Centrum voor Wiskunde en Informatica

REPORTRAPPORT

INS

Information Systems

INS INformation Systems

Domain model enhanced search - A comparison of taxonomy, thesaurus and ontology

K. Schwarz

REPORT INS-E0514 SEPTEMBER 2005

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-3681

Domain model enhanced search - A comparison of taxonomy, thesaurus and ontology

ABSTRACT
This thesis investigates the use of domain models for improving electronic search in the context of an enterprise. Specifically the three domain modeling schemes taxonomy, thesaurus and ontology are compared. The result is intended to support information architects in the decision making process to determine the best solution to a search problem.

2000 Mathematics Subject Classification: 94A99
1998 ACM Computing Classification System: H.3.3
Keywords and Phrases: content management, knowledge engineering

Domain model enhanced search
- A comparison of taxonomy, thesaurus and ontology

Thesis by Katharina Schwarz

Master of Content and Knowledge Engineering
University of Utrecht

Supervisors:
Virginia Dignum, UU
Lynda Hardman, CWI
Remko Helms, UU
Timo Kouwenhoven, CIBIT Consultants | Educators

15 July 2005

Contents

List of Figures

Acknowledgement

I have been very lucky to have had the opportunity of studying the Master of Content and Knowledge Engineering at the University of Utrecht, to have benefitted from the support awarded to foreign students that come from member countries of the European Union, and to have had much moral support from family and friends. I want to thank everybody who has showed interest in my work and welfare in the past 6 months. There are a few people though whom I want to thank specifically.

My team of supervisors have contributed a great deal to this thesis. Timo Kouwenhoven has generously shared his wealth of knowledge and experience with me, from which I have learned very much. Besides an interest in domain modeling, we have found other shared interests. This made the collaboration even more enriching and enjoyable.

Lynda Hardman has guided me for nearly 3 years now, and has taught me a lot about scientific thinking and working. Although these have been important lessons, I treasure more the standard she has set in leading a group by creating an atmosphere of easy communication and joy in research. I hope I will always be able to work in a group like INS2.

Virginia Dignum has been a great source of knowledge and support throughout my whole master program. She has given me the freedom to define my own subject, yet she made sure that I was on the right path, and corrected me when I strayed from it. All teachers of the master program have been very enthusiastic and creative in their lessons. I have learned much from them during the master program.

Frank Nack has again given valuable feedback on this thesis. After having supervised the first thesis I am sure he is now once and for all thoroughly disgusted with my writing...

Timo Kouwenhoven has arranged interviews for me with Peter Nieuwenhuizen, Vincent Huis in 't Veld, Alma Wolthuis and Evert Jagerman. I want to thank them once more for their time and the input they gave to this thesis.

Rob van der Spek from CIBIT has given me the chance to write my thesis in cooperation with one of their consultants in the first place and to learn from their expertise. I want to thank him for this opportunity and the Information Management group for their feedback on my thesis presentation half-way.

I have had useful feedback from some of the presenters of the ARK taxonomy conference in Amsterdam, in particular Annie Wang, Mark Fea, Jack Teuber and Celia Mindelsohn. Also Eero Hyvoenen from the Museo Suomi project has answered thesis questions promptly and in detail. I want to thank them for their time and interest.

With Mari Pulkkinen I have shared many a laugh, a run and a talk. It was a great way to vent frustration and stress, and to get some relief from the daily thesis grind, especially in the middle, when everything was overwhelming and there seemed to be no sense of direction or purpose in the project. We had a good time. Good luck Mari!

Finally, I could never have come this far without the love, unconditional support and positive attitude of my parents and sister. I am very happy that we belong together and have such a good time with each other!

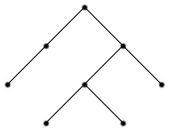
Chapter 1

Introduction

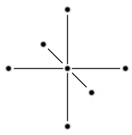
Much corporate information is captured in unstructured and semi-structured resources. It is difficult to find the right information in the abundance of resources that most enterprises cope with [17]. The main problem is to communicate clearly to the system what information is desired. This requires knowledge of both query language and domain terminology. Information Retrieval techniques have become very sophisticated in recent years to compensate for a lack in either [3]. Still a semantic gap remains that separates searchers from the information they seek.

An attempt to bridge the semantic gap between the information need of the searcher and the search query as it is interpreted by the search application is to insert a domain model as mediator. Models are an abstraction of reality. Domain models serve to improve search results by disambiguating the domain concepts and providing an organizational structure [46]. This structure can also be used in the user interface. Research in the field of Human Computer Interaction has improved search interaction styles based on domain models to cater to different types of searchers [13].

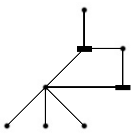
Modeling a domain is a difficult intellectual exercise that requires deep understanding of the domain. Domains can be modeled in many ways. Three of the prevalent domain modeling schemes are *taxonomy*, *thesaurus* and *ontology*, depicted graphically by the icons in the margin. We have chosen these three schemes because at the time of writing they are of great interest to practitioners. Generally there is confusion as to how these schemes differ from each other and what they are best used for.



Taxonomy



Thesaurus



Ontology

A taxonomy is essentially a hierarchical tree structure which models a domain from abstract to specific. A thesaurus is a structured vocabulary that defines each term by 3 types of relationships; hierarchical (as in a taxonomy), associative and equivalent. An ontology is the most formal model. It defines the meaning of concepts by modeling constraints that restrict the number of possible interpretations.

These 3 types of models differ mainly in their degree of precision. The more precise a model is, the more effort goes into making it, and the more features it offers. But what exactly is gained with increasing precision, and which problems do these features solve? These are common questions amongst information architects, domain modelers and practitioners. Our main research questions are:

> How are domain models used to enhance search?
>
> For which problems is each of these domain models especially suited?

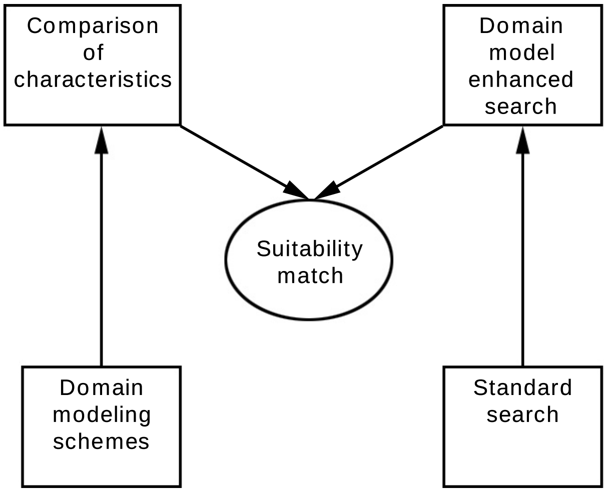


Figure 1.1: Research approach

1.1 Approach

To find the answers we have adopted the approach depicted in figure 1.1 on page 2. On the left side, we have the analysis and comparison of the 3 domain modeling schemes taxonomy, thesaurus and ontology. On the right side we look into the standard search process. We focus on the techniques developed to compensate for the lack of semantics in different phases of the search process. Then we examine the effect of enhancing the standard search process with domain models. Finally we combine both sides. Our goal is to find out for which application in the search process each individual scheme is especially well suited.

We draw relevant information from analyzing the following 12 case studies:

- AON
- Egon Zehnder International
- Flink
- LexisNexis
- McKinsey
- Dutch Ministry of Transport, Public Works and Water Management (hereafter called MinV&W, which is the acronym for the Dutch name of the Ministry)
- Museo Suomi
- Netherlands Institute for Sound and Vision
- PricewaterhouseCoopers
- Sainsburys
- Statoil
- WoltersKluwer UK

Most of the information about these case studies is based on presentations at the ARK taxonomy conference [19] held in Amsterdam in February of 2005. For 3 case studies interviews were arranged with key players. These were the projects at McKinsey, the Dutch MinV&W and the Netherlands Institute for Sound and Vision. Two case studies are research projects of which we studied the documentation and application websites. These are the case studies “Flink” and “Museo Suomi”. They illustrate the use of ontologies for searching. All case studies are described in detail in the appendix, and they are referenced throughout the thesis as examples of certain issues. This research has been presented in a preliminary version at the workshop “Formal Ontologies Meet Industry” [43].

1.2 Structure of the thesis

In chapter 2 we analyze and compare the domain structuring schemes taxonomy, thesaurus and ontology. In chapter 3 we describe the main elements of the search process; resources, search engine and index, result set and user interface, and what role domain models can play in each of these elements to enhance the search result. We conclude with chapter 4 where we present our findings concerning the suitability of each scheme for improving the search process. The appendix contains an overview of commercial search application tools and their functionality, and descriptions of the case studies.

Chapter 2

Domain modeling schemes

A domain model is an abstract representation of a small part of the world. The elements of a domain model are concepts, relationships between these concepts, and properties of the concepts and relationships. Relationships serve to define a concept in the context of other concepts. Properties further specify characteristics of a concept. By modeling a domain, the knowledge about it is captured and the assumptions on which the domain is built are made explicit [37].

A domain model serves to capture a common understanding of the domain to create a basis for unambiguous communication. Each individual has a unique personal conceptual model of the world. Modeling a domain is difficult, because the individual conceptual models of people first need to be elicited, and then reconciled in a single model.

Fundamental decisions that have to be made when modeling a domain concern the following characteristics.

Specificity What is the degree of detail that should be captured in the model?

Choice of terms Which term should be used to describe a concept?

Relationship types What relationships need to be modeled?

Properties What properties need to be modeled?

Monohierarchy or polyhierarchy A hierarchical relationship consists of a so-called parent and its children. Should a child concept be allowed to have a single or several parents? It is sometimes difficult to clearly associate a concept to a single type of parent, e.g. a mule would have two types of parents, the category “donkey” and the category “horse”.

Pre- or post-coordination A domain model has a high level of pre-coordination if it contains many *compound terms*, for example the compound term BACHELOR PARTY, which is made up of the individual terms BACHELOR and PARTY. Categories are more specific in a pre-coordinative model. This increases the maintenance effort of the model, but classifying resources is more straight-forward, as resources can be associated with a single category. On the other hand, in a post-coordinative model several single terms are combined at the moment of classification of a resource. This increases the flexibility of classification, as any term can be combined, and reduces the complexity of the model, since terms can be reused in different combinations (e.g. ANNIVERSARY and PARTY, BIRTHDAY and PARTY etc). However, the model is less structured and requires a higher effort in classification. The type of coordination has effect on the maintenance effort, the classification effort and the search performance.

The intrinsic modeling constructs of the schemes taxonomy, thesaurus and ontology impose different restrictions on the modeling of a domain. The representation language that is used to model a domain further restricts the expressivity of the model. Therefore some of the above mentioned decisions concerning the characteristics of the model are settled beforehand by the choice of modeling scheme and representation language.

Before we discuss these schemes, we need to clarify the relationship between *domain modeling* and *classification*. We have identified 3 ways to understand the term classification.

1. Classifying as a verb is synonymous with domain modeling; it is the act of grouping together similar or related concepts and arranging the resulting groups in a logical way.

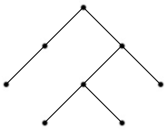
2. Classification as a noun is the resulting domain model. An example of a well known classification is the Dewey Decimal Classification [12], which is used to organize the collections of libraries.

3. A second meaning of classifying as a verb is used in relation to instances. Instances are classified according to an existing domain model in order to organize them, for example classifying individual books in a library according to the Dewey Decimal Classification System. It is important to distinguish the act of classifying on the concept level from the act of classifying on the instance level. In this chapter we are dealing with the former, in the next chapter, we discuss the latter.

In the following we describe the domain modeling schemes taxonomy, thesaurus and ontology. We have chosen these three because they are of great interest in the field of Information Management and Knowledge Representation. Conceptual modeling techniques for software applications and databases such as the Unified Modeling Language and Entity Relationship Modeling are comparable to these domain models, but they differ in their application and purpose. Whereas they deal with structured data and modeling processes, we are concerned with using domain models for improving search for unstructured data.

For each scheme, we give a definition, a description of their modeling characteristics and their application.

2.1 Taxonomy



To illustrate the concept of taxonomy we have taken a small excerpt from two large taxonomies on the Internet, the Yahoo! directory [29] and the Open Directory Project [11] (also called DMOZ, an acronym for Directory Mozilla).

Figures 2.1 and 2.2 on page 7 show the parts of the taxonomies that lead to the term *Salsa*. The numbers behind the term *Salsa* indicate how many web resources about *Salsa* are found.

In the following section we give our definition of a taxonomy which reflects how it is used in practice.

2.1.1 Definition

A taxonomy is a hierarchical domain structure. Parts of a taxonomy are

Hierarchical relationship it relates concepts from general to specific. The hierarchical relationship is transitive: whatever holds for a more general concept also holds for a more specific concept, e.g. music is a type of art. The hierarchical relationship is also called an IS_A relationship.

Level a hierarchy consists of various levels. The highest level is the most abstract. From top to bottom, the elements on the levels become increasingly concrete. All elements on one level should have approximately the same degree of abstraction.

Root this is the top of the structure, usually the domain or the source of the structure.

Node denotes a concept in the structure. Most nodes are both parent (of the lower level) and child (of the higher level).

Top node a concept on the first level below the root of the taxonomy. The first level is very important, because it reflects the chosen fundamental structure of the domain.

Leaf node a node that has no children nodes.

Sibling a node that has the same parent node as another node.

Path the sequence of nodes that are traversed to reach a specific node.

2.1.2 Modeling

There is no standardized approach to modeling a domain as a taxonomy, nor is there a commonly understood definition of taxonomy. The convention is to visualize a taxonomy as a tree structure. We illustrate the domain modeling constructs of such a structure with the example taxonomies.

Specificity In the DMOZ directory, all kinds of music styles are listed under *Styles*, such as country, folk or pop, and an additional category leads to *Regional and ethnic* styles. In the Yahoo! Directory this type of level is skipped in the path leading to *Salsa*. The DMOZ Directory is less specific in this area. The downsides of a high degree of specificity are that it is harder to reach agreement and the taxonomy is increasingly more complex and difficult to manage. Benefits of high specificity are less ambiguity and more precise search results.

Choice of terms In the DMOZ Directory the term *Styles* was used; in the Yahoo! Directory the term *Genres* was used for the same concept. Which term to use depends on the vocabulary of the target audience, but if there are several target audiences with different terminology this is a problem.

Relationship types A taxonomy officially only has the hierarchical relationship. Although this is a fundamental relationship, it is not enough to model a domain with. Besides the hierarchical relationship both directories also model associative relationships. *Music* and *Dance* are two different concepts, but *Salsa* belongs to both. Pointers are inserted to show the connection. In the DMOZ Directory, pointers lead directly to *Salsa* in a different context, whereas in the Yahoo! directory, pointers lead to the terms that define the context *Dance* and *Music*, and from there to the resources about *Salsa*. This is a makeshift solution to compensate for a lack in the modeling constructs of a taxonomy.

Properties Properties cannot be modeled in a taxonomy. Therefore each property that is required becomes a concept. Intuitively *style* would be considered a property of *Music*. In the DMOZ directory it is modeled as a concept. This practice violates the hierarchical relationship. In the example, "Regional and ethnic" is modeled as though it were a kind of music (as child of "Styles" which is a child of "Music"), yet it is a property of music. Again, a practical solution is chosen to make up for a deficit in the modeling constructs of a taxonomy.

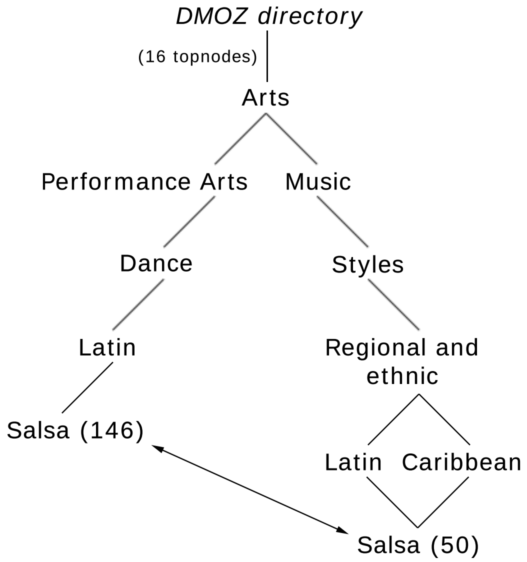


Figure 2.1: Classification of the term *Salsa* in the Open Directory Project

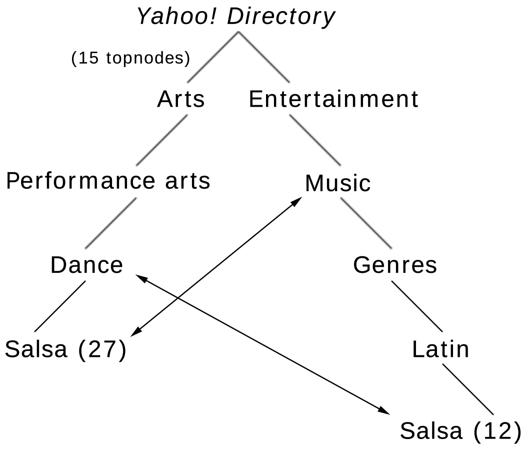


Figure 2.2: Classification of the term *Salsa* in the Yahoo! Directory

Monohierarchy vs polyhierarchy In the Yahoo! Directory, both the categories *Latin* and *Caribbean* lead to the term *Salsa*. This means that it is a polyhierarchy, where concepts can have more than one parent. It is a way of modeling associative relationships. A taxonomy should not be polyhierarchical, because it becomes very complex and difficult to manage.

Pre- and post-coordination A taxonomy is a pre-coordinative model, because the compound terms are defined when the structure is made, and they are assigned a place in the hierarchy. An example of a compound term in the DMOZ Directory is the node *Regional and ethnic*.

It is difficult to press domain knowledge into a rigid model, as the description of the modeling constructs has shown. In practice it is difficult to adhere strictly to the hierarchical relationship. For example in figure 2.1, the next level below the term *salsa* could consist of the concepts *schools*, *clubs*, *accessories*, *history* etc. These concepts do not have a hierarchical relationship with their parent *salsa*, but an associative relationship. Since there is no real restriction to modeling these types of relationships as well, it is done in practice. The type of each relationship in a taxonomy is implicit, but it is usually interpreted correctly by people. The strict definition of taxonomy does not apply in most cases. Rachel Hammond [23] describes a taxonomy in the following way

> It is usually a formal collection of words and/or phrases which describe a set of related concepts and the relationships between them, to a more or less elaborate level.

Another problem is the question of *mono- and multidimensionality* of the domain. With the top nodes of the taxonomy, the domain modeler tries to capture the most fundamental groups of the domain. Often however a domain can be structured along several dimensions. In this case the modeler chooses one dimension for the top nodes, and has to repeat the other dimensions throughout the model. A problem with repeating terms arises when the taxonomy vocabulary is used to classify resources. A term that is repeated in different sections of the model might unintentionally relate the resource to a section that it has nothing to do with.

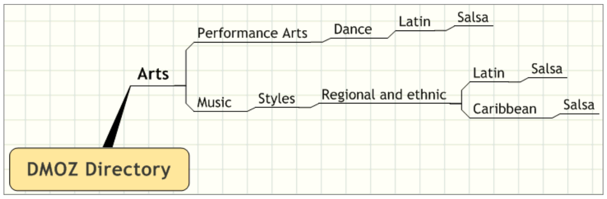


Figure 2.3: An example of the DMOZ taxonomy of *salsa* made with Mindmanager

For example in the DMOZ Directory in figure 2.1 on page 7, the node *Regional and ethnic* and its children nodes *Latin* and *Caribbean* are likely to return in numerous places of the taxonomy. Here they categorize styles of dancing, but they may also be used to categorize types of food, art, business, sports etc. As a matter of fact, *Regional* is also a top node in the DMOZ taxonomy, as can be seen in figure 3.7 on page 33. All other top nodes are repeated as categories under each regional node. It is a cumbersome solution that leads to an unwieldy taxonomy and much maintenance effort. This problem can be solved by identifying the dimensions and making a separate taxonomy of each dimension. This results in a facet classification, which is explained in more detail in section 2.5 on page 18.

The excerpts from the 2 large taxonomies also illustrate the earlier mentioned problem of domain modeling, which is that of varying conceptual models. In this case, the Yahoo! Directory categorizes *Music* under *Entertainment* and *Dance* under *Performance Arts*, whereas the DMOZ Directory categorizes both *Music* and *Performance Arts* under *Arts*. People who are used to the one will be irritated by the different classification in the other, but they can cope. In a digital environment searches are often supported by software agents. These are autonomous or semi-autonomous proactive and reactive programs. Software agents would have trouble recognizing that both concepts of *Music* refer to the same thing.

One of the most important modeling rules is to identify the conceptual model of the users of a domain model, and match it. The common approach to achieve this is to take a group of representatives from the target users to model the domain. Here the difficulty is knowing how well the representatives represent the target audience, and how homogeneous the conceptual models of all users within the target audience are.

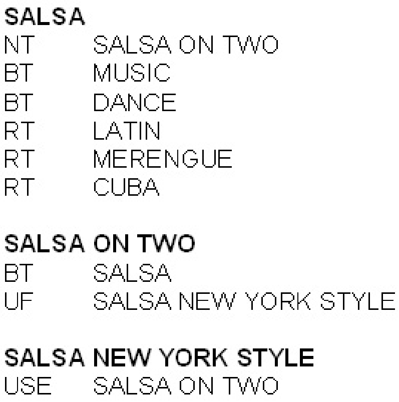
An attempt to solve the problem is to let all users of a domain model influence its structure and content. The Open Directory Project [11] is an interesting example of collaborative and autonomous creation of a taxonomy. The project aims to give structure to the web by categorizing all the websites. The structure is made by and for web users. Any web user can add a part of the taxonomy that is not yet included, and classify sites that deal with that subject. The problem with this approach is total lack of control over the content and structure of the taxonomy.

There is no standardized notation to describe a taxonomy with. A commonly used program for making a taxonomy is MindManager by Mindjet[1]. This is a tool that visualizes a tree structure, as illustrated in figure 2.3 with an extract from the DMOZ Directory leading to the concept of *salsa*. MindManager can export a taxonomy in an XML[2] representation.

2.1.3 Application

The initial application of taxonomies was to structure the natural world around us. The Swedish scientist Carolus Linnaeus is credited to be the first to have created a

[1] http://www.mindjet.com/eu/
[2] http://www.w3.org/XML/



SALSA
NT SALSA ON TWO
BT MUSIC
BT DANCE
RT LATIN
RT MERENGUE
RT CUBA

SALSA ON TWO
BT SALSA
UF SALSA NEW YORK STYLE

SALSA NEW YORK STYLE
USE SALSA ON TWO

Figure 2.4: An illustration of modeling the term *salsa* in a thesaurus

hierarchical classification, called taxonomy, of the flora and fauna based on observable characteristics. The taxonomy serves to create order and relate the concepts to each other, thereby supporting understanding of the domain, e.g. enabling the distinction of roses and cactuses, although both have thorns. In the Linnaeus taxonomy the relationships are strictly hierarchical, in the sense that each concept inherits all characteristics of its parent (genus-species). The aim was to be able to clearly identify every kind of animal or plant by assigning it to a single place in the hierarchy. Taxonomies are still used in the classical sense in the domain of natural sciences.

In a corporate environment there are two common applications for a domain taxonomy. On the one hand it can be used as a structured vocabulary for classifying resources consistently, and for facilitating retrieval. On the other hand, because of its inherent tree structure, it can be used as the basis for a visual navigation structure in the user interface.

2.2 Thesaurus

Most people get to know a thesaurus as a handy book of synonyms. What they do not know is that thesauri have a much wider field of application, and carry much more information than just synonyms. Figure 2.4 is a small illustration of how the term SALSA might be described in a thesaurus.

2.2.1 Definition

A thesaurus is a structured vocabulary. The purpose of a thesaurus is to facilitate retrieval of resources and to achieve consistency in indexing [1]. Widely accepted standards determine precisely the conventions for construing a thesaurus. These are the International Standard ISO 2788, the British Standard BS 5723 and the US Standard ANSI/NISO Z39/19. Fundamental elements of thesaurus construction are

- the form of terms, e.g. grammatical form, order of compound terms, and
- relationships between terms to create structure.

The thesaurus standards state conventions for the form in which the terms are recorded in the thesaurus, e.g. singular or plural (DANCE, ARTS), form of the word (PERFORMANCE, not PERFORMING) and the order of compound terms. This way consistency in indexing is achieved.

The types of relationships that are defined are *equivalence*, *hierarchical* and *associative.*

An equivalence relationship states which term is the preferred term to denote a specific concept, and which terms are also used to describe the same concept. This includes for example synonyms, colloquial terms, transliterations and culturally different terms. The notation for an equivalence relationship is USE - UF (Use For). In the example figure 2.4 on page 9 the terms SALSA ON TWO and SALSA NEW YORK STYLE are defined as being equivalent, and SALSA ON TWO is defined as the preferred term.

The standards [1] for building thesauri define the following 3 types of hierarchical relationships

- genus / species, e.g. dog / Labrador,
- concept / instance_of, e.g. lake / Garda Lake and
- whole / part, e.g. ear / middle ear. The whole-part relationship is usually seen as an associative relationship, but there are four cases in which it is considered a hierarchical relationship, because the part implies the whole in any context. These cases are
  - Systems and organs of the body, e.g. brain - grey matter
  - Geographical location, e.g. Netherlands - North Holland
  - Discipline or field of study, e.g. history - art history - 19th century art history
  - Hierarchical social structure, e.g. methodist church organization - methodist district

A hierarchical relationship can only exist between terms of the same category. Dividing the subject field into categories is a way to provide a fundamental structure. The basic categories defined in the standard ISO 2788 are *concrete entities* (e.g. ANIMALS, FLOWERS, WOOD, GLASS), *abstract concepts* (e.g. PEACE, MATHEMATICS, DIGESTION) and *individual entities* (e.g. THE NETHERLANDS, THE RAIN FOREST, THE EUROMAST). The notation of a hierarchical relationship is BT - NT (Broader Term - Narrower Term). In the example, figure 2.4 on page 9, MUSIC and DANCE are defined as broader terms of SALSA, because Salsa is a kind of music and a kind of dance. SALSA ON TWO is defined as a narrower term of SALSA, because it is a specific form of Salsa dancing.

The associative relationship is used for remaining relationships. In the thesaurus standards, 13 types of associative relationships are defined. Examples are *an action and the product of that action* (e.g. ROADMAKING - ROADS) and *concepts related to their origins* (e.g. WATER - WATER WELLS). The complete list can be found in the thesaurus manual [1]. The notation of an associative relationship is RT (Related Term). In the example figure 2.4 on page 9, related terms of SALSA are LATIN, a property of salsa dancing, MERENGUE, a similar style of dancing and CUBA, the country of origin of salsa dancing.

2.2.2 Modeling

The thesaurus standards give detailed modeling instructions and recommendations.

Specificity Specificity is an important element to control the performance of information retrieval with the help of the thesaurus. The more specific the vocabulary terms are, the better are the chances of matching a search query with the required information. On the other hand, increasing specificity also increases the number of terms in a thesaurus, which leads to higher maintenance effort and might slow down the search.

Choice of terms The problem of having to settle on a single term is solved by the equivalence relationship. All synonyms can be captured and related to the preferred term that is used to express a concept.

Relationship types All relationships in a thesaurus are reciprocal. Each term has a pointer to the related term. When modeling relationships, the standards provide guidance in choosing the right kind of relationship between 2 terms. There are 3 official interpretations of a hierarchical relationship, 13 official interpretations of an associative relationship, and 4 official interpretations of an equivalence relationship [1]. In the end, only these 3 generic relationships are modeled. The reasoning for assigning a specific relationship is lost, e.g. was DANCE related to MUSIC as a RT (Related Term) because of a causal relationship, or a relationship based on origin, or something else? This is important knowledge for people who classify or index resources with the thesaurus, and who maintain it. Since the type of relationship can be a borderline choice, the reasoning behind the choice should be captured in a scope note. Scope notes are used for writing definitions and comments. The standards prescribe the types of comments that may be written in scope notes, e.g. an *indication of restrictions on meaning* or *term histories*.

Properties Properties cannot be modeled explicitly in a thesaurus. There is no construct that identifies a property from a concept. Some types of associative relationships do refer to properties, such as *concepts related to their properties* and *an action and a property associated with it*. However this distinction is not captured in the relationship.

Monohierarchy vs polyhierarchy The polyhierarchical relationship is defined and accepted. This occurs when a term has more than one BT (Broader Term) relationship. Polyhierarchy is common in a thesaurus. This is no problem, because the focus of structure in a thesaurus is on the local term structure, e.g. the relationships pertaining to one term, as can be seen in the example 2.4 on page 9. The global structure does not need to be coherent.

Pre- and post-coordination Thesauri are mainly post-coordinative models. This can best be explained by comparing figure 2.2 on page 7 with figure 2.4 on page 9. In the Yahoo directory, *Latin* is clearly associated to *Genres*, and *Genres* to *Music*. If the term *Latin* should be used in another part of the taxonomy, it has to be repeated in the structure. In a thesaurus, all hierarchical, associative and equivalence relationships of the term *Latin* are specified in the term itself. When the thesaurus is used for classification of resources, the term *Latin* can be applied in any combination with other terms. This combination makes a thesaurus post-coordinative.

2.2.3 Application

A thesaurus can be used in two areas to facilitate the retrieval of information resources: the search query and the index. In the first case, a search query can be expanded to cover equivalent and related terms. In this way, if someone is looking for resources dealing with “music style”, but the terms used in the repository are “music genre”, relevant resources can still be found if “style” was defined as being equivalent to “genre”. The hierarchical relationship of a thesaurus can be used to increase or decrease the result set, depending on the number of hits. If there are too few results, the search engine could also return resources that are related to a parent node, or resources dealing with

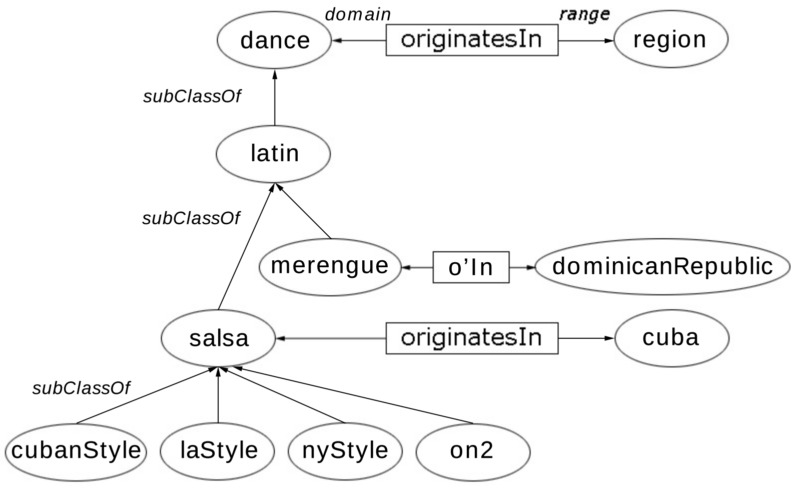


Figure 2.5: An ontology of dance with the focus on *salsa*

associated terms. If there are too many hits, the search engine could refine the search with children nodes. Query expansion is not a feature of every search engine. Search engines are described in more detail in chapter 3 on page 23.

Indexing is the classical application of a thesaurus. An index is a list of terms, where each term points to at least one resource that deals with the concept denoted by the term. Using thesaurus terms in the index is equivalent to adding metadata to a resource.

2.3 Ontology

The concept *ontology* is originally a branch of metaphysics that deals with the nature of being. An ontology captures and structures knowledge so that it can be shared and reused [47]. Figure 2.5 shows an excerpt of a possible ontology of dance. It contains the same terms as the thesaurus example in figure 2.4 on page 9, but it is much more specific in describing the nature of the relationships of these terms, e.g. the relationship of *salsa* with *merengue* (another Latin dance style) and with *Cuba* (the country of origin).

Ontologies have become an important element of the field of Artificial Intelligence, in particular Natural Language Understanding and Knowledge Based Problem Solving. Since a valid domain model is a prerequisite to developing any information system, ontologies are also widely applied in object-oriented software system design and information retrieval systems [9]. Among practitioners there is an aura of ambiguity around ontologies because of their wide application in various fields and the different types of knowledge that are modeled with them, such as objective realities and empirical facts [25].

Mike Uschold and Robert Jasper [49] have developed a framework to help classify ontology application areas and reduce the ambiguity around the term ontology. They have defined four main categories of application areas, one of which is Ontology-Based Search. In this research, we concern ourselves with this category, and focus on finding information with the help of domain models. More specifically, we consider ontologies in the context of the Semantic Web. The goal of the Semantic Web is to leverage the information and knowledge available on the World Wide Web by making the content machine readable [5]. The basic idea is to enable machines to understand the meaning of web content, so that web services and agents can better support its usage. Uschold acknowledges that this is already being done in today's Web. The role of ontologies in the Semantic Web is to formally define the meaning of the used terminology. Software agents can understand the meaning of meta data by looking it up in the associated ontology.

After defining the concept of ontology, we briefly explain the Semantic Web and the role of ontologies, and some of their representation languages. We then look at the modeling constructs provided by one of the representation languages. Finally we consider the application of ontologies.

2.3.1 Definition

Studer et al. [46] (p.25) provide the most commonly cited definition of ontology for the field of knowledge representation:

> An ontology is a formal, explicit specification of a shared conceptualization. A conceptualization refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. Explicit means that the type of concepts used, and the constraints on their use are explicitly defined. For example, in medical domains, the concepts are diseases and symptoms, the relations between them are causal and a constraint is that a disease cannot cause itself. Formal refers to the fact that the ontology should be machine readable, which excludes natural language. Shared reflects the notion that an ontology captures consensual knowledge, that is, it is not private to some individual, but accepted by a group.

The requirement of a formal specification is one of the main differences between an ontology and the two schemes taxonomy and thesaurus. This makes ontologies especially well suited for use in the Semantic Web.

2.3.2 Ontology for the Semantic Web

The Semantic Web is the name given to the development of the World Wide Web that is envisaged and supported by the Web research community. A precise definition of the Semantic Web is missing, but Michael Uschold has identified 2 key characteristics; (1) machine usable content that is (2) associated with more meaning [48].

A common application is searching the Web and comparing prices of products with each other, for example at the website *www.vergelijk.nl*. This is based on information that is captured in tags and added to the web content. An agent can find all tags that are called *price* in association with a specific product, and compare their content. This works because the agent is programmed to search precisely for the tag called *price*. The agent does not know the meaning of the word *price*. The semantics of the word *price* are implicitly understood and agreed upon by the humans who define the tag and use it. The disadvantage of this approach is that different vendors may give the same thing different names, e.g. *endprice*, *saleprice* or *value*, or they might give different thing the same name, e.g. the price before tax, the purchase price or the price including shipping costs.

In the Semantic Web infrastructure, an additional element formally describes the meaning of the word *price*, so that its meaning is unequivocally defined for the software agent. This additional element is an ontology that describes the concept of price, its properties and its relationships. On the website, the price of a product is marked by a tag that refers to the ontology. An agent who comes across the tag can look up the meaning in the ontology, recognize the price of the product and compare it to others.

The advantage of the Semantic Web in this case is the following. Now, all website owners who want their product to be included in the comparison have to make sure that they use a tag defined by the operators of *www.vergelijk.nl.* If a website uses a different tag for the same thing, it is not included. In a Semantic Web scenario, the agreement about which tag to use happens at a higher level; at the level of ontology definition, for example an ontology of web shops. This web shop ontology needs to be publically available. Website owners that annotate their content according to the ontology will be automatically included in the comparison, and the right prices will be compared.

The Semantic Web Community[3] has developed languages for describing ontologies and annotating resources. These are presented in the following section.

2.3.3 Modeling

The *Resource Description Framework (RDF)* is a data model based on XML syntax that serves to describe resources. In RDF, resources and their properties are modeled as statements. Statements are triples that consist of a resource, a property and a value, which is an analogue to the basic syntax of a natural language sentence: subject, predicate and object. Any statement can be made in RDF, there is no control for correctness or contradictions, e.g. " a rose is a cactus". Wrong statements can be discovered when its elements are formally described in an ontology, in this case an ontology that defines that roses belong to a different family than cacti.

The main elements that are modeled in ontologies are concepts, relations between the concepts and properties of those concepts and their values. *RDF Schema* is the simplest ontology representation language that has become a recommendation by the W3C [50]. It can model concepts and properties, and restrict the domain and range of concepts to some degree. The *Web Ontology Language OWL* was developed to provide more expressivity for modeling a domain. In OWL, additional elements are defined such as data type properties, cardinality, more detailed range and domain specifications, types of properties (transitive, symmetric, functional), boolean combinations and enumerations. The following is a small example of how the classes *latin* and *salsa* in the salsa ontology in figure 2.5 on page 12 could be described with OWL. Since OWL builds on RDF Schema, basic commands remain in use.

```
<owl:Class rdf:ID="latin">

   <rdfs:comment>the class of latin dance</rdfs:comment>

   <rdfs:label>latin dance</rdfs:label>

   <rdfs:subClassOf rdf:resource="#dance"/>

</owl:Class>


<owl:Class rdf:ID="salsa">

   <rdfs:comment>the class of salsa dance</rdfs:comment>

   <rdfs:label>salsa dance</rdfs:label>
```

[3] http://www.w3.org/2001/sw/

```
   <rdfs:subClassOf rdf:resource="#latin"/>

   <rdfs:subClassOf>

      <owl:Restriction>

      <owl:onProperty rdf:resource="#originatesIn"/>

      <owl:hasValue rdf:resource="#cuba"/>

      </owl:Restriction>

   </rdfs:subClassOf>

</owl:Class>
```

Another ontology representation language is the ISO standard *Topic Maps*, explained in detail by Steve Pepper [40]. Its modeling constructs are topics, associations and occurrences. The idea is to be able to model a domain intuitively using these constructs to create any concept and relationship that is required.

Specificity An ontology specifies the meaning of concepts with constraints that narrow down the range of possible interpretations of the concept. The degree of specificity determines how precisely a concept is defined. The better the concept is defined, the less ambiguous it is. A characteristic of a good ontology is high specificity, provided that the constraints are correct. A mediocre ontology is much less specific, so there is more room for misinterpretation. This can lead to false agreement when mapping concepts from two ontologies to each other. If the modeled constraints are wrong the ontology is also wrong.

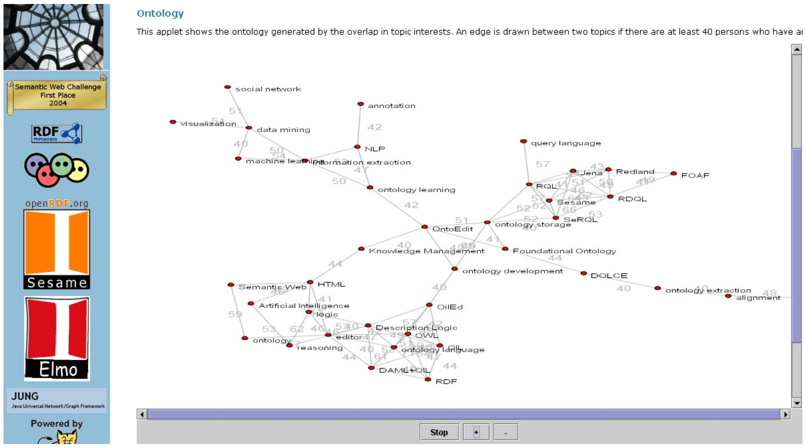
Choice of terms An ontology is intended to define the meaning of a concept, so that it can mediate between different terms [21]. It is not intended to impose a conceptual model and a specific vocabulary onto its users. On the contrary, the point of an ontology is to allow users to maintain their individuals models, yet still be able to communicate with others via an ontology that mediates between them.

Relationship types The hierarchical structure is the core building block of an ontology [9]. Relationships are defined by properties. The hierarchical relationship is defined in RDF Schema with the properties `rdf:type` (instance of), `rdfs:subClassOf` (genus-species) and `rdfs:subPropertyOf` (a specification of a property). Associative relationships are also defined with properties.

Properties Properties are a fundamental construct of modeling ontologies. They are either attributes of a class, or they define the relationship between two classes. There are predefined restricting attributes that constrain the range and domain of a class. When the property is seen as the predicate, e.g. "drives", then `rdfs:domain` restricts the subject, e.g. "human being", and `rdfs:range` restricts the allowed objects, e.g. "car".

Monohierarchy vs polyhierarchy Polyhierarchy is allowed, which means that a subclass can have more than one parent class. This lends more flexibility to modeling, but also more complexity.

Pre- and post-coordination The coordination of an ontology is not easy to define. It is pre-coordinative in the sense that concepts, properties and relationships are



Ontology

This applet shows the ontology generated by the overlap in topic interests. An edge is drawn between two topics if there are at least 40 persons who have a

Figure 2.6: An automatically generated ontology of the research area *Semantic Web* on the website of Flink (page 52)

explicitly defined in the model. On the other hand, when an ontology is populated with instances, properties can be combined in any way that is necessary.

Developing an ontology requires a huge effort. The level of detail that is modeled is much deeper than in taxonomies or thesauri, and the definition states that there has to be consensus about the domain model amongst the group of all users, so a lot of discussion and communication is required before an ontology is passed. With the aim of knowledge sharing and reuse in mind, it is strongly recommended to reuse an existing ontology instead of making a new one. Fundamental ontologies are available that model the world at large, e.g. CYC[4] and WordNet[5], as well as domain specific ontologies in ontology libraries[6].

In the Flink case study (page 52) an attempt is made at generating an ontology on the basis of overlapping topic interests in the Semantic Web community (figure 2.6).

2.3.4 Application

One of the most important features of ontologies is automated inferencing. Because the knowledge captured in the domain model is formalized, agents can understand it and act accordingly. On the Semantic Web this feature is used for intelligent information retrieval, for example, to find resources dealing with related concepts, although the resources were never explicitly related to each other.

Two interesting Semantic Web applications that are both results of research projects which we describe in the appendix (Flink on page 52 and Museo Suomi on page 59). The former shows how existing information on the Web can be reused for semantic search, the latter shows how distributed repositories can be conceptually combined so that semantic recommendations can be made.

[4] http://www.opencyc.org/

[5] http://wordnet.princeton.edu/

[6] http://www.daml.org/ontologies/

2.4 Comparison of taxonomy, thesaurus and ontology

A taxonomy in its classic sense of being a hierarchical structure of genus-species relationships is a constituent of both thesauri and ontologies. Modeling an ontology begins with making a taxonomy of the domain.

In the following table we summarize the differences between the 3 domain models.

Differences between taxonomy, thesaurus and ontology			
	Taxonomy	**Thesaurus**	**Ontology**
Background	Natural Sciences, e.g. biology, chemistry	Library Sciences	Metaphysics, Artificial Intelligence, knowledge engineering
Standard:			
Modeling	none	ISO 2788, BS 5723, ANSI/NISO Z39/19	Some methodologies (e.g. CommonKADS), but no official standard
Notational	Graphical tree structure	Thesaurus symbols (BT, NT, RT, UF, USE)	W3C recommendations (e.g. RDF Schema, OWL)
Modeling constructs:			
Relationships	basically hierarchical, but all types of relationships are modeled using the same notation	untyped hierarchical, associative and equivalence	typed hierarchical and associative
Properties	none	if required, they can be described in scope notes	In RDF Schema: Relationship properties and restricting properties (domain, range)
Application	Classification, navigation, search	Classification, navigation, search	Classification, navigation, search, visualization, automated reasoning
Tools for creation	MindManager	MultiTES	Prot*égé*

The lack of a standard for the modeling and notation of taxonomies gives the modeler much freedom. There are no restrictions or rules to building a taxonomy. In practice, thesaurus-like functionality is often added to taxonomies, such as associative relationships and scope notes with definitions of terms and equivalent terms. The relationships are not always strictly hierarchical. The downside is that such taxonomies are ambiguous and prone to misinterpretation, and cannot be easily reused.

Because thesauri are structured according to a standard, they can be transformed semi-automatically to an ontology representation language, such as RDF Schema. This is a welcome source for reuse, because many thesauri already exist, and the development of ontologies is expensive. Wielinga et al. [51] have tested the conversion of the Arts and Architecture Thesaurus to an ontology described in RDF Schema. Manual intervention is required to check that the hierarchical relationships are correct, and that they strictly obey the rules of inheritance. An example of this problem was given in the case study of Museo Suomi, where the Finnish cultural thesaurus MASA was transformed into the ontology MAO [28]. In the thesaurus, a “make-up mirror” was defined as a narrower term of “mirror”, and “mirror” was defined as a narrower term of “furniture”. In the context of a thesaurus this does not imply that a make-up mirror is a piece of furniture. On the other hand in an ontology, this would indeed lead to the conclusion that a make-up mirror is a piece of furniture, because the hierarchical *subClassOf* relationship is transitive.

The modeling constructs, in particular, give each scheme its individual character. Properties cannot be modeled explicitly in taxonomies or thesauri. In a taxonomy, properties are included in the hierarchy and related to the terms which they describe, but this practice violates the hierarchical relationship. In a thesaurus, properties are usually modeled with the associative relationship, which is a bit more accurate. In an ontology, properties play a fundamental role, as they are the relationships.

Taxonomies are pre-coordinative. This has the advantage of leading to higher precision and the disadvantages of being very restrictive and leading to lower recall. Thesauri are mainly post-coordinative. The advantages are higher flexibility and recall, the disadvantage is a precision loss. Ontologies are both pre- and post-coordinative.

Taxonomies and thesauri are very similar in their application, yet taxonomies are gaining popularity in enterprises and search engine vendors. From practitioners we have repeatedly heard the opinion that the concept of taxonomy has a “sexy” ring to it (McKinsey, page 55, Sainsbury’s, page 64), in comparison to the library-tainted thesaurus or the scientific ontology. However, it also needs de-mystifying for people to understand what it is for and to be able to work with it (Statoil). For this reason the taxonomy developed at Sainsbury’s was called a classification. A popular metaphor for taxonomy is “the glue that ties together the content” (AON, page 50, Statoil, page 65).

2.5 Facet classification

Getting started is the hardest part of modeling a domain. A common approach to deal with the complexity of domains is to start by dividing the domain into rough groups. This is called Facet Classification. The characteristics of facets are that

- they are so general that they can be applied to any domain and

- they produce mutually exclusive, homogeneous groups.

13 Fundamental facets are distinguished by Vanda Broughton [6].

The following table shows a selection of facets in the left column, and illustrates how they are applied to structure the domain of arts in the right column.

Generic facets	**Facets applied to an art collection**
Materials/substances, constituent substances	Media, e.g. book, glass, painting, sculpture, metalwork
Time	Date
Space	Location
Operations (external, transitive actions)	Occupations, e.g. entertainer, leader, professional, worker
Living entities, organisms	Animals and Plants, e.g. birds, creatures and beasts, flowers, insects, parts of plants, trees
Naturally occurring entities	Heaven and Earth, e.g. dawn, dusk, night/ islands, deserts, forests/ mountains, hills, valleys/ rivers, lakes, seas
Artifacts (man-made)	Built places, e.g. bridge, building, part of building, road
End-products	Objects, e.g. clothing, containers, musical instruments
Abstract entities	Themes, e.g. military, mortality, nautical, religion
Attributes: properties/qualities, states/conditions	Shapes, Colors and Scenes, e.g. color, decoration, metal
Agents (performers of action - inanimate and animate)	Artists

Facet classification provides a way to systematically analyze a domain, answering the fundamental questions of *what* is being done, *how* it is being done, *by what means*, *where* and *when* [6]. Making this rough initial structure has the effect of the "divide and conquer" principle. The individual facets are treated as domains and structured according to a specific domain modeling scheme.

The difference between facets and the top level of a taxonomy is that facets are defined by the two characteristics mentioned above, whereas the top level concepts of a taxonomy are not restricted in any way. For example, the top level nodes of the DMOZ directory which can be seen in figure 3.7 on page 33 do not match the facets described above. There are several "end-products", such as the nodes "Games" and "Computers". There are several "abstract entities", like "Business" and "Home", and some nodes do not fit any facet, such as "Reference" and "World". In this way it is impossible to produce mutually exclusive, homogeneous groups. For this reason a taxonomy can contain repeating groups.

Modeling a domain with facets leads to a specific way of adding metadata to resources, and of applying the domain model in the user interface. Each facet can be seen as a metadata field of a resource with which a value gets associated. This is elaborated in section 3.2.1 on page 27. It leads to a particular style of interaction in the user interface, which is explained in detail in section 3.2.4 on page 34.

2.6 Conclusions

In this chapter we discussed domain modeling in general, introduced the 3 domain modeling schemes taxonomy, thesaurus and ontology and described them from a modeling and application perspective. This was followed by a comparison of the 3 schemes. Finally we explained the concept of facet classification, because it is a basic approach to begin with modeling a domain.

We have seen that the main difference between these schemes is their degree of specificity, also called ontological precision. A taxonomy is a rough modeling scheme because it only connects terms to each other with a hierarchical relationship. A thesaurus is more specific, because it also offers equivalence and associative relationships. An ontology is most precise because it defines the meaning of concepts by modeling properties that constrain the possible interpretations of a concept.

In the following chapter we will see how domain models can be used in search.

Chapter 3

Application of domain models in digital search

The best way to get information is to ask somebody who knows the answer. If this approach is not an option, we look for information that has been captured in a resource and stored somewhere. The field of Information Retrieval deals with finding the right information when it is needed [3]. We consider only text-based information in this research.

The digital search process consists of the following elements. *Resources* are created, sometimes annotated with *meta data*, and stored in a repository. Users access a *user interface* to enter their query. A *search engine* processes the query and accesses a repository, often via an *index*, and retrieves the resources that match the search query, which is the *result set*. We have created icons to represent these elements, because they will be used often in this chapter for illustration (see figure 3.1 on page 22).

The main difficulty in this process is conveying the information need of the user to the search engine. Basically, a search engine matches the string of characters in the search query with exactly the same string of characters in resources. This way the meaning of the terms is not taken into account. Since language is ambiguous, e.g. words may have different meanings but be spelled the same (homonyms), or words with the same meaning are spelled differently (synonyms), the search result is likely to be ineffective.

The main effectiveness measures of search results are *recall* and *precision* [18]. A searcher hopes to retrieve all the relevant material for the search question (recall), and at the same time no irrelevant items (precision). To improve these measures for search results and to compensate the lack of meaning in a syntax-based search, different techniques have been developed for each part of the search process. These are based mainly on statistical analysis of the resources. They are described in the first part of this chapter.

However, these techniques can only apply if the resources fulfill all the following requirements:

- the resources consist of computer readable text;
- each resource consists of at least a page of text;
- there is a large number of resources (1.000 is not very many, 10.000 is better, 1.000.000 is ideal);
- the domain vocabulary is heterogeneous (e.g. news articles are heterogeneous, medical article are homogeneous).

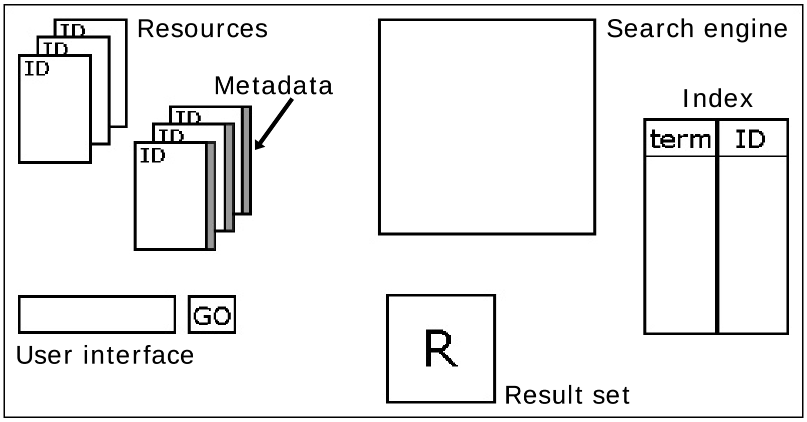


Figure 3.1: Basic elements of the process of retrieving information

Instead of trying to derive the relationships of terms from statistical analysis, they can be explicitly defined. As we have seen in the previous chapter, domain models serve to model terms and their relationships. The knowledge captured in domain models can be applied in the search process to improve the search results. In the second part of this chapter, we show which roles domain models can play in the search process.

3.1 Standard search

This type of information retrieval requires hardly any effort in set-up and maintenance. Although little effort generally results in low quality, there are many ways to improve the quality of the search result. These are described in the following.

3.1.1 Resources

The bulk of corporate information is captured in structured, semi-structured and unstructured resources. Structure can apply at different levels.

- The directory structure of a repository (file system) or the structure of a database
- The internal structure of a resource (e.g. document template in MS Word)
- The structure between resources, for example a CV and a motivational letter together form an application

When talking about structured information, we mean information that resides in a database. The database structure (e.g. Entity Relationship Model) provides an overview over what type of information the database contains, and gives support in formulating a search query.

Semi-structured resources are structured according to organizational standard templates, such as product descriptions, proposals or invoices.

Unstructured resources lack a homogeneous organizational and internal structure. They are structured according to the individual judgment of their creator, so that each resource of the same type is still structured differently, such as emails, presentations or reports.

Semi-structured and unstructured resources are commonly stored in networked repositories, such as servers, databases or public directories of personal desktops.

Creating document templates is a way to improve the searchability of unstructured resources. Much relevant information can be gained about a resource if it is clear which parts of the document denote for example the title, the abstract and the author. Terms that come from these parts of a resource can be extracted and weighted more strongly in the index. The index is explained in the following section.

3.1.2 Search engine and index

As mentioned earlier, a simple search engine just searches for the string of characters of the query in the resources. It does not take the meaning of the search query into account, so the result set can easily be off target. Sophisticated search engines can compensate the lack of domain knowledge with various techniques.

Index An index is a table of terms that point to the resources that contain these terms. A search engine makes an index automatically by crawling through all the resources in the repository. This is done periodically so that the index is up to date. It increases the speed of searching, because instead of having to parse the complete set of resources with each query, the engine only parses the index.

Stoplist A stoplist is a list of terms that appear in nearly every resource. These terms do not serve to distinguish resources from each other. The stoplist prevents the search engine from indexing the terms it contains. In this way the index is reduced in size and the search process is accelerated.

Stemming Terms can exist in various grammatical forms, e.g. infinitive (*swim*), gerund (*swimming*), noun (*swimmer*) etc. To include resources in a result that contain the same term as is used in the search query, albeit in a different grammatical form, all index terms are reduced to their root. This is called stemming. The same is done with the query terms. This way, there are more matches between search query and resources.

Relevance A term that appears only once or twice in a resource is probably not relevant for the subject of the resource. Still, a query for that term will return every resource to which the term points in the index. The index is improved if it recognizes which terms are really discerning for a resource and which are not. An example of an algorithm that calculates the relevance of a term for describing the content of the resource is Claude Shannon's Information Theory. It measures the local term frequency and the global term frequency. An upper and lower bound are defined. The terms that appear less frequently than the lower bound are not considered relevant for the content of the resource, and the terms that appear more frequently than the upper bound do not distinguish one resource from another. Only the terms that are within the upper and lower bounds are used to index the resource with. This way the terms that are indexed hone in on the real subject of a resource.

Proximity Related concepts are discovered by calculating the proximity of terms. If certain terms appear at the same distance from each other disproportionately often, they are considered to be related to each other, and a search query for either term will also return resources containing the other term. A problem with this approach is determining the ideal term distance for drawing meaningful conclusions about term relationships.

Query Matching The problem of finding related resources that do not necessarily contain the terms in the search query is further countered by query matching. Query matching is based on techniques such as Boolean Logic, Vector Space Algorithms and Probabilistic Algorithms. We have not gone deeply into this

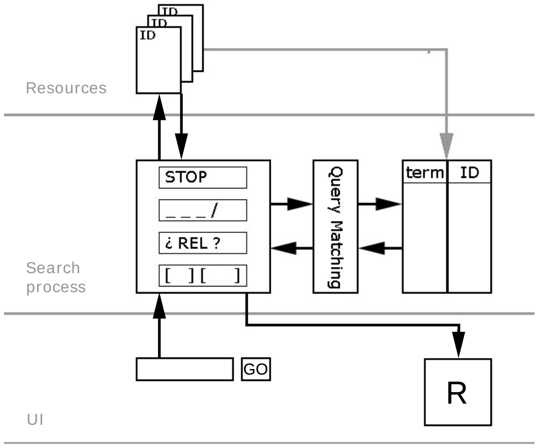


Figure 3.2: Sophisticated techniques for compensating the lack of meaning in standard search engines

topic, but more information can be found in the books by Baeza-Yates et al. [3] and Korfhage [30].

The roles of these techniques in the search process are shown in figure 3.2 on page 24.

3.1.3 Result set

Generally the result set is a list of links to resources with a short description. These resources are ranked according to a criterion, e.g. number of times downloaded. This does not mean however that the resource that was downloaded most is the one that answers the query best. It is difficult to find a good ranking criterion. The page ranking mechanism used by Google on the World Wide Web [39] is to base relevance of a webpage on the number of links to it - the more links pointing to a webpage, the higher its relevance. This is not suitable however in a corporate environment dealing with documents.

Common problems of a result are that it is too big or too small. A result set that is too big can be automatically clustered to structure the result set and provide an overview. An example of a search engine that automatically clusters its result set in hierarchical groups is the meta search engine clusty.com[1] developed by Vivisimo. The difficulty in automatic clustering is finding meaningful headings for clusters. For example, a search for "salsa" in Clusty results in a tree structure with 23 top nodes. Only the first 10 nodes are shown for better overview. These are *Dance*, *Recipes*, *Sauce*, *Order/ style*, *Salsa magazine*, *Congress*, *Natural*, *Genre*, *Join Salsa* and *Album, photo*. In this case the cluster headings are sufficiently informative.

Another approach to deal with a very large initial result set is to reduce it before it is shown to the user. This is called query contraction. Some methods to contract the result list are:

[1] http://clusty.com/

- invoke a rule to restrict the result set to resources where the query term(s) must appear both in the title and the body of the resource,
- if there are several query terms, require them to appear in resources at the same time by combining them with a Boolean “and”.

If the result set is empty or very small, the query can be expanded with similar methods:

- introduce a slop factor. This defines the number of moves that are allowed to change the search query. For example, with a slop factor of 2, the query “salsa dancing” could become “dancing salsa” and “salsa * dancing” and “salsa * * dancing”, but not “dancing * salsa” (to switch the order of 2 words requires 2 moves).
- if there are several query terms, include resources in the result set that contain only one of them with a Boolean “or”.

Further, the result set is an important instrument to learn more about the information need of the user. The search query generally does not give sufficient indication of what the user wants to know. The result set serves as a basis for feedback, in which the user can specify the query. Techniques for doing this are

- suggestions of alternative query terms based on a spelling checker (e.g. Google) or on terms that are syntactically close to the query term, with only 1 or 2 characters difference (e.g. Verity),
- allowing the user to suggest a resource that would answer the query well, which can be used as suggestions to other users,
- adding an option such as “search for related resources” or “more of the same” behind each individual resource of the result set, so that the query can be specified with more keywords taken from that particular resource.

3.1.4 User interface

Usability and Human Computer Interaction research has shown that there are different types of users with different information needs, different search strategies and therefore different requirements of a search interface [4], [24]. Users can be distinguished by the first 2 user model characteristics defined by Peter Brusilovsky [7]; the user’s *goal* and *knowledge* (we do not consider the *background* and *preferences*). The user’s search strategy is influenced by the goal, e.g. *what* is the information need, and the knowledge. The knowledge can be divided into procedural knowledge, e.g. *how* search literate is the user, and declarative knowledge, e.g. *how* well does the user know the domain. Search literacy includes knowledge of *where* to look for a particular resource and *how* to formulate a search query [31].

Many interface dialogue styles have been developed. They can be roughly categorized as either *search* or *browse* [31]. Research has shown that free text based search is the preferred strategy for users who have a precisely defined information need, whereas browsing an information structure, or a combination of search and browse, is preferred by those who have a less constrained information need [13]. Browsing through a domain requires some kind of structure. This type of search is described in section 3.2.4 on page 31.

The common interface for standard search is a text input field, such as the interface of the Google search engine. It allows users to use their own vocabulary to search with, and to specify the query with query language in the advanced search interface. This dialogue style is therefore well suited for users who know the domain well in which they search, and who are familiar with query language. It is less suited for users whose vocabulary differs from that of the vocabulary used in the resources of the domain. If they do not get results, they can not know whether it is because of a terminology mismatch, or whether the information they seek is really not there.

3.1.5 Summary

The main aim of the techniques described above is to simulate knowledge of the meaning of words by calculating probabilities based on statistical analysis. They are effective to a certain degree. However, there are a number of conditions which the resources must fulfill for this to work, as described in the introduction of this chapter. In all other cases, these techniques are not sufficient to attain good search results.

Further drawbacks are that they are not able to reconcile people with distinct conceptual models. This problem typically occurs when the information producers are different from the information consumers, for example domain experts and customers.

Finally, the standard search only caters to the search requirements of users with a well defined information need. All other user types are disregarded.

Using domain models to enhance the standard search can solve these problems. How this is done is explained in the following section.

3.2 Domain model enhanced search

In chapter 2 we have described the positive effect that domain models have on communication. A domain model leads to disambiguation because it makes assumptions explicit. It captures domain knowledge and provides an overview. In a domain model the meaning of concepts becomes clear through the context.

For these reasons domain models can play an important role in improving standard search. They can be applied differently in each part of the search process. How this can be done is described in the following.

3.2.1 Resources and metadata

Metadata are data about data. Metadata consist of fields and values. Fields describe various attributes of a resource. Examples of metadata fields are *topic*, *author*, *industry of relevance*. Which metadata fields should be used are determined by a demand analysis. Who will search for resources, and what are the questions they will ask? For example, do they regularly search for all documentation concerning a specific project? Then an important metadata field would be *project ID*. Or do they search for resources by their *authors*, their *reviewers* or their *target audience*? Then these 3 attributes would become metadata fields.

Metadata values are the values that belong to a metadata field. Metadata values can be controlled, which means that only those terms specified in a list are allowed to be used as values for a certain field. This leads to consistent use of the same term to denote a certain concept. If there are many metadata values for a metadata field, it makes sense to structure them. They are structured in relationships such as hierarchical, associational and equivalent. A structured controlled vocabulary is also called a thesaurus. Examples of such vocabularies are the Medical Subject Headings Thesaurus[2] (MeSH) and the Getty Thesaurus of Geographic Names[3] (TGN).

We have just established that structured metadata values represent a domain model. The domain of these values is the metadata field, e.g. in the examples above

[2] http://www.nlm.nih.gov/mesh/meshhome.html

[3] http://www.getty.edu/research/conducting_research/vocabularies/tgn/

the metadata field of a resource would be *topic* for values from the MeSH thesaurus, and *location* for values from the TGN thesaurus.

However, the metadata fields together can also form a domain model. This is best illustrated with the Dublin Core Metadata Initiative[4]. The main metadata fields (called *elements* in Dublin Core lingo) of the Dublin Core metadata standard are *creator*, *title*, *date*, *resource type*, *relation*, *subject* and *date*. For some of these metadata fields, there are standardized structured controlled vocabularies (called *encoding schemes* or *qualifiers* in Dublin Core lingo) such as the "Dublin Core Type vocabulary", which provides values for the metadata field *resource type* (e.g. collection, data set, event, image, interactive resource, text), or the MeSH thesaurus, which provides values for the metadata field *subject*. Another designation of the Dublin Core Elements is the *Dublin Core ontology*.

Both metadata values and metadata fields can constitute a domain model. These are models at differing levels of abstraction. Ontologies are suited for modeling the top level of a domain, i.e. the metadata fields, because of their expressivity and capacity to represent knowledge. They are comparable to conceptual database models, which also serve to model the elements and relationships of a domain on a high level. Taxonomies can be applied to hierarchically structure the values belonging to single elements of the ontology, i.e. the metadata values. They are monodimensional domain models (for explanation see page 7). The same holds for thesauri. In the case study of Statoil (page 65), a Norwegian oil company, a metadata model was developed that is comparable to an ontology, and the metadata values were structured as taxonomies.

This relationship between ontologies, taxonomies and thesauri leads to the conclusion that facets serve the same purpose as metadata fields and top level ontologies. They are a simple approach to capturing and separating the dimensions of a domain, so that modeling of the domain becomes more structured and simpler.

Adding metadata to resources is also called *tagging*, *annotating*, *coding* or *labeling*. In the case of ontologies it is sometimes called *populating* the ontology with instances. This reflects the view that the domain model is filled with resources. In a physical environment, a resource could only be attached to a single place in the domain model, that is, be annotated with a single metadata value. In a digital environment however, resources can be classified with several values, because regardless of where the resource is stored, it can be retrieved from several points. Tagging a resource with several values from a taxonomy is a way to implicitly model associative relationships [41].

Adding metadata to resources increases their searchability:

- Resources that are distributed over various repositories can be described homogeneously, allowing search across repositories via a single interface (compare case studies AON (page 50), McKinsey (page 55), Museo Suomi (page 59) and The Netherlands Institute for Sound and Vision (page 61)).

- Non-textual resources are described with text and can be searched by their metadata (compare case studies McKinsey (page 55) and The Netherlands Institute for Sound and Vision (page 61)).

- The same resource can be described with metadata from different conceptual models, so that the gap between different target groups is bridged (compare case studies AON (page 50), LexisNexis (page 53), the MinV&W (page 56) and WoltersKluwer UK (page 67)). Each domain modeling scheme has a way of dealing with diverging conceptual models of different target groups. Taxonomies require the development of one taxonomy for each target group, which are then mapped to each other. More about this use of taxonomies is found in section 4.1.1 on page 40. In a thesaurus, synonyms of terms can be defined. These

[4] http://dublincore.org/index.shtml

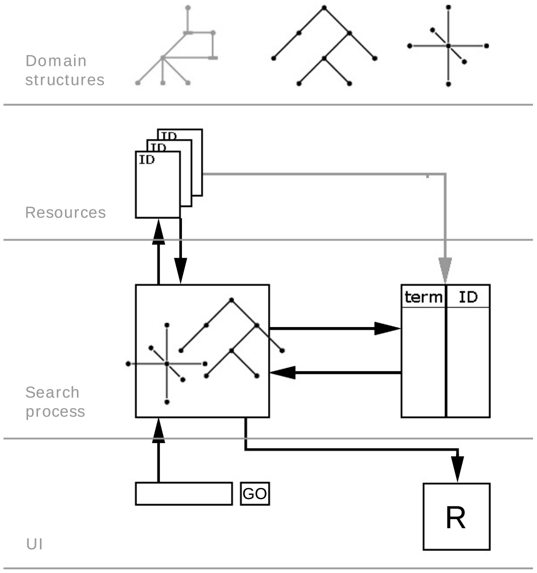


Figure 3.3: Transforming a taxonomy or a thesaurus to search engine rules so that the domain model knowledge is used in processing the search query

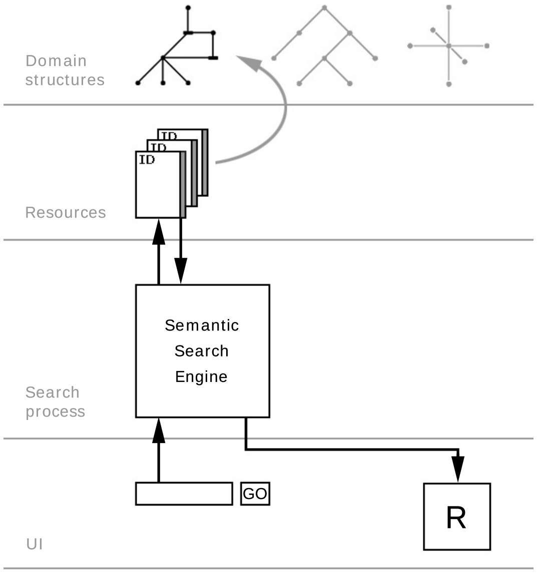


Figure 3.4: Using a semantic search engine to make inferences about the meaning of terms in the search query and in resources based on knowledge captured in the ontology

can capture the vocabulary of different target groups and map them to each other. The ontology modeling language OWL provides a construct to model the equivalence of classes ("equivalentClass").

3.2.2 Search engine and index

A domain model explicitly defines the relationships between terms that the search engine techniques described in section 3.1.2 on page 23 try to approximate statistically. The common representation format of taxonomies and thesauri is not readable by a search engine. For the search engine to make use of the domain model knowledge, each relationship in a taxonomy or thesaurus is required to be translated to a rule in a machine readable format. This is illustrated in figure 3.3.

Some search engine vendors provide the functionality to convert thesauri and taxonomies to a machine readable format. For example, Intellisophic[5] provides taxonomies in the format of the ISO standard XTM Topic Maps [40], and Verity[6] converts thesauri to a proprietary format called Topic Sets. Due to the thesaurus standard, the conversion process can be automatic, but only to a certain extent, as for example composite thesaurus terms require manual intervention. The conversion of taxonomies is even more difficult, since there is no standardized way of modeling or representing them. This results in a maintenance issue, as there will be 2 versions to maintain.

The main difference for searching between taxonomies and thesauri on the one hand, and ontologies on the other hand becomes apparent in this section. Whereas the rules that are programmed into search engines are machine readable, in the sense that they can be processed by the search engine, the engine does not know what it is processing. The content is just an unintelligible string of bits. In contrast, ontologies in the context of Semantic Web technology *define* the concepts in a machine readable way.

[5] http://www.intellisophic.com/

[6] http://www.verity.com/

This means that a semantic search engine and software agents can use this knowledge to include resources in the search result that would not have been retrieved with standard IR techniques. They can *infer* related information based on the knowledge contained in the ontology.

For this to work however the whole Semantic Web infrastructure must be in place. This requires that

- all resources are annotated with metadata, described in RDF and refer to the ontology that defines them,
- all domain models are described in a machine readable format, and
- that search engines are able to process semantics.

This is shown in figure 3.4 on page 28.

An example of an application framework for semantic search is TAP [22], which gives an impression of how a semantic search engine works. The 3 steps that a semantic search engine has to perform are (1) determining which concept is meant with the search query terms, (2) deciding which resources best answer the query and (3) how to present them to the user. One of the difficulties in an open environment like the web, where anybody can contribute data, is that there is no rule defining which metadata are used to describe a resource. In TAP the minimal metadata required to annotate a resource are the type of the resource (*rdf:type*, e.g. Person, Image, Organization) and the common denotation (*rdfs:label*). The ontology used in TAP is the TAP Knowledge Base that describes a range of domains, such as people, organizations, places and products.

The term of the search query is compared to all the available RDF metadata in the repository to find matches. If there is no match, then the semantic technology cannot contribute anything and the search will be a common standard text search. If there is 1 match, the associated type is considered to be the concept of the query. This is called the anchor node. If there are several matches, one of the concepts needs to be chosen to become the anchor node. This can be done on the basis of factors such as the popularity of the concept, knowledge about interests of the user from the user profile or the search context, or they can be offered to the user to make a choice. A query consisting of several query terms may result in several relevant concepts. In TAP the number of allowed relevant concepts has been reduced to 2 to control the complexity of the query and result set.

Next a choice has to be made which associated information of the anchor node to show. The choice can be made automatically by restricting the number of subgraphs from the anchor node to a predefined number, or manually with a predefined choice of properties that will be shown in relation to a specific class or object (e.g. for a person always show the properties homePageURL, hasImage, worksFor and livesIn, provided they are available).

Finally the retrieved information is formatted according to predefined templates that are associated to classes or objects.

As mentioned earlier, for this approach annotation is fundamental. Semantic web technology does not work without annotated resources. Annotation is not a hard requirement when dealing with thesauri or taxonomies. Instead the index can be created automatically in the same way as it is done in standard search. The domain knowledge from a thesaurus or taxonomy is translated to search engine rules and used to better interpret the search query, e.g. by also searching for synonyms of the search query terms. Synonyms are useful for capturing all the different ways in which people name a concept.

The advantage of annotating resources with the terms from the domain model is that the scores for recall and precision increase drastically in the context of the

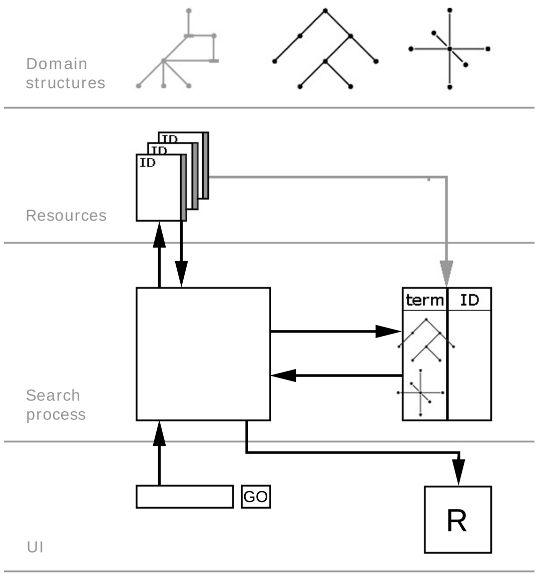


Figure 3.5: Using a taxonomy or a thesaurus to create an index has the same effect as adding metadata to resources

application, provided that the resources have been annotated well. If a certain term is searched for, it is guaranteed that all resources that are annotated with that term are retrieved.

Both approaches have advantages and disadvantages. On the one hand, search engine vendors offer the service of translating a domain model to a proprietary format which is only understood by that search engine. The solution is quick and simple, but the maintenance issue is increased because two versions of the domain model have to be updated each time there are changes, and the interoperability of this format is low. On the other hand is the Semantic Web technology, that requires precise, machine readable domain models, machine readable annotations of resources and a semantic search engine to be effective. This results in a high initial effort and total commitment to this approach. Benefits are improved search results, interoperability through open formats and inferencing capabilities.

term	ID

The domain model can also be applied in the index. Whereas a standard index only contains terms that are found in the resource, a domain model enhanced index uses terms from the domain model (see figure 3.5 on page 30). Resources are categorized according to indexing rules. Such an index conceptually is the same as adding metadata to a resource. At LexisNexis (see case study on page 53) the bulk of content is indexed in this way, only a small subset of high value content is annotated manually. They attain a precision score of 95% this way, which is good, considering that with the manual approach the precision score improves with just 2%.

On page 47 a collection of commercial search engine vendors gives an overview over their functionality. There are just two examples of commercial semantic search engines: Zoom by Semantic Knowledge and RetrievalWare by Convera. In the case of Museo Suomi (page 59) a generic view-based RDF search engine called Ontogator[7] was developed specifically for the project.

[7] http://www.cs.helsinki.fi/group/seco/museums/dist/

3.2.3 Result set

The result set of a standard search is commonly a list of links, as mentioned in section 3.1.3 on page 24. It can be manipulated with query *expansion* and *contraction* techniques. This works even better when a domain model is available to the search engine. A very small result set can be increased by expanding the query with synonyms, related terms or broader terms. This leads to higher recall. If the result set is very large it requires specification. The narrower terms of the term(s) in the search query can be suggested to the user to get feedback on the real information need.

Alternatively, a large result set can be structured according to the underlying domain model. Only the parts of the model that are represented by the resources in the result set are displayed in connection with each other. Irrelevant parts of the model are pruned out, only those nodes that are supposedly of interest to the user remain. This approach is called Dynamic Taxonomy [41]. It requires that the resources are annotated with terms from the domain model.

Ontologies can also be used to visualize the search results in much clearer ways than the common list of links. A technique developed by Frank van Harmelen and Christiaan Fluit [10] is called *Cluster Maps*. It shows the result set in overview, displaying the domain structure, the resources belonging to each element of the domain structure clustered in groups, overlapping groups of clusters (resources that belong to several classes) and the number of resources in each cluster. This visualization technique is employed in the commercial search engine Spectacle[8] from Aduna.

Another improvement to the result set achieved with ontologies is that of offering semantic recommendations. This was one of the main goals in the case study of Museo Suomi (page 59). The standard presentation of a resource of Museo Suomi is an image of the item and its metadata. In addition, relationships to other items from any of the 3 collections are shown. The advantage of using Semantic Web technology in this application is that even relationships between items that have not been explicitly defined with the metadata are revealed through inferences made on the basis of the underlying ontologies. This becomes clear through recommendations made about items that reside in different collections.

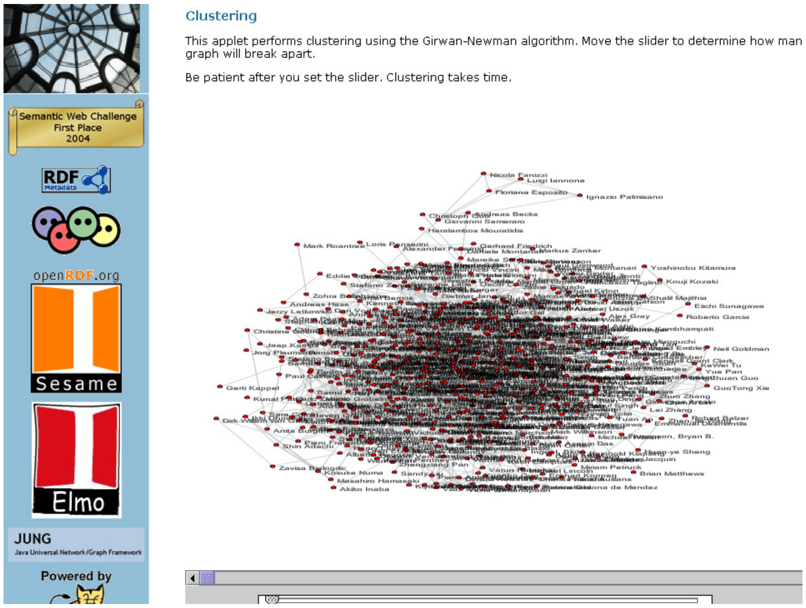
Generally, recall and precision are improved when using domain models for annotating resources and searching by the same terms. All resources that are annotated with a specific term are retrieved when that term is queried. A demand for high precision or recall may be a reason to enhance the standard search with domain models, even if the resources are purely text based and numerous. For example, a total recall score may be required in a judicial environment, where attorneys who look up precedence cases must not miss a single one.

In the case study at Egon Zehnder International (page 51) this was one of the main reasons for making a taxonomy. Egon Zehnder International is an enterprise that specializes in assessing and recruiting high profile candidates for top positions. The people data is their core business knowledge. It is contained in semi-structured files. In this domain it is important to retrieve all candidates that match a certain profile. For this reason a taxonomy was developed. Resources are annotated with metadata from the taxonomy.

3.2.4 User interface

The main drawback of the text field user interface of standard search as described in section 3.1.4 on page 25 is the potential terminology mismatch between the search query entered by the user and the terms in the index, the resources or the metadata. An

[8] http://aduna.biz/products/spectacle/index.html



Clustering

This applet performs clustering using the Girwan-Newman algorithm. Move the slider to determine how man graph will break apart.

Be patient after you set the slider. Clustering takes time.

Figure 3.6: Visualization gone awry in the Flink case study (page 52)

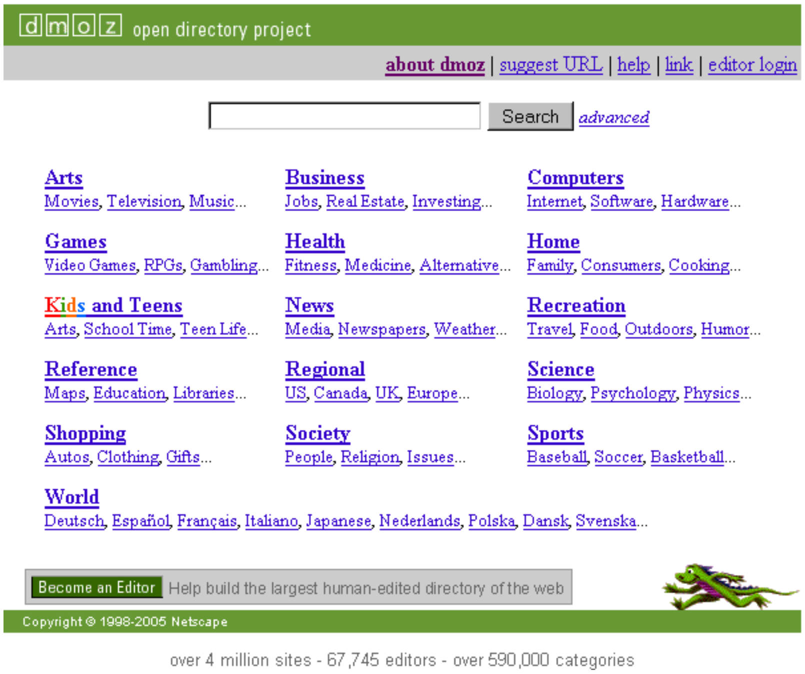
alternative dialogue style is *browsing* or *navigating* through the information space [31]. According to [31] this is suitable for users who

- have little domain knowledge,
- have little affinity with using query language and
- have a loosely constrained information need.

Domain models represent the information space, so they can be used to visualize it for the user. The main difficulty lies in visualizing a large space. Either it becomes unreadable from the amount of information pressed into the available space, or the sense of the complete space is lost by zooming in. The former is the case in the visualization of all the community members in the case study of Flink (page 52), as shown in figure 3.6 on page 32.

A common domain model for navigation is a taxonomy. The taxonomy is exposed so that users can navigate through its nodes via their relationships. Many search engines on the web offer this kind of interaction besides the regular text field. An example of this is the DMOZ Directory, of which the top level and the second level are shown in figure 3.7 on page 33. The bold terms represent the categories on the top level. The terms below them are the categories on the second level. The user can navigate through the tree by clicking on any of the categories, which opens up the next level. At each level, the user only sees the categories that are available one level lower. The path taken by the user is displayed at the top of the page as an orientation guide.

There are 2 options for dealing with metadata; resources are left untouched, or they are annotated with metadata. In the first case the domain model is just used for navigation, and the underlying search mechanism is the standard search. In the second case the domain model is used both for classification and for navigation. By annotating resources they are associated with nodes in the structure. Whenever the user reaches a node that has associated resources, links to them are displayed in the interface. In the case study of MinV&W (page 56) the taxonomy was used for navigation and for classification.



dmoz open directory project

about dmoz | suggest URL | help | link | editor login

Search advanced

Arts
Movies, Television, Music...

Business
Jobs, Real Estate, Investing...

Computers
Internet, Software, Hardware...

Games
Video Games, RPGs, Gambling...

Health
Fitness, Medicine, Alternative...

Home
Family, Consumers, Cooking...

Kids and Teens
Arts, School Time, Teen Life...

News
Media, Newspapers, Weather...

Recreation
Travel, Food, Outdoors, Humor...

Reference
Maps, Education, Libraries...

Regional
US, Canada, UK, Europe...

Science
Biology, Psychology, Physics...

Shopping
Autos, Clothing, Gifts...

Society
People, Religion, Issues...

Sports
Baseball, Soccer, Basketball...

World
Deutsch, Español, Français, Italiano, Japanese, Nederlands, Polska, Dansk, Svenska...

Become an Editor Help build the largest human-edited directory of the web

Copyright © 1998-2005 Netscape

over 4 million sites - 67,745 editors - over 590,000 categories

Figure 3.7: Example of a tree structure interface from the DMOZ website

In either case the user implicitly formulates a search query while browsing the structure. This relieves the user from the burden of using query language to make the search query.

Disadvantages of this approach are

- The domain is usually too big to give an overview. The user has no feeling for the depth or width of the information space.
- The taxonomy is pre-coordinative, so that the structure of the domain is completely defined. Only 1 dimension can be modeled in a taxonomy, as explained in section 2.1.2 on page 7 about the modeling disadvantages of taxonomies. The choice of dimension from which to model is ad hoc. Other dimensions are modeled as repeating groups in the structure. This makes it difficult for a user to guess where to search, especially if the chosen dimension does not match the user's conceptual model. In the MinV&W case study (page 56) this problem was encountered. The dimension used as the top structure was the organizational top level division of the Ministry (policy, execution, inspection), and the dimension that was repeated in many places in the taxonomy was about the business areas (e.g. roads, waterways, rails, air). An additional problem may arise from repeating groups due to poor technology; if the search engine does not take the whole path into account, but just the last term, then the result set for "roads" in the context of "inspection" will also return all other resources concerning "roads", e.g. "policies" and "execution" instructions concerning "roads".
- The user can only move up and down in the model, following hierarchical relationships, not across. If there are associational relationships that allow the user to jump from one path to the other, then the orientation within the model becomes difficult.
- There are only as many paths leading to a resource as there are metadata attached to it. This can be as little as 1 or 2 in a taxonomy, because commonly

resources are assigned a single place in a taxonomy.

- A domain model used for navigation should adhere to general usability guidelines. These guidelines are described in the following table. They are based on heuristics of what the mind can process and memorize during a search [35] and rules of thumb learned from practitioners at the ARK taxonomy conference [19]. Often a domain model does not fit nicely into this structure, so the modeler has to choose between usability and accuracy.

Usability guidelines for a navigation taxonomy	
Breadth	The top nodes of a navigation taxonomy should be no more than 10 if necessary, preferably 7.
Depth	A navigation taxonomy should be no deeper than 5 levels.
Leaf nodes	A parent node in a navigation taxonomy should not have more than 30 child nodes.
Result set	The result set found in the leaf node of a navigation taxonomy should be no smaller than 5 and no larger than 100. If it is smaller than 5, consider deleting the node. If it is larger than 100, consider splitting it up into more specific nodes.

An alternative form of interaction is *faceted navigation*. This is currently very popular in commercial search engines (e.g. Endeca, Verity, Aduna, FredHopper). Faceted navigation relies on a domain model that has a top-level ontology which divides the domain into different groups, also called facets, dimensions or views. The relationship between facets, ontologies and metadata fields is explained in detail in section 3.2.1 on page 27.

A facet is a metadata field. Each facet is composed of values that can be used to describe an aspect of the resources. Facet navigation relies on annotated resources. Not every facet has to be used to describe each resource. In faceted navigation the facets are exposed to the user, allowing the user to combine values from any of these facets to construct a query. The case study Museo Suomi (page 59) uses faceted navigation in the interface. Their interface builds on the approach developed by Marti Hearst et al. [24], which is demonstrated with the website of the Flamenco Fine Arts Search[9].

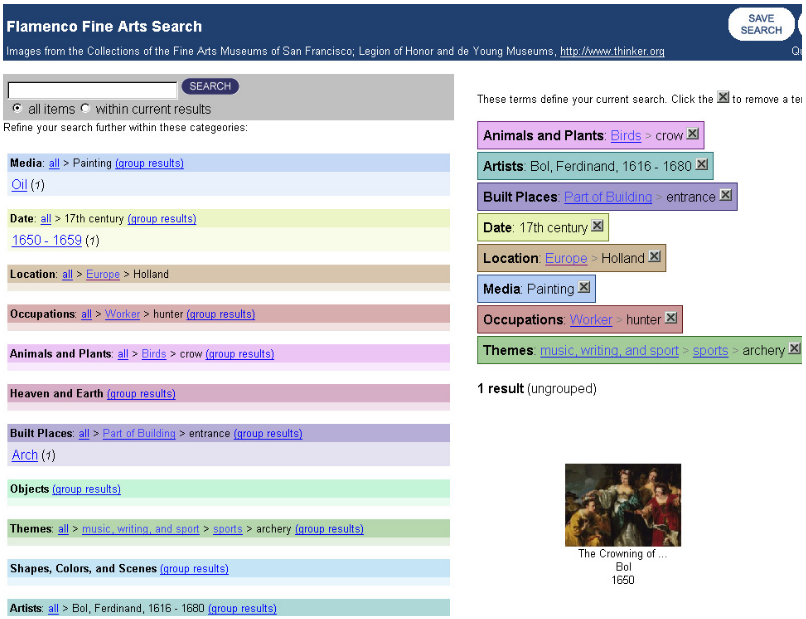
Figure 3.8 on page 35 illustrates the result of a search with faceted navigation. The bold terms in the bounding boxes on the right-hand side of the interface are the facets that were selected in this search. The terms behind them are the selected values. The values within the facets are structured hierarchically. This can be seen for example in the last facet "Themes", where the selected value "archery" is on the third level of the hierarchy, preceded by "music, writing and sport" and "sports". Other examples of websites where facet navigation is implemented are

- www.belvilla.nl, a vacation accommodation guide that allows narrowing down the eligible accommodations by combining facets such as location, price, time of the year, number of people, swimming pool availability, pet friendliness, etc.
- www.funda.nl, a real estate website with facets such as type of house, age of house, suburb, vicinity of, size, etc

Usability studies with this search approach have shown that users who do not know exactly what they are looking for prefer faceted navigation to a text input field and a tree structure navigation [27].

The basic assumption underlying the usage of domain models in the interface for facilitating the access to information is that the meaning of the terms is clear to the

[9] http://orange.sims.berkeley.edu/cgi-bin/flamenco/arts/Flamenco



Flamenco Fine Arts Search

Images from the Collections of the Fine Arts Museums of San Francisco; Legion of Honor and de Young Museums, http://www.thinker.org

SAVE SEARCH

SEARCH

● all items ○ within current results

Refine your search further within these categeories:

Media: all > Painting (group results)
Oil (1)

Date: all > 17th century (group results)
1650 - 1659 (1)

Location: all > Europe > Holland

Occupations: all > Worker > hunter (group results)

Animals and Plants: all > Birds > crow (group results)

Heaven and Earth (group results)

Built Places: all > Part of Building > entrance (group results)
Arch (1)

Objects (group results)

Themes: all > music, writing, and sport > sports > archery (group results)

Shapes, Colors, and Scenes (group results)

Artists: all > Bol, Ferdinand, 1616 - 1680 (group results)

These terms define your current search. Click the ☒ to remove a te

Animals and Plants: Birds > crow ☒

Artists: Bol, Ferdinand, 1616 - 1680 ☒

Built Places: Part of Building > entrance ☒

Date: 17th century ☒

Location: Europe > Holland ☒

Media: Painting ☒

Occupations: Worker > hunter ☒

Themes: music, writing, and sport > sports > archery ☒

1 result (ungrouped)

The Crowning of ...
Bol
1650

Figure 3.8: Example of a facet based search interface from the Flamenco Fine Arts Search website

users. What happens however if the terminology used in the domain model is different from the vocabulary of the users? In this case users still have difficulty searching with this interface, even though they are not required to come up with the terminology themselves. This is a common dilemma. The solution is to create a separate domain model for each target group, reflecting the terminology and conceptual model of the target group. These domain models are mapped to a fundamental domain model. The maintenance effort increases, but searching becomes much easier for a diverse user group. This approach is taken in the case studies WoltersKluwer UK (page 67), AON (page 50) and LexisNexis (page 53).

Generally, it is recommended to always provide a search interface, e.g. a text field, regardless of whether there is an option to browse the information space. On the one hand because it is a popular, well known interaction style that empowers the user. On the other hand, search logs are an important source for learning the vocabulary of the target group, so the text field is an instrument for feedback and improvement of the domain model.

3.2.5 Drawbacks

Enhancing search with domain models solves many problems as described above, but it creates new problems too.

Chapter 2 has shown that modeling a domain requires much effort, no matter which modeling scheme is chosen. The real trouble begins however when the model is put to use. Unforeseen application requirements and changes in the domain demand a constant maintenance effort, which can lead to a decrease in the quality of the model [44]. A well-defined maintenance procedure is required to control the quality of the model.

Adding metadata to resources also requires a great effort. First of all people have to take the time to describe resources with metadata. If people have to add metadata to resources without understanding the need or wanting to at all, they can be reluctant to do it properly. Experiences at WoltersKluwer UK (page 67) were that employees simply copy-pasted metadata from one resource to the other, so that the whole point of capturing the distinct character of a resource was missed. At Statoil (page 65) this is dealt with by shielding employees from adding metadata as much as possible. Most metadata are added automatically. The few that are required to be done manually are suggested by the system, which people just need to adapt or confirm.

Actions to improve manual annotation are (1) *training* to make people understand the importance of accurate metadata, (2) assigning *responsibility* for content and its metadata and (3) creating *feedback loops* in the process of sharing information. For example, statistics that show how often a resource was downloaded can be sent regularly to the creator, as well as a rating assigned to its relevance in the context in which it was retrieved.

As an alternative to manual annotation, resources can be classified automatically. If there is a structured, controlled vocabulary, metadata can be added automatically based on this model with an Automatic Classification tool. There are several approaches of how this works. One approach is to define a category and a set of documents that belong to this category. The tool extracts relevant terms from this set and adds weights to them. These terms constitute the attributes of documents that belong to this category. Subsequently all documents in the repository that match this term profile are added to the category. A human with domain knowledge is required to evaluate whether the documents were classified correctly after the first classification round. Normally this is an iterative process of several rounds. After several rounds the system is trained and can be expected to index resources at a consistent level.

A second approach is to define a category and a set of terms that characterize resources belonging to this category. The tool will look for all resources that contain these terms and classify them in the defined category. Finally, a tool can make clusters of resources without any input and assign metadata to them.

Most tools are black-box tools, they only allow a person to add or remove documents from a category. Some tools allow adjustment of the terms and their weights. Uncertain classifications also require the judgment of a human. At LexisNexis (page 53) the bulk of the content is classified automatically. Only a small subset of high value content is classified manually. For automatic classification, indexing rules are defined based on the taxonomy. The scope notes are an important support for the indexer who builds the rules. When classifying automatically, a fundamental decision must be made whether to focus on recall or precision. At LexisNexis, the focus is on precision, of which they achieve 95% with automatic indexing, and 97% with manual indexing, which is an extremely high rate.

Automatic classification is only an option if there are many text-based resources with large amounts of text. Some commercial tools that can do this are Verity Intelligent Classifier, Inxight Discovery, Convera, Autonomy and Tropes (see table on page 47).

Another issue in adding metadata manually is consistency. The same person may classify the same resource differently from one day to the next. This discrepancy is even more pronounced between different people, as described by Terry Butler et al. [8].

Once metadata are added there is also a maintenance problem. Over the years, meanings of words change (terminology drift). If the meaning of terms in the domain model is redefined, or if terms are deleted or changed, these updates have also to be carried out in all resources that carry the affected terms. This is a great problem at the Netherlands Institute for Sound and Vision (page 61), because the metadata of all legacy resources have to be adapted to the new thesaurus. An example of terminology drift is the music category R&B. Its original meaning is Rhythm and Blues, a musical marketing term that was introduced in the late 40's. The music style became immensely popular in the following decades, and slowly lost momentum in the 60's when it was replaced by soul music. Modern R&B defines a different style of music. Only the abbreviation is used to denote this music, not the full expression. This could lead to a search result that scores low on precision if both old and new songs were annotated with the expression R&B.

3.3 Summary of conclusions

Standard search techniques are based mainly on statistical analyses of text. They can provide good search results if the resources fulfill a number of requirements. These are:

- the resources consist of computer readable text;
- each resource consists of at least a page of text;
- there is a large number of resources (1.000 is not very many, 10.000 is better, 1.000.000 is ideal);
- the domain vocabulary is heterogeneous (e.g. news articles are heterogeneous, medical article are homogeneous).

In other cases, domain models can be used to enhance standard search and attain the required quality of search results. Domain models provide a structured controlled vocabulary with which to annotate resources. The structure of domain models and metadata added to resources increases their searchability:

- Resources that are distributed over various repositories can be described homogeneously, allowing search across repositories via a single interface.
- Non-textual resources are described with text and can be searched by their metadata.
- Domain models can facilitate access to information for users who have little procedural search knowledge, little domain knowledge and a loosely-constrained information need. The structure of the model is used to enable browsing through the information space.
- The annotated resources of the result set can be structured according to the domain model to provide a better overview.
- The same resource can be described with metadata from different conceptual models, so that the gap between different target groups is bridged.
- The recall and precision of search results are improved.

However, besides benefits domain models also bring many drawbacks. Creating a domain model requires a large effort. Committing to use a domain model is a big step, because it requires constant maintenance and quality control. The same applies to adding metadata to resources. Adding metadata requires discipline from people to do it conscientiously and consistently. It can also be done automatically with Automatic Classification tools. Maintenance of metadata becomes a problem if the domain model changes, or the meaning of the terms changes with time (terminology drift).

An alternative to adding metadata is indexing. Indexing with the help of a domain model is conceptually a form of adding metadata, although they are not added to the resource physically. The index is created automatically, and a new index can be made each time there are changes to the model. However, precision and recall of search results are generally lower than with manual annotation.

The decision to use domain models has to be a well-balanced one. Standard search techniques and domain models are complementary, as they can often be used together.

- Besides searching using metadata, a standard automatic index is useful as backup.

- An interface that allows browsing should also provide a text field for free text search. Users appreciate it as a last resort, and it supplies useful feedback about the terminology of the target group.
- A domain model that is just used for navigation still requires standard search techniques in the background to process the search query.

The 3 domain modeling schemes taxonomy, thesaurus and ontology have different areas for which they are especially suited. In the following chapter we will describe the application for enhancing search for each model individually.

Chapter 4

Suitability match

We present our conclusions about the best way to apply each of the 3 schemes taxonomy, thesaurus and ontology for improving search.

The application requirements determine whether a domain model solution is necessary. The most important requirement is the problem that needs to be solved. Problems that are typically solved with domain models have been listed in the conclusions of the previous chapter.

If the need for a domain model has been established, the following factors need to be analyzed to be able to choose which domain model or combination of models is best suited.

Demand analysis - Who are the people that will search for information? Which different target groups can be distinguished? What are their questions when searching? By what characteristics do they typically locate a resource? What is their vocabulary?

Supply analysis - What are the characteristics of the resources? Important characteristics are their types (pdf, word, ppt), volume, languages, modality (e.g. text, graphic, audio), quantity, quality (distinctions made with quality indicators such as those used at McKinsey, page 55) and life expectancy. How are the resources organized? Examples are databases, legacy systems, content management systems, document management systems and digital asset management systems.

Effort - What is the effort that is required for the problem solution in relation to the expected benefits? The effort is seen as the time invested in the development and maintenance of the domain model for the search application. The main expected benefit is a more efficient search process, which may lead to a reduction in time spent searching and more satisfied and productive employees. These indicators are difficult to measure. This is generally one of the main problems of justifying such a solution.

For each domain modeling scheme, we describe what it is suitable for, what it is not suitable for and what the effort and benefits are of applying it. The section structure reflects the organization of the thesis. We summarize the suitability of each domain modeling scheme for modeling and application in a search solution. In particular, we look at the suitability in individual parts of the search process; classifying resources with metadata, using the domain model in the search engine, visualizing the result set or enabling a browse or navigate dialogue style in the user interface.

4.1 Taxonomy

4.1.1 Suitable

Due to their hierarchical structure and precoordination, taxonomies are well suited for navigation in the graphical user interface. We describe 3 general applications of a navigation taxonomy:

1. A taxonomy per target group for navigating the interface
2. A taxonomy for navigating the interface and for classification of resources
3. Several taxonomies; at least 1 for navigation and 1 for classification

User interface

A taxonomy that is used just for navigating the interface must adhere to usability guidelines. These are described in the table on page 34. This taxonomy reflects the conceptual model of the searchers. The advantage of a pure navigation taxonomy is that it is easy to adapt to changes, because nothing else is affected by changes to the structure. If several target groups are identified, a navigation taxonomy should be developed for each of these groups to reflect their conceptual models. In a corporate environment the login of an employee can be used to determine the appropriate interface.

A good way to learn what the deficiencies of the taxonomy are is to offer a text field search interface and to analyze the search logs. These provide statistical evidence of the typical vocabulary and queries of the users.

Resources and metadata

A taxonomy can be used to classify resources. This is the situation in the case study of the MinV&W on page 56.

A combined navigation and classification taxonomy leads to high recall and precision scores, because the resources in the result set are specifically linked to the term that was clicked. There are however some disadvantages connected to this approach. It is difficult to reconcile the shape of a navigation taxonomy to the shape of a classification taxonomy. A classification taxonomy needs to be precise and detailed. Domain experts use it to classify their resources. A navigation taxonomy, on the other hand, should adhere to usability guidelines. These influence the shape of the taxonomy in that they pose restrictions, such as a limit to the number of children to a parent node and to the number of levels in the taxonomy.

When the same taxonomy is used for navigation and classification, the number of resources influence the shape of the taxonomy. One of the usability guidelines for navigation taxonomies states that a result set should not contain more than 100 resources. If significantly more than 100 resources are allocated to a node, this node should be split up into more specific categories.

Further, the terms used in the navigation taxonomy may not be the most ideal and precise terms to classify resources with. Content creators are domain experts that have their own terminology, which is usually difficult to understand for users from a different domain. In a corporate environment, almost every department has a different terminology, e.g. R&D vs Sales vs Production vs Marketing. If these terms are used in the taxonomy, it will be very difficult to use for target groups that are not domain experts. On the other hand, domain experts have trouble classifying their resources with a vocabulary that is too colloquial, because it does not reflect their conceptual model. A decision must be made whom to favour with the taxonomy: the consumers or the producers. Whichever group is chosen, the other will have trouble using the taxonomy.

Using the same taxonomy for classification and navigation leads to conflicts between modeling goals and usability goals. A third possibility is to make one taxonomy for classification and as many as needed for navigation. The domain experts become an additional target group. This is the most expensive approach, as it requires the most creation and maintenance effort, but it is also very effective. It bridges the semantic gap that separates content creators from content consumers. This is done in the cases of AON (page 50), LexisNexis (page 53) and WoltersKluwer UK (page 67).

Result set

Taxonomies are also suited for visualizing the result set. An approach developed by Giovanni Sacco [41] is to determine the relevant parts of the taxonomy from the search query and display only these parts to the user in the result set. Sacco calls it Dynamic Taxonomy. The reduced taxonomy gives the user a good overview of the area of interest. This is achieved by showing the nodes containing resources that were specifically queried by the user, plus the nodes that are associated to these resources because they are also part of their metadata. It requires that resources are annotated with more than one term. In this way, associative relationships are derived from combinations of metadata without explicitly being defined.

4.1.2 Unsuitable

Modeling and application

A taxonomy provides very primitive modeling constructs, so a domain can only be modeled roughly. The only relationship provided in a taxonomy is the hierarchical relationship, and properties cannot be modeled at all. In practice, taxonomy tools provide additional modeling constructs borrowed from thesauri, such as scope notes and associative and equivalence relationships. Because there is no standard for modeling or for notation, each taxonomy tool has a proprietary format which is difficult to translate to others. Therefore taxonomies score low on interoperability.

Also, domains are often too complex to model with a single taxonomy. This results in problems such as repeating groups, because a domain cannot be pressed into a simple structure. A common approach to manage the complexity of a domain is to create a top level ontology of the domain, for example by defining facets, and to model each facet like a taxonomy. Facets are used in the case studies AON (page 50), Egon Zehnder International (page 51), Museo Suomi (page 59), Netherlands Institute of Sound and Vision (page 61), Statoil (page 65) and WoltersKluwer (page 67).

4.1.3 Effort and benefits

The following table gives an overview over the statistics of taxonomy development and maintenance from the case studies. The information is not complete for every case study (FTE = Full time employee, PTE = Part time employee).

Overview of statistics for taxonomy development and maintenance from case studies						
Case	*# of terms*	*# of levels*	*# of resources*	*team size*	*duration of development (months)*	*maintenance time*
AON (page 50)	900	1-4	5,000	16 FTE	9	15 h per month
LexisNexis (page 53)	500	2	2,000,000	3 FTE + domain experts	12	0.5 FTE
McKinsey (page 55)	2,000	4	22,000	-	24	-
MinV&W (page 56)	3,939	8	2,000	2 FTE + domain experts	4	-
PwC (page 63)	25 taxonomies, varying between 4 and 150 terms	2	no precise number, at least 150,000	8 PTE	9	1.5 PTE
Sainsbury's (page 64)	-	4	-	3 + domain experts	3	-
Statoil (page 65)	-	2	20 Terabyte	Proof of concept phase 30, later 70 (50 FTE, 20 PTE)	60	-
Wolters-Kluwer UK (page 67)	2,450	2	5,000	1 + 9 domain experts	3	1 PTE normally, more when a new product is developed

The effort of building a taxonomy varies, but it is comparatively small. A taxonomy can be built and applied quickly. The fastest case study performed at Sainsbury's was pushed through with very tight deadlines. The project at Statoil took longest because it was enormous, and it is not finished yet. They completely reorganized their content management, one of the incentives being a change in laws requiring an increase in traceability of information for business purposes (Sarbanes Oxley act 404[1]).

Drawbacks of taxonomies are that they are a subjective mirror of a specific time or place, they may be unstable and become obsolete quickly.

There are many tools that support the creation, application and maintenance of taxonomies in industry (see table on page 47).

[1] http://www.sarbanes-oxley.com

4.2 Thesaurus

4.2.1 Suitable

Modeling

The focus of structure in a thesaurus is on the local term structure, e.g. the relationships pertaining to one term. The global structure does not need to be coherent. In fact a thesaurus can also contain so called orphan terms, which have no relationship with other terms. For this reason a thesaurus can cope well with a very large number of terms. An example of this is the thesaurus of the case study at the Netherlands Institute of Sound and Vision (page 61), which captures a vocabulary dealing with the whole world, since it is used to describe radio and television broadcast items. In a thesaurus properties are modeled as associative relationships.

Resources and metadata

The classical purposes of a thesaurus are indexing [1] and classifying. A thesaurus is well suited for classifying resources because it is post-coordinative. Any combination of metadata can be made with a thesaurus, whereas the other models are more restrictive due to the fact that each term is embedded in the structure and defined by the larger context.

Search engine

A thesaurus is well suited to specify the search query typed into a text field by the user. The synonym relationship can compensate differences between the vocabularies of all target groups. All 3 relationships serve to improve the result set by providing context information, for example by expanding or contracting the query. This requires that the thesaurus is translated to rules which the search engine can understand. Because thesauri are built according to well defined standards the translation process can be done semi-automatically.

Application

Thesauri are combined with taxonomies in the case studies of AON (page 50) and WoltersKluwer UK (page 67). In these cases the role of the thesaurus is to manage the vocabulary, the role of the taxonomy is to structure the domain. In the DOPE Project [45] the thesaurus was used as a basis for accessing information in a large document repository with Semantic Web technology.

4.2.2 Unsuitable

Result set

A thesaurus is not suited very well for visualizing a domain because it is not necessarily a coherent global structure. Furthermore, the hierarchical relationship is not transitive.

4.2.3 Effort and benefits

Building thesauri is an established craft. There are well defined standards and methodologies for building thesauri, and many tools for their creation and maintenance. Also there are many existing thesauri that can be reused, found for example in the Taxonomywarehouse[2].

The standardization of thesauri is beneficial for their interoperability. Thesauri can be translated to other formats like RDFS or OWL [51].

4.3 Ontology

4.3.1 Suitable

According to qualitative research after the application of ontologies in industry [36], the main applications of ontologies in industrial settings are solutions for *data integration* and *semantic search*. Semantic search requires machine readable information, which allows software agents to make inferences about the information based on domain knowledge supplied by ontologies.

Modeling

In an ontology the meaning of terms is explicitly defined, whereas in a taxonomy or thesaurus it is presupposed that the meaning of the terms is clear to the users. Meaning is defined by specifying constraints of the concepts, which narrows down the possible ways of interpreting the meaning. This allows software agents to interpret the meaning of concepts.

Ontologies are well suited for modeling domain knowledge, provided that the ontology language is expressive enough. Generally ontology languages allow a domain to be described very specifically. Semantic Web technology further provides standards for representing ontologies in a machine readable format.

Resources and metadata

The Semantic Web extends the information that is searchable from digital resources to real world objects [22]. All resources, both digital files and real world objects, are referred to with unique ID's. Therefore the information that is encoded in RDF is mainly structured data. For example in the Museo Suomi case study (page 59) the information consists mainly of RDF triples and URI's that point to art objects. Another project shows how this technology is applied to search semantically in large document repositories of unstructured resources [45]. Ontologies can be used well for organizing both structured and unstructured resources.

All resources must be annotated in a machine readable format (RDF) in the Semantic Web. Every resource has to be defined as an instance of a concept in the associated ontology. This is also called "populating" the ontology.

Result set

Ontologies are suited for visualizing the information space and the result set [10]. Since all resources are annotated, none can get lost in the virtual space, and the type of each resource is explicitly defined.

Interface

Part of the Semantic Web idea is to be able to browse through an information space that is semantically connected. There are already some prototypes of Semantic Web

[2] http://www.taxonomywarehouse.com/

browsers such as Haystack[3] and mSpace [4]. mSpace has a browsing style that is comparable to facet navigation, it provides different dimensions by which the user can search for resources of a particular domain.

Application

Agreed upon standards in hardware, software and content mark-up languages were fundamental for the growth of the World Wide Web. Standards are also being developed for the Semantic Web by the World Wide Web Consortium[5]. If these are adopted by users, developers and producers, then machine readable ontologies will be highly interoperable.

4.3.2 Unsuitable

Ontologies are unsuitable for a quick solution. Building an ontology from scratch takes a lot of effort, more than for taxonomies or thesauri, because it is a much more precise model of a domain.

On the other hand, if the usage of ontologies as domain models in the context of the Semantic Web becomes more popular, there will be libraries of ontologies that can be reused. Reuse and sharing of existing domain models is a fundamental concept of ontologies, as can be read in the definition on page 13, and of the Semantic Web.

However, if indeed anybody can make an ontology and put it on the web, then the quality of these ontologies is probably not very high. Further an ontology may describe a domain from a slightly different perspective from that of the potential user. In this case the ontology has to be adapted, which also leads to much effort.

Finally, mapping ontologies to each other is a difficult task. So far this requires human knowledge and common sense to draw the right connections. A danger of mapping ontologies is that a similarity between 2 concepts is detected which is not really a similarity.

4.3.3 Effort and benefits

The expected benefits of using ontologies for semantic search are undeniable. The boring task of sifting through extensive search results is relegated to software agents that use the domain knowledge from the ontology to make intelligent decisions. This should lead to drastically improved values for recall and precision, combined with a reduction of time spent searching. Semantic recommendations are made in relation to the retrieved results as demonstrated in the case study of Museo Suomi (page 59). Further the result set can be visualized and structured by the types of the resources, providing a much better overview than the common list of links.

Ontologies are not intended to force a particular way of organizing knowledge onto people. On the contrary, people should be allowed to organize their knowledge in any way they feel is most logical. An ontology serves to mediate between the different ways of organizing knowledge, by providing a common ground to refer to.

The price is high effort for development of the ontology and annotation of the resources. The greatest impediments of a large-scale application of ontologies according to [36] are the development of ontologies, the extraction of knowledge and the mapping of ontologies. This has been recognized widely by the research community. There is a strong focus on reducing the effort to model ontologies, and to make the deployment of Semantic Web technology as simple as writing web pages in HTML, so that more

[3] http://simile.mit.edu/hayloft/index.html

[4] http://www.mspace.fm

[5] http://www.w3.org/2001/sw/

people will use it. An improvement in tool support and infrastructure is to be expected. However, modeling a domain in itself is and will always be difficult, as it requires deep understanding of a domain and strong analytical capabilities.

4.4 Conclusions

The results of this thesis are intended to support the information architect in designing a solution for improved search in a corporate environment. Specifically we have examined the type of search problems that require a domain model to enhance the search process.

There are several approaches to modeling a domain. We have considered 3 different types of domain modeling schemes; taxonomy, thesaurus and ontology. The intention is to support the information architect in making an informed choice between one or more of these schemes.

In our opinion the main criteria for this choice are the modeling characteristics of a scheme and the suitability for application in the search process. The second chapter is a discussion of modeling characteristics of each scheme, followed by a comparison between them. This should give an information architect an idea of which aspects of a domain can be modeled with each scheme. What is missing here is an indication of the effort required to model a domain with each scheme. There are too many factors that influence the amount of required effort, ranging from measurable factors like domain size and resource characteristics to cultural matters such as the willingness to share knowledge and the existence of a project champion in the team to keep the project running.

The third chapter shows what role domain models can play in each part of the search process. This gives an idea of the problems that domain models can solve. We have split the search process into individual parts to show that domain models can be applied very differently in the process.

The fourth chapter makes recommendations about the suitability of each individualdomain modeling scheme for improving search. Each scheme has particular characteristics that make it especially suitable for a domain or a search problem.

In the appendix each case study is described in detail. These descriptions are intended to serve as a benchmark. The current problem of the enterprise can be compared to those described to see which case study is most similar, which solution was chosen, which problems arose and how they were dealt with.

An important issue that we have not touched upon in this thesis is that of maintenance. The real problems of a domain model are revealed when it is applied in a search system and its deficits and wrong assumptions become clear. Adaptation and maintenance are always required. Unfortunately we have not been able to glean sufficient information about maintenance issues from our case studies to draw any meaningful conclusions.

Chapter 5

Appendix

5.1 Tools

The following table provides an overview of search applications that are currently on the market and the functionality they offer. It is not an exhaustive collection.

Overview of functionality of search applications							
Tool	*Provider*	*Build Model (editor)*	*Translation domain structure to search engine rules / classification rules*	*Automatic Classification (adding metadata)*	*Indexing*	*Clustering*	*Faceted Navigation Frontend*
AquaBrowser	MediaLab Solutions	x	x	x	x	x	-
Autonomy	Autonomy	-	-	x	x	x	-
Brain Enterprise Knowledge Platform	The Brain Technologies Corp.	x	x	-	x	-	-
Clearforest Semantic Tagger	Clearforest	x	x	x	x	-	-
Collexis	Collexis	x	x	x	x	-	x
dtSearch	dtSearch UK	-	-	-	x	-	-
FAST Enterprise Search Platform	FastSearch and Transfer	-	-	x	x	x	-
Fredhopper	Fredhopper	x	x	-	x	x	x
GridWalker	Gridline	x	x	-	x	-	-
Intelligent Classifier	Verity	x	x	x	-	x	-

Overview of functionality of search applications

Tool	*Provider*	*Build Model (editor)*	*Translation domain structure to search engine rules / classification rules*	*Automatic Classification (adding metadata)*	*Indexing*	*Clustering*	*Faceted Navigation Frontend*
Lucene	Open Source	-	-	-	x	-	-
Metadata-Server	Aduna	x	x	x	x	-	x
Ontopia Knowledge Suite	Ontopia	x	x	-	x	-	-
Parametric Search	Verity	-	-	-	x	-	x
ProFind	Endeca	x	x	x	x	-	x
Retrieval-Ware	Convera	x	x	x	x	x	-
Semio Knowledge Engineering Workbench	Entrieva	x	x	x	x	x	-
Semio- Tagger	Entrieva	-	-	x	x	-	-
Semio- Taxonomy	Entrieva	x	x	-	-	-	-
Smart- Discovery	Inxight	-	x	x	x	x	-
Spectacle	Aduna	-	-	-	x	-	-
StarTree Viewer	Inxight	x	-	-	-	-	-
Stratify Discovery	Stratify	-	-	x	-	-	-
TermChoir, ViewChoir and MetaChoir	WebChoir	x	x	x	x	-	-
TermTree	Active Classification Solutions	x	x	-	-	-	-
Thesaurus Master and Machine aided indexer	Data-harmony	x	x	x	x	-	-

Overview of functionality of search applications							
Tool	*Provider*	*Build Model (editor)*	*Translation domain structure to search engine rules / classification rules*	*Automatic Classification (adding metadata)*	*Indexing*	*Clustering*	*Faceted Navigation Frontend*
Tropes	Semantic Knowledge	-	-	x	-	-	-
Ultraseek	Verity	-	-	-	x	-	-
Verity K2	Verity	-	-	-	x	-	-
Zoom	Semantic Knowledge	-	-	x	-	x	-

5.2 Case Studies

The purpose of these case studies is to give an impression of how domain structures are used in practice. They are ordered alphabetically by name of the executing company or group. The icon in the margin shows what domain model(s) are used in the case study.

The descriptions are structured as follows.

Context The global context of the case is described; the domain of the enterprise, the number of employees and the number of offices worldwide.

Project dependencies The important characteristics of the project, which influence the choice of domain model, have been captured in the second table. These are the subject matter of the domain of the project, the goals of the project, the target groups and the types of resources that are dealt with.

Solution The third table describes the solution that was chosen. We indicate which domain models were used and what for, which search strategies are supported in the project, whether resources are tagged with metadata manually or automatically and whether facets were applied to model the domain.

Approach After the chosen solution, we describe the approach that was taken in the project. This includes a specification of the roles that were represented in the development team, which tools were used, how were legacy resources dealt with, what was re-used for modeling the domain, and how is maintenance organized.

Statistics In the last table we give an overview of relevant statistics. This is useful for getting a feeling for the dependencies between size, duration and human resources in these projects. The statistics we have gathered are the number of terms in the model, the number of levels (in the case of a taxonomy), the number of resources, the team size, the duration of development and the maintenance effort.

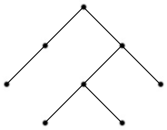
Assorted Finally, if available, we shortly describe encountered problems and solutions, benefits of the project, lessons learned and best practices.

We have not got the complete information for every case study.

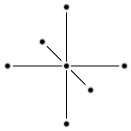
5.2.1 AON

Source: Presentation by Annie Wang at ARK Conference [19].

Context	
Enterprise domain	Risk management, insurance brokerage, reinsurance, human capital, management consulting, outsourcing
# of employees	48.000
# of offices	500 in +120 countries



Taxonomy



Thesaurus

Project dependencies	
Project domain	Business knowledge captured in shared knowledge management repositories
Goals	- Improve search across different repositories - Ease content management process - Integrate all the offices worldwide by providing multiple views of the same content
Target groups	Employees (125 countries) and clients
Resource modalities	Mainly text, structured (e.g. product data, people data, client data) and unstructured (e.g. sales presentation, research report, white paper, case study, sales sheet, lessons learned)

Solution	
Taxonomy used for	- Mediating between different conceptual models: a global taxonomy was created that contains every concept, and local views with adapted terminology and a selection of the relevant concepts were created for each business unit. - Providing organizational structure of resources - Providing content in context - Navigation
Thesaurus used for	Vocabulary
Ontology used for	/
Supported search strategies	- Free text search - Facet based/view based/criteria based search - Display of related items - Predefined queries beneath links, so that a
Metadata	added manually to content via a content submission form
Facets	8 facets in global taxonomy

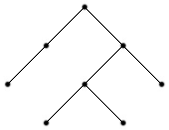
Approach	
Development team roles	Interviews with 16 business unit and country knowledge managers, brainstorm sessions with content managers, content submitters and site managers
Tools	TeamSite, Structured Content Filter Application
Legacy	/
Re-use	Knowledge Management repositories contain for example global insurance guide, phone directory and resource directory, libraries of case studies, solutions and intellectual capital, people finder, contact database
Maintenance	Maintenance of all taxonomies is handled from a central database. There is a change management process in place. It reserves the right to make certain changes for the global organization, and local taxonomists provide report of their changes. A global task force meets monthly.

Statistics	
# of terms	900
# of levels	Between 1 and 4
# of resources	5000, daily increase
Team size	16 FTE
Duration of development	9 months
Maintenance time	ca. 15 h per month per person

5.2.2 Egon Zehnder International

Source: Presentation by Ben Simmonite at ARK Conference [19].

Context	
Enterprise Domain	Assessing and recruiting executives
# of employees	300
# of offices	+59 in 38 countries



Taxonomy

Project dependencies	
Project domain	CV information - education, employment, recommendations
Goals	Be better able to find relevant information
Target groups	Employees of EZI
Resource modalities	Text based, semi - structured (CV type data, contacts and evaluations)

Solution	
Taxonomy used for	Standardized way to describe the employment history of candidates at a good level of granularity, unify world view of employees
Thesaurus used for	/
Ontology used for	/
Supported search strategies	- Free text search - Navigation
Annotation approach	Manual
Facets	Yes, faceted hierarchical classification

Approach	
Development team roles	External expert for quality assurance
Re-use	External thesauri
Maintenance	Based on regular user surveys

Statistics	
Duration of development	2 years

Assorted	
Problems	**Solutions**
Existing system does not integrate the taxonomy very well (no support for adding metadata, no sophisticated search function)	Use an open system architecture where possible
Taxonomy is devised by information specialists - non-specialists have trouble using it to add metadata to content because it does not reflect their world view	Involve all users in developing the taxonomy, not just experts
Benefits	
Structure is being used in more in-house applications than intended	

5.2.3 Flink

Sources: Website[1] and technical documentation by Peter Mika [34].

Ontology

Context	
Enterprise Domain	Semantic Web research
# of employees	/
# of offices	/

Project dependencies	
Project domain	Semantic Web community
Goals	- locate the persons with much knowledge in a specific area - provide a tool for Social Network Analysis - identify the roles of community members based on their connectivity with other members - automatically create an ontology of the semantic web domain
Target groups	Social network analysts, semantic web community members, researchers
Resource modalities	Text and images that are distributed all over the web and not initially intended for this application, such as - HTML information on websites - FOAF profiles - Emails - Bibliography items

[1] http://prauw.cs.vu.nl:8080/flink/

Solution	
Ontology used for	FOAF[2]: describing the actors in the community and their relationships
Supported search strategies	Different browsable views on a community (people relationships, subject relationships based on a shared interest in the same subjects by at least 40 people, geographic location of people)
Annotation approach	Mainly reuse of existing annotations (FOAF profiles), partly manual (bibliography items)

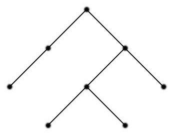
Approach	
Development team roles	/
Tools	- Web scutter for finding FOAF profiles - Web mining via google to calculate tie strength between network members based on how often their names co-occur on the web - Sesame database for storing all the RDF data - JUNG programming toolkit for visualization of social network
Legacy	/
Re-use	- HTML information on the web - FOAF profiles - Emails - Bibliography items
Maintenance	/

Statistics	
# of terms	Ca 60 terms in foaf
# of resources	Ca 550 people, 5152 publications, 8185 messages (21.06.2005)
# of levels	/
Team size	/
Duration of development	/
Maintenance time	/

5.2.4 LexisNexis

Source: Presentation by Mark Fea at ARK Conference [19].

Context	
Enterprise Domain	Legal branch of Reed Elsevier publisher
# of employees	13.300
# of offices	80



Taxonomy

Project dependencies	
Project domain	Common law for UK, Canada and Australia
Goals	Improved access to content
Target groups	News and Business customers, Legal customers worldwide
Resource modalities	Unstructured text documents

Solution	
Taxonomy used for	Integration of content - core common law taxonomy, mapped to local taxonomies
Thesaurus used for	/
Ontology used for	/
Supported search strategies	- Free text search: results in all nodes of taxonomy that are related to resources containing the search terms (dynamic taxonomy) - Navigation of hierarchy: user can expand/collapse nodes for better overview - Clustering: Result set is categorized for better overview and contextualization - Information push: customers can request automatic updates on content related to index terms
Annotation approach	Manual annotation for a small subset of high-value content Automatic/semi-automatic indexing for bulk
Facets	/

Approach	
Development team roles	- 4 full-time taxonomy specialists - 10 senior editorial staff and subject specialists identified from workshops
Tools	Example-based and rules-based software for automatic indexing
Maintenance	Maintenance of taxonomy limited, focus on indexing maintenance (adaptation of indexing rules to subtle changes in legislation, re-testing when there is a bulk of new content)

Statistics	
# of terms	currently 500, plans to expand
# of levels	2, expected to expand to 4
# of resources	2 million, expected yearly increase of 100.000
Team size	3 FTE's, plus occasional subject experts
Duration of development	1 year
Maintenance time	currently half 1 person's job, expected to increase with taxonomy

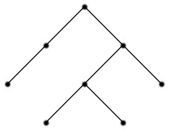
Assorted	
Problems	**Solutions**
Recall is hard to measure when dealing with a large and relatively unknown dataset (due to automatic indexing)	Make hitlists of documents known to be in the database to test the recall performance of the indexing

Benefits
- Improved Speed of retrieving the right information - Higher confidence that all the available information has been retrieved - Information discovery in new subject areas through context
Best Practices
- Put related terms in scope notes - Choose between focus on either precision or recall for programmatic indexing - here: precision (95% precise) - Precision can be measured by randomly sampling at least 100 documents from the result set - Do collaborative design, prototyping and usability testing for best design and usability

5.2.5 McKinsey

Source: Interview with Evert Jagerman, head of the Information and Research Group.

Context	
Enterprise domain	Consultancy
# of employees	ca. 11.500
# of offices	82 in 42 countries



Taxonomy

Project dependencies	
Project domain	Corporate intranet
Goals	- Improve transparency of sources (author, reviewer, quality, status) - Integrate sources from all repositories
Target groups	Employees
Resource modalities	Unstructured text and presentations containing many graphics

Solution	
Taxonomy used for	Structuring domain, classifying resources, navigation
Thesaurus used for	/
Ontology used for	/
Supported search strategies	- Free text search - Navigation
Annotation approach	Manual annotation of all resources, because they are mainly non textual (ppt) and because there are not enough resources to do it automatically
Facets	no

Approach	
Development team roles	/
Tools	Mindmanager, Verity
Legacy	Thesaurus
Re-use	Keyword lists of practices
Maintenance	/

Statistics	
# of terms	2000
# of levels	4
# of resources	22.000
Team size	/
Duration of development	2 years
Maintenance time	/

Assorted	
Problems	**Solutions**
Ever growing amount of resources, policy to preserve them all	Classify resources in categories, e.g. - Knowledge object - Reviewed - Useful with limitations - Archived
How to deal with candidate terms	Besides the metadata field "keywords" (obligatory) add a metadata field "free terms" to provide a space for terms that are not in the controlled vocabulary. These terms are reviewed by the maintenance commission, who decide whether to add them to the taxonomy or not.

Best Practices
- Make sure that the person responsible for maintenance has a strong personality and is able to resist pressure from colleagues to make changes to the structure too quickly. The maintenance person should be pragmatic but still abide by the rules of the taxonomy. Otherwise the quality of the structure is surrendered! - Be economical with associational relations - Allow concepts to appear in the taxonomy multiple times - Do not allow the label "other" for a node in the taxonomy - Do not let nodes that are empty appear in the interface (empty nodes are the result of a top-down approach. They indicate which resources should be there, not necessarily what is there. An empty node can reveal a knowledge gap) - Use the taxonomy just for domain specific subject matter, record all further vocabulary in lists.

5.2.6 Ministry of Transport, Public Works and Water Management (MinV&W)

Source: Interview with Peter Nieuwenhuizen, information specialist of the Bouwdienst.

Context	
Enterprise Domain	Dutch mobility policy (traffic via roads, waterways, railsways and by air) and protection against floods or falling water tables
# of employees	-
# of offices	-

Taxonomy

Project dependencies	
Project domain	Subject matter of the MinV&W, of which the top 3 tasks are policy making, execution of projects and inspection of the areas passenger transport, freight transport, civil aviation, water affairs, public works and water management and the Royal Netherlands Meteorological Institute (KNMI).
Goals	Develop an intranet for the employees of the MinV&W. The intranet should fulfill the following requirements: • Provide a single point of information for employees of the MinV&W • Make the information available digitally via the intranet, not just references to it • Make the information easy to find for all employees, so that the need for information specialists is reduced • Enable other employees than information specialists to add information to the repository (The previous system "catalog" called V&WLIS (Verkeer & Waterstaat Library Information System) contained only the metadata on the resources, not the resources itself. Hardly anyone besides the information specialists used V&WLIS; it was not trivial to search through without understanding the thesaurus that was used for classification, therefore information specialists were needed. When employees required information, they sent a request to the information specialists. These retrieved the resources and passed them on if they were available digitally, otherwise a notification was sent out to pick up the information.) • Plan for the intranet to be extended across all departments of the MinV&W, and eventually to become available to the public (Dutch law [WOB] requires the MinV&W to be transparent to the public)
Target groups	Researchers, policy makers, information specialists
Resource modalities	Unstructured text such as policy documents, technical reports, technical drawings, scientific reports and news articles (clippings - every day a report of all news clippings related to the MinV&W is made)

Approach	
Development team roles	1 taxonomy specialist 1 information specialist 1-2 subject specialists from each field
Tools	Mind Manager for building taxonomy
Legacy	Thesaurus, physical historical library

Statistics	
# of terms	2865 (+1074 synonyms)
# of levels	8
# of resources	ca. 2000
Team size	2 - 5
Duration of development	4 months
Duration integration	4 months
Date live	01.11.2004
Maintenance time	No information on maintenance yet

Planning

The approach was to start building a draft taxonomy at the level of policy making. The people involved in the beginning were

- 1 taxonomy specialist (facilitator)
- 1 information specialist, who held peer reviews with each new version of the draft taxonomy.

This draft taxonomy was presented to the heads of the 4 main policy areas. From these areas 1 to 2 communication employees improved their part of the taxonomy. After this the taxonomy of the policy level was finished.

The next step was to build taxonomies for the processes of execution and of inspection in accordance with the taxonomy at policy level. To get this started, there was a kick-off workshop to present the policy making taxonomy and to explain why and how to build a taxonomy. The purpose of this workshop was not just to teach and inform, but especially to convince. There was one information specialist from each unit present. They were asked to each make a proposal for a taxonomy of their area.

One unit was very fast in presenting a first proposal (early adopter), this encouraged the others to give it a try. The project got momentum, and one of the information specialists took on the role of coordinator. This was very important, because the level of the taxonomy got too specific for the facilitator to be able to make decisions about it. With the help of the coordinator the individual taxonomies were joined into one. Early adopters are key to the success of a project.

Techniques for developing the taxonomy

- Try to have an example of how the application will work, so that people can have a better idea of how to structure their domain. Take the example from a completely different domain!
- Consider not making the taxonomy reflect the complete repository, but only those concepts whose resources are being used frequently (this might require a mind-shift from the information specialists who are inclined to believe that the whole repository is vital). This is only possible if statistics of the usage of the repository are available.
- Print the taxonomy and hang it on the wall where everybody can see it. Ask all people who come by and are part of a target group to take a look and give feedback.
- Check taxonomy term against thesaurus and add existing synonyms (as a note) replace legacy thesaurus to avoid double maintenance.

- For navigation purposes 5 levels of depth are recommended (according to the usability guideline by Nielsen: no more than 3 clicks away, plus 2 click bonus for the motivated searcher).

  Problem what to do when the fifth level is reached and there are more than 100 documents attached to the node?

  Solution manually create overview documents that provide a short overview of all the information available in this node and point to specific documents. A virtual 6th level is created, yet the user has a feeling of having accomplished an important step in the search process. The added value is the breakdown of the bulk of resources into categories to serve as overview.

- Do not use abbreviations, question all terms as to their universal meaning outside of the context of the group/area

Anticipated conflicts

- When fitting together taxonomies made by separate groups there will be overlapping terms/concepts in the individual structures. Let coordinator decide where to place each term, but it is important that this is done in agreement with the authors of the individual taxonomies

- When there are several possibilities of placing a concept in the structure, ask how the end user searches. In this case, the information specialists knew how end users search by the questions they get daily. However, it was difficult to get them to think from that perspective, and not from their own background. This required a mind shift that took about 2 weeks.

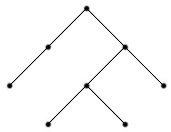
- If everyone can add resources to the repository it will result in a big mess and inconsistent metadata, rendering the whole system useless. It is important to have a quality control procedure, for example by information specialists.

- Authors will complain that the structure does not reflect their area adequately, that they cannot classify their resource in it. This can have two reasons. Either the author accredits too high importance to his/her document and demands a level of detail that is not compatible with the big picture, this requires mind-shift from the author. Or the structure reflects an ideal situation (e.g. there ought to be resources concerning concept xyz) instead of reality (e.g. there are in fact a lot of resources about concepts a and b, but hardly any on concepts c and d, and none on e). If this is the case, either the structure should be adapted or it can be used as a control measure to give guidance on where to focus next.

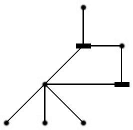
5.2.7 Museo Suomi

Source: Website Museo Suomi [26] and documentation [27], [28].

Context	
Enterprise Domain	Finnish cultural heritage
# of employees	/
# of offices	/



Taxonomy



Ontology

Project dependencies	
Project domain	Collection items from 3 different Finnish musea
Goals	- Collection interoperability - exposing global semantic associations between collection items of different collections - Allow browsing of all archives via a single interface
Target groups	Art professionals, researchers, the general public
Resource modalities	Digital images of the collection items, textual descriptions

Solution	
Taxonomy used for	Structuring the facets
Thesaurus used for	/
Ontology used for	Machine readable annotation of collection items for automatically inferring relationships between items from all three collections
Supported search strategies	- Free text search - Facet based search
Annotation approach	Manual initial annotation of information objects in proprietary format, semi-automatic translation to RDF
Facets	9 facets

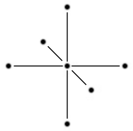
Approach	
Development team roles	/
Tools	- Semantic search engine “Ontogator” - Word based search engine for keyword search - Logic server “Ontodella” - Protege 2000 ontology editor (advantage: simple enough to be used by museum personnel) - “Terminator” used for creating term cards that define a mapping between words and expressions used at the XML level and the corresponding ontological concepts - “Annomobile” transforms XML to RDF - semi-automatic, requires human editor to make final decisions in some cases, these are identified by Annomobile
Legacy	/
Re-use	- Dublin Core Metadata - Finnish cultural thesaurus MASA
Maintenance	The material was not extended so no maintenance was needed.

Statistics	
# of terms	4721 in pilot version, based partly on larger cultural ontology with 6768 classes
# of levels	/
# of resources	176,602 triplets, 50,056 unique URIs or literals
Team size	2-3 persons, most of the actual ontology editing was done by a museum curator
Duration of development	The project started in the beginning of 2002 and the pilot was released in March 2004, but the project did many other things, too.
Maintenance time	/

5.2.8 Netherlands Institute for Sound and Vision

Source: Interview with Vincent Huis in 't Veld and Alma Wolthuis.

Context	
Enterprise Domain	Audiovisual public broadcast
# of employees	-
# of offices	1, the Media Park in Hilversum



Thesaurus

Project dependencies	
Project domain	All subjects of television and radio, the whole world
Goals	- Handle a single vocabulary for various audiovisual archives - Allow search in all archives via a single interface
Target groups	Broadcast professionals, researchers, information specialists
Resource modalities	High quality video and audio

Solution	
Taxonomy used for	/
Thesaurus used for	Vocabulary for adding metadata to resources
Ontology used for	/
Supported search strategies	- Free text search - Navigation
Annotation approach	manual
Facets	7 facets in thesaurus (called axis)

Approach	
Development team roles	1 representative from each partaking archiving institute, a full time information specialist from the Netherlands Institute for Sound and Vision, a taxonomy expert
Tools	MultiTES
Legacy	All old resources that have metadata from the previous thesauri. They are classified as B-documents and will be revised to become A-documents (metadata compliant with new thesaurus)
Re-use	Individual catalogues from each institute, merged into one thesaurus
Maintenance	There is a commission that convenes every fortnight to decide whether candidate terms proposed by documentalists will be added to the thesaurus. The decision is made by the head of thesaurus maintenance.

Development

The thesaurus was developed based on a merger of the existing controlled vocabularies and thesauri. Together these vocabularies consisted of ca. 12.000 terms that were reduced to about a third by eliminating double terms. Choices for preferred terms were based on an analysis of terms actually used in the existing catalogs. A guideline for defining a descriptor was its usage in the descriptions of a minimum of ca 20 resources. A further guideline was the expected searching behaviour of users. This could be derived from the customer service employees, who search for material requested by customers.

For the “subject term” axis, a structure of 16 top categories covering all subjects was developed. These categories were cross-checked by documentalists (= employees who actually write descriptions of audiovisual resources) that were very familiar with the specific category. As a rule of thumb, terms were allowed to be categorized in multiple categories, but no more than 3.

Considering that the audiovisual archives are characterized by a lack of clear domain borders, one might expect that existing domain specific thesauri would have been reused and integrated. This however has hardly been done. Domain specific thesauri have varying, usually quite high levels of detail. This would make the thesaurus unbalanced and much larger than necessary. However, existing thesauri were analyzed to get inspiration for a good structure.

Statistics	
# of terms	Subject term axis: ca 4.500, people’s names axis: ca 90.000
# of resources	700.000 hours in the archive, growing with ca. 40.000 hours each year
Team size	4-5
Duration of development	ca 5 years
Maintenance time	Not yet known

Problems and solutions

All the descriptions that were made with the old thesauri and controlled vocabularies are inconsistent with the new thesaurus. These are classified as B-documents. A-documents are those which were made on the basis of the GTAA. Once the thesaurus is more or less finished, the task of revising all the B-documents and turning them into A-documents needs to be tackled. This will be a matter of manually checking and adapting each document to the GTAA. Expected conflicts between the old and new thesaurus are

- new descriptors cover multiple old keywords
- old keywords are divided into multiple new descriptors
- old keywords that have become obsolete have had no replacement
- syntactically matching old keywords and new descriptors may have different semantics

Evaluation and extension

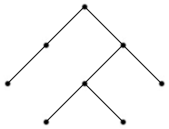
The GTAA fulfills its goal of unifying the catalogues of the audiovisual institutes that joined the initiative. A lot of effort has been put into the development of this thesaurus. Other institutes within the media park have already expressed interest in sharing in the use of the thesaurus. Beeld en Geluid intends it to become an indexing standard within the whole Media Park.

In the future, it is meant to be extended to an earlier link in the production chain. It would be desirable if at the time of production a large part of the description were already provided. For the documentalists this would mean that they do not describe a resource from scratch, they would rather edit and improve an existing description.

Further the collection is intended to be made accessible to public, as far as rights allow it. A web based search interface is planned for external clients. To allow different search modalities, parts of the thesaurus will be converted to a navigation taxonomy to support intuitive browsing of the collection. This is especially interesting for the museum that Beeld en Geluid are currently building.

5.2.9 PricewaterhouseCoopers

Source: Presentation by Jack Teuber at ARK Conference [19].



Taxonomy

Context	
Enterprise Domain	Audit and assurance, tax and advisory services
# of employees	+122.000
# of offices	in +120 countries

Project dependencies	
Project domain	Business knowledge
Goals	Enhance search, navigation and information retrieval
Target groups	Employees
Resource modalities	Text documents in many different languages

Solution	
Taxonomy used for	Search, navigation and information retrieval
Thesaurus used for	/
Ontology used for	/
Supported search strategies	- Free text search - Navigation of hierarchy
Annotation approach	/
Facets	25

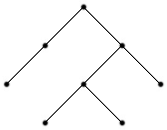
Approach	
Development team roles	/
Tools	Lotus Notes databases for storing content, Verity K2 product for indexing
Legacy	/
Re-use	Lotus Notes database taxonomies
Maintenance	/

Statistics	
# of terms	Between 4 and 150 in each facet (25)
# of levels	2
# of resources	no exact figure available due to distributed nature of environment. Rough estimate: 500.000
Team size	8 of which none full time
Duration of development	ca. 9 months
Maintenance time	1.5 people part-time, + a small group of domain experts that are responsible for different parts of the taxonomy

Assorted
Best Practices
- People do not use more than the first 2 levels of the taxonomy for navigating. To support this search strategy, create overview documents on the second level that provide a summary of the information that can be found within the chosen category, and links to it - Engineer documents to facilitate information retrieval (meaningful page titles, meaningful link names and alt tags, meaningful headings, interlink related documents). - Analyze search log files to find out what users want to know, special attention for "null result" terms, compare with documents with least hits. Are they not interesting enough or just not found?

5.2.10 Sainsbury's

Source: Presentation by Shelley Hardcastle at ARK Conference [19].



Taxonomy

Context[3]	
Enterprise Domain	Food retail
# of employees	150.000
# of offices	564 Sainsbury's Supermarkets in 2004

Project dependencies	
Project domain	Sainsbury's business functions and processes (business direction and performance, business support, customers, products and services, suppliers and supplying, selling)
Goals	Define corporate domain model/ business language to promote information sharing and collaboration
Target groups	- Ca 350 content publishers - all employees using the portal
Resource modalities	Unstructured text resources

Solution	
Taxonomy used for	Classification structure for organizing content in the new corporate portal
Thesaurus used for	/
Ontology used for	/
Supported search strategies	/
Annotation approach	manual, strictly only one concept per resource
Facets	no

Approach	
Development team roles	Information specialist, technical expert, Knowledge Management expert
Tools	Verity
Legacy	old intranet content, migrated to new portal
Re-use	No re-use of off-the-shelf taxonomies, because they were too specialized (e.g. just food classification), whereas the requirement was for an individual corporate taxonomy
Maintenance	Major business changes are anticipated, so a bulk change facility is built into the software. Regular maintenance is performed by an editorial panel. There is a process for adding/deleting terms or changing existing ones. Concepts without resources are deleted, and terms are adapted to the user's feedback.

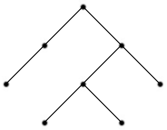
Statistics	
# of terms	/
# of levels	4 levels deep, 6 top-level concepts
# of resources	/
Team size	3 people, plus representatives from all business areas to participate in the workshops
Duration of development	12 weeks
Maintenance time	/

Assorted
Best Practices
- Rules for defining the terms to be used in the taxonomy: no acronyms, no names of departments, no jargon, and just plain English - Do not structure the taxonomy according to departmental names and hierarchy, because this is liable to change, and it carries no meaning for other people - Try to make the taxonomy as durable as possible at the highest level, without making it too rigid to restrict adapting to the natural change of the organization - Taxonomy breadth: 7 top terms, depth: 4 levels - Each resource can only be associated with one node in the taxonomy
Lessons Learned
- One approach to identifying categories for the taxonomy is to derive it from the document population of the intranet. However, this can only be done if the collection is representative of the current business. At Sainsbury's the collection was historic and did not represent the business very well. - Another approach to identify the taxonomy categories is to hold workshop sessions with representatives of each part of the business to determine the terms and structure.

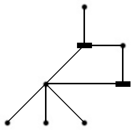
5.2.11 Statoil

Source: Presentation by Anne Kleppe at ARK Conference [19].

Context	
Enterprise Domain	Oil production and trade
# of employees	23.899
# of offices	29 countries



Taxonomy



Ontology

Project dependencies	
Project domain	Business processes and value chain of oil company
Goals	- Increase traceability of information for business purposes and legal and statuatory requirements (Sarbanes Oxley act 404) - Prevent duplication of data by making it easier to find consistently - Improve results of "Work Environment Organization Survey" concerning ease of finding required information - Automatically indexing legacy information for better searchability - Manage large amount of information by automating as much as possible (classification, distribution, retention, processes)
Target groups	- Employees
Resource modalities	Text

Solution	
Taxonomy used for	Subset of metadata model, used to structure the content classification vocabulary.
Thesaurus used for	-
Ontology used for	Here a metadata model was developed that is similar to an ontology, in that it defines all relevant data fields, their definition, data type and allowed values.
Supported search strategies	- Free text search - Facets
Annotation approach	automatic
Facets	yes

Approach	
Development team roles	/
Tools	MS Office 2003 - Sharepoint (Portal toolkit) - Teamsite - Meridio (Content Management platform, partner chosen by Microsoft) - FAST (enterprise search solutions) - Stratify (unstructured data management)
Legacy	Ca 6000 lotus notes databases with content. The policy is to avoid migration. The content is frozen, it can be accessed but not downloaded.
Re-use	Existing standards - Analisys of the existing corporate language - External taxonomies were included where possible (industry standards)
Maintenance	Policy not defined at time of writing, but planned.

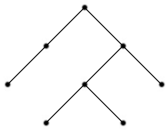
Statistics	
# of terms	/
# of levels	2 levels at corporate level, more detail required at organizational level.
# of resources	Legacy information: 20 TB. Growth of 300.000 information objects per month
Team size	30 people at the beginning, in the “proof of concept” phase. Later 70, of which 50 FTE and 20 PTE.
Duration of development	Ca 5 years so far (version 1)
Maintenance time	/

Assorted
Best Practices
- Shield employees from adding metadata as much as possible! At Statoil, Team Site was used to create basic document templates that add context metadata to newly created documents automatically. The user is only required to add the subject, taking the values from the taxonomy. In this way, even if the author does not add any metadata, the context of the resource is automatically there (e.g. author, organizational unit). - Distinguish metadata elements from taxonomy values. - “buy don’t build” policy for all tools.
Lessons Learned
Not just the creation of the taxonomy is important, but also the import of the taxonomy into the system for utilisation

5.2.12 WoltersKluwer UK

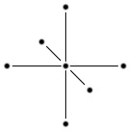
Source: Presentation by Celia Mindelsohn at ARK Conference [19].

Context	
Enterprise Domain	Multi-media publisher of legal, tax and regulatory business
# of employees	+20.000
# of offices	-



Taxonomy

Project dependencies	
Project domain	Products and processes at Wolterskluwer
Goals	- Encapsulate common vocabulary - Content reuse - Accelerate product development - Create product catalogue - Provide business support helplines - Improve customer satisfaction through faster speed-to-market
Target groups	Editors, employees, customers
Resource modalities	XML/SGML encapsulated content in the form of “information objects”



Thesaurus

Solution	
Taxonomy used for	Customer specific vocabulary mapped to corresponding thesaurus term. Taxonomies are product specific and new taxonomies are created as new products dealing with different areas of subject matter are developed. Examples: - Health and safety (12 top level terms, 130 second level) - Human Resources (12 top level terms, 120 second level - Education management (12 top level terms, 90 second level) - Facilities management (10 top level terms, 60 second level)
Thesaurus used for	- Assigning metadata to content - Creating table of content in search interface - Generating index of thesaurus terms - Displaying related terms/topics
Ontology used for	-
Supported search strategies	- Free text search - Hierarchical navigation structure
Annotation approach	Metadata assigned manually
Facets	yes, in master thesaurus

Approach	
Development team roles	- Coordinator - information consultant - subject matter experts - review board
Tools	- MultiTES for thesaurus - Interface: Automatically create TOC, index, "related topics" overview
Legacy	-
Re-use	- Indexes - Catalogue - Industry taxonomies - Marketing categorization - Existing classification scheme
Maintenance	Procedures in place

Statistics	
# of terms	2000 in thesaurus, 450 in taxonomies
# of levels	2 in taxonomies
# of resources	5000
Team size	1 team leader, 9 subject matter experts
Duration of development	ca. 3 months
Maintenance time	1 person acts as Thesaurus manager/coordinator who deals with requests for new and changed terms, referring to review board where required. The thesaurus is in place for over a year so there are very few requests of this nature – perhaps 3 or 4 a month.

<table>
<tr><th colspan="2">Assorted</th></tr>
<tr><th>Problems</th><th>Solutions</th></tr>
<tr><td>Terms change often</td><td>Assign unique ID to concepts, this stays the same, then terms can be adapted to user's language</td></tr>
<tr><td>Annotations are of bad quality</td><td>Focus on training of employees, everybody needs to understand the benefit of the new system for themselves.</td></tr>
<tr><th colspan="2">Benefits</th></tr>
<tr><td colspan="2">- Editorial efficiency
- Accelerated product development
- Improved customer experience (content easier to find for editors and customers)
- Cross referencing between topics
- Faster time-to-market of products through content re-use, prototyping and automated navigation build</td></tr>
</table>

Bibliography

[1] Jean Aitchison, Alan Gilchrist, and David Bawden. *Thesaurus construction and use: a practical manual.* Aslib, London, 1997.

[2] G. Antoniou and F. Van Harmelen. *A Semantic Web Primer.* The MIT Press, 2004.

[3] Ricardo A. Baeza-Yates and Berthier A. Ribeiro-Neto. *Modern Information Retrieval.* ACM Press / Addison-Wesley, 1999.

[4] M. J. Bates. The design of browsing and berrypicking techniques for the online search interface. *Online Review*, 13(5):407–424, 1989.

[5] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *ScientificAmerican.com*, 2001.

[6] Vanda Broughton. Faceted classification as a basis for knowledge organization in a digital environment: the bliss bibliographic classification as a model for vocabulary management and the creation of multidimensional knowledge structures. *The New Review of Hypermedia and Multimedia*, 7(1):67–102, 2001.

[7] Peter Brusilovsky. Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction*, 6(2-3):87–129, 1996.

[8] Terry Butler, Sue Fisher, Greg Coulombe, Patricia Clements, Isobel Grundy, Susan Brown, Jean Wood, and Rebecca Cameron. Can a team tag consistently?: experiences on the orlando project. *Markup Languages*, 2(2):111–125, 2000.

[9] B. Chandrasekaran, John R. Josephson, and V. Richard Benjamins. What are ontologies, and why do we need them? *IEEE Intelligent Systems*, 14(1):20–26, 1999.

[10] Frank van Harmelen Christiaan Fluit, Marta Sabou. Ontology-based information visualisation: Towards semantic web applications.

[11] Web Community. Open directory project. *http://dmoz.org/.*

[12] Melvil Dewey. *Abridged Decimal Classification and Relative Index.* OCLC Online Computer Library Center; 14th edition, 2004.

[13] Jennifer English, Marti Hearst, Rashmi Sinha, Kirsten Swearingen, and Ka-Ping Yee. Hierarchical faceted metadata in site search interfaces. In *CHI '02: CHI '02 extended abstracts on Human factors in computing systems*, pages 628–639, New York, NY, USA, 2002. ACM Press.

[14] Jennifer English, Marti Hearst, Rashmi Sinha, Kirsten Swearingen, and Ka-Ping Yee. Flexible search and navigation using faceted metadata. *Technical Report, University of Berkeley, School of Information Management and Systems*, 2003.

[15] Marti Hearst et al. Flamenco fine arts search. *http://orange.sims.berkeley.edu/cgi-bin/flamenco/arts/Flamenco.*

[16] D. Fensel. *Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce*. Springer-Verlag, 2001.

[17] Alan Gilchrist. Thesauri, taxonomies and ontologies - similarities and differences. *Presentation at Informatie Professional, vakblad voor informtiewerkers, IP Lezing, Amsterdam, 20 Januari*, 2005.

[18] Michael Gordon and Praveen Pathak. Finding information on the world wide web: the retrieval effectiveness of search engines. *Inf. Process. Manage.*, 35(2):141–180, 1999.

[19] Ark Group. Practical taxonomy design and application conference, amsterdam. *http://www.ark-group.com/home/dwnld/p%2Dtaxonomy/*, 2005.

[20] T. R. Gruber. Towards principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands, 1993. Kluwer Academic Publishers.

[21] N. Guarino. Formal ontology and information systems. In N. Guarino, editor, *Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98*, pages 3–15, Trento, Italy. IOS Press.

[22] R. Guha, R. McCool, and E. Miller. Semantic search. In *Proceedings of the 12th international conference on World Wide Web*, pages 700–709. ACM Press, 2003.

[23] R. Hammond. Taxonomy - the science of classification - 1. *Professional Webmaster*, pages 16–22, 2000.

[24] Marti Hearst, Ame Elliott, Jennifer English, Rashmi Sinha, Kirsten Swearingen, and Ka-Ping Yee. Finding the flow in web site search. *Commun. ACM*, 45(9):42–49, 2002.

[25] Clyde W. Holsapple and K. D. Joshi. A collaborative approach to ontology design. *Commun. ACM*, 45(2):42–47, 2002.

[26] Eero Hyvoenen. Museum finland - finnish museums on the semantic web. *http://www.cs.helsinki.fi/group/seco/museums/.*

[27] Eero Hyvoenen, Miikka Junnila, Suvi Kettula, Eetu Mkel, Samppa Saarela, Mirva Salminen, Ahti Syreeni, Arttu Valo, , and Kim Viljanen. Finnish museums on the semantic web: The users perspective on museumfinland. 2004.

[28] Eero Hyvoenen, Mirva Salminen, Miikka Junnila, and Suvi Kettula. A content creation process for the semantic web. 2004.

[29] Yahoo Inc. Yahoo! *http://www.yahoo.com.*

[30] Robert R. Korfhage. *Information storage and retrieval.* John Wiley & Sons, Inc., New York, NY, USA, 1997.

[31] Timo Kouwenhoven. Zoeken en navigeren is vinden. In *Jaarboek Stichting Archiefpublicaties [SAP] 2005 - thema: Audiovisuele archieven*, 2005.

[32] Timo Kouwenhoven and Peter Nieuwenhuizen. Presentation: Navigating taxonomies. *ARK-Group Conference on Taxonomies, Amsterdam*, 2005.

[33] Lexico Publishing Group LLC. Online dictionary. *http://dictionary.com.*

[34] Peter Mika. Flink: The who is who of the semantic web. *Technical Documentation for Semantic Web Challenge 2004 http://prauw.cs.vu.nl:8080/flink/flink-doc.pdf*, 2004.

[35] George A. Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *The Psychological Review*, 63:81–97, 1956.

[36] Lyndon Nixon, Alain Leger, Francois Paulus, and Pavel Shvaiko. Towards a successful transfer of knowledge-based technology to european industry. In *Workshop 'Formal Ontologies Meet Industry', Castelnuovo del Garda, Italy, June 9-10*, 2005.

[37] Natalya F. Noy and Deborah L. McGuinness. Ontology development 101: A guide to creating your first ontology. Technical Report KSL-01-05, Knowledge Systems Laboratory, Stanford University, March 2001.

[38] National Information Standards Organization. Ansi/niso z39.19 - 2003 guidelines for the construction, format, and management of monolingual thesauri. *National Information Standards Organization*, 2003.

[39] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[40] S. Pepper. The tao of topic maps - finding the way in the age of infoglut. *XML Europe 2000, Paris.*

[41] Giovanni Maria Sacco. Dynamic taxonomies: A model for large information bases. *IEEE Transactions on Knowledge and Data Engineering*, 12(3):468–479, 2000.

[42] T. Saracevic and Paul B. Kantor. Studying the value of libary and information services. part ii. methodology and taxonomy. *Journal of the American Society for Information Science*, 48(6):543–563, 1997.

[43] Katharina Schwarz, Timo Kouwenhoven, Virginia Dignum, and Jacco van Ossenbruggen. Supporting the decision process for the choice of a domain modeling scheme. In *Workshop 'Formal Ontologies Meet Industry', Castelnuovo del Garda, Italy, June 9-10*, 2005.

[44] Rodolfo Stecher and Claudia Niedere. Ontology fitness - supporting ontology quality beyond logical consistency. In *Workshop 'Formal Ontologies Meet Industry', Castelnuovo del Garda, Italy, June 9-10*, 2005.

[45] H. Stuckenschmidt, F. van Harmelen, A. de Waard, T. Scerri, R. Bhogal, J. van Buel, I. Crowlesmith, Ch. Fluit, A. Kampman, J. Broekstra, and E. van Mulligen. Exploring large document repositories with rdf technology: The dope project. *IEEE Intelligent Systems*, 19(3):34–40, 2004.

[46] R. Studer, V. R. Benjamins, and D. Fensel. Knowledge engineering: Principles and methods. *Data and Knowledge Engineering (DKE)*, (25):161–197, 1998.

[47] William Swartout and Austin Tate. Guest editors' introduction: Ontologies. *IEEE Intelligent Systems*, 14(1):18–19, 1999.

[48] Michael Uschold. Where are the semantics in the semantic web? *AI Mag.*, 24(3):25–36, 2003.

[49] Mike Uschold and Robert Jasper. A framework for understanding and classifying ontology applications. In V.R. Benjamins, B. Chandrasekaran, A. Gomez-Perez, N. Guarino, and M. Uschold, editors, *Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5), Stockholm, Sweden*, 1999.

[50] W3C. W3c recommendation 10 february 2004, rdf vocabulary description language 1.0: Rdf schema. *http://www.w3.org/TR/rdf-schema/*.

[51] B. J. Wielinga, A. Th. Schreiber, J. Wielemaker, and J. A. C. Sandberg. From thesaurus to ontology. In *K-CAP 2001: Proceedings of the international conference on Knowledge capture*, pages 194–201, New York, NY, USA, 2001. ACM Press.

[52] Inc. Wikimedia Foundation. Taxonomy. *http://en.wikipedia.org/wiki/Taxonomy*.

Disclaimer: We do not guarantee the validity of the URLs.