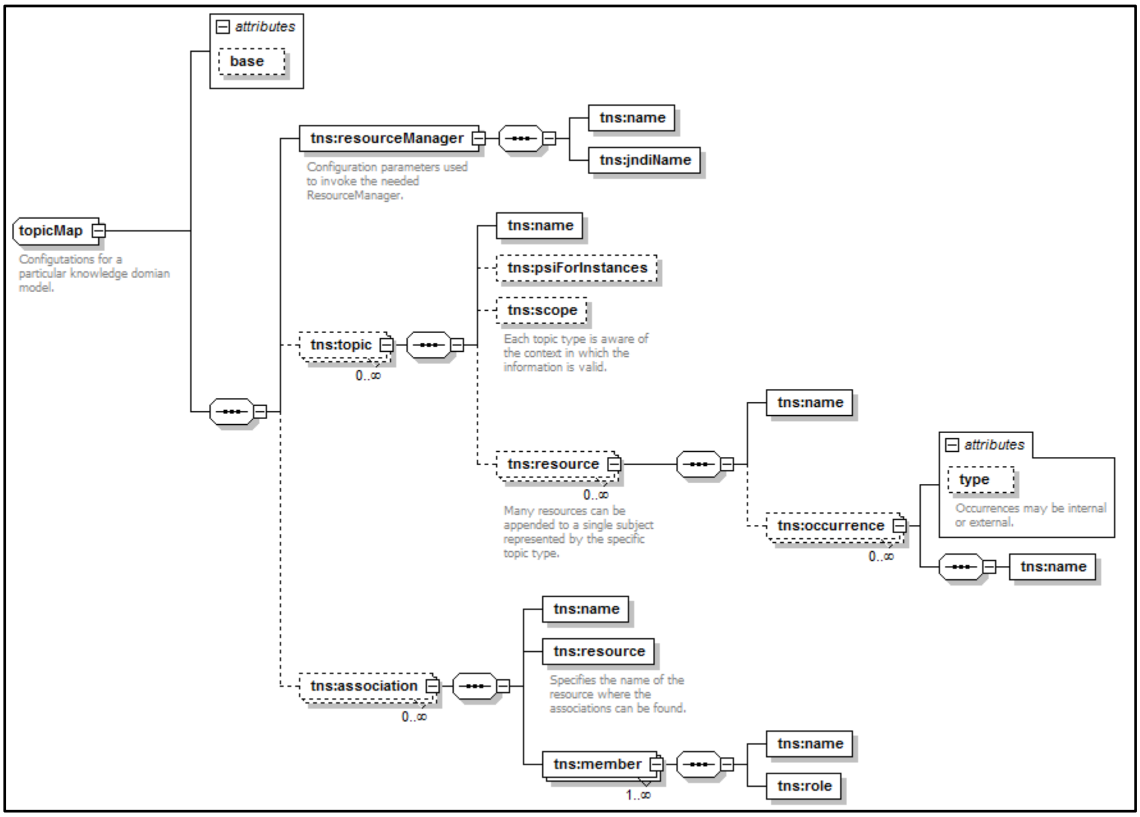

attributes
base
topicMap
Configutations for a particular knowledge domian model.
tns:resourceManager
Configuration parameters used to invoke the needed ResourceManager.
tns:name
tns:jndiName
tns:topic
0..∞
tns:name
tns:psiForInstances
tns:scope
Each topic type is aware of the context in which the information is valid.
tns:resource
0..∞
Many resources can be appended to a single subject represented by the specific topic type.
tns:name
tns:occurrence
0..∞
attributes
type
Occurrences may be internal or external.
tns:name
tns:association
0..∞
tns:name
tns:resource
Specifies the name of the resource where the associations can be found.
tns:member
1..∞
tns:name
tns:role

Figure 3-13: Graphical representation of the XML schema defining the format of the configuration files used in `geknowme.syntax.knowledgedomain` packages.

A `SyntaxConfig` file describes all topic type and association type components mapped to a particular knowledge domain. For each single component, configuration parameters are defined showing which resource wrappers deliver information about the subject or about the relationships between the subjects represented by the topic type or association type. However, the syntax components do not communicate directly with the single resource wrappers, instead only with a ResourceManager. The invocation parameters for this communication are also specified in the configuration file.

Each resource provides different type of information about the subject, thus for each linked resource a list of occurrence types is defined. Additionally, the members playing a part in an association type are described with their matching roles. This kind of information is used by the `TMGenerator` to construct for each syntax-EJB the so-called *base topic map* containing all involved typing topics (compare *Figure 2-18*). During the parsing of the retrieved results, these typing topics are used for the generation of the found topics and associations. At the end of each topic or association search, each syntax component delivers a topic map fragment in the form of an XTM document, containing all related information found about the search subjects or interrelations between subjects (compare *Figure 3-5*).

3.4.3 Semantic Package

The components within the `geknowme.semantic` package control the semantic information integration. The major task in this process is to manage the defined knowledge domain models and is performed by a component called semantic manager. The belonging interface `SemanticManager` with the public offered operations is presented in the class diagram of the semantic package (see *Figure 3-14*). Its realization with all involved classes is organized in the `geknowme.semantic. semanticmanager` package.

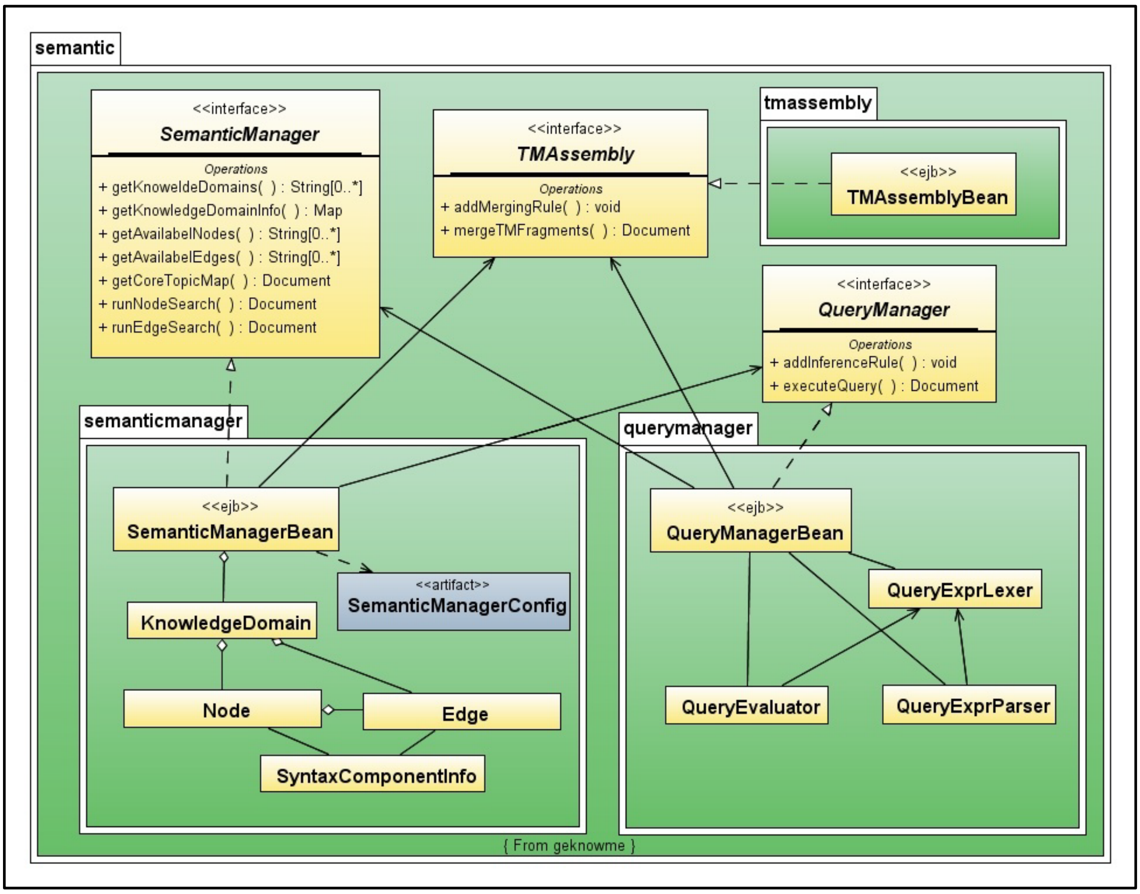


Figure 3-14: Class diagram depiction of the components within the `geknowme.semantic` package.

Each knowledge domain model represents a network mapping the concepts a certain scientific community is interested in. Such a network is represented by an instance of a `KnowledgeDomain` class, which aggregates composing `Node` and `Edge` objects. These objects contain information about the implementation of the corresponding syntax components realizing the `geknowme.syntax.TopicType` and `geknowme.syntax.AssociationType` interfaces. The EJB implementation `SemanticManagerBean` controls a set of such predefined knowledge domains and plays a role as an enhanced facade object for the syntax components providing the semantic annotation. The mapping of the knowledge domain nodes and edges to the corresponding topic types and association types is configurable in a `SemanticManagerConfig` XML file (its schema is illustrated in *Figure 3-15*).

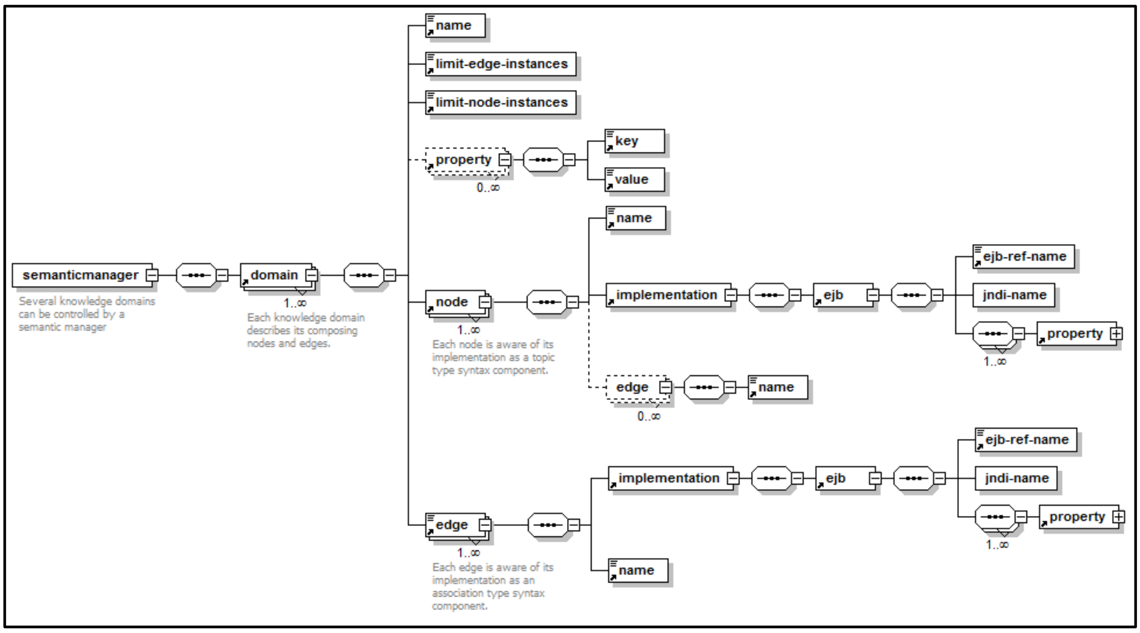

name
limit-edge-instances
limit-node-instances
property
0..∞
key
value
name
semanticmanager
Several knowledge domains can be controlled by a semantic manager
domain
1..∞
Each knowledge domain describes its composing nodes and edges.
ejb-ref-name
node
1..∞
Each node is aware of its implementation as a topic type syntax component.
implementation
ejb
jndi-name
property
1..∞
edge
0..∞
name
ejb-ref-name
implementation
ejb
jndi-name
edge
1..∞
Each edge is aware of its implementation as an association type syntax component.
property
1..∞
name

Figure 3-15: Graphical representation of the XML schema defining the format of the configuration file used by a `SemanticManagerBean` in the `geknowme.semantic.semanticmanager` package.

Another important task in the semantic integration process is to combine the results received from the syntax component in the form of topic map fragments. Since the retrieved information is already semantically described, it is easy to assemble it correctly to a complete topic map. For this purpose, the semantic manager uses the merging functionality provided by the `TMAssemblyBean`, which can be adjusted by defining specific merging rules. The TMAssembly component is encapsulated from the semantic manager, because the merging functionality can be utilized also by other components and thus helps to increase the system reusability.

Another advanced functionality available in the semantic package is offered by the *Query Manager* component. With the provided information about the structure of the different knowledge domains by the semantic manager, the `QueryManagerBean` is capable of building and executing expanded queries, which can infer new insights by finding indirect relations between subjects. This task is achievable by using the own designed and developed *Knowledge Querying Language (KQL)*, which is rather similar to the *Tolog*[7] querying language but adjusted to the dynamic semantic annotation implemented in the GeKnowME framework. The functionality of the KQL is

[7] *Tolog* is a querying language for static topic maps based on the Prolog programming language.

implemented by the QueryExprLexer, QueryExprParser, and QueryExprEval classes within the geknowme.semantic.querymanager package (see *Figure 3-14*).

At this stage, it is significant to point out the aspect of low coupling and high cohesion of the developed modules within the GeKnowME system implementing the integrative logic. For instance, the components within the geknowme.semantic package are aware of which syntax components are involved in the representation of a certain knowledge domain. Beneficially, they do not have to consider the diverse distributed information resources mapped to the knowledge domain. Thus, they do not have to be modified, if a new resource is plugged in to the GeKnowME system and provides additional occurrence space for the topic map model. In such case, only the related components within the corresponding geknowme.syntax.knowledgemodel package have to be adapted. In the same way, if something changes in an information resource such as its data structure or connection parameters, usually only the equivalent resource wrappers have to be modified and not the involved syntax components. Additionally, the functional cohesion is high, since the system is designed in a modular way and facade classes are imposed. Consequentially, the reusability of the developed modules is very high and the maintenance efforts are respectively low.

3.4.4 Presentation Package

The presentation package implements the final developmental steps in the GeKnowME system. The main tasks of the components within the geknowme.presentation package are to provide the necessary graphical user interface for the exploration of the semantically integrated life-science information space and thus it depends on the underlying semantic manager (see *Figure 3-9*). The GUI portlet components are organized in the geknowme.presentation.gui package and use the functionality provided by the *Cache Manager* component to persist the generated topic map models.

Currently, the GeKnowME portal is composed of three portlets: SearchForm, ResultView, and ModelCanvas as shown in *Figure 3-16*. All of them communicate with a semantic manager and a cache manager over a SessionFacade object. The communication between the client and the presentation components is bound to a session object and involves multiple messages in both directions. The results retrieved as a dynamically generated topic map representing a semantic network are mapped to Java objects of the classes Node and Edge, which are aggregated into a Graph. This data is kept for a certain period of time in the *TMCache* database (compare *Figure 3-8*) and if a session expires, by default it is thrown away. Logged-in users are allowed to store their exploration results.

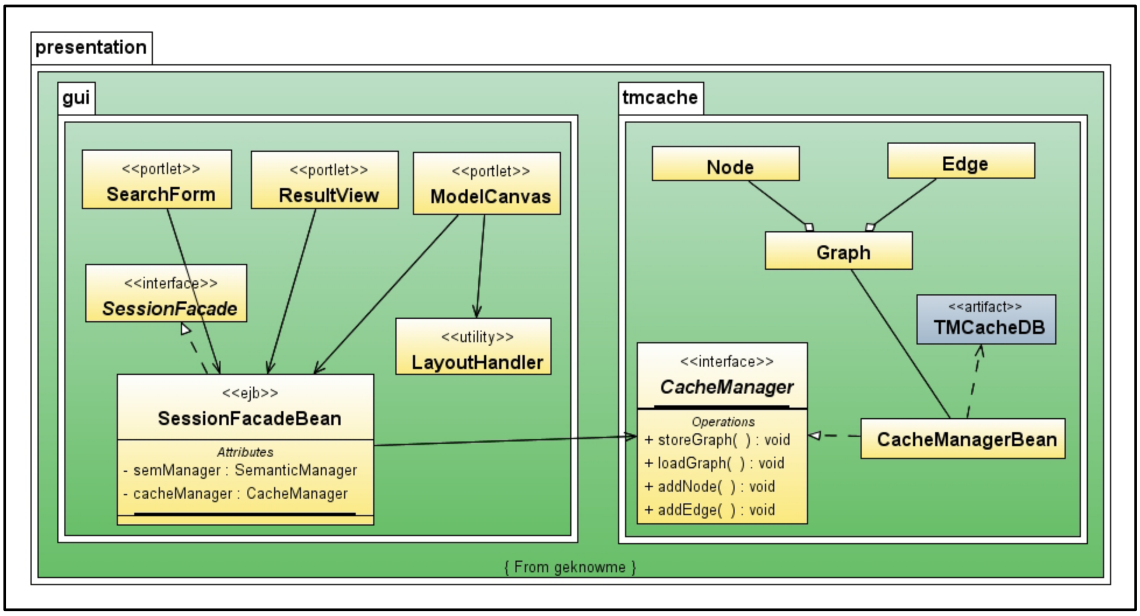
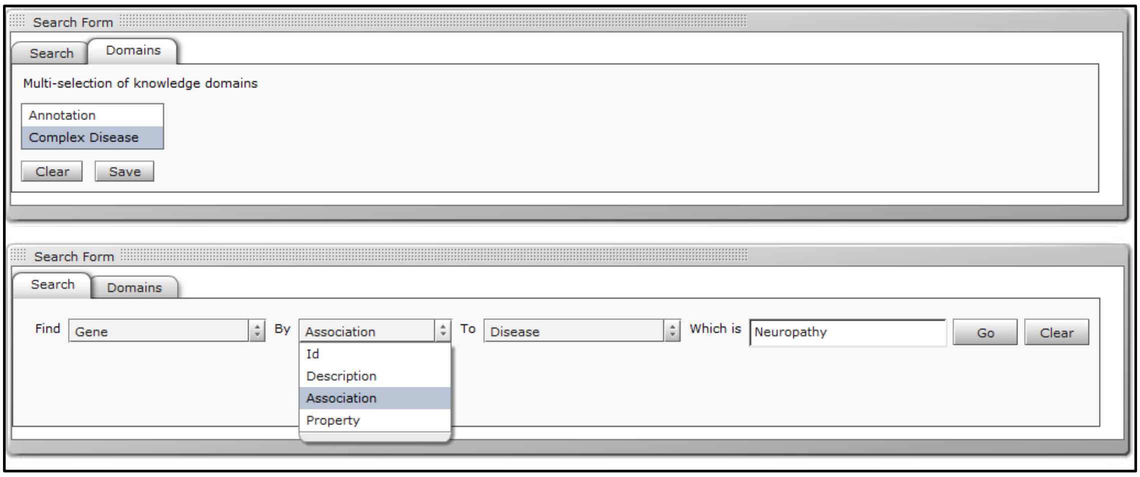


Figure 3-16: Class diagram of the involved components within the `geknowme.presentation` package

In the `SearchForm` portlet, shown in *Figure 3-17*, users can specify the search criteria for the exploration through the life-science information space. In the portlet's tab *Domains* the user can select the knowledge domains of his interest and thus to reduce the information complexity. If multiple knowledge domains are chosen and there are connection points between them, then the topic map models are combined to provide a broader subject matter search. Depending on to which community a logged-in user belongs, only the community related knowledge domains are selectable. For each knowledge domain the corresponding graphical representation of the model with the mapped information resources is obtainable.

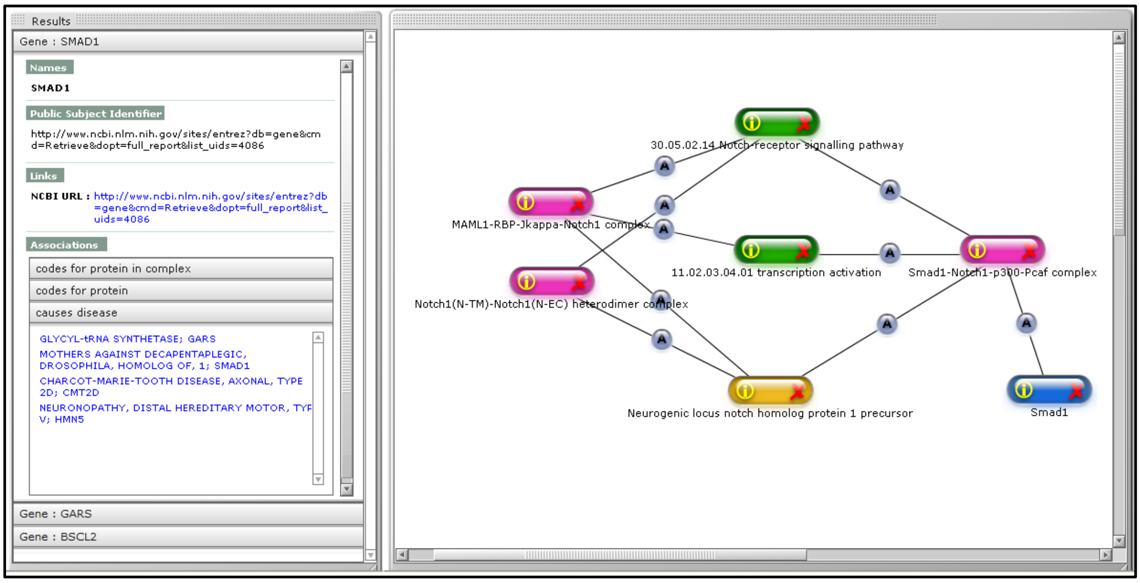
Once the desired knowledge domains are selected, the topic map models are loaded into the system and the involved topic types and association types are available in GUI, exactly in the *Search* tab of the `SearchForm` portlet (see *Figure 3-17*). The user can specify the type of the subject, he is interested in, and the further search criteria. It is possible to search for a certain topic (in this case a biological entity) by specifying its id, name or a property if known. Additionally, topics can be found, if the user specifies with which other entity they are associated (e.g. one can look for genes associated with a certain disease such as Neuropathy). Depending on which subject type has been selected, only the related topic types as defined in the knowledge domains are obtainable in the association-related exploration. The entered search criteria are bound to the client session and passed to the corresponding `SessionFacadeBean` object, which invokes the semantic manager component.



Find: Gene
By: Association
To: Disease
Which is: Neuropathy

Figure 3-17: Graphical representation of the GeKnowME *Search Form* portlet depicting the views of the both portlet's tabs under each other.

The found subjects fulfilling the entered search criteria are extracted from the generated topic map and listed in the `Results` portlet as represented in *Figure 3-18*. For the search defined criteria, three genes SMAD1, GARS, and BSCL2 have been found. Then the user can explore single entities by viewing their exact characteristics and associated entities. Usually, they have been extracted from distributed information resources and semantically annotated by the corresponding syntax components on-the-fly. As already mentioned, not only all found entities, but also their associated topics are temporary maintained during the exploration process. The user can navigate through the generated networks and hop from one entity to another related one and expand the semantic network. This is achievable, since all associations are described bidirectional.



Results

Gene : SMAD1

Names

SMAD1

Public Subject Identifier

http://www.ncbi.nlm.nih.gov/sites/entrez?db=gene&cmd=Retrieve&dopt=full_report&list_uids=4086

Links

NCBI URL : http://www.ncbi.nlm.nih.gov/sites/entrez?db=gene&cmd=Retrieve&dopt=full_report&list_uids=4086

Associations

codes for protein in complex

codes for protein

causes disease

GLYCYL-tRNA SYNTHETASE; GARS

MOTHERS AGAINST DECAPENTAPLEGIC, DROSOPHILA, HOMOLOG OF, 1; SMAD1

CHARCOT-MARIE-TOOTH DISEASE, AXONAL, TYPE 2D; CMT2D

NEURONOPATHY, DISTAL HEREDITARY MOTOR, TYP V; HMN5

Gene : GARS

Gene : BSCL2

30.05.02.14 Notch receptor signalling pathway
MAML1-RBP-Jkappa-Notch1 complex
11.02.03.04.01 transcription activation
Smad1-Notch1-p300-Pcaf complex
Notch1(N-TM)-Notch1(N-EC) heterodimer complex
Neurogenic locus notch homolog protein 1 precursor
Smad1
A

Figure 3-18: Graphical representation of the GeKnowME *Results* and *Model Canvas* portlets depicted next to each other. In the Model Canvas, the magenta icons represent protein complexes, the green ones biological functions, the blue ones genes, and the yellow icon is a protein.

If a scientist finds an essential entity (topic), he can place it on the `ModelCanvas` portlet and start building models as depicted in *Figure 3-18*. The topics are represented as icons, which have different colors depending on their topic types. If a further entity is laid on the model canvas and in the corresponding graph object there is a known relation between these two entities, then an interconnecting edge is drawn. The advantage of this approach is that the user can decide which entities are relevant for his research and include them in the graph representation. Since the `ModelCanvas` portlet is implemented using the RIA representation techniques, the user can organize the topics by drag and drop in a preferred way. The GeKnowME user is supported by powerful techniques such as navigation though dynamically generated semantic network in the `Results` portlet and building models including the entities of interest in the `ModelCanvas` portlet to discover hidden relationships between biological entities. During the exploration process, the user has a feeling of navigation though coherent information space, instead of single distributed information resources, which is provided by the subject-centric semantic integration.

In order to be able to achieve this kind of knowledge representation, it is necessary to describe what happens actually within the system. Therefore, an overview follows describing the interactions between all above discussed components, which have been developed to fulfill the system design.

3.5 Process View

The last view describing the system architecture is the *process view*, which focuses on the representation how the system accomplishes the required goals by applying the designed concepts and developed software components. As the name states, GeKnowME is a generic system and can be extended and adapted to any scientific knowledge domain of interest. Before the explorative features of the system can be used by the scientific communities, a defined developmental process has to be performed. As already pointed out in the GeKnowME use case view, there are two types of actors interacting with the framework: developers and scientists. Accordingly, two separate interaction processes from both points of view are considered.

3.5.1 Developmental Process

The description of the developmental process summarizes the main steps performed by a GeKnowME developer to make a knowledge domain model defined by a scientific community available for exploration. Most of the procedures concern primarily configuration or extension of existing software components within the GeKnowME framework. An overview of the involved developmental steps is represented as an activity diagram in *Figure 3-19*.

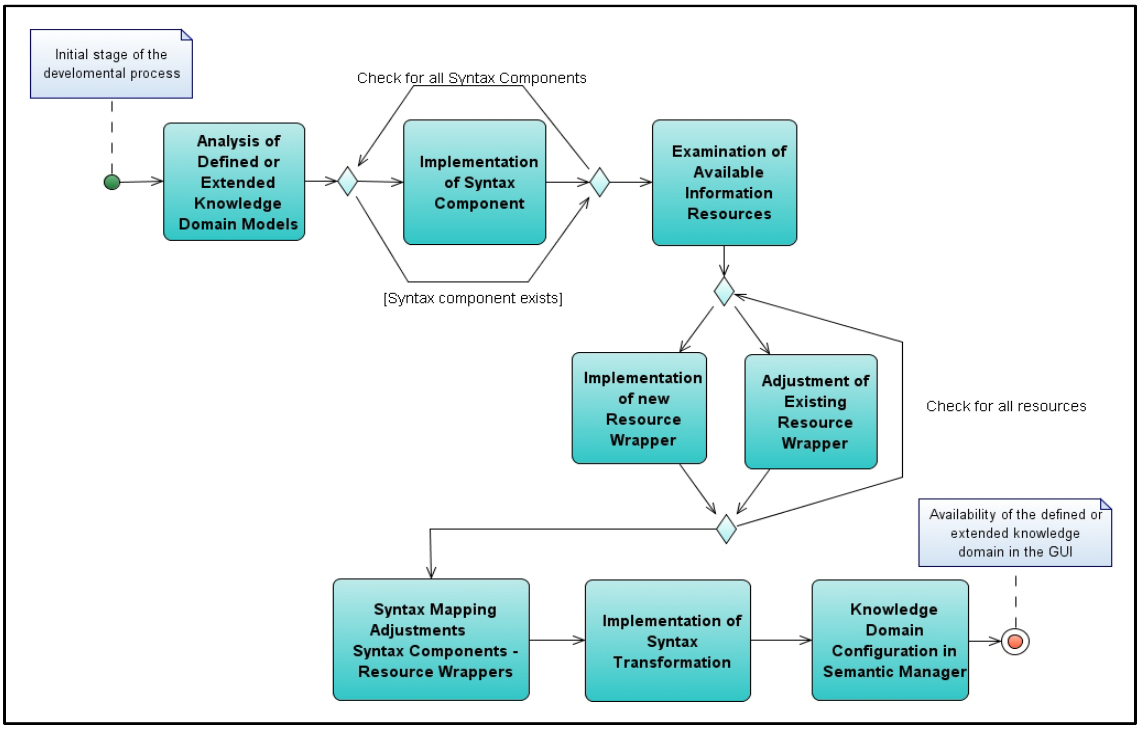

Initial stage of the developmental process
Analysis of Defined or Extended Knowledge Domain Models
Check for all Syntax Components
Implementation of Syntax Component
[Syntax component exists]
Examination of Available Information Resources
Implementation of new Resource Wrapper
Adjustment of Existing Resource Wrapper
Check for all resources
Availability of the defined or extended knowledge domain in the GUI
Syntax Mapping Adjustments Syntax Components - Resource Wrappers
Implementation of Syntax Transformation
Knowledge Domain Configuration in Semantic Manager

Figure 3-19: UML activity diagram illustrating the steps involved in the GeKnowME developmental process for a generic knowledge domain.

Once a scientific community has defined the knowledge domain of interest as a topic map model, the developer can start considering which topic type and association type components are needed for the syntax integration. All comprising syntax components have to be developed. Some of them may have been already developed and thus can be just reused; others have to be newly implemented. The implementation of new topic type or association type components is not too laborious, since according abstract classes implement their fundamental functionalities (compare *Figure 3-12*).

For each generated syntax component, the occurrence space providing the information related to the subject or subject-relations has to be identified and described. For this reason, either existing resource wrappers have to be adjusted, or new ones have to be developed. The last ones have to be registered by the involved resource manager. After the configuration of which resource wrappers are mapped to a particular syntax component, the implementation of its parsing procedures for all these related resources follows. The last developmental step to be performed considers the semantic integration. All comprising syntax components of the developed knowledge domain model have to be notified to the participating semantic manager. Once all above described steps are executed for a newly defined or extended knowledge domain model, the semantically integrated information resources are available for exploration by scientists.

3.5.2 Scientific Exploration Process

From the scientist’s point of view the GeKnowME systems offers powerful tools for exploration and navigation though correlated information entities. Independently from the researched knowledge domain, scientists perform similar sequence of activities in the investigation process. The main steps of this exploration process are depicted in *Figure 3-20* as a UML activity diagram.

In the first step, scientists have to decide in which information space they want to perform their investigations. With this step they restrict the life-science information space to the knowledge domains of interest and therewith they reduce the information complexity. The next scientific activity is to execute the search procedure according to the entered criteria. In general, the results represent the enter point to a semantic network built up by biological entities, which are connected in a bidirectional way. From a resulting entity scientists can navigate to further related entities and investigate the semantic network. If an entity seems to be essential for the scientific research, it can be attached to a model representing the entities and their interrelations of the subject matter of interest. All preceding steps can be repeated to retrieve relevant entities. The involved entities in the generated model can be arranged in a desired way for a better expressiveness. If further explorations are necessary afterward, the generated models can be stored and provided to scientists at a later point. Desirably, at the end of an exploration process, scientists have gained novel insights or have inspirations for new hypotheses, which can be additionally evaluated (e.g. experimentally verified).

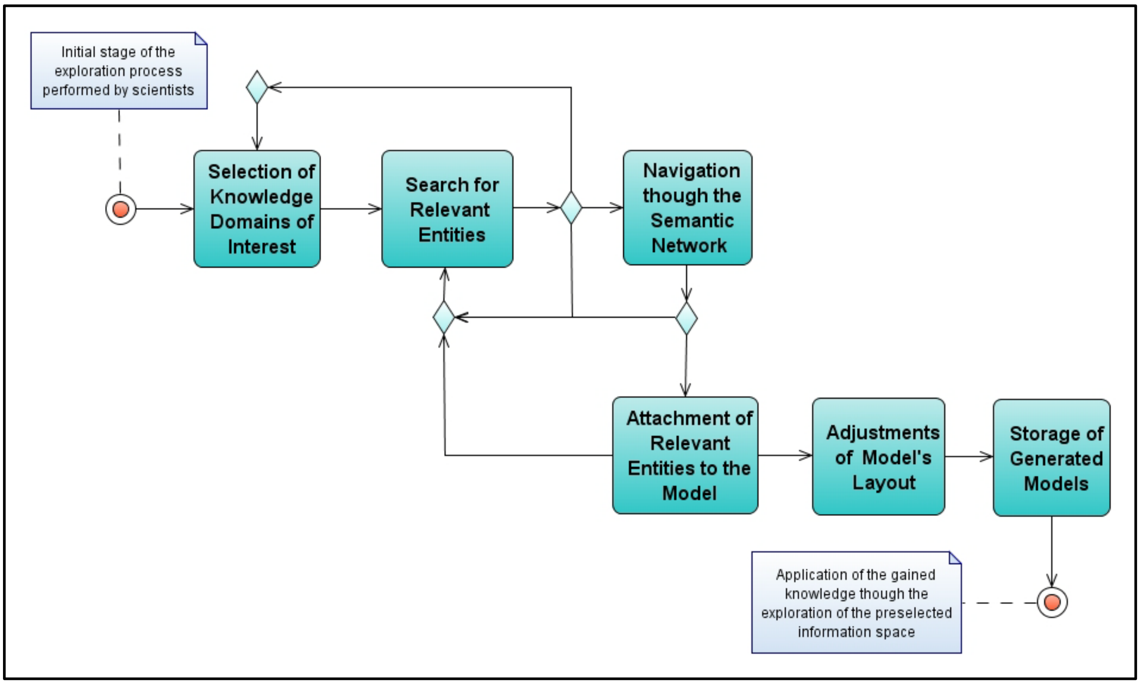

Initial stage of the exploration process performed by scientists
Selection of Knowledge Domains of Interest
Search for Relevant Entities
Navigation though the Semantic Network
Attachment of Relevant Entities to the Model
Adjustments of Model's Layout
Storage of Generated Models
Application of the gained knowledge though the exploration of the preselected information space

Figure 3-20: UML activity diagram illustrating the steps involved in the exploration process for the generation of a model representing the entities involved in the subject matter of scientific interest.

The descriptions above capture the main steps in the exploration process from the scientist’s point of view. Moreover, the process can be illustrated through the interactions between the involved GeKnowME components and their execution order. The exact sequence of the executed events in the communication between the single components is depicted in *Figure 3-21* corresponding to runtime scenario of the scientific process. The UML sequence diagram represents the participants with their lifeline and triggered events. It captures the interactions between the components described in details in the system’s developmental view. The diagram involves components from the upper presentation layer down to components from the lowest integration layer. The actor triggering the illustrated process is the scientist, who loads all available knowledge domain models within the used browser. Since the sequence diagram is quite self-explanatory, the exact handling of the message communication is deliberately omitted.

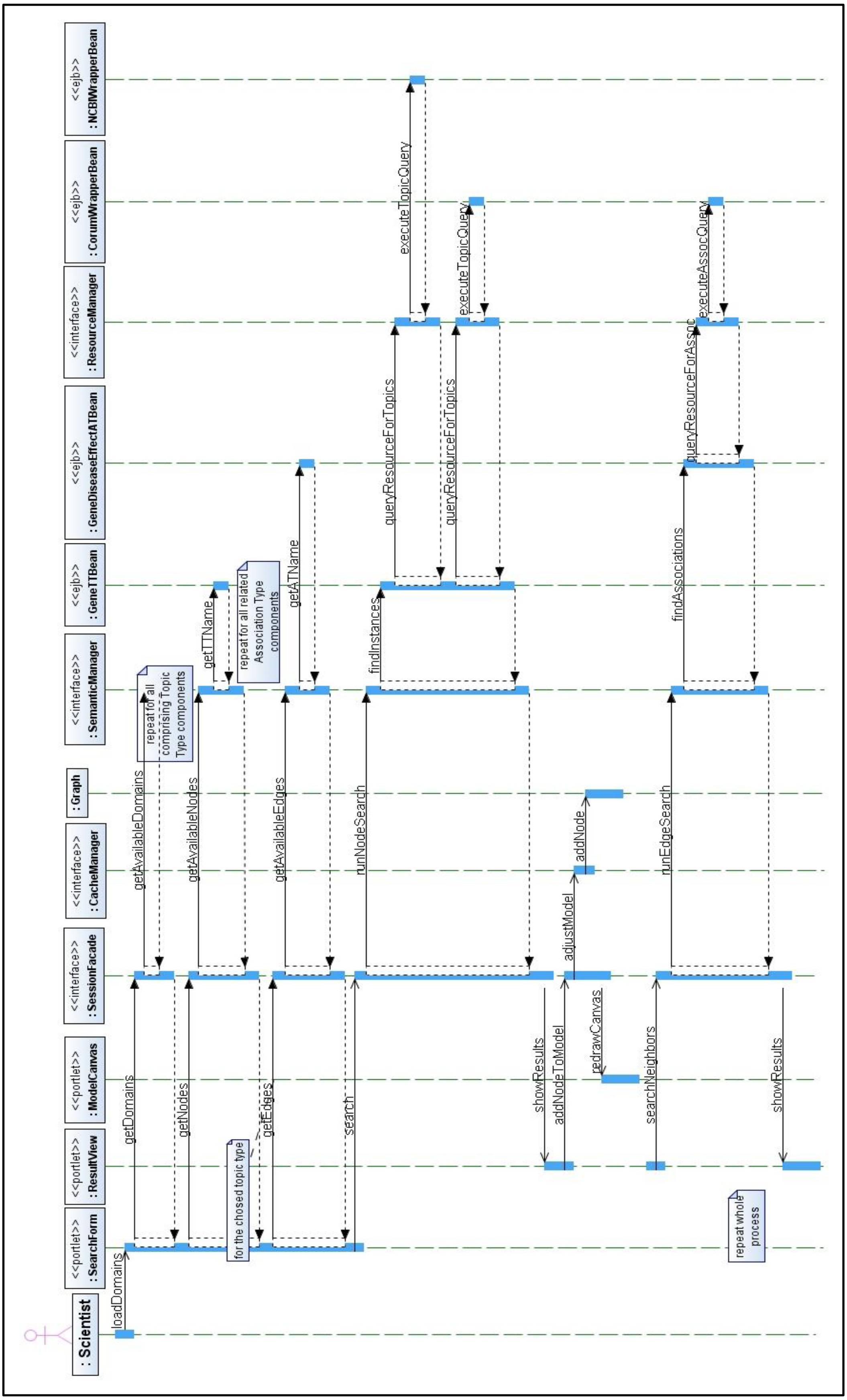


Figure 3-21: UML sequence diagram representing the communication between the GeKnowME components during the scientific exploration process.

The potency of the GeKnowME framework is its generic applicability. Any arbitrary knowledge domain can be defined within the system. Developers can easily associate existing information resources to these ontological models and provide preselected exploration information space. For the correct integration of the distributed and heterogeneous information, concepts for dynamic semantic annotation have been established. Their implementation is based on design pattern approaches, thus the developmental and maintenance efforts are kept low. To demonstrate the power of the introduced integrative concepts behind the GeKnowME framework, I applied the system for few life-science related studies.

4 Applications

> *"Die Praxis sollte das Ergebnis des Nachdenkens sein, nicht umgekehrt."*
>
> ***Hermann Hesse (1877 - 1962)***

The analysis of the human genome concerning genetic disorders and the resulting hereditary disease phenotypes has been deliberately chosen as a scientific area to exemplify the subject-centric integrative approach of the GeKnowME framework. As already mentioned in *section 2.2*, the main reason for the emergence of the tremendous amount of information resources available in life-science nowadays is the scientific aspiration to understand the basis of the human health and causes for morbidity. This has been the key goal of the HGP for the past two decades (Freimer, et al., 2003). The results of the diverse elaborate investigations in this field are very heterogeneous and organized in highly distributed information resources. Nevertheless, the generated information regarding the examinations of the correlations between genotypes and phenotypes is still quite overlapping. The process of understanding various disease mechanisms demands access and combination of diverse distributed pieces of information[8]. Therefore, the exploration of the human genetic disorders is a very suitable field to demonstrate the importance of semantic information integration for the management of the available knowledge and how it can be achieved by utilizing the GeKnowME framework.

Before starting with the illustration of the undertaken studies, an overview of the GeKnowME utilization process is given to emphasize how the framework can be used in general. One can consider the GeKnowME system as a "gate" to a giant virtual knowledge network, which contains the entities involved in the predefined knowledge domains (see *Figure 4-1*). The knowledge network represents an n-partite graph of

[8] In *Figure 4-4* and its corresponding description, the correlation between different disease factors is discussed according to the paper by *Loscalzo et al.* (Loscalzo, et al., 2007).

semantically annotated entities. Depending on the performed explorations, real sub-networks can be generated containing only the research relevant entities. The size of such knowledge sub-networks can vary a lot, for instance networks containing very few related entities or really expanded ones. These sub-networks are represented in an XTM format and can be used for further processing. If necessary, a knowledge sub-network can be converted into another desired data format or transformed to a graph containing a certain set of partitions (e.g. a bipartite or tripartite graph). Subsequently, three analysis types can be undertaken: large-scale, mid-scale, and small-scale studies. While the large-scale and mid-scale analyses are executed in a Java EE client application, the small-scale ones are performed within a web browser (compare *Figure 3-8*).

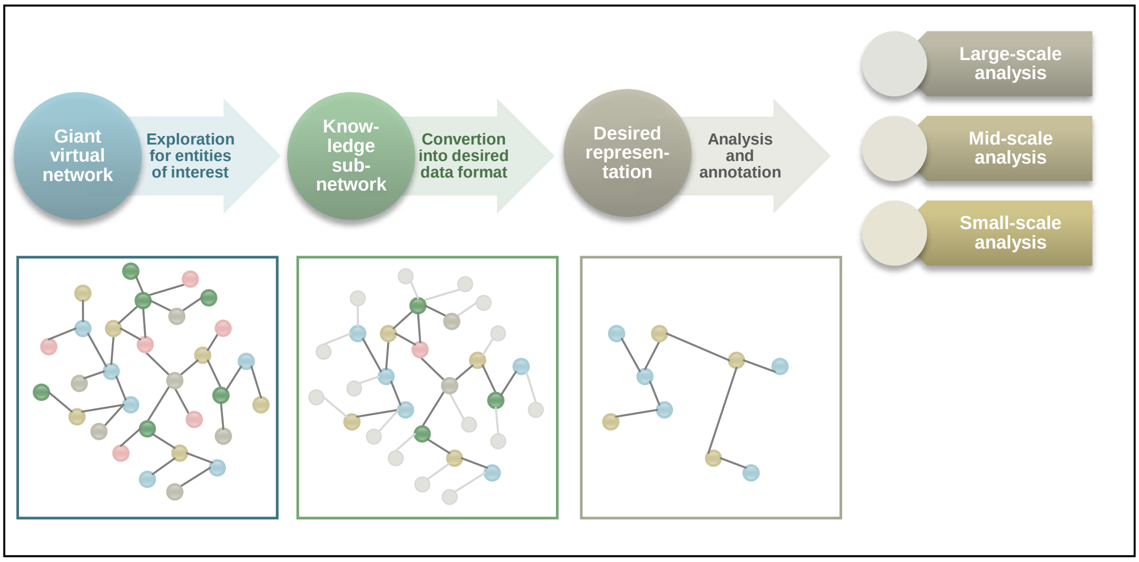


Figure 4-1: Utilization process and analysis types of the GeKnowME framework.

An example of each analysis type is introduced in the following subsections. Beforehand, the considered knowledge domain and the associated mapped information resources are concisely discussed.

4.1 Human Genetic Diseases

In the past two decades the study of the human hereditary diseases has achieved substantial results. Over 2.000 genes have been identified and associated with human disorders or phenotypic traits (NCBI - OMIM, 2008). The research of the correlation between genes and pathogenic phenotypes gives the opportunity to understand the molecular and physiological basis of human genetic diseases and achieve progress in their therapies. Since 1966 human diseases, known having a Mendelian inheritance, have been compiled in the *Mendelian Inheritance in Man* (MIM) catalog. *Online Mendelian Inheritance in Man* (OMIM) represents the most complete and up-to-date online repository containing information on genetic disorders and genes (Hamosh, et al., 2005). Currently, OMIM includes more than 6.000 diseases, which are categorized by their inheritance pattern and annotated whether the underlying DNA sequence is known (see *Table 4-1*). Some disorders are linked to the identified genes, while others to chromosomal regions. However, there is a great part of genetic disorders with unknown molecular basis.

Diseases with	Autosomal	X-Linked	Y-Linked	Mitochondrial	Total
Molecular basis known	2.118	199	2	26	**2.346**
Molecular basis unknown	1.480	137	5	0	**1.622**
Suspected Mendelian basis	1.943	140	2	0	**2.085**
Total	**5.541**	**476**	**9**	**26**	**6.053**

Table 4-1: OMIM statistics about the number of human disease entries (NCBI - OMIM, 2008).

The determination of the DNA sequences that cause specific traits in an intact organism is a tedious, labor-intensive activity (Botstein, et al., 2003). Even for monogenetic diseases, the finding of the causative mutation of the disease under consideration is a tedious task. This situation is further complicated by the fact that a mutation in the same gene can influence also multiple phenotypic traits (pleiotropy) (Griffiths, et al., 2000). The process of discovering the correlation between phenotypes and genotypes becomes even more demanding when complex diseases are considered. Generally, complex, or multifactorial, diseases are influenced by more than one gene or environmental factor and thus do not exhibit a simple mode of inheritance (Ghosh, et al., 1996). Examples of multifactorial diseases include neurodegenerative diseases such as Alzheimer and Parkinson, or metabolic ones such as diabetes and hypertension.

Recent publications have shown that there is an increasing evidence for strong interrelationships between genetic diseases at a higher level of organization such as the cell-, tissue-, organ-, and organism-level (compare *Figure 2-3*). Since the study of such interconnections can reveal unexpected and novel genetic links, these relationships can be of vital importance for the better understanding of complicated pathogenic mechanisms and thus for the improvement of clinical practices. *Oti et al.* introduced the terms *phenotype space* and *gene space* to illustrate the relationships among individual phenotypic traits and their underlying genes as shown in *Figure 4-2* (Oti, et al., 2008). The phenotype space represents the human phenome landscape associated to all known human genetic diseases. Conversely, the gene space includes genes associated with DNA disorders causing human genetic diseases. By building disease and gene networks, the interactions between the phenotype and gene space can be analyzed qualitatively.

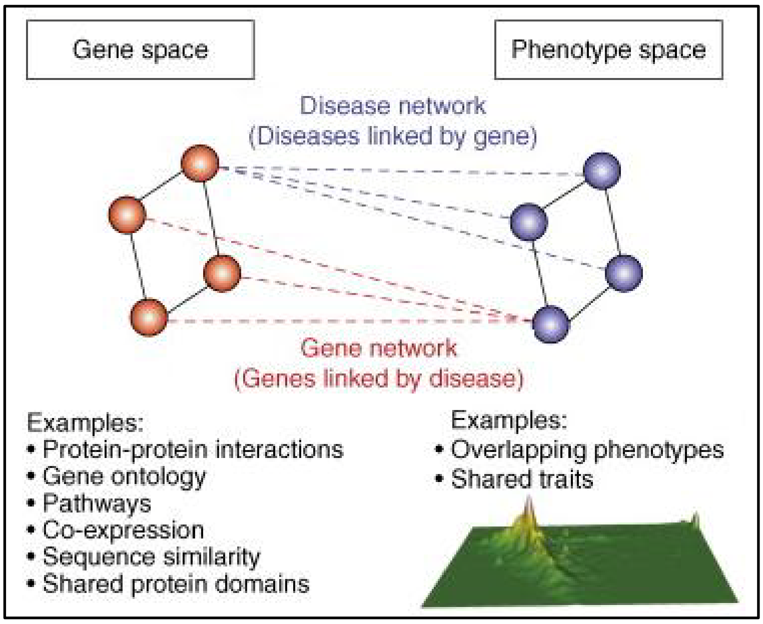


Figure 4-2: Relationships between gene space and phenotype space with examples how the corresponding gene and disease networks can be expanded with further biological information (Oti, et al., 2008).

A disease network, marked in blue in the figure above, can be derived by linking multiple phenotypic traits or genetic disorders associated with the same gene. Contrarily, gene networks can be built by linking all genes related to a single disease. Additionally, observations in the human phenome and other model organisms suggest that similar phenotypes are caused by mutation in functionally related genes. These genes may be involved in different types of biological modules (Oti, et al., 2007). For instance, such modules can be a multi-protein complex, a pathway, a functional module based on possible protein-protein interactions, etc. The gene networks can be then expanded and the modular nature of the genetic diseases can be considered. Consequentially, this type of modular information can be used to greatly increase the likelihood of finding potential candidates for disease causing genes.

In the study of *Goh et al.*, the combination of the whole human gene and phenotype space is referred to *human diseasome* (Goh, et al., 2007). It can be represented as a bipartite graph of disorders and disease genes linked by known phenotype-genotype associations based on the OMIM dataset (see *Figure 4-3*). The results regarding the analysis of the human diseasome show that most diseases are not isolated but rather form part of continuum of interconnected diseases. A large genetic overlap is also supposed to be the cause for these disease interconnections. The genetic overlap may result in shared pathogenesis of diverse genetic diseases evident by phenotypic overlap. These concepts have been also analyzed and confirmed by further separate studies (Rzhetsky, et al., 2007), (Xu, et al., 2006), and (van Driel, et al., 2006).

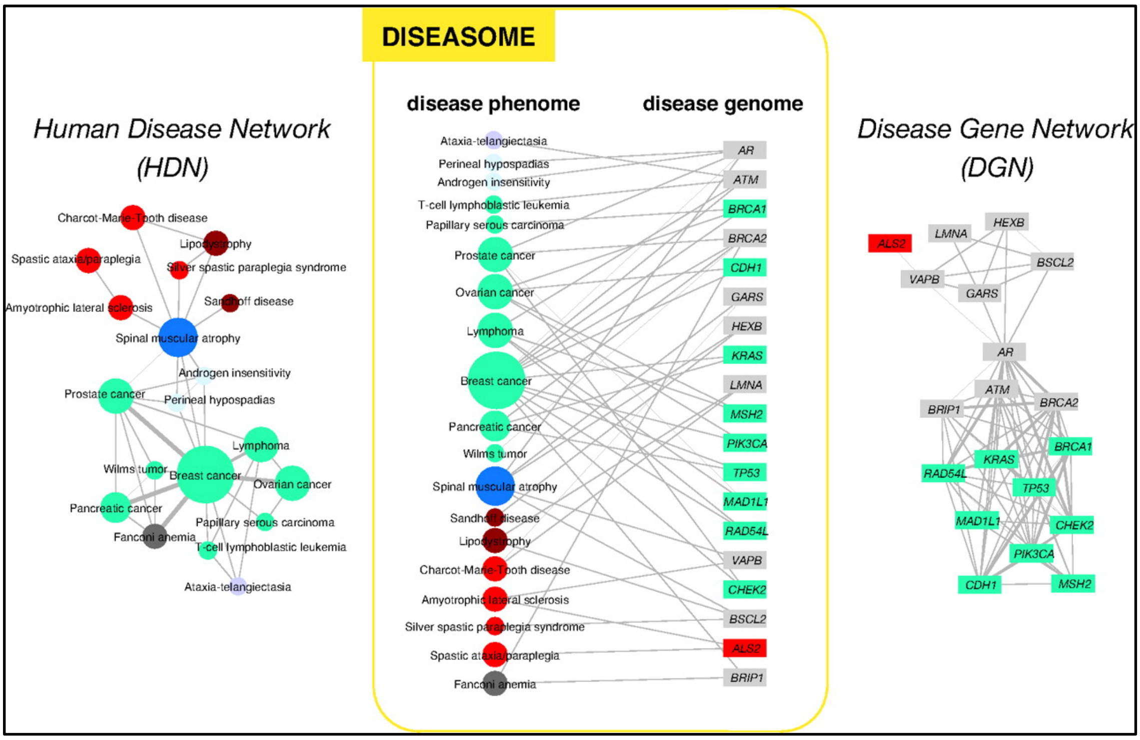


Figure 4-3: A small subset of the OMIM-based disorder gene association network called diseasome. Two biological relevant networks projections *Human Disease Network* and *Disease Gene Network* can be derived out of the primary diseasome (Goh, et al., 2007).

Generally, the relationship between a disease phenotype and the underlying genotype is not trivial. Multiple factors influence the final patho-phenotypes of genetic diseases. *Loscalzo et al.* grouped these factors into four different modular networks, as shown in *Figure 4-4* where they are marked in green (Loscalzo, et al., 2007). The nodes within these networks, such as genes, proteins, physiological properties, or environmental factors, interact with each other to yield into *patho-physiological states* signed as *PS*, which, in turn, underlie all disease phenotypes (*patho-phenotypes*, *P*). In this classification of factors influencing diseases, the disease-modifying genes are subcategorized into two groups, those that cause a disease by a primary genetic mutation (*primary disease genome, G*), and those that influence the disease indirectly (*secondary disease genome, D*) by reflecting generic response to organism stress evoked by mutation or environmental exposure. A further modular network consists of *intermediate phenotypes, I*, representing the variations in disease expression and clinical representation. Additionally, *environmental determinants, E,* may affect the patho-physiological states.

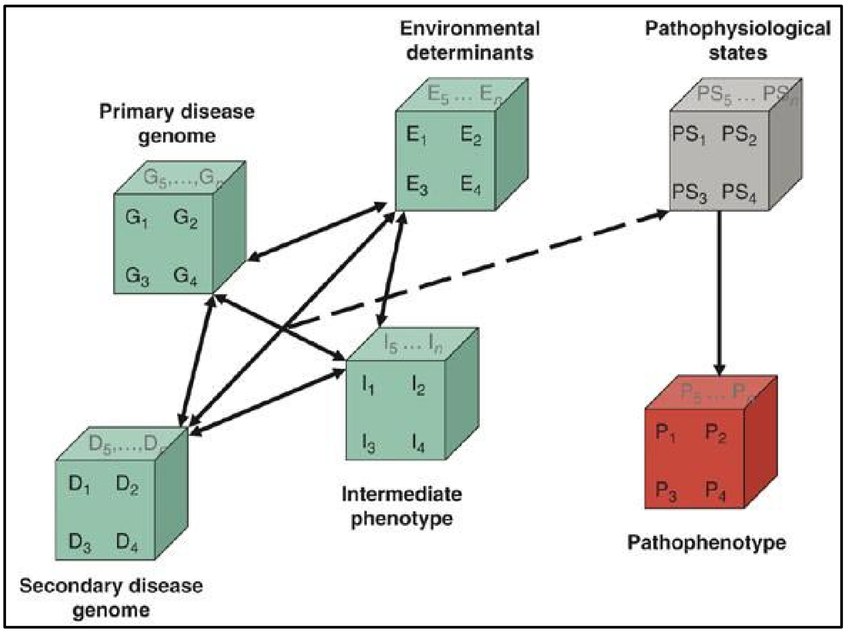


Figure 4-4: Illustration of the correlation between influencing factors causing disease phenotypes (Loscalzo, et al., 2007).

More and more information resources are available containing potentially useful knowledge for the analysis of all these influencing factors and their interconnections. Life-scientists are confronted with very large amount of significant information spread over many distributed resources. A more effective and efficient management of the gained knowledge in this field is desirable. The investigation process of the factors influencing genetic diseases can be optimized by utilizing the GeKnowME framework.

4.2 Knowledge Domain Model

A knowledge domain model, called *Human Genetic Diseases*, has been defined representing the structure of the scientific field concerning the exploration of factors affecting genetic disorders (see *Figure 4-5*). The model represents significant biological concepts (topic types) and their interrelations (association types) with the adequate association labels. Concepts' characteristics (occurrence topic types) obtainable from the bound information resources are also illustrated in the model. The general characteristics *name*, *synonyms*, and *ids* are deliberately omitted in the figure for a more clear depiction. It is important to mention, that the defined model does not include all involved biological concepts playing a role in the analysis of the human genetic diseases and their underlying causes. The purpose for this decision is to keep the model simple in order not to overload researchers with too much information during the exploration of the bound information space and to mainly demonstrate how the GeKnowME system can be utilized. If further important biological concepts and interrelations are required, the implemented knowledge domain model can be easily extended, or even new models can be defined and merging mechanisms can be applied (compare *Figure 3-3*).

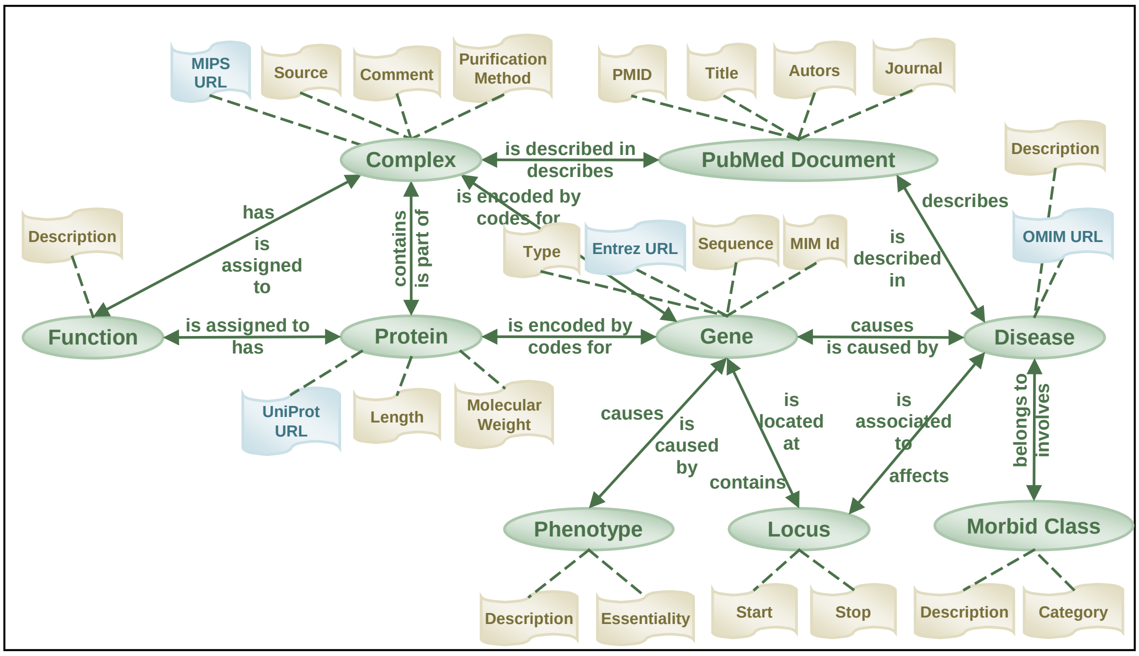


Figure 4-5: *Human Genetic Diseases* knowledge domain model. Biological concepts (topic types) are represented as green ovals. Available characteristics (occurrence types) are linked to concepts by dotted lines, where the blue-colored sheets represent external occurrences and the brown-colored are internal ones.

The bound life-science information resources to the knowledge domain model *Human Genetic Diseases* are represented in the *Figure 4-6*. In general, they provide the properties associated to the subjects of investigation. Each biological entity such as a particular human gene like BRCA1 is identifiable by a preselected *Public Subject Indicator*, compare *subsection 2.4.3 Topic Maps*. For instance, the chosen PSIs for the instances of the topic type *gene* are the references to the NCBI Entrez Gene database, which contains a recognized set of gene identifiers. Therefore, the PSI for the BRCA1 gene is *http://www.ncbi.nlm.nih.gov/sites/entrez?Db=gene&TermToSearch=672*. Consequently, many properties coming from distributed resources can be attached to an identifiable topic. For example, the property *type* of a gene with the value *"protein coding"* available from the NCBI Entrez Gene database and the property *MIM Id 113705* obtainable from OMIM, are appended to the gene topic BRCA1. Therefore in the figure below, some topic types are linked to more than one information resource.

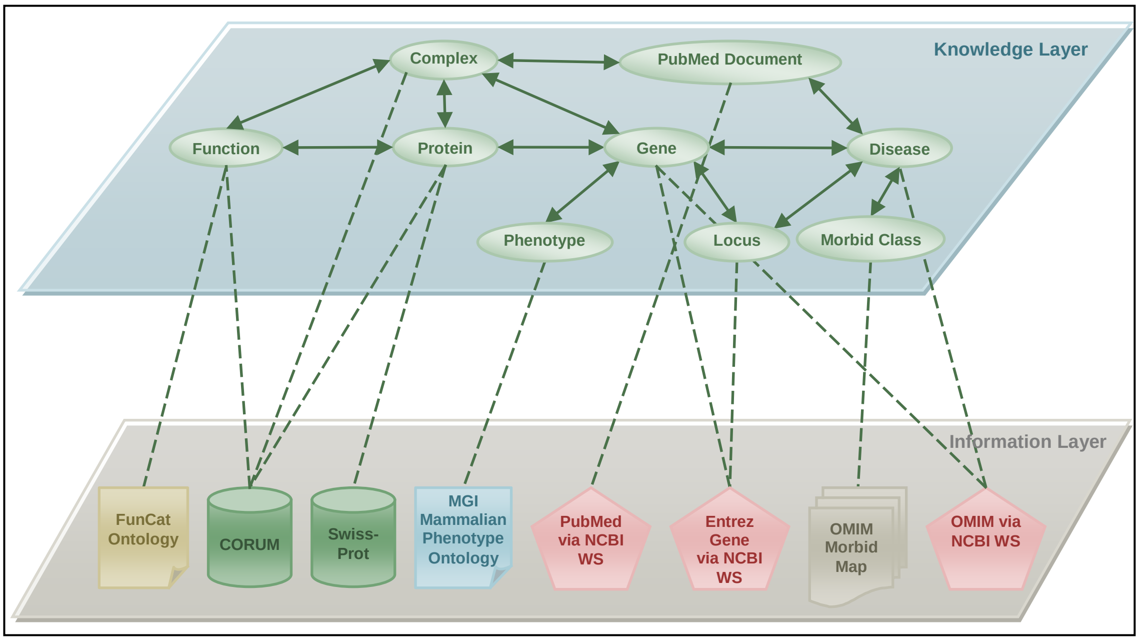


Figure 4-6: Mapping of the defined *Human Genetic Diseases* knowledge domain model to existing information resources. The dotted lines represent which resource contains information related to the instances of the defined concepts.

All biological concepts defined in the knowledge domain of investigation and the linked information resources are discussed briefly to point out their relevance to the analysis of the human genome concerning genetic disorders and the resulting hereditary disease phenotypes.

- *Disease:* In the chosen context, the topic type *disease* refers to subjects representing a morbid condition caused by abnormalities in the genome. Only human genetic disorders are considered in the undertaken studies. As already mentioned, the most complete and up-to-date public resource containing information about human genetic diseases is the OMIM repository provided by

NCBI. It includes both monogenetic and complex diseases. The disease related information is accessed by utilizing the NCBI Utilities Web Services (NCBI, 2008).

- *Morbid Class:* A *morbid clas*s represents an upper category built by merging subtypes of a single genetic disease. For instance, *"Alzheimer disease"* is a morbid class, which includes fourteen complementation diseases described in OMIM (see *Appendix A*). The morbid classes are included in the domain model to support a higher level of disease exploration. The morbid classification was introduced and applied in Goh's *et al.* studies of the human diseasome. It was generated by semi-automated annotation of the OMIM Morbid Map (Goh, et al., 2007). Furthermore, the morbid classes are classified into 20 primary disorder categories based on the human physiological system. Disorders having multiple clinical features are assigned to the category *Multiple* and disorders with unclear phenotypes to the category *Unclassified*. The complete list of all disorder categories is represented in *Appendix B*. In the GeKnowME system, the morbid classes and their categories are available as a flat-file.

- *Locus:* Instances of the topic type *locus* represent fixed positions on a chromosome of a certain genome. Not only genes, but also numerous genetic disorders are linked to such chromosomal locations. Information about loci is available through the NCBI Utilities Web Services.

- *Gene:* The topic type *gene* represents a unit of inheritance, which specifies a biological function and refers to a DNA locatable region. Information about genes is available in numerous resources. In the GeKnowME system, the NCBI Entrez Gene database, as one of the most well curated databases containing gene-centric information, delivers the main occurrence space for the instances of the topic type *gene*. Currently, it includes over 24.000 *Homo sapiens* protein coding genes (Maglott, et al., 2007). Additionally, the OMIM database is bound to the topic type *gene*, since it contains disease-related information for more than 12.000 protein coding genes.

- *Phenotype:* In the defined knowledge domain, a description of the observable state of an individual with respect to some inherited characteristic is considered as a *phenotype*. Unfortunately, in the OMIM database there is no definite set of terms describing patho-phenotypes. The associated phenotypes to a disease are involved in its free text description. The MGI Mammalian Phenotype Ontology offers such controlled vocabulary and is involved in the provided information space for the analyzed knowledge domain (Bult, et al., 2008).

- *Protein: Proteins* as gene products perform indispensable functions within cells. The disturbance of their expression may lead to modified phenotypes and cause disease states. The Swiss-Prot database is used as a primary resource for retrieving protein related information and is accessible using predetermined URLs. Almost 20.000 *Homo sapiens* proteins are included in the latest release of Swiss-Prot 55.6 (Boeckmann, et al., 2003).

- *Complex:* Protein *complexes* represent essential molecular entities that integrate multiple gene products to achieve particular functions within the cell. For the analysis of human genetic diseases, the consideration of protein complexes can give insights at a higher level of organization. The CORUM database is the major resource containing information about experimentally verified mammalian protein complexes (Ruepp, et al., 2008). Currently, at about 2.500 mammalian complexes are available in the CORUM database and are accessible in the GeKnowME framework via JDBC connectivity.

- *Function:* Generally, all functions of living organisms are related with proteins. Each protein or protein complex is responsible for its own specific *function*. Both the GO (Gene Ontology Consortium, 2006) and the FunCat (Ruepp, et al., 2004) ontologies offer controlled vocabulary describing the roles of gene products. For the analysis of the *Human Genetic Disease* knowledge domain the FunCat catalogue is preferred to organize the protein and protein complex space into biologically meaningful subsets, which are significant but not too specific.

- *PubMed Document: PubMed documents* represent scientific articles published in diverse biomedical journals. They may include relevant information regarding examinations of human genetic diseases. The occurrence space for the PubMed documents is provided by the PubMed repository accessed via the NCBI Web Services (Wheeler, et al., 2008).

In order to be able to express the context, in which the biological entities are valid, different scoping topic types are introduced. The *organism* name is used as a main scoping topic type to allow correct integration of distributed information. For instance, integrated genes are scoped with their correct genome names like “*Homo sapiens*” or “*Mus musculus*”. The scoping topic *organism* is also related to the topics from type *protein*, *complex*, *disease*, and *locus*.

The determined knowledge domain model can be considered as a map over the chosen life-science information resources. By utilizing this map, one is able to navigate intentionally from one biological entity to a next one, since the associations between the entities are bidirectional. In general, these associations are obtainable from the mapped information space. For instance, the OMIM resource delivers information whether a gene is associated with a particular disease and if so with which one. However, the Topic Maps approach provides more powerful possibilities of investigation, since one can continue navigating for instance from a found gene further to a protein and then to a complex or a function and perhaps infer new significant insights by finding indirect relations between the involved topics.

According to the defined knowledge domain, within the GeKnowME framework all required software components were implemented to enable the subject-centric integration of the associated information space. For each information resource, such as the CORUM database, a corresponding *ResourceWrapper* component was developed to encapsulate the procedures needed for the effective retrieval of the mapped information. Additionally, for each topic type and association type, defined in the “Human Genetic Diseases” knowledge domain, a syntax component was developed implementing the necessary methods for the dynamic semantic annotation. Finally, all these software components were configured in the GeKnowME system to provide the possibilities to explore more efficiently the information space related to human genetic diseases and to build models explaining the complexity of their development.

Generally, the GeKnowME system offers access to a virtual semantic network of the entities available from the mapped information space (compare *Figure 4-1*). The virtual network of the considered knowledge domain includes, for instance, over 6.000 disease topics, over 24.000 human gene topics, or over 6.000 phenotypic descriptions. Undoubtedly, novel knowledge can be acquired from this giant knowledge network of interconnected biological entities. Depending on how many entities one considers in the scientific exploration, the process of identifying new insights can be performed in three different ways: large-, mid-, and small-scale analysis. For each analysis type, an example is illustrated in the following sections. The descriptions of the examples emphasize not on the technical implementation but more on the results of the undertaken studies.

4.3 Large-Scale Analysis

Generally, in the large-scale analysis the entire knowledge network or a major part of it is explored to identify common relations and general features on a conceptional level to support creation or verification of scientific assumptions. The generation of such a large network is rather time consuming task, since it contains all or almost all available entities from the associated distributed information resources to the predefined knowledge domains and the entities have to be semantically integrated following the subject-centric approach. Therefore, the generation of a large knowledge sub-network out of the fundamental virtual one is executed within a Java EE client application.

4.3.1 Extended Human Diseasome

The giant virtual semantic network corresponding to the “Human Genetic Diseases” knowledge domain represents the currently known relationships between genes and genetic disorders. However, it involves also information about protein complexes. In recent years, it has been shown by systematic experiments that the large majority of the gene products do not act as isolated entities but form transient or stable interactions with other proteins. Certainly, protein complexes, as the basic representatives of functional modules fulfilling higher-level cellular tasks, can be used to examine their disease relevance. A comparable research was performed by *Lage et al*. They generated and analyzed a human phenome-interactome network of protein complexes implicated in genetic disorders (Lage, et al., 2007). Since the network was based only on 506 complexes, the investigations were focused on particular diseases and on identification of disease-causing genes, and not on drawing general conclusions about the modular nature of genetic diseases. Additionally, the considered protein complexes were computationally generated from protein-protein interaction data and not experimentally verified.

To investigate the protein relevance to human genetic diseases, a sub-network was generated out of the giant virtual knowledge network including only the entities of interest (compare *Figure 4-1*). Since it is rather similar to the human diseasome network but considering the diseasome at a higher level of organization, the generated network is called *extended human diseasome*. For this purpose, all available protein complexes were included and also all proteins involved in at least one protein complex. The proteins, members of all mammalian complexes, were associated to the orthologous *Homo sapiens* genes. In addition, all diseases, known to be associated to the complex coding genes, were included in the extended human diseasome with their morbid classes. Prenatal or postnatal-lethal phenotypes were also considered. The exact numbers of the involved biological entities in the extracted sub-network are represented in *Table 4-2*.

Topic types	Number of topics
Mammalian protein complexes	2.090
Proteins involved in at least one protein complex	3.767
Human genes coding for the involved proteins	2.908
Human genetic diseases related to the involved genes	640
Morbid classes assigned to the involved diseases	402
Prenatal or postnatal phenotypes	12

Table 4-2: Statistics about the number of involved topics in the generated *extended human diseasome.*

The construction of the extended human diseasome network allows enhanced analyses of the human genetic diseases by considering additional influencing factors. Few undertaken studies are described in the following sections to demonstrate the acquisition of essential findings by analyzing such semantic networks.

4.3.2 Recurrences of Proteins in Complexes

The extended human diseasome network contains more than 2.000 mammalian protein complexes that are coded by almost 3.000 different genes. If one assumes a number of about 20.000 to 25.000 protein-coding genes in the human genome, as estimated by *Levy et al.* (Levy, et al., 2007), or the 24.000 human protein-coding genes available in NCBI Entrez Gene database, the analyzed gene set mapped to the corresponding *Homo sapiens* orthologs covers about 13%.

One of the main characteristics of protein complexes is their modularity. Several studies about the evolution of protein complexes have shown that some genes code for proteins, which tend to be shared across different complexes (Pereira-Leal, et al., 2007), (Hernández, et al., 2006). By analyzing the extended human diseasome, I looked into the recurrence of proteins in different complexes and their relevance to genetic diseases[9]. Since the human genetic diseases are associated to causing genes and not to the gene products, in the analysis it is considered how often a gene codes for protein complexes. The histogram depicted in *Figure 4-7* shows that 40% of all involved genes code for just one protein complex. Another 40% of the gene set code for proteins recurring rarely, i.e. 1, 2, or 3 times. The other 20% of the genes code for proteins that tend to be very often shared across multiple protein complexes.

[9] The term *recurrence* refers throughout this chapter to the property of genes to code for proteins, which recur in multiple protein complexes.

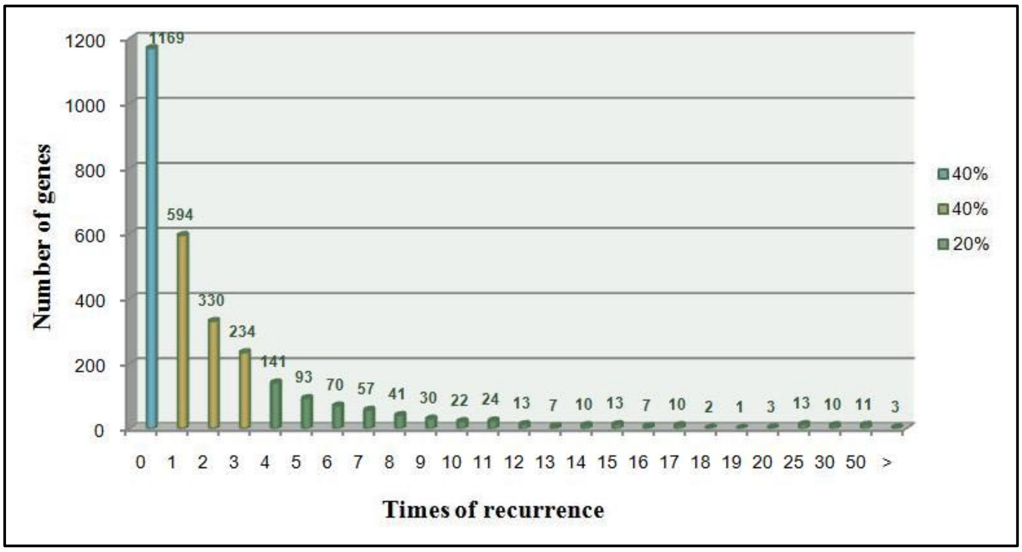


Figure 4-7: Depiction of the frequency how often a gene codes for protein complexes. The blue bar represents the genes coding for just one complex, the brown bars show the genes coding for proteins recurring not so often (1, 2, or 3 times), and the green bars correspond to the genes coding for proteins recurring multiple times.

To examine the correlations between these three groups of genes, an interaction network was generated out of the extended human diseasome, in which genes coding for the same protein complex were connected. The network includes 2.908 gene nodes and 39.480 interactions. The topological character of the network was analyzed by calculating the corresponding degree distribution and clustering coefficient (see the illustrations of these measures in *Figure 4-8*). According to *Barabasi* (Barabási, et al., 2004), the generated interaction network shows a scale-free topology denoted by the power-law distribution of the degree, whereby the degree distribution shows a scattering for the higher degrees. Furthermore, the network shows a tendency towards hierarchical topology, which is expressed by a diffused power-law distribution concerning the mean clustering coefficient *C(k)*. Since the network is considered to be of biological relevance, this topology was expected and gives a hint on the reliability of the data.

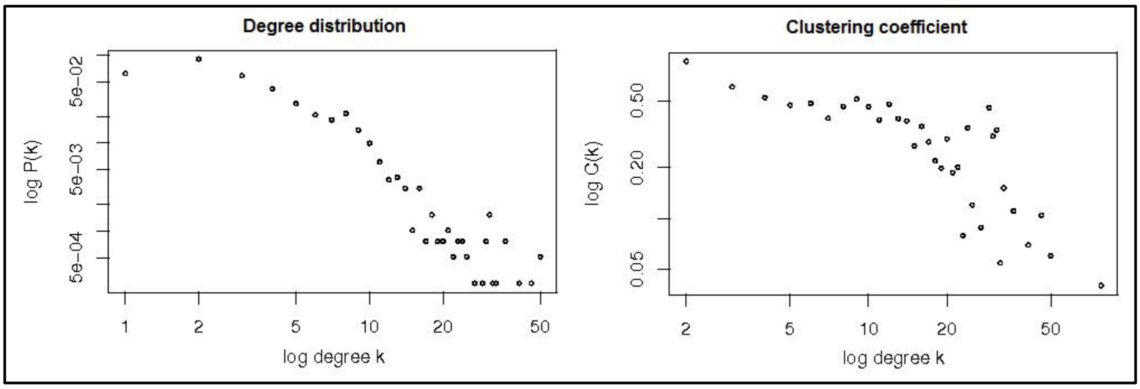


Figure 4-8: Topology check of the generated interaction network of human genes, which shows a scale-free topology and a hierarchical tendency as expected for biological networks.

4.3.3 Protein Recurrence in Relation to Genetic Diseases

In the extended human diseasome, 640 different genetic disorders are associated to the analyzed gene set. Overall 463 genes (16%) are assigned to these diseases, where the number of distinct diseases associated to a gene range from one to nine disorders. This information gives rise to analyze how the degree of diseases related to a gene reflects to the recurrence of the coded proteins. *Figure 4-9* resents the distribution of the disease related genes according to their attributes recurrence and number of diseases involved in. It indicates that genes related to multiple diseases are less likely to code for proteins that recur very often. Actually, there is an outlier observable in the set of genes connected to many diseases. The gene *TP53* codes for 23 protein complexes and is related to 9 diseases, which are categorized to the disorder class *cancer* (see *Table 4-3*). However, the main function of the TP53 gene is to regulate the cycle of cell division by keeping cells from growing and dividing too fast or in an uncontrolled way (Vousden, et al., 2005). Thus it functions as a tumor suppressor and its recurrence is even beneficial for processes involved in preventing cancer. Nevertheless, the 3D distribution shows that, the protein recurrence decreases further for genes associated with more than one disorder. In principle, this result indicates a probable sign of an evolutionary advantage, which is shown by the fact that proteins coded by genes involved in multiple diseases are rarely reused.

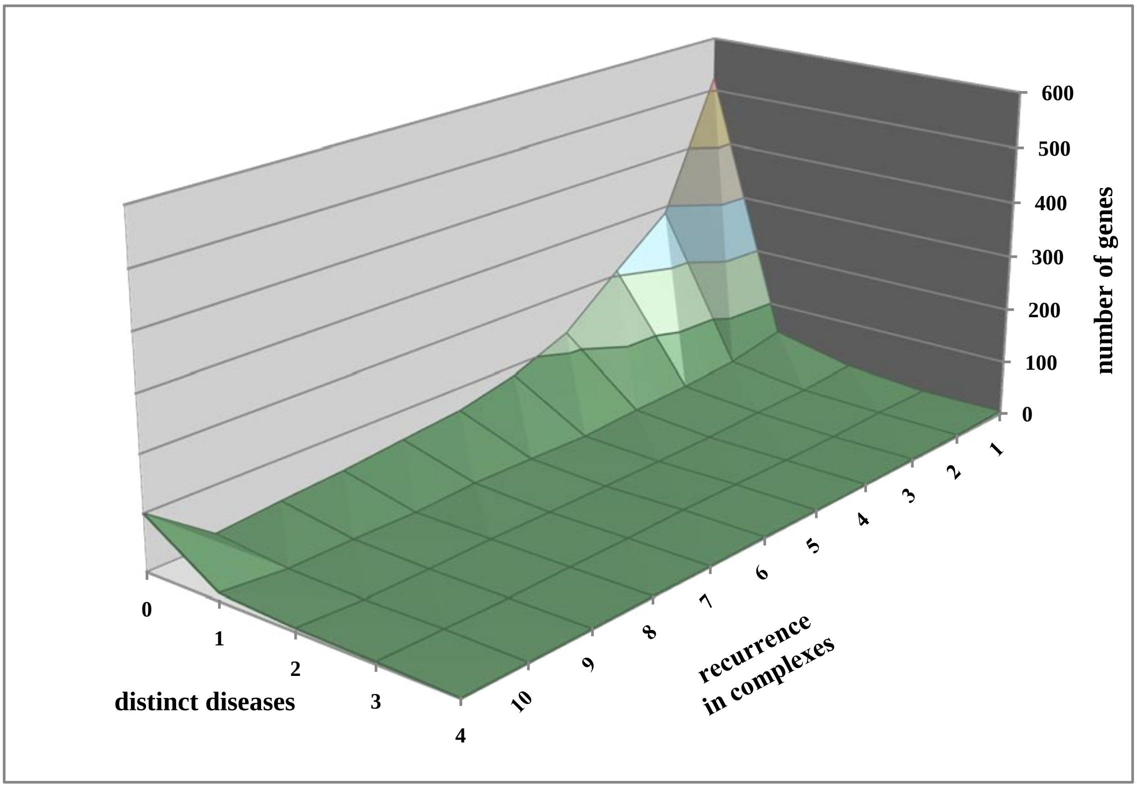


Figure 4-9: 3D plot depicting the distribution of all disease related genes in relation to their characteristics recurrence and number of associated diseases.

gene	# recurrences	# diseases	# morbid classes	# disorder categories	Disorder categories
FGFR2	0	9	9	6	cancer; skeletal; developmental; connective tissue; unclassified; multiple
TP53	22	8	8	1	cancer
COL1A1	0	8	4	2	bone; connective tissue
BRCA2	3	7	6	2	multiple; cancer
PAX6	0	7	6	1	ophthalmological
TGFBI	0	7	1	1	ophthalmological
MECP2	5	6	6	3	neurological; psychiatric; developmental
GNAS	2	6	5	4	endocrine; multiple; bone; cancer
MYH9	1	6	6	3	hematological; ear, nose, throat; multiple
RET	1	6	5	3	respiratory; cancer; gastrointestinal

Table 4-3: Top scoring disease related genes ordered by number of diseases involved in.

4.3.4 Protein Recurrence in Relation to Essentiality

In the extended human diseasome, the analyzed gene set is also associated to phenotype entries of the MPO. Another undertaken study regarding the protein recurrence in complexes is the analysis of its relevance to essentiality. Overall 721 genes, or 25% of the gene set, are annotated with at least one lethal phenotype. The categories MP:0005373 *lethality-postnatal* and MP:0005374 *lethality-prenatal/perinatal* with their subcategories have been considered (overall 12 distinct entries). The fractional distribution of the genes with assigned lethality-phenotypes against the corresponding recurrence property is illustrated in *Figure 4-10*. The histogram shows that two third of the genes annotated with lethal phenotypes code for proteins recurring frequently in protein complexes.

To observe the correlation between essentiality and recurrence, a network including only the protein complexes was extracted out of the extended human diseasome (see *Figure 4-11 A*). The network was generated by linking protein complexes, if they share one or more proteins. Additionally, for each edge a weight was calculated representing an essentiality factor. The calculated weights range from 0 to 1, where the value 0 means that no shared proteins are essential and 1 that all common proteins are essential. By removing all edges with no essentiality, the analyzed network does not change a lot (compare *Figure 4-11 A* and *B*). The low number of edges with no essentiality (approximately 22%) indicates that proteins shared across several complexes have an increased tendency to be essential. *Pereira-Leal et al.* came to the same assumption in their research of the origins and evolution of functional modules by analyzing experimentally defined complexes in yeast (Pereira-Leal, et al., 2007).

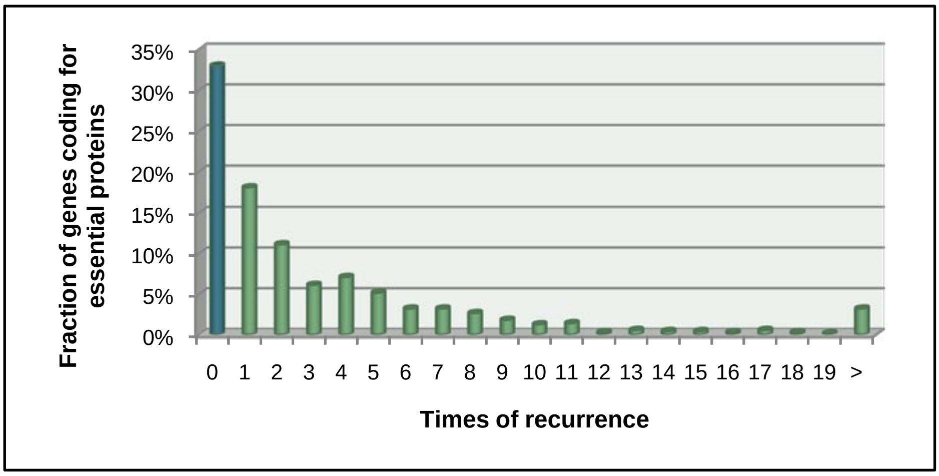


Figure 4-10: Representation of the fraction how often a gene with an assigned lethal phenotype codes for protein complexes.

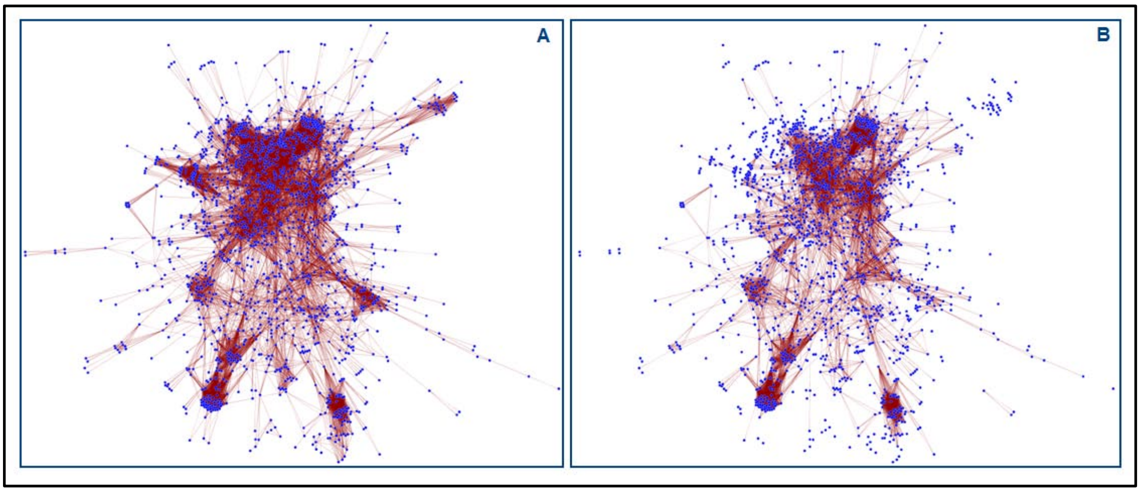


Figure 4-11: A. Network of protein complexes generated by interconnecting complexes, if they have at least one protein in common. An essentiality factor is assigned to each edge. **B.** The same network after removal of edges with no essentiality.

Generally, the introduced examples above illustrate how the GeKnowME framework can be utilized to organize and combine heterogeneous information from distributed information resources on a large scale by generating semantically correct knowledge networks. These large networks of scientific interest can be analyzed systematically to gain or verify novel insights on a conceptual level, in this case in the field of human genetic diseases.

4.4 Mid-Scale Analysis

The knowledge domains defined within the GeKnowME system can be also used to perform mid-scale explorations by restricting the associated information space to a particular set of entities. For instance, within a chosen knowledge domain one can focus the research on a set of genes involved in a particular metabolic pathway or members of a certain gene family. Knowledge networks, similar to networks generated during a large-scale analysis, can be extracted out of the virtual semantic network containing only the entities of the chosen context. Usually, these networks are not too large and subject specific evaluations can be performed. The *Notch* signaling pathway has been chosen as a representative example for a common mid-scale analysis.

The *Notch* signaling pathway is among the most commonly used communication channels in mammalian cells. It plays a key role in neuronal processes and functions at all stages of development to regulate cell proliferation, survival, and differentiation (Chiba, 2006). Studies in model organisms have demonstrated that the Notch signaling is essential during early embryonic development. Additionally, it’s known that the Notch signaling is dysregulated in many cancers, and faulty Notch signaling is implicated not only in several monogenetic diseases such as the *Aortic Valve Disease* or the *Alagille Syndrome*, but also in complex diseases such as the *Alzheimer Disease* (Bolós, et al., 2007).

The main idea behind the undertaken investigation is to explore the disease space associated to the genes involved in the Notch signaling pathway (compare *Figure 4-2*). As already mentioned in *section 4.1*, studies of the human phenome and other model organisms have shown that similar phenotypes are caused by mutations in functionally related genes (Oti, et al., 2008). In order to be able to increase the possibilities to find unknown relationships between genetic diseases, the gene space can be expanded with further functionally related biological information. For instance, one can involve all genes coding for protein complexes, which are known to be involved in the Notch signaling pathway.

Consequently, a tri-partite knowledge sub-network of the topic types *Gene*, *Complex*, and *Disease* was generated out of the information space associated to the knowledge domain “Human Genetic Diseases”. The generated network includes 49 genes and 72 protein complexes, which are annotated with the functional category *30.05.02.14 Notch-receptor signaling pathway*. Additionally, the network was expanded by considering further protein complexes, which are not annotated with the *FC 30.05.02.14*, but are coded by genes involved in the pathway. Analogously, all genes coding for the protein complexes involved in the signaling pathway were also included. Overall, the generated network includes 317 genes and 111 protein complexes.

Subsequently, diseases known to be associated to this gene set were integrated in the network. A graphical representation of the network is depicted in *Figure 4-12*.

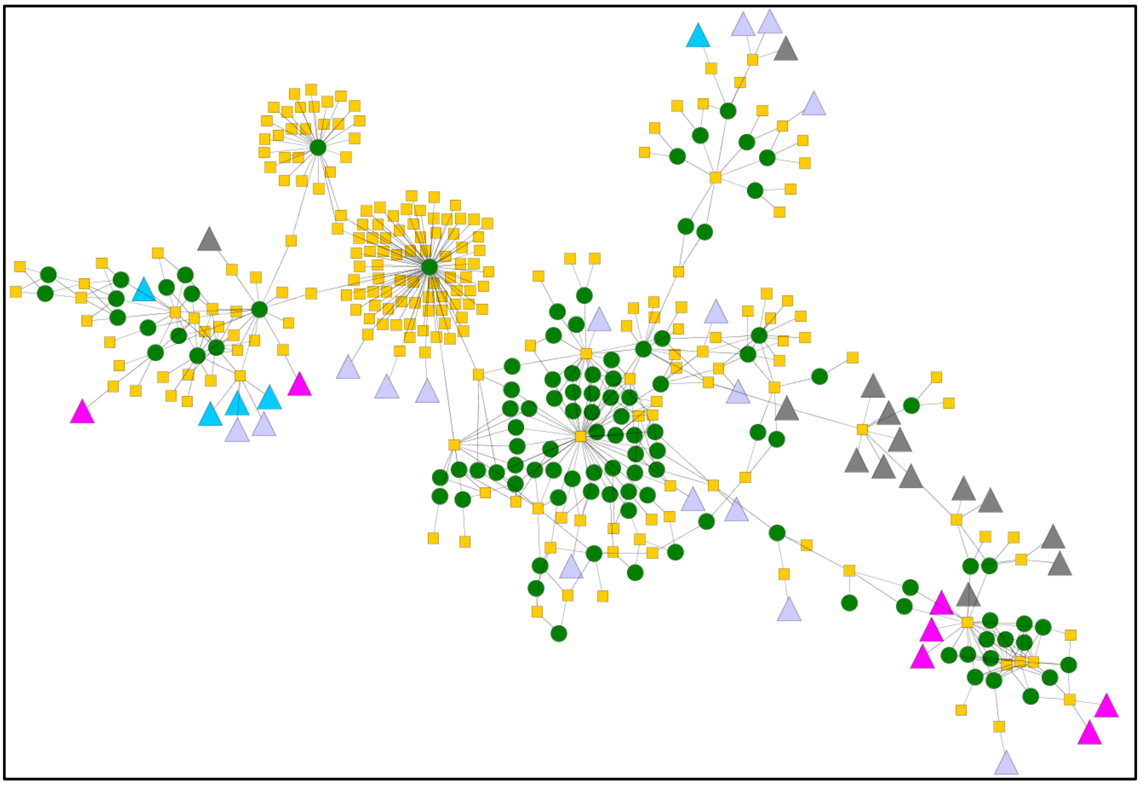


Figure 4-12: 3-partite graph representing the genes and protein complexes involved in the Notch signaling pathway with the associated genetic diseases. Protein complexes are depicted as green circles, genes as yellow squares, and diseases as triangles. The diseases assigned to the disorder category cancer are colored in dark gray, to the category neurological in magenta, and dermatological in blue.

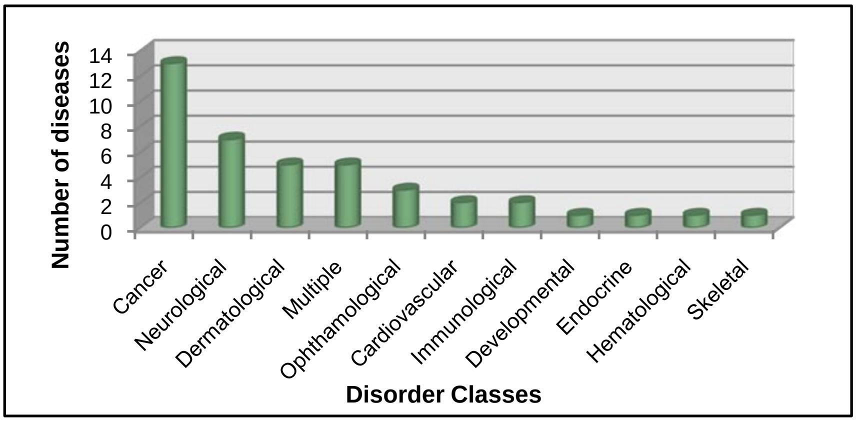


Figure 4-13: Distribution of the diseases associated to the gene set involved directly and indirectly to the Notch signaling pathway.

The disease space associated to the analyzed gene set includes overall 41 distinct genetic diseases. As expected, half of them belong to the disorder classes *cancer* and *neurological,* as shown in the histogram depicted in *Figure 4-13*. Moreover, this disease set indicates interrelationships between the disorder classes *cancer*, *neurological* and *dermatological*, which are not evident in the disease network created by *Goh et al.* (Goh, et al., 2007). The reason for this differentiation may be the consideration of the gene diseases associations at the higher level of organization. In comparison to the *Goh's* disease network, the generated disease space is built by considering not only the direct gene disease associations, but also the indirect ones derived from the protein complexes.

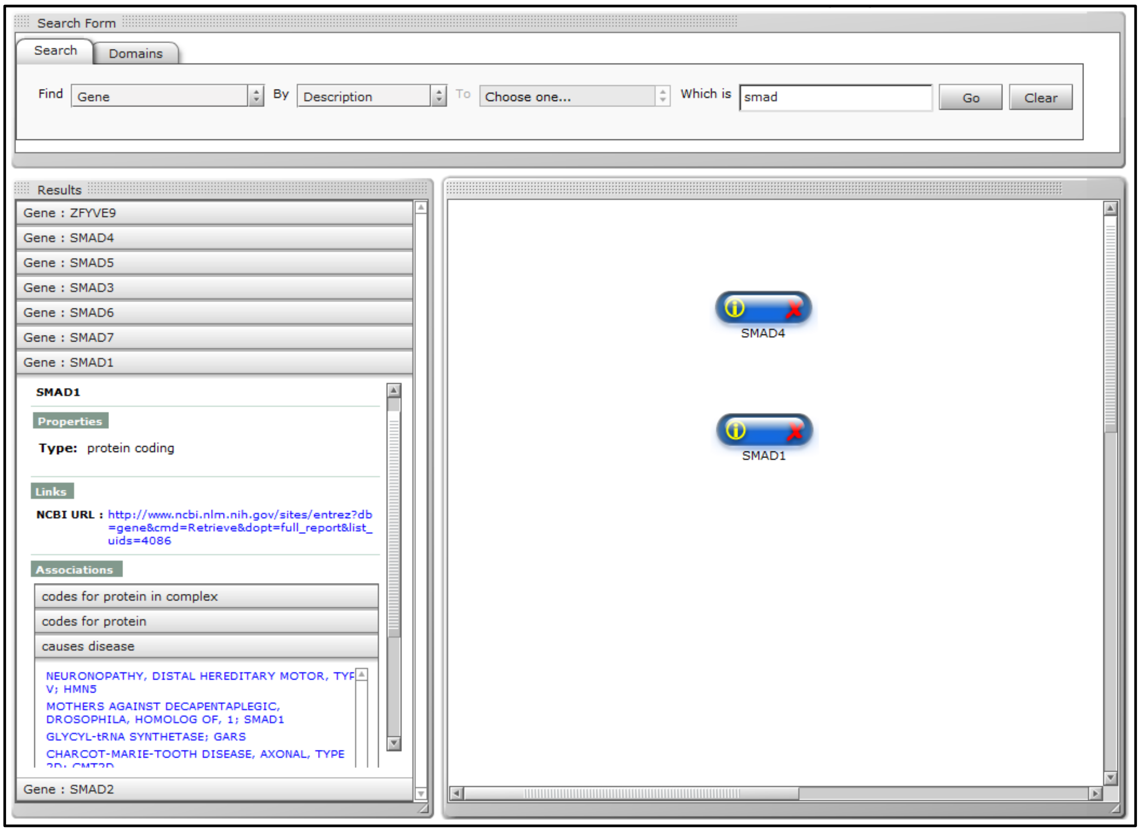
In general, mid-scale or large-scale knowledge networks are generated as XML documents within a Java EE client application by querying the virtual semantic network using the functionalities of the Semantic Manager component (compare *Figure 3-4*). Nevertheless, the virtual semantic network can be also explored by using the GeKnowME web portal, where the user can navigate through the coherent information space and generate small-scale network models.

4.5 Small-Scale Analysis

In comparison to the other two analysis types where the network generation is executed automatically or semi-automatically, in the small-scale analysis the researcher creates manually the models representing the interrelations between the semantically integrated biological entities. Driven from the special scientific interests, a life-scientist can start exploring the information space mapped to a specific knowledge domain by using the navigation techniques provided within the graphical user interface (compare *Figure 3-18* and *section 3.4.4*). He can decide which entities are relevant for his investigations and consider them during the model generation. By building models in the *Model Canvas* of the web portal, he can represent and analyze the interactions between the semantically integrated entities of particular research significance to derive novel insights on an instance level. It is important to mention that the queried information is always up-to-date and semantically correct, since the integration is based on dynamical information retrieval and subject-centric annotation. An additional advantage during the exploration process is that the user is able to expand the investigated information space by considering additional knowledge domains (compare *Figure 3-3*). A brief description of a chosen example follows to illustrate generally the utilization of the GeKnowME framework for a common small-scale analysis.

The biologists, *Kiyono* and *Shibuya*, studying the inhibitory transcription factors of *SMAD* genes and their impact on arterial endothelial cells supposed in their published scientific results that "*The Delta–Notch pathway is a good candidate for the transcriptional activator of SMAD genes in arterial endothelial cells*" (Kiyono, et al., 2006). A GeKnowME user looking for supporting facts is able to search for the subjects of scientific interest, in this case *SMAD* genes, in the preselected knowledge domain "Human Genetic Diseases" within the web portal (see *Search Form* in *Figure 4-14*). Since the user is interested in genes, the required information is searched and retrieved from the related information resources to the association type "Gene", in this case from the Entrez Gene database via the NCBI web service (compare *Figure 4-6*).

Single topics fulfilling the entered search criteria can be explored by viewing their exact properties and associated entities. For instance, the user can retrieve further information related to the *SMAD1* gene as shown in the *Result* portlet depicted in *Figure 4-14*. This information is extracted from distributed resources and semantically integrated by linking it to the subject (topic) *SMAD1*, which is performed within the syntax component *Gene Topic Type*. For example, the properties *Type* and *NCBI URL* are retrieved from the Entrez Gene database, the related protein complexes are extracted from the CORUM database using JDBC connectivity within the corresponding *Resource Wrapper* component, the proteins coded by this gene are delivered from the Swiss-Prot database, and the associated diseases are obtainable from the OMIM database.



Find: Gene
By: Description
To: Choose one...
Which is: smad

Gene : ZFYVE9
Gene : SMAD4
Gene : SMAD5
Gene : SMAD3
Gene : SMAD6
Gene : SMAD7
Gene : SMAD1

SMAD1

Properties

Type: protein coding

Links

NCBI URL : http://www.ncbi.nlm.nih.gov/sites/entrez?db=gene&cmd=Retrieve&dopt=full_report&list_uids=4086

Associations

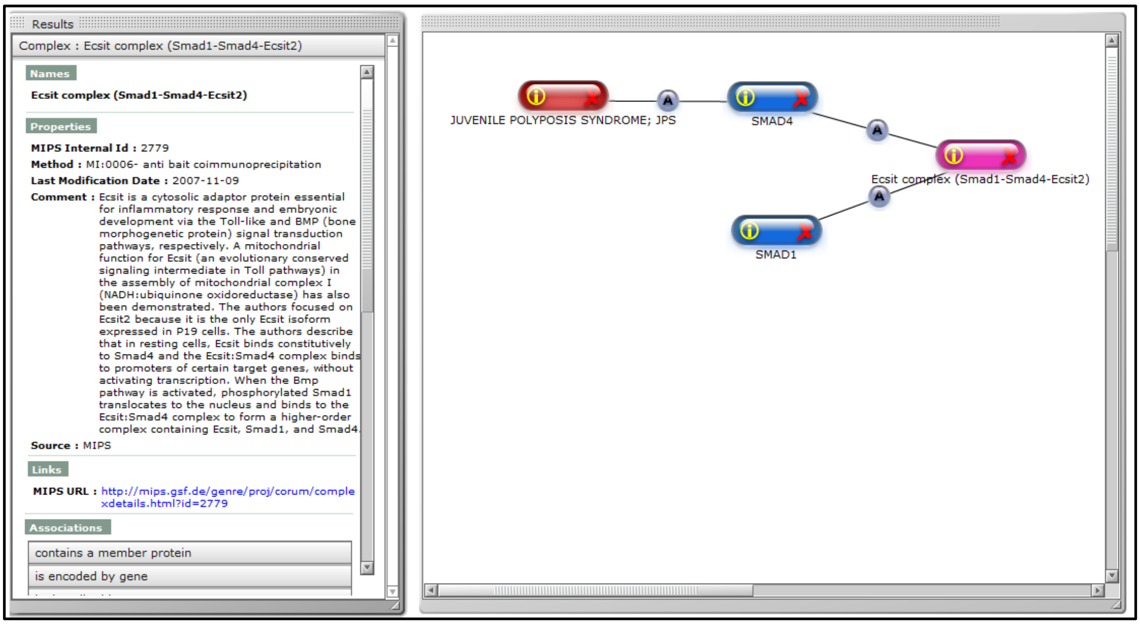
codes for protein in complex
codes for protein
causes disease

NEURONOPATHY, DISTAL HEREDITARY MOTOR, TYPE V; HMN5
MOTHERS AGAINST DECAPENTAPLEGIC, DROSOPHILA, HOMOLOG OF, 1; SMAD1
GLYCYL-tRNA SYNTHETASE; GARS
CHARCOT-MARIE-TOOTH DISEASE, AXONAL, TYPE 2D; CMT2D

Gene : SMAD2

Figure 4-14: Screenshot of the starting steps performed in the exploration process for SMAD genes and their associated entities within the GeKnowME web portal.

Once the user finds significant topics such as the genes *SMAD1* and *SMAD4*, possibly because of their known effect on genetic diseases, he can place them on the model canvas and start building a network of related entities. In the next steps, he can explore the neighbors of the chosen topics, such as the protein complex *Ecsit* (see *Result* portlet in *Figure 4-15*). Depending on their relevance to the analyzed subject matter, they can be appended to the model as shown in *Figure 4-15*. Since the associations between the topics are bidirectional, it is possible to navigate from each topic to the next related entities.


Results
JUVENILE POLYPOSIS SYNDROME; JPS
SMAD4
Ecsit complex (Smad1-Smad4-Ecsit2)
SMAD1

Complex : Ecsit complex (Smad1-Smad4-Ecsit2)

Names

Ecsit complex (Smad1-Smad4-Ecsit2)

Properties

MIPS Internal Id : 2779

Method : MI:0006- anti bait coimmunoprecipitation

Last Modification Date : 2007-11-09

Comment : Ecsit is a cytosolic adaptor protein essential for inflammatory response and embryonic development via the Toll-like and BMP (bone morphogenetic protein) signal transduction pathways, respectively. A mitochondrial function for Ecsit (an evolutionary conserved signaling intermediate in Toll pathways) in the assembly of mitochondrial complex I (NADH:ubiquinone oxidoreductase) has also been demonstrated. The authors focused on Ecsit2 because it is the only Ecsit isoform expressed in P19 cells. The authors describe that in resting cells, Ecsit binds constitutively to Smad4 and the Ecsit:Smad4 complex binds to promoters of certain target genes, without activating transcription. When the Bmp pathway is activated, phosphorylated Smad1 translocates to the nucleus and binds to the Ecsit:Smad4 complex to form a higher-order complex containing Ecsit, Smad1, and Smad4.

Source : MIPS

Links

MIPS URL : http://mips.gsf.de/genre/proj/corum/complexdetails.html?id=2779

Associations

contains a member protein

is encoded by gene

Figure 4-15: Screenshot representing how a user can building step by step a model, which involves the biological entities of scientific interest and their interconnections by exploring the integrated information space. For instance, he can add two further topics (*Juvenile Polyposis Syndrome* and *Ecsit complex*) on the canvas and if there are known associations between all represented topics, they are displayed.

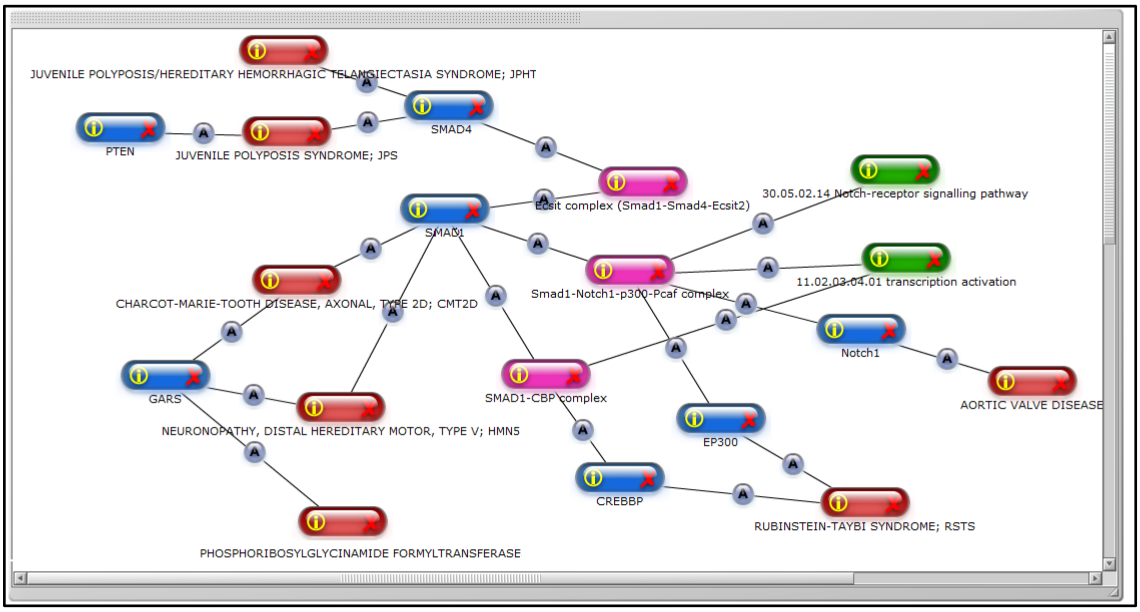

JUVENILE POLYPOSIS/HEREDITARY HEMORRHAGIC TELANGIECTASIA SYNDROME; JPHT
PTEN
JUVENILE POLYPOSIS SYNDROME; JPS
SMAD4
Ecsit complex (Smad1-Smad4-Ecsit2)
30.05.02.14 Notch-receptor signalling pathway
SMAD1
CHARCOT-MARIE-TOOTH DISEASE, AXONAL, TYPE 2D; CMT2D
Smad1-Notch1-p300-Pcaf complex
11.02.03.04.01 transcription activation
Notch1
GARS
SMAD1-CBP complex
AORTIC VALVE DISEASE
NEURONOPATHY, DISTAL HEREDITARY MOTOR, TYPE V; HMN5
EP300
CREBBP
RUBINSTEIN-TAYBI SYNDROME; RSTS
PHOSPHORIBOSYLGLYCINAMIDE FORMYLTRANSFERASE
A

Figure 4-16: A biological model generated during the exploration process for *SMAD* genes. It provides an evidence for the assumption that the Delta–Notch pathway is a good candidate for the transcriptional activator of *SMAD* genes.

The model shown in *Figure 4-16* is generated by following this approach and represents a supporting evidence for the above stated assumption by *Kiyono* and *Shibuya*. In this case, an indirect connection could be found between the *SMAD1* gene and a protein complex involved in the *Notch signaling pathway*. Additionally, the representation of further subject-related entities and their interrelations such as associated genetic diseases may lead to generation of new assumptions and initialization of additional investigations. For instance, one can start analyzing, whether the Notch signaling pathway impacts the *Rubinstein-Taybi Syndrom*.

The illustrated examples in the three groups of analysis types (large-scale, mid-scale, and small-scale) have shown that in the contemporary process of scientific research it is very essential to assemble knowledge coming from adjacent scientific disciplines. Through the consideration of the knowledge domain “Human Genetic Diseases”, it has been demonstrated that by utilizing the GeKnowME framework life-scientists can accelerate the knowledge discovery process, since the significant knowledge can be extracted from distributed information resources that are relevant for the research field and it can be combined in a subject-centric way.

5 Discussion

> *"Alles Wissen und alles Vermehren unseres Wissens endet nicht mit einem Schlusspunkt, sondern mit einem Fragezeichen."*
>
> ***Hermann Hesse (1877 - 1962)***

In the two preceding chapters, it has been described how the GeKnowME framework implements the approach of subject-centric semantic integration, which I have developed to overcome the technical and conceptual challenges regarding the more effective exploitation of existing life-science information resources. It is important to explain which integrative obstacles have been more or less successfully resolved, to discuss the strengths and limitations of the system, and to address the directions of possible future extensions.

5.1 Dynamic Information Retrieval

Since the developed concepts for information integration follow the FDBMS integrative approach, in particular the usage of resource wrappers for dynamic information retrieval (compare *section 2.3.2*), one of the most significant advantages of the GeKnowME system is that the explored information is always up-to-date. The utilization of Web Services, EJBs, ODBC or JDBC connectivity allows the remote information access and querying execution. Unfortunately, there are still communities in the field of life-science that provide no automated access mechanisms to their data collections but offer them just as flat-files for download. Therefore, the inclusion of such information resources into the framework requires not only additional processing procedures such as data indexing for faster access, but also regular downloads and updates. Additionally, if the structure of the provided flat-files changes, the maintaining efforts increase. Nevertheless, since the framework is designed in a modular way, only the corresponding resource wrapper component has to be adjusted and kept consistent. The syntax components associated to such resources do not need any modifications and thus the overall maintenance efforts remain relatively low.

One of the main drawbacks of the dynamic information retrieval concerns the response time of complex queries, because their processing is based on queries distribution and cleansing procedures are performed on-the-fly. The realization of the “Human Genetic Diseases” knowledge domain in combination with implementations of additional test cases has shown that in general the response time is acceptable for queries, which are distributed among up to five distributed resources. Currently, the query distribution is executed sequentially, since some of the sub-queries may rely on others. An important improvement of the system regarding the acceleration of the response time would be the development of more enhanced mechanisms for queries distribution and response processing. For this purpose, mechanisms of the so called *message-oriented middleware* can be adopted to the GeKnowME system architecture. They include the configuration of communication channels for asynchronous message processing, which can increase not only the response time, but also the system’s reliability. For instance, by using messaging, sub-queries can be executed in parallel and obstacles regarding network reliability can be reduced in a straightforward manner.

To cope with the limitation of long responses, it is recommendable to define a group of clear knowledge domain models including only the specific domains of interest and not all possible ones to all potential users. Since the information is semantically annotated, several separate models can be queried simultaneously and the results can be merged together. Depending on the exploration needs, the user can decide how broad the information space for exploration should be and consequently influence the response time.

5.2 Dynamic Subject-Centric Semantic Annotation

The novel approach of dynamic subject-centric semantic annotation has been introduced and implemented in the GeKnowME system to provide a consistent information space for the more effective knowledge discovery by correctly integrating relevant information. Its main advantage is that information coming from distributed resources and having heterogeneous format but regarding the same subject of investigation can be assembled. One of the discussed factors concerning the slow acceptance of the Semantic Web technologies in the life-science domain is the lack of a consistent set of life-science identifiers. In the GeKnowME system, this obstacle is not too crucial, because one can concentrate on a set of predefined knowledge domains and agree on particular identifiers. Therefore, during the syntax processing, the retrieved information can be correctly merged by using known mappings of biological identifiers.

The adoption of the Topic Maps concept for top-down semantic annotation leads to another significant advantage. It is no longer necessary to annotate the large number of existing information resources in the life-science domain with appropriate RDF statements to achieve the vision of the Life-Science Semantic Web. Nevertheless, one can inquire whether the subject-centric approach can be applied for really large-scale information integration. For the realization of such goal, I would suggest a possible future extension of the GeKnowME approach, which key concepts are depicted in *Figure 5-1*.

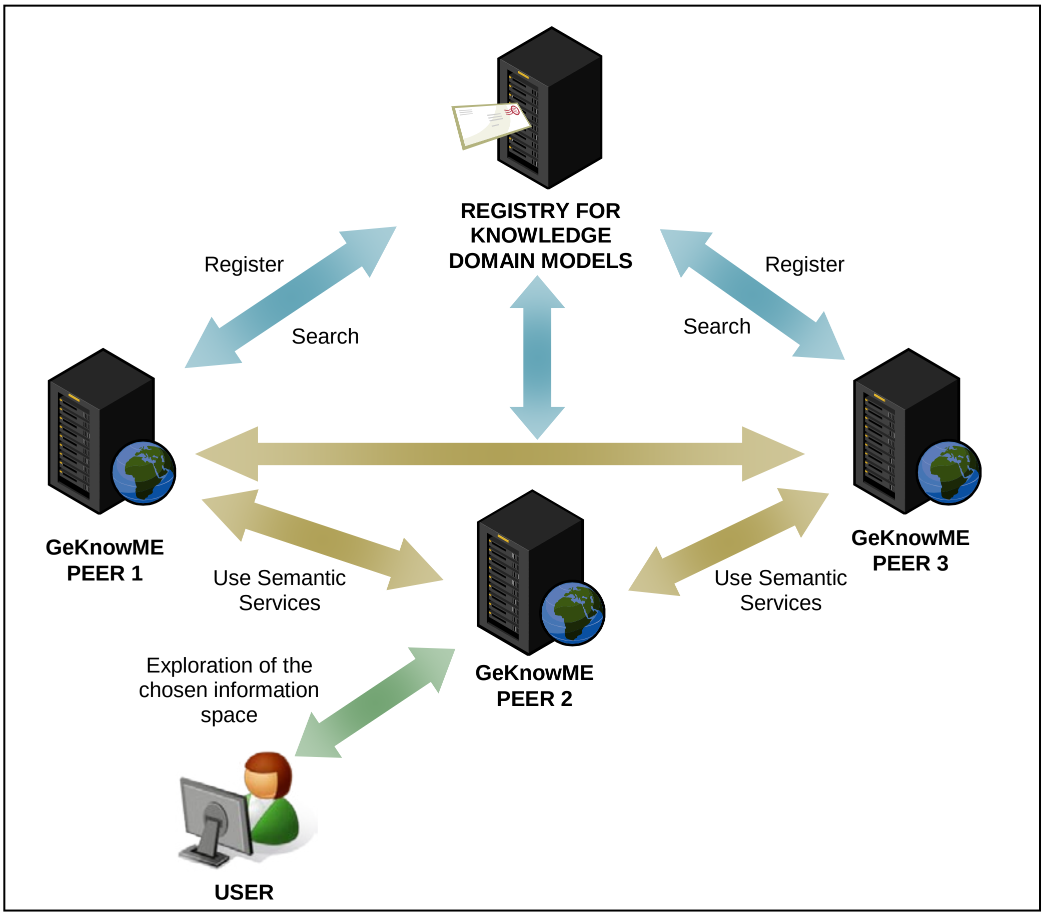


Figure 5-1: Introduction of a public available registry, where knowledge domain models can be registered and looked up by separate GeKnowME nodes over the Internet. If a knowledge domain offered by another peer is relevant for the exploration, its semantic services can be used and the user can explore much broader information space.

Inspired from both the *Universal Description, Discovery and Integration* (UDDI) specification, which defines a registry service for Web Services, and the *Peer Data Management Systems* (compare *section 2.3.2*) one can extend the subject-centric approach by introducing a registry, where separate GeKnowME peers each having an independent semantic manager can register the provided knowledge domains. The main task of the registry is to enable the GeKnowME peers to publish their offered knowledge domains with the involved semantic services. Subsequently, the peers can search the registry for related knowledge domains and retrieve metadata how to utilize them. Once a GeKnowME peer is aware how to interact with another peer over the internet, the offered semantic services can be used (shown as brown arrows in the figure above). For the user, who decides how broad the information space for exploration should be, this interactions remain hidden.

5.3 Knowledge Representation

The GeKnowME framework is designed to represent knowledge in the form of semantic networks of interrelated entities. One of the strength of this kind of representation is that the user can decide which entities are relevant for his exploration and consider them during the network generation. Currently, the user has to arrange manually the entities of the generated small-scale network models. The arranging functionalities of the graphical interface can be improved by offering automated layout mechanisms for the network representation. Additionally, enhanced techniques can be introduced for browsing associated biological ontologies.

A restriction regarding the generation of mid-scale or large-scale semantic networks refers to the fact that currently such kind of networks can be created only in Java client applications. The GeKnowME portal can be extended by implementing corresponding portlets, where the user can specify the criteria for the generation of such networks. The processing of requests for the generation of mid- or large-scale networks has to be executed asynchronously, since the semantic integration tasks for the large amount of distributed information are very time consuming.

An additional improvement of the GeKnowME system may be the support of network analysis methods. The CABiNet (Comprehensive Analysis of Biomolecular Networks) software suite is a generic network analysis system that provides a semi-automatic network processing pipeline for complex analyses (Oesterheld, et al., 2007). New possibilities for examination and exploration open up when these two generic frameworks are bound together. The generated networks in the GeKnowME system can be investigated then in the CABiNet suite. For instance, if necessary the networks can be manipulated, their topological features can be figured out, clustering techniques can be applied, etc..

5.4 Applications

Although diverse biomedical knowledge domains are well suited for the utilization of the developed semantic integrative concepts, the approach of subject-centric information integration implemented in the GeKnowME framework is not exclusively restricted to them. The potency of the GeKnowME system is its generic applicability. The knowledge domain "Human Genetic Diseases" has been introduced mainly to demonstrate the utilization process of the GeKnowME framework. However, this implemented knowledge domain can be used actualy by biologists studying particular human genetic diseases. To provide new knowledge perspectives, further compatible knowledge domain models can be defined and implemented. Such models can represent further influencing factors causing disease phenotypes and be associated to additional relevant information resources. For instance, the knowledge domain model "*Post-transcriptional Regulation by miRNA*", shown in *Figure 5-2*, can be considered in the exploration of miRNA influences on human genetic disorders. Instances of the topic types *Gene* and *Locus* can be used as connection points between the two knowledge domains.

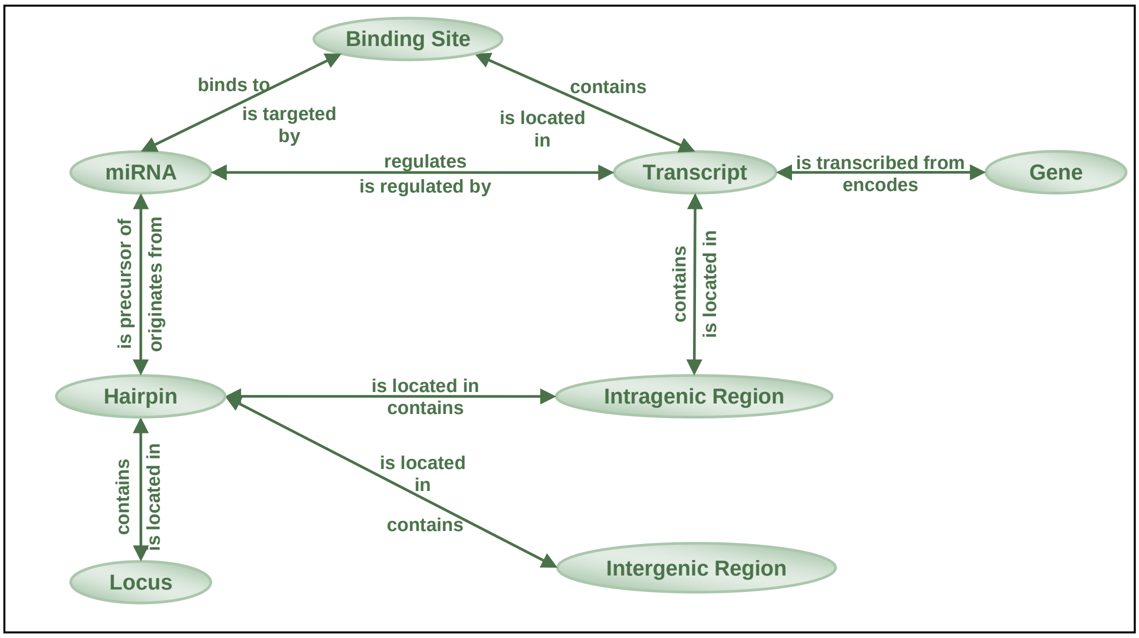


Figure 5-2: *"Post-transcriptional Regulation by miRNA"* knowledge domain model.

A further knowledge domain model, which can be merged not only with the above mentioned models but in general with life-science related domains, is the "*Text Mining*" model (compare *Figure 2-19*). It represents the results generated by the EXCERBT text mining engine, which extracts semantic relations between biological entities from biomedical text. GeKnowME users can decide, whether to explore just the extracted relationships from the literature for the biological entities of interest, or they can use the information space associated to the “Text Mining” knowledge domain for finding evidences for discovered associations in other domains.

6 Conclusion

> *“ Was wir sind, ist nichts,*
> *was wir suchen, ist alles.”*
>
> ***Johann Christian Friedrich Hölderlin***
> ***(1770-1843)***

A novel approach for subject-centric semantic integration applicable to the life-science information space has been introduced in this thesis. The GeKnowME software system, which I implemented for its realization, allows scientists not only to search for biological entities relevant for their research but also to investigate the distributed information space related to these subjects of study.

The designed integration approach bridges effectively the gap between the unconnected islands of biological knowledge represented by the distributed autonomous information resources available in the area of life-science and fulfills the scientific requirements for a coherent information space for exploration. This kind of integration is achievable, because the developed concepts reflect the human associative way of thinking by allowing different scientific communities to define abstract models, which represent only the area of their research in the form of concepts and relationships between them. These knowledge domain models can be associated only to resources containing relevant information concerning the research of interest and consequently they reduce the overall information complexity. The correct interconnection of the entities available from the distributed heterogeneous information resources is achievable by the introduction of on-the-fly semantic annotation, which adopts the Topic Maps knowledge representation techniques. Moreover, a model can be simply combined with other models sharing the same concepts and thus a scientific community can obtain a much broader overview of the subject matter if needed. To provide not only consistent but also up-to-date information, modern integration technologies for remote information querying and access are implemented in the GeKnowME framework.

The GeKnowME system is designed to be applicable for a broad range of use cases not only in the diverse fields of life-science, but also in other knowledge based areas. This significant generic feature is attainable, because the implementation of the framework follows well established software design patterns and applies state-of-the-art software development technologies. The functional complexity is decomposed in five segregated layers to separate the conceptual principles. Additionally, each functional level is based on software components that encapsulate specific functionality. This multi-tier, component-oriented architecture allows the reusability of already developed modules and reduces the maintenance efforts. Moreover, this approach increases the flexibility in the realization or extension of new or existing knowledge domain models and makes the system highly scalable, e.g. a new information resource can be straightforwardly plugged in to the GeKnowME system and mapped to the correspondent knowledge domain model.

Scientists studying the human genome regarding genetic disorders and the resulting hereditary disease phenotypes can use the GeKnowME framework to explore simultaneously significant information resources such as the OMIM, SWISS-Prot, and CORUM databases in the form of a giant semantic network of interconnected biological entities. They can investigate the interrelations of the subjects of scientific interest and find indirect relations between the involved entities, which are not obvious by looking for them into the single information resources. The inference of such novel insights improves the process of knowledge discovery and can be performed in three different ways: large-, mid-, and small-scale analysis, depending on how many entities are considered in the scientific exploration.

The implementation of the “Human Genetic Diseases” knowledge domain model along with the potential for the realization of further biological models provides a suitable example how the GeKnowME system with the embedded subject-centric semantic integration approach can be utilized. The GeKnowME system supports biologist in their research endeavors to understand the complexity of life and in particular the mechanisms of diseases by allowing them to model, organize, and assemble knowledge coming from various sub-disciplines and available in existing distributed life-science information resources.

List of Abbreviations

ACL	Agent Communication Language
API	Application Programming Interface
BIS	Biological Information Systems
CORUM	Comprehensive Resource of Mammalian protein complexes
DBMS	Database Management System
DHTML	Dynamic HTML
DNA	Deoxyribonucleic acid
DW	Data Warehouse
EAI	Enterprise Application Integration
EJB	Enterprise JavaBean
ETL	Extraction, Transformation, and Loading
EXCERBT	EXtraction of Classified Entities and Relations from Biomedical Text
FC	Functional Category
FDBMS	Federated Database Management System
FunCat	Functional Catalogue
GeKnowME	Generic Knowledge Modeling Environment
GO	Gene Ontology
GUI	Graphical User Interface
HGNC	HUGO Gene Nomenclature Committee
HGP	Human Genome Project
HTML	Hypertext Markup Language
HTTP	Hypertext Transport Protocol
HUGO	The Human Genome Organization
ICD-10	International Classification of Diseases Version 10
IGD	Integrative Genome Database
IIOP	Internet Inter Object request broker Protocol
IIS	Information Integration System
ISO	International Organization for Standardization
Java EE	Java Platform, Enterprise Edition; formerly Java 2 Platform, Enterprise Edition (J2EE)

JAX-WS	Java API for XML Web Services
JDBC	Java Database Connectivity
JNDI	Java Naming and Directory Interface
JPA	Java Persistence API
JSR-168	Java Standardization Request-168
KR	Knowledge Representation
KQL	Knowledge Querying Language
LSSW	Life-Science Semantic Web
MAS	Multi-Agent System
MEMEX	MEMory EXtender
MeSH	Medical Subject Headings
MGI	Mouse Genome Informatics
MIPS	Institute for Bioinformatics and Systems Biology, formerly Munich Institute for Protein Sequencing
MPO	MGI Mammalian Phenotype Ontology
NAR	Nucleic Acid Research
NCBI	US National Center for Biotechnology Information
NHGRI	National Human Genome Research Institute
NIH	US National Institute of Health
NLM	US National Library of Medicine
NLS	oN-Line System
OBO	Open Biomedical Ontologies, formerly Open Biological Ontologies
ODBC	Open Database Connectivity
OWL	Web Ontology Language
P2P	Peer To Peer
PDMS	Peer Data Management Systems
PSI	Published Subject Indicator
PSI-MI	Proteomics Standards Initiative – Molecular Interaction
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
RIA	Rich Internet Application
RMI	Remote Method Invocation
RNA	Ribonucleic acid
SAX	Simple API for XML
SIMAP	Similarity Matrix of Proteins
SKP	Semantic Knowledge Representation
TAO	Topics, Associations, and Occurrences
TLR	Toll-Like Receptor
TM	Topic Maps
TM4J	Topic Maps For Java
TMAPI	Topic Map Application Programming Interface

UDDI	Universal Description, Discovery and Integration
UML	Unified Modeling Language
UMLS	Unified Medical Language System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
W3C	World Wide Web Consortium
WHO	World Health Organization
WS	Web Service
WWW	World Wide Web
XHTML	Extensible Hypertext Markup Language
XML	Extensible Markup Language

List of Figures

List of Tables

References

Baclawski Kenneth and Tianhua Niu Ontologies for Bioinformatics [Book]. - [s.l.] : The MIT Press, 2006. - pp. 35-38. - ISBN: 0262025914.

Bader Gary D, Betel Doron and Hogue Christopher WV BIND: the Biomolecular Interaction Network Database. [Journal] // Nucleic Acids Res. - 2003. - Vol. 31. - pp. 248-250.

Baker Christopher J.O. and Cheung Kei-Hoi Semantic Web Revolutionizing Knowledge Discovery in the Life Sciences [Book]. - [s.l.] : Springer Science + Business Media, LLC, 2007. - ISBN: 0387484361.

Barabási Albert-László and Oltvai Zoltán N Network biology: understanding the cell's functional organization. [Journal] // Nat Rev Genet. - 2004. - Vol. 5. - pp. 101-113.

Barnickel Thorsten [et al.] Semantic role labeling with neural networks [Article] // Bioinformatics. - 2008. - in processing, submitted Sep.2008.

Benson Dennis A [et al.] GenBank. [Journal] // Nucleic Acids Res. - 2008. - Vol. 36. - pp. D25--D30.

Berman Helen [et al.] The worldwide Protein Data Bank (wwPDB): ensuring a single, uniform archive of PDB data. [Journal] // Nucleic Acids Res. - 2007. - Vol. 35. - pp. D301--D303.

Berners-Lee Tim and Cailliau Richard WorldWideWeb: Proposal for a HyperText Project. - [s.l.] : CERN, 1990.

Berners-Lee Tim Semantic Web Road Map. - 1998.

Boeckmann Brigitte [et al.] The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. [Journal] // Nucleic Acids Res. - 2003. - Vol. 31. - pp. 365-370.

Bolós Victoria, Grego-Bessa Joaquín and de José Luis Notch signaling in development and cancer. [Journal] // Endocr Rev. - 2007. - Vol. 28. - pp. 339-363.

Botstein David and Risch Neil Discovering genotypes underlying human phenotypes: past successes for mendelian disease, future approaches for complex disease. [Journal] // Nat Genet. - 2003. - Vol. 33 Suppl. - pp. 228-237.

Brayner Angelo, Meirelles Marcelo and Filho Jose de Aguiar Integrating Heterogenous Data Sources in the Web [Book Section] // Web Data Management Practices: Emerging Techniques & Technologies / book auth. Athena Vakali and George Pallis. - [s.l.] : IGI Publishing, 2006. - ISBN: 1599042282.

Buchholz William Ontology [Book Section] // Encyclopedia of Knowledge Management / book auth. Schwartz David. - [s.l.] : Idea Group Inc., 2006. - ISBN 1591405734.

Bult Carol J [et al.] The Mouse Genome Database (MGD): mouse biology and model systems. [Journal] // Nucleic Acids Res. - 2008. - Vol. 36. - pp. D724--D728.

Burger Albert Agent Technologies in Life Sciences [Book Section] // Semantic Web Revolutionizing Knowledge Discovery in the Life Sciences / book auth. Baker Christopher J.O. and Cheung Kei-Hoi. - [s.l.] : Springer Science + Business Media, LLC, 2007. - ISBN: 0387484361.

Bush Vannevar As We May Think [Article] // Atlantic Monthly. - 1945.

CERN The website of the world's first-ever web server [Online]. - 2008. - Mai 05, 2008. - http://info.cern.ch/.

Channabasavaiah Kishore, Holley Kerrie and Tuggle Edward Jr. Migrating to a service-oriented architecture [Journal] // IBM developerWorks. - 2003. - SOA and Web services.

Cheung Kei-Hoi [et al.] YeastHub: a semantic web use case for integrating data in the life sciences domain. [Journal] // Bioinformatics. - 2005. - Vol. 21 Suppl 1. - pp. i85--i96.

Chiba Shigeru Notch signaling in stem cell systems. [Journal] // Stem Cells. - 2006. - Vol. 24. - pp. 2437-2447.

Chute Christopher G. Medical Concept Representation [Book Section] // Medical Informatics Knowledge Management and Data Mining in Biomedicine / book auth. Chen Hsinchun [et al.]. - [s.l.] : Springer Science + Business Media, Inc., 2005. - ISBN: 0387243811.

Clark Tim and Kinoshita June Alzforum and SWAN: the present and future of scientific web communities. [Journal] // Brief Bioinform. - 2007. - Vol. 8. - pp. 163-171.

Clark Tim Special Issue: Knowledge Integration and Web Communities [Article] // Briefings in Bioinformatics. - 2007. - 3. - Vol. 8.

Collins Francis S. and McKusick Victor A. Implications of the Human Genome Project for medical science. [Journal] // JAMA. - 2001. - Vol. 285. - pp. 540-544.

Crasto Chiquito J [et al.] SenseLab: new developments in disseminating neuroscience information. [Journal] // Brief Bioinform. - 2007. - Vol. 8. - pp. 150-162.

Croasdell David. and Wang Y. Ken Virtue-Nets [Book Section] // Encyclopedia of Knowledge Mangement / book auth. Schwartz David G.. - [s.l.] : Idea Group Inc., 2006. - ISBN: 1591405734.

Daconta Michael C. Understanding Ontologies [Book Section] // The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management / book auth. Daconta Michael C., Obrst Leo J. and Smith Kevin T.. - [s.l.] : John Wiley & Sons, LTD, 2003. - ISBN: 0471432571.

Daconta Michael C., Obrst Leo J. and Smith Kevin T. The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management [Book]. - [s.l.] : John Wiley & Sons, LTD, 2003.

Davies John, Fensel Dieter and van Harmelen Frank Towards the Semantic Web: Ontology-driven Knowledge Management [Book]. - [s.l.] : John Wiley & Sons Ltd., 2003. - pp. 1-9. - ISBN: 0470848677.

Feigenbaum Lee [et al.] Boca: an open-source RDF store for building Semantic Web applications. [Journal] // Brief Bioinform. - 2007. - Vol. 8. - pp. 195-200.

Fortier Jean-Yves and Kassel Gilles Organizational Semantic Webs [Book Section] // Encyclopedia of Knowledge Management / book auth. Schwartz David. - [s.l.] : Idea Group Inc., 2006. - ISBN: 1591405734.

Fowler Martin Patterns of Enterprise Application Architecture [Book]. - USA : Addison Wesley, 2002. - pp. 1-9. - ISBN: 0321127420.

Freimer Nelson and Sabatti Chiara The human phenome project. [Journal] // Nat Genet. - 2003. - Vol. 34. - pp. 15-21.

Galperin Michael Y. The Molecular Biology Database Collection: 2008 update [Journal] // Nucleic Acid Res. - 2008. - Vol. 36. - pp. D2-4.

Gamma Erich Design Patterns Elements of Reusable Object-Oriented Software [Book]. - [s.l.] : Addison-Wesley, 1995. - pp. 1-29. - ISBN: 0201633612.

Gene Ontology Consortium The Gene Ontology (GO) project in 2006. [Journal] // Nucleic Acids Res. - 2006. - Vol. 34. - pp. D322--D326.

Gerstein Mark B [et al.] What is a gene, post-ENCODE? History and updated definition. [Journal] // Genome Res. - 2007. - Vol. 17. - pp. 669-681.

Ghosh Soumitra and Collins Francis S. The geneticist's approach to complex disease. [Journal] // Annu Rev Med. - 1996. - Vol. 47. - pp. 333-353.

Gillman Daniel W. Data Semantics [Book Section] // Encyclopedia of Knowledge Management / book auth. David Schwartz. - [s.l.] : Idea Group Inc., 2006. - ISBN: 1591405734.

Goh Kwang-Il [et al.] The human disease network. [Journal] // Proc Natl Acad Sci U S A. - 2007. - Vol. 104. - pp. 8685-8690.

Good Benjamin M and Wilkinson Mark D The Life Sciences Semantic Web is full of creeps! [Journal] // Brief Bioinform. - 2006. - Vol. 7. - pp. 275-286.

Gribble Steven [et al.] What Can Databases Do for Peer-to-Peer? [Conference]. - Santa Barbara, California, USA : WebDB Workshop on the Web and Databases , 2001.

Griffiths Anthony J. F. and et.al. Gene Interaction: From genes to phenotypes [Book Section] // An Introduction to Genetic Analysis. - [s.l.] : W.H.Freeman & Co Ltd, 2000. - ISBN: 071673771X.

Groth Philip [et al.] PhenomicDB: a new cross-species genotype/phenotype resource. [Journal] // Nucleic Acids Res. - 2007. - Vol. 35. - pp. D696--D699.

Hamosh Ada [et al.] Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. [Journal] // Nucleic Acids Res. - 2005. - Vol. 33. - pp. D514--D517.

Heese Ralf [et al.] Self-Extending Peer Data Management [Conference]. - Karlsruhe, Germany : Datenbanksysteme in Business, Technologie und Web (BTW 2005), 2005.

Hendler James Communication. Science and the semantic web. [Journal] // Science. - 2003. - Vol. 299. - pp. 520-521.

Hernández Helena [et al.] Subunit architecture of multimeric complexes isolated directly from cells. [Journal] // EMBO Rep. - 2006. - Vol. 7. - pp. 605-610.

HGNC HUGO Gene Nomenclature Committee [Online]. - July 16, 2007. - April 21, 2008. - http://www.genenames.org/.

Holland R. [et al.] BioJava: an open-source framework for bioinformatics. [Journal] // Bioinformatics. - 2008. - Vol. 24. - pp. 2096-2097.

ISO Topic Maps. - [s.l.] : International Organization for Standardization - Information technology, 11 07, 2007.

Jacobs Ian and Walsh Norman Architecture of the World Wide Web [Article] // W3C Recomondations. - 2004. - Vol. 1.

Jensen Lars Juhl, Saric Jasmin and Bork Peer Literature mining for the biologist: from information retrieval to biological discovery. [Journal] // Nat Rev Genet. - 2006. - Vol. 7. - pp. 119-129.

Keele John W and Wray James E Software agents in molecular computational biology. [Journal] // Brief Bioinform. - 2005. - Vol. 6. - pp. 370-379.

Kiyono Mari and Shibuya Masabumi Inhibitory Smad transcription factors protect arterial endothelial cells from apoptosis induced by BMP4. [Journal] // Oncogene. - 2006. - Vol. 25. - pp. 7131-7137.

Kruchten Philippe The 4+1 View Model of Architecture [Journal] // IEEE Software. - 1995. - Vol. 12. - pp. 42 - 50.

Lage Kasper [et al.] A human phenome-interactome network of protein complexes implicated in genetic disorders. [Journal] // Nat Biotechnol. - 2007. - Vol. 25. - pp. 309-316.

Lambrix Patrick [et al.] Biological Ontologies [Book Section] // Semantic Web Revolutionizing Knowledge Discovery in the Life Sciences / book auth. Baker Christopher J.O. and Cheung Kei-Hoi. - [s.l.] : Springer Science + Media, LLC, 2007. - ISBN: 0387484361.

Landow George P. Hypertext 2.0: The Convergence of Contemporary Critical Theory and Technology [Book]. - [s.l.] : The Johns Hopkins University Press, 1997. - ISBN: 0801855861.

Leser Ulf and Naumann Felix Informationsintegration: Architekturen und Methoden zur Integration verteilter und heterogener Datenquelle [Book]. - [s.l.] : Dpunkt Verlag, 2006. - pp. 7-10. - ISBN: 3898644006.

Levy Samuel [et al.] The diploid genome sequence of an individual human. [Journal] // PLoS Biol. - 2007. - Vol. 5. - p. e254.

Liebel Urban, Kindler Bjoern and Pepperkok Rainer Bioinformatic "Harvester": a search engine for genome-wide human, mouse, and rat protein resources. [Journal] // Methods Enzymol. - 2005. - Vol. 404. - pp. 19-26.

Loscalzo Joseph, Kohane Isaac and Barabasi Albert-Laszlo Human disease classification in the postgenomic era: a complex systems approach to human pathobiology. [Journal] // Mol Syst Biol. - 2007. - Vol. 3. - p. 124.

Louie Brenton [et al.] Data integration and genomic medicine. [Journal] // J Biomed Inform. - 2007. - Vol. 40. - pp. 5-16.

Mader Sylvia S. Textbook: Biology [Book Section]. - New York : McCraw-Hill, 2004. - ISBN: 0072418826.

Maglott Donna [et al.] Entrez Gene: gene-centered information at NCBI. [Journal] // Nucleic Acids Res. - 2007. - Vol. 35. - pp. D26--D31.

Manola Frank and Miller Eric RDF Primer [Article] // W3C Recommendation. - 2004.

Mazzocchi Stefano, Garland Stephen and Lee Ryan SIMILE: Practical Metadata for the Semantic Web [Article] // O'Reilly XML.com. - 2005.

Microsoft Corporation .NET Framework Developer Center [Online]. - Microsoft Corporation, 2008. - 05 28, 2008. - http://msdn.microsoft.com/en-us/netframework/default.aspx.

Mishra Gopa R [et al.] Human protein reference database--2006 update. [Journal] // Nucleic Acids Res. - 2006. - Vol. 34. - pp. D411--D414.

MIT Haystack: Research on Information Access, Analysis, Management, and Distribution [Online] // Massachusetts Institute of Technology. - Computer Science & Artificial Intelligence Laboratory, 2008. - Mai 06, 2008. - http://groups.csail.mit.edu/haystack/.

NCBI - OMIM OMIM Statistics for July 6, 2008 [Online]. - June 06, 2008. - July 07, 2008. - http://www.ncbi.nlm.nih.gov/Omim/mimstats.html.

NCBI Entrez Utilities Web Service [Online]. - Januar 28, 2008. - April 20, 2008. - http://www.ncbi.nlm.nih.gov/entrez/query/static/esoap_help.html.

NCBI Model of Entrez Databases [Online]. - September 28, 2007. - October 11, 2008. - http://www.ncbi.nlm.nih.gov/Database/datamodel/index.html.

Neumann Eric A Life Science Semantic Web: Are we there yet? [Journal] // Science STKE. - 2005. - pe22 : Vol. 283.

Neuroscientific Net The Semantic Synapse Project [Online] // Weaving the Biomedical Semantic Web. - Januar 30, 2008. - Mai 06, 2008. - http://neuroscientific.net/semantic.

Newcomb Steven R. A Perspective on the Quest for Global Knowledge Interchange [Book Section] // XML Topic Maps Creating and Using Topic Maps for the Web / book auth. Park Jack. - [s.l.] : Addison-Wesley, 2003. - ISBN: 0201749602.

NHGRI Understanding Our Genetic Inheritance [Online] // Human Genome Project's Five-Year Plan (1991-1995). - National Human Genome Research Institute, 1990. - 04 15, 2008. - http://www.genome.gov/10001477.

NLM Medical Subject Headings [Online]. - January 14, 2008. - Mai 04, 2008. - http://www.nlm.nih.gov/mesh/.

NLM Semantic Knowledge Representation [Online]. - December 13, 2007. - Mai 04, 2008. - http://skr.nlm.nih.gov/.

NLM Unified Medical Language System [Online]. - April 11, 2008. - Mai 04, 2008. - http://www.nlm.nih.gov/research/umls/.

Nyce James M. and Kahn Paul From Memex To Hypertext [Book]. - [s.l.] : Academic Press Inc., 1991. - ISBN: 0125232705.

Oda Kanae and Kitano Hiroaki A comprehensive map of the toll-like receptor signaling network. [Journal] // Mol Syst Biol. - 2006. - Vol. 2. - p. 2006.0015.

Oesterheld Matthias, Mewes Hans W. and Stümpflen Volker Analysis of integrated biomolecular networks using a generic network analysis suite [Journal] // Journal of Integrative Bioinformatics. - 2007. - Vol. 4(3).

Orchard Sandra [et al.] Annual Spring Meeting of the Proteomics Standards Initiative 23-25 April 2008, Toledo, Spain. [Journal] // Proteomics. - 2008.

Oti Martin and Brunner Han G. The modular nature of genetic diseases. [Journal] // Clin Genet. - 2007. - Vol. 71. - pp. 1-11.

Oti Martin, Huynen Martijn A and Brunner Han G Phenome connections. [Journal] // Trends Genet. - 2008. - Vol. 24. - pp. 103-106.

Pepper Steve Everything is a Subject [Conference]. - Norway, Oslo : The Second International Topic Maps Users Conference, 2008.

Pepper Steven The TAO of Topic Maps: finding the way in the age of infoglut. [Article] // Proceedings of XML Europe 2000. - 2000. - pp. 11-01.

Pereira-Leal Jose B [et al.] Evolution of protein complexes by duplication of homomeric interactions. [Journal] // Genome Biol. - 2007. - Vol. 8. - p. R51.

Philippi Stephan and Köhler Jacob Addressing the problems with life-science databases for traditional uses and systems biology. [Journal] // Nat Rev Genet. - 2006. - Vol. 7. - pp. 482-488.

Quan Dennis Improving life sciences information retrieval using semantic web technology. [Journal] // Brief Bioinform. - 2007. - Vol. 8. - pp. 172-182.

Roberts Eric J IV. - Plato's View of the Soul [Article] // Mind (Oxford Jounal about Humanities). - 1905. - XIV. - Vol. 3. - pp. 297-440.

Ruepp Andreas [et al.] CORUM: the comprehensive resource of mammalian protein complexes. [Journal] // Nucleic Acids Res. - 2008. - Vol. 36. - pp. D646--D650.

Ruepp Andreas [et al.] The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. [Journal] // Nucleic Acids Res. - 2004. - Vol. 32. - pp. 5539-5545.

Rzhetsky Andrey [et al.] Probing genetic overlap among complex human phenotypes. [Journal] // Proc Natl Acad Sci U S A. - 2007. - Vol. 104. - pp. 11694-11699.

Shedroff Nathan An Overview of Understanding [Book Section] // Information Anxiety 2. - [s.l.] : Que, 2001. - ISBN: 0789724103.

Sigel Alexander Topic Maps in Knowledge Organization [Book Section] // XML Topic Maps Creating and Using Topic Maps for the Web / book auth. Park Jack. - [s.l.] : Addison-Wesley, 2003. - ISBN: 0201749602.

Smith Barry [et al.] The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. [Journal] // Nat Biotechnol. - 2007. - Vol. 25. - pp. 1251-1255.

Smith Kevin T. Understanding Taxonomies [Book Section] // The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management / book auth. Daconta Michael C., Obrst Leo J. and Smith Kevin T.. - [s.l.] : Wiley Publishing, Inc., 2003. - ISBN: 0471432571.

Sowa John Knowledge Representation: Logical, Philosophical, and Computational Foundations [Book]. - [s.l.] : Course Technology, 1999. - ISBN: 0534949657.

Stein Lincoln D Integrating biological databases. [Journal] // Nat Rev Genet. - 2003. - Vol. 4. - pp. 337-345.

Sun Microsystems, Inc. JSR 244: Java Platform, Enterprise Edition 5 (Java EE 5) Specification. - [s.l.] : Java Specification Requests, 2006.

Sun Microsystems, Inc. The Java EE Tutorial [Book]. - [s.l.] : Sun Microsystems Press, 2007. - Part No: 819-3669-10.

van Driel Marc A [et al.] A text-mining analysis of the human phenome. [Journal] // Eur J Hum Genet. - 2006. - Vol. 14. - pp. 535-542.

Vousden Karen H and Prives Carol P53 and prognosis: new insights and further complexity. [Journal] // Cell. - 2005. - Vol. 120. - pp. 7-10.

W3C World Wide Web Consortium Official Website [Online]. - W3C, Mai 02, 2008. - Mai 05, 2008. - http://www.w3.org/.

Waltz Edward Knowledge Management in the Intelligence Enterprise [Book]. - [s.l.] : Artech House, 2003. - pp. 50-60. - ISBN: 1580534945.

Wang Xiaoshu, Gorlitsky Robert and Almeida Jonas S From XML to RDF: how semantic web technologies will change the design of 'omic' standards. [Journal] // Nat Biotechnol. - 2005. - Vol. 23. - pp. 1099-1103.

Wedberg Anders A History of Philosophy. Volume 1: Antiquity and the Middle Ages [Book]. - [s.l.] : Clarendon Press, 1982.

Westphal Daniel and Bizer Chris RAP - RDF API for PHP [Online]. - Februar 29, 2008. - Mai 06, 2008. - http://www4.wiwiss.fu-berlin.de/bizer/rdfapi/.

Wheeler David L [et al.] Database resources of the National Center for Biotechnology Information [Journal] // Nucleic Acids Res.. - 2008. - Vol. 36. - pp. D13 - D21.

WHO International Classification of Diseases (ICD) [Online]. - April 05, 2006. - Mai 04, 2008. - http://www.who.int/classifications/apps/icd/icd10online/.

Xu Jianzhen and Li Yongjin Discovering disease-genes by topological features in human protein-protein interaction network. [Journal] // Bioinformatics. - 2006. - Vol. 22. - pp. 2800-2805.

Zarri Gian Piero Knoweledge Representation [Book Section] // Encyclopedia of Knowledge Managemen / book auth. Schwartz David. - [s.l.] : Idea Group Inc., 2006. - ISBN: 1591405734.

Appendix A

Subset of the OMIM Morbid Map containing all genetic disorders assigned to the morbid class “*Alzheimer disease*”.

Disorder name	Gene symbols	Chromo-some
Alzheimer disease-1, APP-related (3)	APP, AAA, CVAP, AD1	21q21
Alzheimer disease-2, 104310 (3)	APOE, AD2	19q13.2
Alzheimer disease-4, 606889 (3)	PSEN2, AD4, STM2	1q31-q42
Alzheimer disease, late-onset, 104300 (3)	APBB2, FE65L1	4p14
Alzheimer disease, late-onset, susceptibility to, 104300 (3)	NOS3	7q36
Alzheimer disease, late-onset, susceptibility to, 104300 (3)	PLAU, URK	10q24
Alzheimer disease, susceptibility to, 104300 (3)	ACE, DCP1, ACE1	17q23
Alzheimer disease, susceptibility to, 104300 (3)	MPO	17q23.1
Alzheimer disease, susceptibility to, 104300 (3)	PACIP1, PAXIP1L, PTIP	7q36
Alzheimer disease, susceptibility to (3)	A2M	12p13.3-p12.3
Alzheimer disease, susceptibility to (3)	BLMH, BMH	17q11.2
Alzheimer disease, type 3, 607822 (3)	PSEN1, AD3	14q24.3
Alzheimer disease, type 3, with spastic paraparesis and apraxia, 607822 (3)	PSEN1, AD3	14q24.3
Alzheimer disease, type 3, with spastic paraparesis and unusual plaques, 607822 (3)	PSEN1, AD3	14q24.3

Appendix B

Disease Categories
Bone
Cancer
Cardiovascular
Connective tissue
Dermatological
Developmental
Ear, Nose, Throat
Endocrine
Gastrointestinal
Hematological
Immunological
Metabolic
Muscular
Neurological
Nutritional
Ophthalmological
Psychiatric
Renal
Respiratory
Skeletal
Multiple
Unclassified

Curriculum Vitae

MSc(CompSc) Karamfilka Krasimirova Nenova

Personal Information

Birthday:	November 14, 1979
Place of Birth:	Pomorie, Bulgaria
Nationality:	Bulgarian
Marital Status:	Single

Employment History

Since May 2005
HelmholtzZentrum München
Institute for Bioinformatics and Systems Biology
Position: Ph.D. Student
Focus: Semantic information integration for the life-science knowledge domain

Education

Sep. 2002 – Feb. 2005
University of Applied Sciences Darmstadt
Study: Joint International Master of Computer Science
Exchange semester: University of Wisconsin-Platteville, USA
Qualification: Master of Computer Science
Thesis: “Data Mining as Part of Medical Controlling to support the Introduction of the German-Diagnosis-Related-Groups System”
Grade: 1.0 (granted the Computer Science Department Award)

Sep. 2000 – Aug. 2002
University of Applied Sciences Darmstadt
Study: Computer Science
Qualification: Bachelor of Computer Science
Thesis: „Evaluation eines Software-Entwicklungs-Tools auf Basis eines Java-Applikation-Servers anhand einer Realisierung einer WEB-Applikation“
Grade: 1.4

Sep.1998 – Aug. 2000
University of National and World Economy, Sofia, Bulgaria
Study: Business Information Systems

Sep.1992 – Jul. 1998
High School for Natural Sciences and Mathematics, Burgas
Majors: Mathematics, Physics, Computer Science, English
Grade: 1.0

Practical Experience

Sep. 2004 – Feb. 2005	**University of Applied Sciences Darmstadt**	
	Position:	Tutor in courses *Information Integration* and *Databases*
Apr. 2004 – Oct. 2004	**University Hospital Clinic for Anesthesiology**, **Erlangen**	
	Position:	Research Assistant
	Focus:	Data Mining for Medicine Controlling
Oct. 2002 – Jun. 2003	**UBS Deutschland, Frankfurt/Main**	
	Department:	Business & Information Technology
	Position:	Intern
	Focus:	Business processing, sales management and project management assistance
Sep. 2001 – Feb. 2002	**Logica PDV GmbH, Mainz**	
	Position:	Intern
	Focus:	Evaluation of the software development tool *jBear (Lintec Solution GmbH)*

Computer Skills

Programming:	Java, JavaEE, C/C++, C#
Web Development:	HTML, XML/XSL, JSP, JSF, OpenLaszlo, Liferay, Portlets
UML-Tools:	MS Visio, Visual Paradigm, Together, Rational Rose
OS:	Microsoft Windows, Unix (Linux), DOS
Databases:	MySQL, PostgreSQL, MS Access, Oracle
IDE:	NetBeans, Eclipse, InteliJ IDEA, XML Spy, C# .IDE
AS:	GlassFish, JBoss, Tomcat
Data Mining Tools:	SPSS Clementine

Language Skills

Bulgarian:	mother tongue
German:	business fluent
English:	business fluent
Russian:	fluent
Korean:	solid basics

Further Activities

1992 – 1998	An active handball player at national and international level

Interests

Jogging, Origami, Traveling, Sports

Publication Record

Publications as primary author

Stümpflen, Volker; Barnickel, Thorsten; Nenova, Karamfilka *Large Scale Knowledge Representation of Distributed Biomedical Information Scaling Topic Maps.*, Lecture Notes in Artificial Intelligence, Vol. 4999, pp. 116-127, *Scaling Topic Maps*, Third International Conference on Topic Map Research and Applications, TMRA 2007 Leipzig, Germany, October 11-12, 2007, ISBN: 978-3-540-70873-5

Stümpflen, Volker; Gregory, Richard; Nenova, Karamfilka *From Biological Data to Biological Knowledge.*, Lecture Notes in Artificial Intelligence, Vol. 4438, pp. 62-66, *Leveraging the Semantics of Topic Maps*, Second International Conference on Topic Maps Research and Applications, TMRA 2006, Leipzig, Germany, October 11-12, 2006, ISBN: 978-3-540-71944-1

Publications as co-author

Walter, Mathias; Rattei, Thomas; Arnold, Roland; Güldener, Ulrich; Münsterkötter, Martin; Nenova, Karamfilka; Kastenmüller, Gabi; Tischler, Patrick; Wölling, Andreas; Volz, Andreas; Pongratz, Norbert; Jost, Ralf; Mewes, Hans-Werner; Frishman, Dmitrij *PEDANT covers all complete RefSeq genomes.*, Nucleic Acids Research, Database issue 2009, (in press)

Declaration / Erklärung

Ich erkläre an Eides statt, dass ich die der Fakultät für Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt der Technischen Universität München zur Promotionsprüfung vorgelegte Arbeit mit dem Titel:

Towards Effective Biomedical Knowledge Discovery through
Subject-Centric Semantic Integration of the Life-Science Information Space

im Institut für Bioinformatik und System Biologie des Helmholtz Zentrums München

unter der Anleitung und Betreuung durch

Prof. Dr. Hans-Werner Mewes

ohne sonstige Hilfe erstellt und bei der Abfassung nur die gemäß § 6 Abs. 5 angegebenen Hilfsmittel benutzt habe.

( ) Ich habe die Dissertation in dieser oder ähnlicher Form in keinem anderen Prüfungsverfahren als Prüfungsleistung vorgelegt.

( ) Die vollständige Dissertation wurde in ....
.... veröffentlicht. Die Fakultät für
.... hat der Vorveröffentlichung zugestimmt.

( ) Ich habe den angestrebten Doktorgrad noch nicht erworben und bin nicht in einem früheren Promotionsverfahren für den angestrebten Doktorgrad endgültig gescheitert.

( ) Ich habe bereits am .... bei der Fakultät für
.... der
Hochschule ....
unter Vorlage einer Dissertation mit dem Thema...
....
die Zulassung zur Promotion beantragt mit dem Ergebnis:..........................

Die Promotionsordnung der Technischen Universität München ist mir bekannt.

München, den ....

Karamfilka K. Nenova