Chapter 11
Case Study I: Ontology-Based Multi-Agent System for Human Disease Studies

11.1 Introduction

In this chapter, we shall put the Onto-Agents Methodology into practice and use it step by step to develop a Generic Human Disease Ontology (GHDO) and to design a multi-agent system that can use the designed GHDO for intelligent information retrieval.

In Section 2, we discuss the domain of human diseases, the purpose of the ontology, the community of users and agents for which this ontology is being developed, and applications based on the designed ontology. The aligning and merging of existing medical ontologies against the defined ontology structure is discussed in Section 3. Section 4 describes how to design the ontology base, while Section 5 explains the design of the ontology commitment layer. In Section 6, we discuss the evaluation of the GHDO.

In Section 7, we identify and describe different groups of agents according to their functions and responsibilities within the system. In Section 8, we describe a mechanism by which GHDO ontology is used in this system during the processes of problem solving, task and result sharing, and assembling of results. In Section 9, we focus on the structural organization of the agents within the system and define agents’ collaborations. Individual agent structure and agents’ components are described in Section 10. In Section 11, we discuss security issues within the multi-agent system. Examples of the use of the designed systems are given in Section 12.

11.2 Generalization and Conceptualization of the Medical Domain

In this section, we discuss the following points:

- Community associated with the ontology
- Purpose of the ontology
- Domain that needs to be represented by the ontology
- Application based on the designed ontology

We will also give a top-level hierarchy of the proposed Generic Human Disease Ontology (GHDO).
11.2.1 Community Associated with the Ontology

Firstly, we will discuss the community of users of the system.

The first group of users is faced with a situation of a disease and is interested, for example, in disease symptoms for the purpose of correct identification and diagnosis of this disease, and/or they may be interested in possible treatments for a particular disease. This group of users consists mainly of patients and physicians.

The second group of users is interested in disease causes, disease prevention and/or how to improve available disease treatments. The second group of users is comprised mainly of medical researchers whose goal is to make a disease situation easier for patients and physicians. Medical researchers may look for disease causes and on the basis of that, develop or discover more effective treatments.

Note that there are no strict boundaries here. A physician may also be interested in the genetic causes of a disease. She/he can screen the patient’s DNA for the purpose of detecting any (mutated) gene to confirm the presence of a disease. Also, a medical researcher may be interested in information about disease treatments, especially if she/he is working on the drug design.

Secondly, we will mention the community of agents that are committed to, and are using, the designed ontology during the process of intelligent information retrieval. We identified four different groups of agents. Interface agents assist the user in forming queries as well as returning the retrieved and assembled information to the user. Interface agents communicate a user’s request to the Manager agents. Those agents then assign specific tasks to the Information agents in order to gather the requested information in the most efficient way. The Information agents retrieve the requested information from a wide range of biomedical databases. Each Information agent may have a set of assigned databases it needs to visit in order to gather the requested information. The Information agents hand over the retrieved information to Smart agents. These agents select appropriate information, assemble it correctly, and hand it over to the other agents which will direct it back to the user as an answer to his/her query.

Due to the complexity and size of the body of knowledge represented by the human diseases ontology, we decided to partition the ontology into its subontologies, and for each of the subontologies, design agents with the four abovementioned functionalities. Such a system has a hierarchical structure and will be discussed in more detail in the second part of this chapter.

11.2.2 Purpose of the Ontology

The medical knowledge contained in various databases needs be intelligently extracted in order to improve medical research, disease control and patient care. Also, new medical information is being constantly added to the existing pool of knowledge. Researchers and medical practitioners need mechanisms to capture and diffuse domain-specific knowledge. This requires structured and standardized organization of data.
We believe that the use of ontologies would facilitate the sharing and reusing of the medical information, cooperation, querying of the information in an intelligent way, embracing all the knowledge shared by the medical community into a unifying framework and offering the possibility of one common ontology being used by various applications.

11.2.3 Ontology Domain

We need to understand the general structure and organization of the medical domain before we start building software models. The complexity of the medical domain is due to different factors:

- structure and content of information resources. The information resources are autonomous and a huge number of them contain raw and heterogeneous data. The information is distributed across various information resources and a thorough search is impossible. The medical domain is dynamic because new information is constantly being added and the number of users increases. Consequently, network traffic also increases. There is the need for an intelligent and dynamic information system to enable efficient information retrieval.
- complexity of medical research experiments. Medical research experiments are complex due to the long DNA sequence, the heterogeneity of study groups, and the multiple factors that together may cause a disease such as in the case of psychiatric illnesses.
- social issues such as security and privacy issues, social and professional acceptance, safety critical issues and legal issues need to be effectively addressed within the medical domain, especially because of the sensitive nature of patient information.

In this chapter, we focus on the medical domain regarding human diseases. We aim to embrace all the information that may be helpful in the study and control of human diseases. This may include information regarding laboratory experiments, drugs used to treat human diseases, disease symptoms, case studies etc. We believe that by combining all this information into a common framework, it becomes easier for the users to identify and/or recognize conceptual relationships in the data and identify possible links between originally specially distributed and heterogeneous information.

For example, medical research experiments may provide the results that may be useful during the drug design process. We may need to combine the information of drug treatments with disease symptoms in order to effectively examine, for example, the effect that a new drug has on disease symptoms. We believe that all subdomains (such as examining genetic causes of diseases, drug design technology, monitoring symptoms of diseases etc.) of the domain of human diseases need to be unified as they are targeting the same goal but at different knowledge levels.
11.2.4 Application Based on the Designed Ontology

We aim to design an ontology-based, multi-agent system that can make use of the Human Disease Ontology for the purpose of intelligent and dynamic information retrieval. Ontology is used by the agents at different levels to decompose the overall problem into smaller problems, to locate and retrieve the significant information, to communicate with each other, to cooperate and coordinate their actions, to analyze and manipulate the retrieved information, and to present this information to the user in a meaningful way.

11.2.5 Top-Level Hierarchy of GHDO

We identified two main user categories of the system:

- medical researchers who are mainly interested in the causes of a disease; and
- physicians and patients who are faced with the situation of a disease and are mainly interested in symptoms and treatment of a disease.

We believe that the ontological model of the Generic Human Disease Ontology (GHDO) needs to have four main branches:

1) disease types, describing different types of a disease;
2) phenotype, describing symptoms of a disease;
3) causes of a particular disease which can be environmental or genetic. Microorganisms, such as bacteria and viruses, can also cause a large number of diseases.
4) treatments, giving an overview of all treatments possible for that particular disease.

The GHDO top-level hierarchy is illustrated in Figure 11.1. The information presented in this figure indicates that a disease may have different types that may be further divided into subtypes and sub-subtypes. For each disease, there is a corresponding phenotype or observable characteristics of an ill individual, namely the symptoms of a disease. Each disease is caused by factors(s) which can be genetic (genotype), environmental, or caused by a microorganism. Information about the genetic causes of a disease can be about a mutated gene, a complex of genes or a DNA region of interest. A DNA region of interest is a region in the DNA sequence that potentially contains a gene responsible for the disease. This region needs to be further examined in order to correctly locate the mutated gene. Environmental causes of a disease can be stress, climate, drugs or family conditions. Microorganisms that may cause a disease may be viruses or bacteria. Possible treatments for a disease can be drug therapy, chemotherapy, surgery, psychotherapy or physiotherapy.

The four different branches (subontologies) of the GHDO ontology can serve as a reference point against which the concepts from the existing medical ontologies can be reorganized, aligned and merged.
11.2 Generalization and Conceptualization of the Medical Domain

The four subontologies (disease types, symptoms, causes and treatments) are different from each other in the knowledge they represent and in the way their concepts are organized. We say that these different subontologies are orthogonal to each other.

The Types subontology is more a classifying ontology and is strongly hierarchically supported. It provides information about different types and subtypes of a specific disease. For example, psychiatric diseases have different types such as depression, schizophrenia, bipolar disorder etc. Those different types can be further divided into subtypes. For example, for bipolar disorder type, we have Bipolar I and Bipolar II subtypes. This subontology is based on classification and it does not provide a user with much scientific information.

The Phenotype subontology is more descriptive than the others. It is based on observation and diagnosis of the characteristics of an ill individual. Usually, physicians are in direct contact with patients and are able to provide this information.

The Cause subontology provides a user with scientifically proven facts and is strongly based on scientific research. Most of this information results from laboratory experiments and is provided by medical researchers. For example, bacterial and viral characteristics and their role in disease development are examined in medical laboratories.

The Treatment subontology is a combination of classification and research ontology. Comparing different characteristics of different drugs targeting the same disease and working on a drug design is medical research work but, for example, all the discovered drugs can be classified according to a hierarchical structure.

All four dimensions are different from each other and each dimension is unique. But jointly they provide an overall picture and a good overview of knowledge about a human disorder.
### 11.3 Aligning and Merging Existing Medical Ontologies

Researchers in the medical ontology-design field have developed different terminologies and ontologies in many different areas of the medical domain. A large number of medical ontologies cover overlapping domains. The need for sharing and reusing this body of knowledge becomes increasingly critical. It would be of great value to construct a common terminology for all different applications within the same knowledge domain and so enable the integration and exchange of information between different applications.

This stage of ontology design focuses on the process of building a new ontology by reusing and adapting ontological terms from other ontologies. In this section, we will focus on the following points:

- identifying suitable ontologies that can be reused
- defining merge and alignment tool
- importing source ontologies
- identifying correspondences between different ontologies
- aligning and merging ontologies

#### 11.3.1 Identifying Ontologies Suitable to Be Reused

Rather than creating a new terminology, we suggest reusing existing medical ontologies. When designing GHDO, other medical ontologies, such as LinkBase (Ceusters et al. 2001), can be reused. The way that these concepts are organized within the existing medical ontologies is not suitable for use by our system. Moreover, LinkBase contains more than one million concepts and only those concepts suitable for, and needed by, our information system need to be selected. TAMBIS ontology (Stevens et al. 2002) represents general knowledge in regard to proteins and associated genes. If a gene is mutated, it may result in a non-functional protein. The presence of a non-functional protein in the human body can cause a disease. This is the reason that some biomolecular ontologies, such as TAMBIS ontology, would be suitable for that part of our ontology that relates to genetically caused diseases. UMLS (Bodenreider 2004) and SNOMED (Ceusters et al. 2004) terminologies can be used to validate the chosen concepts.

It is thus possible to use concepts from the existing ontologies but to organize these concepts according to the four ontology dimensions (disease types, symptoms, causes and treatments) mentioned in the previous section. The resulting ontology can then be used for our purpose and by our information system.

#### 11.3.2 Define Alignment and Merge Tool

Because we adopted the DOGMA principle (separation of the ontology base from the ontology commitment layer) in our ontology design, we will use the DOGMA-Modeler tool (Spyns 2005) to design a Generic Human Disease Ontology (GHDO).
It seems logical then, to use an ontology alignment and merge tool associated with DOGMAModeler to align and merge source ontologies with GHDO.

11.3.3 Import Source Ontologies

The identified ontologies suitable for reuse should be imported into the new ontology design environment. As different ontologies may be designed using different tools and may have different formats, it may be necessary to convert all these ontologies into one format in order to use them efficiently. This standardized format needs to be supported by the chosen ontology alignment and merge tool i.e. DOGMAModeler in our case.

11.3.4 Identify Correspondences between Different Ontologies

The ontology structure of the source ontologies, the manner and order in which the terms are organized within these ontologies may be similar but is almost always different from the structure of the ontology being designed. The same fact may be represented differently by two different ontologies. This is the reason that correspondences among the source ontologies need to be established. The set of overlapping concepts, concepts that are similar in meaning but have different names, and concepts that are unique to each of the sources, need to be determined.

11.3.5 Align and Merge Ontologies

By aligning the source ontologies, links will be established between the aligned ontologies. This will allow the designer to have more control over the ontology design process and to get a clear overview about the knowledge represented by different ontologies. In places where different ontologies overlap, the ontology designer is able to choose the most suitable concept definitions/relationships. In other places where there is no overlap between different ontologies, the ontology designer can either accept the suggested knowledge representation, or define its own.

After making and acting upon these choices, a single ontology will be created as a merged version of the original ontologies.

11.4 Formal Specification of Human Disease Domain Conceptualization

Formal ontology representation enables ontologies to be used by computers. In this section, we will focus on the formal specification of conceptualization while in the following section we will focus on the formal specification of ontology commitments.
In this section, the aim is to define:

- Ontology concepts
- Relationships between concepts
- Lexons
- Relationships between lexons
- Groups of related lexons

### 11.4.1 Ontology Concepts

At this stage, we need to identify and precisely define ontology concepts needed to represent the domain of human diseases. Every concept definition is dependent on other concepts and we have to rely on the commonly accepted understanding of some basic terms. The main ontology concepts are shown in Figure 11.1.

### 11.4.2 Relationships between Concepts

This step involves identifying intentional and extensional ontology relationships:

1) Intentional relationships (conceptual relationships) map every concept from the domain of human disease to the Generic Human Disease Ontology (GHDO) structure. The output of this function is a set of chosen ontology concepts that is going to be used to design GHDO.

2) The GHDO structure consists of extensional relations between the chosen domain concepts representing the domain knowledge. An overview of the extensional relations is given in Figure 11.1 and is more precisely defined through the use of lexons.

### 11.4.3 Lexons

An ontology base is (Jarrar and Meersman 2002):

- A set of context-specific binary fact types, called lexons. Notation: \(<\mu, t_1, r, cr, t_2>\). Here is \(\mu \in \Delta\) an abstract context identifier chosen from a set \(\Delta\); \(t_1, r, cr, t_2\) are term1, role, co-role and term2 respectively. The lexical terms (\(t_1, r, cr, t_2\)) are constructed from a given alphabet.

- For each \(\mu \in \Delta\) and each term \(t\) occurring in a lexon, the pair \((\mu, t)\) specifies exactly a unique concept.

- Lexons are conceptual constructs expressing a binary conceptual relationship that is agreed to hold within a given context (among all the parties involved in the ontology).

For example, in ontology designed for representing the knowledge of human diseases, we have the following lexons expressing binary relationships:
11.4 Formal Specification of Human Disease Domain Conceptualization

- **Lexon1**: <human diseases, disease, has, is of, type> of the form $<\mu, t_1, r, cr, t_2>$.

  In this expression, we have:
  
  a. context identifier $\mu$ is “human diseases”
  b. term $t_1$ is “disease”
  c. role $r$ is “has”
  d. co-role $cr$ is “is of”, and
  e. term $t_2$ is “type”

  We describe in the context of human diseases, two complementary binary relationships of the forms $<t_1, r, t_2>$ and $<t_2, cr, t_1>$: $<\text{disease, has, type}>$ and $<\text{type, is of, disease}>$ respectively. This means that in the context of human diseases, “disease has type” and “type is of disease”.

- **Lexon2**: <human diseases, disease, shows, characterizes, symptom> of the form $<\mu, t_1, r, cr, t_2>$.

  In this expression, we have:
  
  a. context identifier $\mu$ is “human diseases”
  b. term $t_1$ is “disease”
  c. role $r$ is “shows”
  d. co-role $cr$ is “characterizes”, and
  e. term $t_2$ is “symptom”

  We describe in the context of human diseases, two complementary binary relationships of the forms $<t_1, r, t_2>$ and $<t_2, cr, t_1>$: $<\text{disease, shows, symptom}>$ and $<\text{symptom, characterizes, disease}>$ respectively. This means that in the context of human diseases, “disease shows symptom” and “symptom characterizes disease”.

- **Lexon3**: <human diseases, disease, is caused by, causes, cause> of the form $<\mu, t_1, r, cr, t_2>$.

  Here, we have:
  
  a. context identifier $\mu$ is “human diseases”
  b. term $t_1$ is “disease”
  c. role $r$ is “is caused by”
  d. co-role $cr$ is “causes”, and
  e. term $t_2$ is “cause”

  We describe in the context of human diseases, two complementary binary relationships of the forms $<t_1, r, t_2>$ and $<t_2, cr, t_1>$: $<\text{disease, is caused by, cause}>$ and $<\text{cause, causes, disease}>$ respectively. This means that in the context of human diseases, “disease is caused by cause” and “cause causes disease”.

### 11.4.4 Relationships between Lexons

The binary nature of relationships serves as a conceptualization constraint and enables modelling of the most basic and atomic fact within the domain of human
disease. To allow for the introduction of higher order semantic relationships, it is possible to treat a lexon as a term in another lexon.

Consider the following example:

- **Lexon 4**: \(<\text{causes, Lexon3, is, is of, genotype}>\) of the form \(<\mu, t_1, r, cr, t_2>\).

  Here, we have:
  
  a. context identifier \(\mu\) is “causes”
  b. term \(t_1\) is “Lexon3”
  c. role \(r\) is “is”
  d. co-role \(cr\) is “is of”, and
  e. term \(t_2\) is “genotype”

We describe in the context of disease causes, two complementary binary relationships of the forms \(<t_1, r, t_2>\) and \(<t_2, cr, t_1>\): \(<\text{Lexon3, is, genotype}>\) and \(<\text{genotype, is of, Lexon3}>\) respectively. This means that in the context of causes of a disease, \(<\text{“disease is caused by cause” and this “cause is genotype”}>\) and \(<\text{“genotype is of cause” and this “cause causes disease”}>\).

Consider the following lexon:

- **Lexon 5**: \(<\text{causes, genotype, is, is a, gene mutation}>\) of the form \(<\mu, t_1, r, cr, t_2>\).

  Here, we have:
  
  a. context identifier \(\mu\) is “causes”
  b. term \(t_1\) is “genotype”
  c. role \(r\) is “is”
  d. co-role \(cr\) is “is a”, and
  e. term \(t_2\) is “gene mutation”

We describe in the context of causes of a disease, two complementary binary relationships of the forms \(<t_1, r, t_2>\) and \(<t_2, cr, t_1>\): \(<\text{genotype, is, gene mutation}>\) and \(<\text{gene mutation, is a, genotype}>\) respectively. This means that in the context of causes of a disease, “genetic (cause) is gene mutation” “gene mutation is a genetic (cause)”.

Medical researchers are searching the human DNA to find situations like this, while the physicians use this information as they can screen human DNA for the presence of this mutation and evidence of a disease.

**11.4.5 Groups of Related Lexons**

Context identifiers group related lexons in an intended conceptualization of a domain. A context can be defined as a mapping from \(\Delta\) to a collection of sources, such as a corpus of documents on the same topic. In the example of Lexon 4, we can cluster together all binary relationships expressing facts regarding disease causes and map this cluster to a corpus of documents on the topic of causes of human diseases. Note that sometimes a disease is also partly affected by the environmental conditions such as stress, climate, family conditions etc.
Lexons are always true and free of further interpretations.Specifications of improbable or impossible (contradictory) worlds are also possible, especially in the early stages of engineering an ontology, but in practice no applications can commit to them. For example, it is impossible for an application not to commit to Lexon3 and to commit to Lexon4 that is based on Lexon3.

All lexons in the ontology base are free of any specific interpretation and all rules and constraints implied on the lexons are moved to the commitment layer.

11.5 Formal Specification of Human Disease Ontology Commitments

The commitment layer is organized as a set of ontological commitments. Each commitment is a consistent set of rules, axioms in a given syntax that provide a specific interpretation to a subset of lexons in the ontology base. Within a commitment, we distinguish:

- **Intra-commitments**: rules that constrain and attribute specific interpretations to a selected subset of lexons contained within the lexon base (Jarrar et al. 2003), and
- **Inter-commitments**: a set of mappings that link elements of this subset of lexons to elements of specific applications (Deray and Verheyden 2003).

In this section, we aim to:

a. identify intra- and inter-commitments
b. formalize the ontology commitments
c. identify reusable knowledge components

11.5.1 Identify Intra- and Inter-Commitments

We will take examples from the previous section:

Lexon1: <human diseases, disease, has, is of, type>
Lexon2: <human diseases, disease, shows, characterizes, symptom>
Lexon3: <human diseases, disease, is caused by, causes, cause>

The intra-commitments are rules that constrain and attribute specific interpretations to this selected subset of lexons contained in the ontology base of the human diseases ontology. As the intra-commitments for those lexons of the ontology base, we have respectively:

- each disease may have a disease type
- each disease shows at least one symptom
- each disease is caused by at least one factor

Inter-commitments characterized on this subset of lexons are a set of mappings that link these lexons to elements of specific applications. For example, the lexons associated with these commitments can be linked to an application that retrieves information on the topic of human diseases.
11.5.2 Formalize the Ontology Commitments

Given a lexon $<\mu, t_1, r, cr, t_2>$, its commitment can take the form of two perspectives $<t_1, r, t_2>$ or $<t_2, cr, t_1>$. The first element is referred to as the theme, the second as the transition, and the third as the rheme (Jarrar and Meersman 2002).

In the examples of:

Lexon1: $<\text{human diseases}, \text{disease}, \text{has}, \text{is of}, \text{type}>$
Lexon2: $<\text{human diseases}, \text{disease}, \text{shows}, \text{characterizes}, \text{symptom}>$
Lexon3: $<\text{human diseases}, \text{disease}, \text{is caused by}, \text{causes}, \text{cause}>$

As the intra-commitments of the type $<t_1, r, t_2>$, we have respectively:

Commitment1: $<\text{each disease}, \text{has}, \text{zero-or-more type}>$
Commitment2: $<\text{each disease}, \text{shows}, \text{at least one symptom}>$
Commitment3: $<\text{each disease}, \text{is caused by}, \text{at least one cause}>$. 

Note that a commitment of the form $<t_1, r, t_2>$ or $<t_2, cr, t_1>$ operating on a lexon of the form $<\mu, t_1, r, cr, t_2>$ from the ontology base does not mean that $t_1, t_2, r$ and $cr$ are the same in the lexon and in the commitment. For example, Commitment1 $<\text{each disease}, \text{has}, \text{zero-or-more type}>$ operates on the Lexon1 $<\text{human diseases}, \text{disease}, \text{has}, \text{is of}, \text{type}>$. For the commitment layer, $t_1$ is “each disease” and $t_2$ is “zero-or more type”. For the ontology base, $t_1$ is “disease” and $t_2$ is “type”.

11.5.3 Identify Reusable Knowledge Components

The ontological commitments may be seen as a set of reusable knowledge components. In practice, similar applications reuse or inherit commitments from each other. New applications can commit to, and use, the existing ontology. This is a very useful feature that can be effectively used within multi-agent systems. Cooperatively working agents differ from each other, but require minor differences in their ontology as they are all working towards the same goal. We can use the same ontology base combined with different commitment layers to design slightly different ontologies for different agents.

Assume that Ag1 is an Interface agent and Ag2 is an expert on general knowledge regarding human diseases. The commitments of these two different agents regarding the same lexon $<\text{human diseases}, \text{disease}, \text{has}, \text{is of}, \text{cause}>$ will be different as these agents have different responsibilities within the system. For Interface agents, it is not necessary for each user of the system to require information regarding diseases causes. For example, a user may be interested only in treatments for a disease. In this case, commitment of the form $<t_1, r, t_2>$ for agent Ag1 that operates on the lexon $<\text{human diseases}, \text{disease}, \text{is caused by}, \text{causes}, \text{cause} >$ is $<\text{each disease}, \text{has}, \text{zero-or-more cause}>$. On the other hand, commitment of Ag2 that operates on the same lexon is $<\text{each disease}, \text{is caused by}, \text{at least one cause} >$. Ag2 is expert on general knowledge of human diseases and it needs to know that a disease has at least one cause.
11.6 Human Disease Ontology Evaluation

The constructability of GHDO can be evaluated mathematically using Set Theory (Wouters et al. 2004), and practically through design of a prototype.

11.7 Classification of Agents According to their Responsibilities within the Human Disease Information Retrieval System

The outcomes of this step are analogous to the examples shown in Section 8.3.

11.7.1 Establish Intuitive Flow of Problem Solving, Task and Result Sharing

Intuitive flow of problem solving, task and result sharing can be accomplished in the following sequence of actions:

query formulation → task sharing → information retrieval → result sharing → result analysis → result assembly and result presentation

11.7.2 Identify Corresponding Agent Functions

We need agents to help query formulation, to decompose the overall problem and assign task to various agents that need to retrieve the needed information. We also need agents to analyze the retrieved information and assemble the selected information to be presented to the user.

11.7.3 Identify Corresponding Agent Types

We will use a Sequence Diagram where Composite Classes have more than one port and represent different roles of the same agent (Wongthongtham et al. 2008). This will enable us to model agents of GHDO-based multi-agent system which play more than one role concurrently. For example, Smart agent plays 2 different roles: collection of information from various Information agents and sending SHDO to the Interface agent.

In the examples shown in Figure 11.2, a number of agents play multiple roles which is represented by multiple ports. Information agent plays four roles; it retrieves information about disease types, symptoms, causes and treatments. Smart agent plays two roles; it collects the retrieved information and sends the SHDO to the Interface agent. Depending on which role the agent is acting in when it sends/receives messages, the sequence diagram shows arrows to/from a particular lifeline for the agent.
We illustrate a system that has four agents. The first agent is called Interface agent. The user initially specifies the disease that they want information about by sending a ‘Specify Disease’ message to the Interface agent. The agent confirms that this message has been received. The next step is that the user specifies their specific interest in the disease. That is, do they want information about disease types, causes, symptoms or treatments? Again, the Interface agent confirms that this message has been received. Once the Interface agent builds a SHDO template (called a Specific Human Disease Ontology Template) using Generic Human Disease Ontology, there is a structure established which can later be filled out with data. For example, the user may specify that they want information about Diabetes, and only on causes and symptoms, not treatments.

The SHDO template is immediately sent to the Manager agent, whose responsibility is to direct requests to Information agents who actually retrieve the data required to fill in the template. Thus, the Manager agent needs to have the empty SHDO template, in order to know which Information agents to activate. The SHDO template is also sent to the Smart agent. The responsibility of the Smart agent is to collect (from Information agents), analyze, select relevant information and fill out the SHDO template with data returned.
Once the SHDO template has been sent by the Interface agent, this agent makes a request to the Manager agent for Disease Data. The Manager agent then successively sends messages to an Information agent about Disease Type, Symptoms, Causes and Treatments. Each different type of information will require that the Information agent play a different role in order to gather it. Hence the messages go to one of four distinct ports, each of which models a distinct role for the Information agent.

Each time the Information agent gathers information it sends it to the Smart agent, who is responsible for collecting the information in order to fill out the SHDO template. Once the Smart agent has all the SHDO data collected and the template has been filled out it plays a different role to send the information back to the Interface agent. The Interface agent then returns the data to the user.

11.8 Identify the Need for Human Disease Ontology to Support Agents’ Intelligence

11.8.1 Problem Decomposition and Task Assignments

The overall problem to be solved is constructed as a Specific Human Disease Ontology (SHDO) template by Interface agents. Retrieving and adding of relevant information to this SHDO template results in Specific Human Disease Ontology (SHDO). This is shown in Figure 11.3.

The SHDO template is decomposed into smaller subproblems by Manager agents. This kind of decomposition is hierarchical and the subproblems are further decomposed into smaller sub-subproblems, and so on. The SHDO template is first decomposed into its four subontologies (disease types, symptoms, causes and treatments). These subontologies are further decomposed into smaller sub-subontologies. The goal of this problem decomposition is to reach a stage where subproblems can be solved by individual Information agents.

A task assigned to an individual Information agent can be composed of more atomic actions. Information agents continue problem decomposition until the subproblems represent atomic actions that cannot be decomposed any further. The different levels of decomposition will often represent different levels of problem abstraction. For example, ‘genetic cause of a disease’ is on a higher abstraction level than ‘gene’.

Agents must have appropriate expertise to decompose the overall problem and assign corresponding tasks to the agents of the system. They must have knowledge of the task structure and must know how the task is put together. For example, the Manager agent needs to know which Information agents are suitable for performing a specific task so that the Manager agent can assign this task to appropriate Information agents when the need arises. Also, the Information agent needs to know how and where to perform atomic actions of the overall task that was assigned to it. This is the reason why the ontology can be used to represent domain knowledge as well as the task structure.
As described above, a problem is decomposed into smaller subproblems by the Manager agent. Because the Manager agent knows exactly which Information agent is appropriate for the execution of a particular task, tasks are allocated to different Information agents. In Figure 11.4, we show an example where four different types of Information agents exist. Each agent is responsible for retrieving information on a specific topic, namely, on disease types, symptoms, causes or treatments. The directed contract is then established between Manager and Information agents. One Information agent is also aware of tasks assigned to other Information agents. Because different databases are assigned to different agents, it
is possible that one Information agent finds information significant to other Information agents. This is a situation where sharing of the information between different Information agents becomes important.

Generally, messages within a multi-agent system can be implemented as request and as information messages. In the case of simple requests for information, the construction of an SHDO template is not needed. The request message can then be used to encode a straightforward request for information. The information message can be used as a response to a request message. Some examples of the messaging within a multi-agent system include situations when a user has a simple request that needs activation of only one Information agent or situations between different Information agents when one Information agent finds information that is significant for another Information agent.

### 11.8.2 Information Retrieval

In this section, we will discuss two related features: atomic problems solutions and result sharing.

In the stage of atomic problems solution, subproblems identified during the problem decomposition phase are individually solved by Information agents. Usually, a task assigned to individual Information agents addresses a specific problem and is composed of more atomic actions. From the example given in Figure 11.4, one Information agent has the task of retrieving information about the effect of drugs on the onset of a specific disease. One aspect of this problem can be related to the medicines used to treat some other disease while the other aspect can be related to substance abuse. Each of these subproblems can be classified further into, for example, natural or synthetic drugs. The Information agents will perform the atomic actions and migrate from one database to another in order to accomplish their overall task, e.g., to retrieve information on the negative association between drugs and a specific disease.

Because different databases are assigned to different agents, sharing of the information between different Information agents within the system can be very useful. Agents share information relevant to their subproblems. The cooperative exchange of information covering different areas of the originally defined SHDO template enables the solution to be developed progressively. Solutions to small problems are gradually refined into larger, more abstract solutions resulting in the final result which is presented to the user.

### 11.8.3 Agent Communication

The agents are able to interact, freely share, and combine their results in the most accurate and efficient way only when communicating by use of a common language. All agents within this multi-agent system need to agree and commit to this common ontology language.

The ontology of an individual agent has two main components. One component is internal and more stable. This ontology component includes domain, procedural, task, cooperative, environment knowledge and similar. We note this component as
Conversely, the other component is dependent on the user’s query and is easily changed. This is the $\mathcal{AO}_e$ component. The agent ontology is then written as $\mathcal{AO}$, where

$$\mathcal{AO} = \mathcal{AO}_i \cup \mathcal{AO}_e.$$  

The difference between the ontologies of different types of agents (such as Interface, Manager, Information, Smart agents) is larger than the difference between individual ontologies of the different agents of the same type.

The component $\mathcal{AO}_i$ is different for different types of agents since they have different tasks assigned to them. The same type of agents may also have minor differences in the $\mathcal{AO}_i$ component since different information resources are assigned to different Information agents.

We need to mention that the difference in $\mathcal{AO}$ for the same task of the same agent is possible. This is the result of the dynamic multi-agent system where it is possible for the same action performed twice in apparently identical circumstances to have different effects.

Ontologies that represent medical knowledge as well as users’ queries about this knowledge can be derived from a single generic ontology (Generic Human Disease Ontology, GHDO). We defined Template Constructor Function $\tau$ to map:

- generic ontology base relations $\mathcal{R}$ between terms $t_1$ and $t_2$ to true or false values, and
- generic ontology commitments $\mathcal{C}$ to true or false values.

We say that the specific ontology templates Specific Human Disease Ontology (SHDO) templates) are formed from the Generic Human Disease Ontology (GHDO) by Template Constructor Function $\tau$,

$$\tau : [((\mathcal{R} \rightarrow \{true, false\}) \land (\mathcal{C} \rightarrow \{true, false\})].$$

We represent this as $<\text{GHDO}, \tau> \rightarrow \text{SHDO template}$.

The Interface agent gives the resulting SHDO template over to the Manager agent. The Manager agent assigns different tasks to different Information agents. This would mean that the SHDO template needs to be partitioned into smaller tasks, $\mathcal{AO}_e$. $\mathcal{AO}_e$ represents the task assigned to an individual Information agent. $\mathcal{AO}_e$ is constructed from the SHDO template using the Subtemplate Constructor Function $\tau_a$ which maps relationships and commitments of the SHDO template to true and false values,

$$\tau_a : [((\mathcal{R}_{\text{SHDO template}} \rightarrow \{true, false\}) \land (\mathcal{C}_{\text{SHDO template}} \rightarrow \{true, false\})].$$

If there are $n$ Information agents, then $n$ of such $\mathcal{AO}_e$ are formed so that ideally the whole information within the SHDO template is covered, or

$$\mathcal{AO}_{e1} \cup \mathcal{AO}_{e2} \cup \ldots \mathcal{AO}_{en-1} \cup \mathcal{AO}_{en} = \text{SHDO template}.$$  

All the resulting $\mathcal{AO}_e$ are subsets of SHDO template.

We represent this as $<\text{SHDO template}, \tau_1, \tau_2\ldots \tau_{n-1}, \tau_n>$, where $\tau_1, \tau_2\ldots \tau_{n-1}$, and $\tau_n$ are $n$ different formation functions for agents $Ag_1, Ag_2\ldots Ag_{n-1}, Ag_n$ respectively, or $<\text{SHDO template}, \tau_1, \tau_2\ldots \tau_n> \rightarrow \mathcal{AO}_{e1}, \mathcal{AO}_{e2}\ldots \mathcal{AO}_{en}$.
Now that their task has been specified and formatted as $A\Omega e$, the Information agents are able to search for and retrieve the requested information. The information retrieved by Information agents is given to the Smart agent. This agent analyses this information and selects relevant information to be added to the SHDO template. Addition of the selected information to the SHDO template results in an SHDO which is then presented back to the user.

There are three different parties involved in communication within this information system: user, agent and environment. The Interface agents communicate to the user during the process of query initiation and also during the presentation of results. The Information agents communicate with the environment and must move between different environments in order to access data. Communication between agents, such as communication between Smart and Information agents, allows agents to share information and coordinate activities, thereby enabling them to perform collaborative work.

### 11.8.4 Information Analysis and Manipulation

In this phase, the Smart agent analyses information provided by different Information agents. In the example from Figure 11.4, information regarding “DNA region of interest” is coming from three different Information agents. This information may be different or the same. If different information covers the same topic, information with the highest value needs to be selected by the Smart agent and incorporated into the SHDO template. In our case, “DNA region of interest” contains information about regions of human DNA which may contain a gene responsible for the onset of a particular disease if mutation (abnormal change of gene structure) of this gene occurs.

| Table 11.1 Information retrieved by different Information agents regarding DNA region of interest |
|-----------------------------------------------------------|
| Agent1 | 2, p13-19 | 10, q21-24 | 17, q11-14 | 17, q11-12 | X, q24-27 |
| Agent2 | 10, q21-26 | 10, q21-25 | 12, q23-24 | 17, q11-13 | X, q24-25 |
| Agent3 | 2, p13-17 | 2, p13-16 | 12, q23-24 | 12, q23-26 | 17, q11-13 |

The Information agents may provide, for example, the following information for the case of manic-depression (Craddock and Jones 2001; Liu et al. 2003). A part of this information is presented in Table 11.1. The numbers represent chromosomes in human DNA that may contain the gene of interest (2, 10, 12, 17 and X chromosome) followed by the precise region of this chromosome where this gene is positioned (p13-16, q21-24, q23-24, q11-12, q24-25 etc.).

The Smart agent compares this information on two levels:

1) On the first level, the Smart agent clusters the information according to the chromosome. For example from Table 11.1, we would have 5 clusters for chromosomes 2, 10, 12, 17 and X.
2) On the second level, the Smart agent compares information regarding chromosome regions for each of the chromosomes. In Table 11.1, for chromosome 17 we have regions: q11-13 (information provided by Agent 2) and q11-12 and q11-14 (information provided by Agent 1). In this context, a smaller DNA region of chromosome means being closer to the gene of interest. For this reason, smaller regions of chromosomes are selected by the Smart agent to be incorporated into the SHDO template. In the example of chromosome 17, region q11-12 would be selected.

The selected information is assembled into the SHDO template resulting in SHDO. In the example from Table 11.1, the following information would be selected and incorporated into the SHDO template: chromosome 2, region p13-16; chromosome 10, region q21-24; chromosome 12, region q23-24; chromosome 17, region q11-12 and chromosome X, region q24-25.

### 11.8.5 Meaningful Information Presentation

Solutions to atomic problems are integrated into the overall solution by Smart agents. As in problem decomposition, this stage is hierarchical with partial solutions assembled at different levels of abstraction.

The use of ontology for meaningful representation of a knowledge domain is equally important in this stage. Information retrieved by Information agents is compared, analyzed and assembled according to the SHDO template that was constructed at the beginning by the Interface agent. This step results in SHDO which is presented to the user as the answer to his/her query.

### 11.9 Define Agent’s Collaboration within the Intelligent Human Disease Information Retrieval System

#### 11.9.1 Establish Efficient Organization of Agents

We need to focus on responsibilities and functions of different agents, and on the nature of communications within a multi-agent system in order to establish the most efficient way of organizing agents within a multi-agent system.

As GHDO has a hierarchical structure, we believe that a multi-agent system based on this ontology functions best when forming itself a hierarchy of agents corresponding to the four different ontology branches. We propose a GHDO-based Holonic Multi-agent Structure (GHMS) (Hadzic et al. 2006) as a nested hierarchy of four holarchies in which each of the four GHDO dimensions (Disease types, Symptoms, Causes and Treatments) is associated with one holarchy (see Figure 11.5). Highest in the agent hierarchy is the Disease Mediator Agent. For each of the four holarchies, we have corresponding Mediator, Specialist and Representative Agents. The information is interpreted and analyzed at the higher levels of the hierarchy while collection of the data happens at the lower level holarchy.
11.9.2 Establish Correspondence between Agent Types and Their Organization

In this section, we will discuss different agent types within the GHDO-based Holonic Multi-agent Structure (GHMS).

Disease Mediator Agent (DMA). The GHMS has Disease Mediator Agent as its main entry point.

On the basis of the SHDO template provided by the Interface agent, DMA decides which of the four holarchies needs to be engaged in order to generate the SHDO. For example, sometimes a user may be interested only in the causes of a disease so that there is no need to engage the Disease types, Symptoms or Treatments holarchy.

DMA has the function of a Manager agent. As there are also other agents from lower levels of the holonic multi-agent system that have the function of the Manager agents, we say that the DMA corresponds to the first level Manager agent.

Specialist Agents (SAs). Holarchy inner nodes represent Specialist Agents (SAs). We differentiate Disease types, Symptoms, Causes and Treatments Specialist Agents (D-SA, S-SA, C-SA and T-SA).

They represent decision makers and are specialists in a specific topic of the corresponding subontology. For example, one C-SA may be specialized in the genetic causes of a disease while another C-SA may be a specialist in the environmental causes of a disease.
SAs assign different tasks to different RAs. For example, C-SA that is specialized in the genetic causes of a disease activates one RA to look for “DNA regions of interest” and another RA to look for a specific “gene” that, when mutated, causes the disease in question. SAs correspond to the third level Manager agents.

After subordinate agents (RAs) have returned their data, SAs interpret, compare, and evaluate these data. In this way, all the delivered data are properly ranked and prepared to be assembled into the SHDO template. SAs also correspond to the third level Smart agents. Note that DMA and MAs correspond to the first level Smart agent and the second level Smart agents respectively.

Not only do they define a proper ranking among all the delivered data, but also an important function of SAs is to interpret the incoming data and conclude whether there is sufficient information retrieved in regard to the SHDO template. If not, the SA has to decide - on the basis of the delivered information - whether it makes sense to consult other RAs.

Representative Agents (RAs). The leaves are so-called Representative Agents (RAs). We differentiate Disease types, Symptoms, Causes and Treatments Diseases Representative Agents (D-RA, S-RA, C-RA and T-RA).

Each RA is an expert on the lowest level concept within the ontology. Note that RAs differ from SAs in that they need to recognize the significant information inside the appropriate database and retrieve that information. RAs correspond to the Information agents. The retrieved information is then passed over to the SA which will do the analysis, comparison and assembly of the retrieved information which is then passed over to the respective mediator agents.

For example, article_1 claims that a gene located somewhere on chromosome 6 is responsible for a disease in question, while article_2 gives more precise information regarding the gene of interest such as location 6p11-p17. C-RA retrieves both articles while C-SA gives to the C-MA only information from article_2. C-MA will do the matching and assign the value ‘6p11-p17’ to the concept “DNA region of interest” of the SHDO template, telling the user that the DNA sequence positioned on chromosome 6 in the region between p11 and p17 potentially contains a gene which may be causing the specific disease. In this way, selection of information is performed and only significant information is presented to the user. This is especially important when a great deal of information regarding a specific topic is available.

11.9.3 Query Processing and Information Integration within GHMS

A user’s query in the form of an SHDO template is firstly constructed by the Interface agent. This query is sent to the Disease Mediator Agent (DMA).

DMAs partition the SHDO template into four subontologies templates: \( \mathcal{A}Oe \) (disease types), \( \mathcal{A}Oe \) (symptoms), \( \mathcal{A}Oe \) (causes) and \( \mathcal{A}Oe \) (treatments). The resulting subtemplates of the form \( \mathcal{A}Oe \) (subontology) are sent to corresponding Mediator Agents (MAs).
Define Agent's Collaboration

MAs also receive information regarding other subontologies templates, but it needs to process only its own. For example, Causes-Mediator Agent (C-MA) receives all four subontology templates but it further processes only the $\mathcal{A}O_e$ (causes) template. The $\mathcal{A}O_e$ (subontology) is further partitioned by MAs into sub-subontologies and parts of them. For example, $\mathcal{A}O_e$ (drug therapy), $\mathcal{A}O_e$ (surgery), $\mathcal{A}O_e$ (chemotherapy), $\mathcal{A}O_e$ (physiotherapy) and $\mathcal{A}O_e$ (psychotherapy) in the case of Treatments-Mediator Agent (T-MA), and $\mathcal{A}O_e$ (genetic) and $\mathcal{A}O_e$ (environmental) in the case of Causes-Mediator Agent (C-MA). The resulting sub-subtemplates are in the form of $\mathcal{A}O_e$ (sub-subontology).

The process is shown in Figure 11.6. According to the sub-subtemplates, tasks are assigned to Specialist Agents and/or Representative Agents. SA and RA also received information regarding other parts of the SHDO template and are aware of tasks assigned to other agents.

During the information assembly process, the DMA on the higher level and MAs on the lower level of the hierarchy function as Smart agent. We mentioned that they correspond to the first level Smart agent and the second level Smart agent respectively.

MAs combine information coming from RAs and SAs in the form of $\mathcal{A}O_e$ (sub-subontology), and present it to DMA as a single unit in the form of $\mathcal{A}O_e$ (subontology). Four different $\mathcal{A}O_e$ of the four different subontologies are combined by DMA: $\mathcal{A}O_e$ (disease types) $\cup$ $\mathcal{A}O_e$ (symptoms) $\cup$ $\mathcal{A}O_e$ (causes) $\cup$ $\mathcal{A}O_e$ (treatments) = $\mathcal{S}O$. This step results in Specific Human Disease Ontology (SHDO) that after final information reorganization is presented to the user.
11.10 Construction of Individual Agents of the Human Disease Information Retrieval System

11.10.1 Identify Required Agents’ Components

An agent interacts with users of the system via the agent interface and constructs the SHDO template (from GHDO) according to the user’s requests. As all agents of the system interact and communicate with each other, they all have an agent interface.

Procedural knowledge of an agent is also constructed as ontology. It contains information regarding problem solving and the goal prioritization method. For example, the SA needs to know how to compare all information delivered by RAs and select significant information that needs to be incorporated into the SHDO template. In the example from Table 11.1, SA needed to choose the smallest DNA regions to be passed over to the MA and further incorporated into the SHDO template.

Communication component processes messages. Messages are written in ontology specification language. Agents need to speak the same language as they are exchanging information and cooperatively working towards the same goal. All agents within the system agree and commit to this common language.

Cooperative knowledge component, also constructed as ontology, enables agents to cooperate with each other. An agent needs to know how to negotiate with other agents, coordinate its own actions with the actions of other agents, and make various decisions. For example, an agent needs to decide for itself whether it is appropriate, in a specific situation, to contact other agents. These decisions are made on the basis of information regarding the organization of agents within the system and on the basis of information regarding the functions of these agents within the system. Agents that need to be contacted can be from the same, higher or lower levels within the holarchy. For instance, the Mediator Agent (MA) needs to know how and when to contact a specific Specialist Agent (SA), and whether it is appropriate to contact it in regard to the overall task that needs to be accomplished.

Task knowledge of the agents, written as ontology, contains knowledge regarding tasks assigned to that particular agent. This knowledge enables an agent to decompose its task into smaller tasks when needed and perform atomic actions. Note that task knowledge regarding tasks assigned to other agents of the system is in cooperative knowledge module. The activation of the cooperative knowledge and task knowledge module is connected as all agents within the system are cooperatively working towards the same goal. For example, Disease Mediator Agent (DMA) needs to decompose overall task (SHDO template) into four parts and assign those subtasks to the corresponding MAs. Through the decomposition process, part of this task is assigned to each agent. An agent needs to be aware of tasks assigned to other agents and to contact them if needed.

An agent also relies on past experiences. These files are stored in the history files component that is also written as ontology. Files are stored according to
The reason for this is that one SHDO requested by one user may contain information partly covering four different GHDO ‘dimensions’. Another SHDO queried for the same disease by another user, may complement this information by covering other ‘dimensions’ or extending existing ‘dimensions’. Covering as much as possible of the information requested by different users, the latest version of the SHDO regarding a particular disease is saved in the history files module. Note that in most cases, the SHDO last requested covers only a part of the SHDO from the history files. This is illustrated in Figure 11.7. If a difference is found between the SHDO last requested and SHDO from history files, the new SHDO should be checked for consistency. If the difference is consistent, the new SHDO should be used to update the corresponding part of SHDO file. In Figure 11.7, ‘f’ needed to be replaced by ‘l’. This latest version of the SHDO is stored in the history file module and used next time for matching. We call this kind of SHDO update an ‘update own files’ update. We also have an ‘update suggestion’ update. This is specifically required in the case where the environment of agents is dynamic and changes within this environment occur constantly. Here, an agent can be designed to detect these kinds of updates and suggests an update of its environment knowledge module to system designers.

Domain knowledge of an agent, written as ontology, contains general information regarding human diseases. In our model, this is Generic Human Disease Ontology (GHDO). A user specifies which part of this knowledge is of interest to him/her. The Interface agents then construct the SHDO template from GHDO on the basis of user’s query.

Environment knowledge is also constructed as ontology. It contains information regarding information content and structure within each of the different databases assigned to different Information agents. Most of the databases within the biomedical community are autonomous, heterogeneous and different from
each other. It would be very beneficial if information content of these information resources would be described using ontology (annotation of information resources). This would enable agents to ‘understand’ the content of these information resources and retrieve relevant information. Some of the information resources have already begun annotation. This is important to the biomedical community as it enables other services to efficiently access and share information contained within these information resources.

We can model the goal-driven aspect of the agent by a Composite Structure Diagram with Parts, and Ports. Each part represents a distinct area of processing within the agent. Each port represents a different role played by the agent (Wongthongtham et al. 2008).

The <<Agent>> stereotype based on the Composite Structure Diagram can be used to model the Smart agent. The <<Agent>> stereotype must have a name, at least a Controller part which controls the efforts of the Agent to achieve a goal, and at least one port, which relates to its playing a role.

**Fig. 11.8** Composite Structure Diagram representing Goal-driven characteristic of the Smart agent

We use a Composite Structure Diagram to represent the goal-driven nature of an agent. In the case of the Smart agent shown in Figure 11.8, we have two ports which correspond to two different roles of this agent, and four parts which show distinct areas of information processing within the agent. Note that the same two ports (Collection and Send) that were present in the sequence diagram (see Figure 11.2) are also present here. Each of the ports is a construct which enables the agent to interact with other agents, namely Information agent and Interface agent. Next to the Controller object, the Smart agent also has Analyze, Select and Assembly objects. The Smart agent collect the retrieved information through the Collection port. This information is analyzed, the relevant information is selected and assembled into the SHDO template resulting in the SHDO. The SHDO is sent to the Interface agent via Send port.
11.10.2 Construct Various Agents

In Chapter 8, we discussed that a variety of agents within a multi-agent system can be achieved in three different ways: (1) different components that are used to construct different agents can be the same, but the content of the components may be different for different agents; (2) content of components used to construct different agents may be the same, but different agents are constructed through a different combination of used components; and (3) different agents differ in the combination of components used to construct them and in the content of these components.

In order to design different agents of a GHDO-based multi-agent system, we will adopt the third principle.

11.11 Protect the Human Disease Information Retrieval System by Implementing Security Requirements

11.11.1 Identify Security Requirements

We need to consider the environment in which the GHMS will be situated in order to identify security requirements. Various agents located over different biomedical databases search for specific information when requested by a user. Their environment is the database environment of the biomedical community. The characteristics of this environment are that it is inaccessible, non-deterministic, dynamic and continuous (Wooldridge 2002).

In such an environment, all the security properties of authentication, availability, confidentiality, non repudiation and integrity (Mouratidis et al. 2003) as well as compliance, service and dedication should be taken into consideration.

11.11.2 Implement the Security Requirements

Because the security of a multi-agent system is closely linked to functions that agents have within the system, we will discuss system security according to the functions of different agents. We discuss Interface, Manager, Information and Smart agents in the singular to represent all agents having the same function.

Authentication: Proving the identity of an agent

Each agent needs to prove its identity to another agent when communicating and exchanging information. The whole system operates as a sequence of different actions of different agents: Interface agent (Ag1) \(\rightarrow\) Manager agents (Ag2) \(\rightarrow\) Information agents (Ag3) \(\rightarrow\) Smart agents (Ag4) \(\rightarrow\) Interface agent (Ag1). The Interface and Information agents operate inside as well as outside the system and are more critical than other agents with respect to security. The Manager agent and Smart agent operate only with the agents of the multi-agent system.

The “communication” module of the agents processes incoming identification messages. On the basis of validity of identification, the agent will either continue performing its action or its action will be cancelled.
The “procedural knowledge” component of an agent contains information regarding the problem solving and goal prioritization method. This component takes two inputs: the task that needs to be performed and information regarding the security of performing this task. On the basis of these two inputs, the agent makes the decision whether to proceed further or to cancel its action (see Figure 8.7).

This decision is communicated to security agents. These agents are specialized in the security of the system. Security agents validate the decision made by the individual agent with regard to continuing with or cancelling its action.

Two original inputs, the final decision and the consequence of this decision are stored in the agent’s “history files” for its own reference. This becomes the third input on the basis of which the decision will be made next time this agent is found in a similar situation.

If the decision made by an agent had negative consequences, results are sent further to the “update suggestion” component. Changes in values given to the security of performing the action will be suggested.

Availability: Guaranteeing the accessibility and usability of information and resources to authorized agents

Only Information agents of the multi-agent system have access to information resources. These agents need to prove their identity to external agents of the information resources. After their identification validation, they are able to access all, or part of, an information resource.

Confidentiality: Information is accessible only to authorized agents and inaccessible to others

Only those agents that can prove their authority over information have access to information. This is true for inside as well as for outside the system. Information agents need to prove their identity to external agents of the information resources. Also, the Smart agent needs to prove its identity to the Information agents in order to be able to receive information from Information agents. The Interface agent needs to prove its identity to the Smart agent in order to be able to receive the SHDO after data selection and assembly.

Non-repudiation: Confirming the involvement of an agent in certain communication

Not only do Smart agents need to show their identity to Information agents, but Information agents also need to prove to Smart agents that the information they provide is from the correct source. A similar process applies to other agents of the system.

The Smart agent needs to provide evidence to the Interface agent that all the information contained within the SHDO is trustworthy and is provided by Information agents.

Integrity: Assuring that the information remains unmodified from source entity to destination entity

The query formed by the Interface agent may not be modified by the Manager agent and subqueries assigned by the Manager agent to Information agents may not be modified by Information agents.
The integrity property needs to be guaranteed to Information agents by agents of information resources. Retrieved information should not lose its meaning once found outside the information resource.

The Interface agent presents the assembled SHDO to the user and may not modify or change the meaning of the information originally assembled by Smart agents.

Compliance: acting in accordance with the given set of regulations and standards.

Manager, Information and Smart agents must be given clear instructions on how they should perform and carry out their tasks within the system. These agents must comply with the given set of laws, regulations and standards. Following instructions set up by their designer(s) will enable the agents to be successful in performing their actions for their own benefit and for the benefit of the whole multi-agent system. For example, Information agents must make information available to authorized Smart agents when requested.

Service: agents need to serve one another for mutually beneficial purposes.

Manager, Information and Smart agents need to serve one another. For example, if some Information agents are under malicious attack, other Information, Manager, and Smart agents need to make their best efforts to protect the attacked Information agents and the whole multi-agent system.

Dedication: complete commitment of the agents to the multi-agent system goal and purpose.

Manager, Information and Smart agents are working on different aspects of the overall goal; in this case, efficient retrieval of the information about human disease. All agents must commit to this goal and purpose, and function in unity for the maximum success.

11.12 Examples of Use of the Intelligent Human Disease Information Retrieval System

The importance of ontology and agent-based technologies has been recognized within the biomedical community (Marchetti and Lanzola 2001). The Onto-Agent methodology can be used to design an ontological model as well as a multi-agent system (based on the designed ontological model) for the purpose of intelligent information retrieval of knowledge regarding human diseases.

The system can be used to support scientists in gathering information on highly specific research topics and to allow users on a world-wide basis to intelligently access new scientific information much more quickly. The knowledge domain can be easier understood and new knowledge discovered. Shared knowledge improves research efficiency and effectiveness. By being aware of research done by other teams, scientists will be able to progress much faster, coming closer to answers for their common problems. In particular, being able to have shared understanding of concepts facilitates information and knowledge exchange. Having an overview of available information, it is much easier to identify research areas where more information and more examination are needed.
Some advantages of such system are that it:

- creates a unifying framework for human disease knowledge which is of great importance for medical researchers, physicians and patients. Embracing all information about human diseases makes it possible to fully understand the knowledge domain.
- enables dynamic knowledge discovery, especially in cases where there is uncertainty about the type of disease or it is not easily diagnosable. This system may also help in early identification of new diseases such as SARS.
- improves research efficiency and effectiveness, as it helps to avoid unnecessary redundancy in performing the same experiments, such as examination of the same region of a DNA sequence and/or determination of part of DNA sequence that needs to be further examined in order to find a gene mutation causing a disease.
- constructs data patterns which combine, for instance, different genetic and environmental causes and different disease types. This helps to sort out the exact combinations of genetic and environmental factors involved, as well as their individual influences on a specific complex disease type. Once factor(s) responsible for a human disease are known, it would be much easier for doctors to diagnose, treat and possibly prevent that disease.

We consider below a number of possible scenarios from the biomedical area in which such a system would be helpful. Researchers are constantly searching for, and adding more information to, the existing pool of knowledge. Physicians are directly in contact with patients and are using all available information to help and treat their patients. Especially when a new disease starts spreading to epidemic proportions, researchers and physicians are strongly connected because they are working towards the same goal, but on different knowledge levels.

### 11.12.1 Example 1: Help Physician to Identify Disease

If a physician queries a system, she/he will be mainly interested in symptoms and possible treatments of a particular disease.

There are some exceptions to this rule such as in the case of a new disease being encountered by a physician. Namely, a physician may have a patient showing some symptoms of a disease but he may not be able to identify the disease. By entering symptoms of a disease into the system, a doctor may be able to retrieve some information regarding that disease.

It is also possible that two different diseases are manifesting the same or similar symptoms so that different outputs are possible. This is shown in Figure 11.9. In such cases, it may be useful to look for some significance in the causes of the disease as we explain in the sequel.

- **Use case 1a: disease causes are not known**

On the basis of key symptoms, the doctor will choose one (set of) disease(s). This disease becomes the doctor's working hypothesis, her/his most likely choice. The doctor then starts to gather evidence in support of the working hypothesis, always keeping in mind the set of alternative hypotheses.
Such a process relies on all kinds of information such as information that is gained by interrogating the patient or by conducting necessary (physical or instrument- or implement-based) examinations and tests.

It will be assumed that all this data and information will be stored in medical records for patients and that all necessary/available medical information about a patient is kept in exactly one comprehensive ontology-based computer-readable patient record. This enables the patient record to be further processed by the system.

- Use case 1b: disease cause is known, e.g. a gene mutation.

From Figure 11.9 it follows that in case of disease X, gene X is mutated and this causes disease X. And disease Y is caused by the mutation of gene Y. Even though the two different diseases are caused by the mutation of a different gene, they can still show similarities in their symptoms. The physician can screen the patient’s DNA to check whether gene X or gene Y is mutated. If mutation is found in gene X, the patient has disease X; and if gene Y is mutated, the patient suffers from disease Y.

![Diagram](image-url)  
Fig. 11.9 Two different diseases caused by mutations of different genes and treated by different methods showing same symptoms
Only when the patient has been correctly diagnosed, can the physician consider possible treatments. In Figure 11.9, we see that disease X is treated by drug X. Disease Y can be simply treated by physiotherapy. Therefore, our information system also reduces risks of misdiagnosis.

### 11.12.2 Example 2: Support Physician to Choose Disease Treatments

It is common to have more than one treatment possible for a particular disease. A medical professional might consult the information system in order to do a one-component search (treatments).

Figure 11.10 provides an example where three different drugs are being used to target the same disease. Different drugs have different active ingredients and show differences in their characteristics. Some drugs are herbal; others are synthetic. Usually, herbal drugs are not as effective as synthetic drugs and do not give results as quickly. However, in most cases, healing with herbal drugs is more effective as long-term irreversible healing. Another advantage is that most herbal drugs show minor or no side effects.

![Drug Diagram](image)

**Fig. 11.10** Different drugs target same disease

A physician may wish to look at all available options before choosing one. Choosing medication is also an individual thing because not all people respond in the same way to the same medications.

### 11.12.3 Example 3: Help Patients and General Public to Prevent a Disease

Some complex diseases, such as manic-depression, are caused by different factors (genetic and environmental). All the different factors that cause this illness have
not yet been identified. Also, the individual role of each causal factor in the onset of manic-depression is not yet known.

A person may know that manic-depression occurs frequently in his/her family line. It may be possible for him/her to inherit genetic cause(s). A patient cannot change his/her DNA, but he/she can influence the effects of environment on him/herself. In order to prevent manic-depression, the patient may wish to use the information system to retrieve information regarding environmental factors causing, or contributing to, manic-depression. In this case, the patient may alter his/her environment and reduce the risk of developing manic-depression. In Figure 11.11, we show that environmental factors such as climate, drug intake, stress and adverse family conditions may have an effect on the onset of this disease.

11.12.4 Example 4: Help Medical Researchers to Identify Disease Causes

Medical researchers may use the system to retrieve information regarding disease causes, such as in the case of manic-depression (Figure 11.12).

By querying the system and obtaining relevant information systematically represented, a researcher may be able to identify some regions of interest in the DNA sequence such as regions 2p13-16, 10q21-24, 12q23-24, 17q11-12 and Xq24-26 on chromosomes 2, 10, 12, 17 and X respectively (Craddock and Jones 2001; Liu et al. 2003).
Fig. 11.12 Genetic causes of manic-depression (current research)

Fig. 11.13 Genetic causes of manic-depression, future research if gene of interest found on chromosome X
These DNA regions need to be further examined in order to find a gene and a mutation inside that gene responsible for the onset of this disease. For example, mutation of gene GRK3, which is positioned on 22q11, is responsible for development of manic-depression in 10% of disease cases (Barrett et al. 2003).

Because of the agreed semantics in shared ontology, it will be easier for the next person to continue research in the same direction and possibly locate the gene of interest. For example, further research on 2p13-16 DNA region of interest, may allow scientist to narrow down the region to, for example, 2p14-15 and finally precisely locate the gene of interest on 2p15. This aspect of cooperation between different teams also increases productivity, and saves time and resources.

Given the length of the DNA sequence, it is obviously much easier for a researcher to target a specific area of a chromosome such as 10q21-q24 than the whole of chromosome 10. If a new Gene Z is found on 10q21-q24, for example on position 10q22, our model will have one less instance of the term ‘DNA region of interest’ and one more instance of the term ‘Gene’ (see Figure 11.13). Note that this example of Gene Z is given just for illustrative purposes.

11.12.5 Example 5: Help Medical Researcher Study Complex Diseases

As we previously mentioned, complex diseases are characterized by different types, such as depression, manic-depression and schizophrenia types of psychiatric diseases. Also, different environmental factors such as stress, adverse family conditions, climate etc. are, together with different genetic factors (e.g. gene mutation), responsible for the onset of such diseases (Verheyen et al. 1997). All different disease causing factors have not yet been identified. Also, the influence of each identified factor on the development of disease is not yet known.

Let e1, e2, e3 and e4 be the environmental causes, g1, g2, g3, g4 and g5 be the genetic causes and t1, and t2, and t3 be different types of a complex disease. We worked out 4 different possible hypotheses in this case.

Hypothesis 1: The same causes influence the phenotype (observable characteristics of an organism) differently.

Different influences can be expressed by different percentages and different influences explain different types of these diseases.

For example, the same causes e1, e2 and g1 are responsible for the types t1, t2 and t3 but have different influences on the phenotype. This is shown in Table 11.2.

| Types/Factors | e1 | e2 | g1          |
|---------------|----|----|-------------|
| t1            | K% | L% | 100 – (K + L)% |
| t2            | M% | N% | 100 – (M + N)% |
| t3            | X% | Y% | 100-(X + Y)%  |
In this case, different diseases types result from the different individual influences of the same factors.

Hypothesis 2: Different factors influence the phenotype equally.

For example, type t1 is influenced by factors e1, e2 and g1; type t2 by factors e3, g2 and g3; type t3 by factors e4, g4 and g5. This is presented in Table 11.3.

Table 11.3 Different factors, same influences

| Factors / Types | e1 | e2 | e3 | e4 | g1 | g2 | g3 | g4 | g5 |
|-----------------|----|----|----|----|----|----|----|----|----|
| t1              | X% | Y% |    |    | 100-(X+Y)% |    |    |    |    |
| t2              |    |    | X% |    | Y% | 100-(X+Y)% |    |    |    |
| t3              |    |    |    | X% |    | Y% | 100-(X+Y)% |    |    |

In this case, different types result from different factors associated with each type.

Hypothesis 3: Different factors influence the phenotype differently.

This is shown in Table 11.4.

In this case, different types are a result of both different causes, and different individual influences of these causes, on the development of that specific disease type.

Table 11.4 Different factors, different influences

| Factors / Types | e1 | e2 | e3 | e4 | g1 | g2 | g3 | g4 | g5 |
|-----------------|----|----|----|----|----|----|----|----|----|
| t1              | X% | Y% |    |    | 100-(X+Y)% |    |    |    |    |
| t2              |    |    | K% |    | L% | 100-(K+L)% |    |    |    |
| t3              |    |    |    | M% |    | M% | 100-(M+N)% |    |    |

Hypothesis 4:

All different factors have a cumulative effect and there is a threshold value that needs to be reached before the disease develops.

Note that it is possible for individual influences of different factors to have the same value. For instance, it is possible that X = Y.
In Table 11.5, we give an example where the development of diseases type t1 requires the presence of factors e1, e2, g3 and g5; of disease type t2, presence of e1, e2, g1 ad g4 factors; and of disease type 3, presence of e3, e4, g2 and g5 factors. If one of the required factors is missing, the corresponding disease type will not develop.

| Factors | e1 | e2 | e3 | e4 | g1 | g2 | g3 | g4 | g5 |
|---------|----|----|----|----|----|----|----|----|----|
| Influences | M% | N% | P% | R% | S% | X% | Y% | Y% | Z% |
| t1      | x  |    |    |    |    |    |    |    |    |
| t2      | x  | x  |    |    |    |    |    |    |    |
| t3      |    | x  | x  |    |    |    |    |    |    |

We illustrated here the importance of an intelligent information system in the information retrieval process for the purpose of testing hypothesis(es). It may be possible for medical researchers to retrieve all information regarding genetic and environmental factors as well as different types of a complex disease. Having all available information structurally organized, makes it much easier to identify relationships between the data, identify possible dependencies between different factors and prove one (or more) hypothesis(es).

### 11.13 Conclusion

In this chapter, we illustrated how to use the Onto-Agents Methodology to develop a Generic Human Disease Ontology (GHDO) and a multi-agent system that can use the designed GHDO for intelligent information retrieval. We also discussed numerous advantages of this system, and illustrated how physicians, medical researchers and patients can all make use of this system.

### References

1. Barrett, T.B., Hauger, R.L., Kennedy, J.L., Sadovnick, A.D., Remick, R.A., Keck, P.E., McElroy, S.L., Alexander, M., Shaw, S.H., Kelsoe, J.R.: Evidence that a single nucleotide polymorphism in the promoter of the G protein receptor kinase 3 gene is associated with bipolar disorder. Molecular Psychiatry 8, 546–557 (2003)
2. Bodenreider, O.: The Unified Medical language System (UMLS): integrating biomedical terminology. Nucleic Acids Res. 32, 267–270 (2004)
3. Ceusters, W., Martens, P., Dhaen, C., Terzic, B.: LinkFactory: an Advanced Formal Ontology Management System. Interactive tools for Knowledge Capture, KCAP-2001 (2001), http://www.isi.edu/~blythe/kcap-interaction/papers/LinkFactoryXWhiteXPaperXfinal.doc (retrieved: May 26, 2004)
4. Ceusters, W., Smith, B., Kumar, A., Dhaen, C.: Ontology-Based Error Detection in SNOMED-CT. In: Medinfo 2004, pp. 482–486 (2004)
5. Craddock, N., Jones, I.: Molecular genetics of bipolar disorder. The British Journal of Psychiatry 176, 128–133 (2001)
6. Deray, T., Verheyden, P.: Towards a semantic integration of medical relational databases by using ontologies: a case study. In: Meersman, R., Tari, Z. (eds.) OTM-WS 2003. LNCS, vol. 2889, pp. 137–150. Springer, Heidelberg (2003)

7. Mouratidis, H., Giorgini, P., Manson, G.A.: Modelling secure multiagent systems. In: The second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2003), pp. 859–866 (2003)

8. Hadzic, M., Ulieru, M., Chang, E.: Soft Computing agents for e-Health Applied to the Research and Control of Unknown Diseases. Information Sciences 176, 1190–1214 (2006)

9. Jarrar, M., De Mey, J., Meersman, R.: On Reusing Conceptual Data Modeling for Ontology Engineering. In: Meersman, R., Tari, Z. (eds.) OTM-WS 2003. LNCS, vol. 2889, pp. 185–207. Springer, Heidelberg (2003)

10. Jarrar, M., Meersman, R.: Formal Ontology Engineering in the DOGMA Approach. In: Meersman, R., Tari, Z., et al. (eds.) CoopIS 2002, DOA 2002, and ODBASE 2002. LNCS, vol. 2519, pp. 1238–1254. Springer, Heidelberg (2002)

11. Liu, J., Juo, S.H., Dewan, A., Grunn, A., Tong, X., Brito, M., Park, N., Loth, J.E., Kanyas, K., Lerer, B., Endicott, J., Penchasazadeh, G., Knowles, J.A., Ott, J., Gilliam, T.C., Baron, M.: Evidence for a putative bipolar disorder locus on 2p13-16 and other potential loci on 4q31, 7q34, 8q13, 9q31, 10q21-24, 13q32, 14q21 and 17q11-12. Mol. Psychiatry 8, 333–342 (2003)

12. Marchetti, D., Lanzola, G., Stefanelli, M.: An AI-Based Approach to Support Communication in Health Care Organizations. In: Quaglini, S., Barahona, P., Andreassen, S. (eds.) AIME 2001. LNCS (LNAI), vol. 2101, pp. 384–394. Springer, Heidelberg (2001)

13. Spyns, P.: Object Role Modelling for Ontology Engineering in the DOGMA Framework. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM-WS 2005. LNCS, vol. 3762, pp. 710–719. Springer, Heidelberg (2005)

14. Stevens, R., Baker, P., Bechofer, S., Ng, G., Jacoby, A., Paton, N.W., Goble, C.A., Brass, A.: TAMBIS: Transparent Access to Multiple Bioinformatics Information Sources. Bioinformatics, 184–186 (2002)

15. Verheyn, G., Raeymaekers, P., Van Broeckhoven, C.: De genetica van stemmingssstoornissen. Cns Review 2, 7–12 (1997)

16. Wooldridge, M.: An Introduction to Multiagent Systems. John Wiley and Sons, Chichester (2002)

17. Wongthontham, P., Dillon, D., Dillon, T.S., Chang, E.J.: Use of UML 2.1 to model multi-agent systems based on a goal-driven software engineering ontology. In: Proceedings of the 4th international conference on semantics, knowledge and grid (SKG 2008), China (2008)

18. Wouters, C., Dillon, T.S., Rahayu, J.W., Chang, E., Meersman, R.: Ontologies on the MOVE. In: Lee, Y., Li, J., Whang, K.-Y., Lee, D. (eds.) DASFAA 2004. LNCS, vol. 2973, pp. 812–823. Springer, Heidelberg (2004)