

Center for

eBusiness@MIT

<http://ebusiness.mit.edu>



A research and education initiative at the MIT Sloan School of Management

Incorporating Generalized Quantifiers into Description Logic to Improve Source selection

Paper 110

June 2001

**Steven Y. Tu
Stuart E. Madnick**

For more information,

please visit our website at <http://ebusiness.mit.edu>

or contact the Center directly at ebusiness@mit.edu or 617-253-7054



Incorporating Generalized Quantifiers into Description Logic to Improve Source Selection

Steven Y. Tu and Stuart E. Madnick

Context Interchange Systems Laboratory
Sloan School of Management
Massachusetts Institute of Technology

June 2001

Abstract

Source selection allows the users to express what they want while the system automatically performs the identification and selection of relevant sources to answer the query request. To automate that process, the system must be able to represent the contents of data sources in a description language. Descriptions of source contents can be characterized by the two concepts of scope and size. This paper builds upon and extends the concept language, description logic (DL), to propose a novel representation system to achieve that goal. We point out that there are technical barriers within description logic limiting the types of data sources that can be represented. Specifically, we show that (1) DL is awkward in representing sufficient conditions, and (2) DL can describe properties of a concept itself only in the case of existential quantification. These barriers limit expressions of size information in source descriptions and thus cause us to extend DL with the notion of generalized quantifiers. We improve the previous results of generalized quantifiers to make them inter-operable with traditional logic. The proposed formalism integrates the nice features of generalized quantifiers into description logic, and hence achieves more expressive power than previous representation systems based purely on description logic. It is also shown that the proposed language preserves those mathematical properties that traditional logic-based formalisms are known to hold.

1. Introduction

1.1 The Importance of Source Selection

Advances in computing and networking technologies now allow huge amounts of data sources to be accessed and shared on an unprecedented scale. The numbers of structured and semi-structured sources appearing on the World Wide Web (WWW) have exhibited unparalleled growth rates in the recent past. At the same time industrial and governmental organizations have experienced a similar explosion in the numbers of internal and external data sources including databases, data feeds, intranets, electronic messages, etc. With this increasing degree of physical connectivity, it is now technologically feasible that users can gain access, beyond their local reach, to many other remote data sources for topics ranging from business, education, to purely leisure. Users of such complex *heterogeneous information systems*¹ however are often assumed to possess familiarity with the content, scope, coverage degree as well as the physical access mechanisms for the sources. These assumptions, however, are challenged because of the large number and because of the rapid growth of data sources on the net. Conceivably, as more and more data sources are added, it will be increasingly formidable for the users to perform source management and selection by themselves. It is an important issue that theories be developed and systems be designed to aid source selection.

There is an increasing need for intelligent software to be developed to effectively manage such large numbers of sources. Such software, namely an *information agent* (Konopnicki and Shmueli 1998) at the representational level needs to transform source contents into useful knowledge in the right form. At the reasoning level, an information agent needs to help users search, gather, select, retrieve, and consolidate information sources in an intelligent manner, in order to be of maximal values to its users. An information agent for source selection – a system that provides a uniform query interface to multiple sources – must then allow the users to express what they “want” while the system automatically performs the identification and selection of relevant sources to answer the query request. To automate that process, the system must be able to represent the contents of data sources in a formal language. More importantly, this formal language should enable finer-grained distinction of source contents such that the users’ queries can be answered more effectively without having to experience query failures through many irrelevant data sources.

1.2 The Importance of Source Description

A data source contains a set of populated data items related to a topic domain. For each source, there is a degree of coverage with respect to the domain. The coverage degree can be very general or narrow, and can be revealed from a data source’s descriptions. The descriptions, different from the real data in an extensional form, capture the source content in an intensional form. Descriptions indicate in what domain those data are and to what extent those data cover. By indicating the coverage degree, the descriptions of a data source can be very useful for the users to get a sense of the content of the source before actually accessing or querying it. More

¹ “Heterogeneous information systems” is a term that encompasses a vast body of literature describing research that appeared in the past under the titles of “heterogeneous databases” (Scheuermann et al., 1990), “federated database systems” (Sheth and Larson, 1990), or “multidatabase systems” (Bright et al., 1994).

often than not, web sites contain not only the online access services to a data source but also the descriptions of the corresponding data source in the same site.

As an example, the home page of *Hoover's Online* (www.hoovers.com) provides a free service to test-use an online data source called *Company Capsules*. In the same page, the following texts are available to describe what kind of data *Company Capsules* contains.

"The capsules contain information on about 11,000 companies, including the largest and fastest-growing companies in the US and around the world."

A more detailed version of the descriptions following the above statement further explains what *Capsules* includes:

- *Every major publicly listed US company traded on the three major stock exchanges.*
- *More than 1,000 of America's largest private companies.*
- *More than 1,000 other private companies.*
- *More than 500 of the most important foreign companies.*
- *Most of the largest law firms and advertising agencies.*

Being a useful device to understand the content of *Capsules*, the above descriptions point to a phenomena; that is, data sources presently found on the Internet usually have contents that are very difficult to be described in a precise fashion. Indeed, it is uncommon that two data sources, both about companies, will have exactly the same contents. They differ either dramatically or subtly in terms of at least their *scopes* and *sizes*. For example, one data source called *CANADA/CD* contains Canadian companies, which differs from *Capsules* because of the geographical scopes being covered. Another data source *Worldscope* might also contain US and international companies, but covers more foreign companies (e.g., most of the important foreign companies) than *Capsules*. In this context, if a user is interested in knowing the financial information about a particular US company, say 'General Motors', then it makes more sense for him/her to select *Company Capsules* as opposed to *CANADA/CD* because of the scope differences, assuming other difference factors immaterial here.

Including size characteristics into source descriptions offers several benefits. It enhances the degree of abstraction by attaching to a set of data its cardinality information such that the content of a data source does not have to be described by explicit enumeration. It also renders intelligent information search because, with this additional size information, relevance of data sources is finer-granular such that the systems can make more intelligent suggestions. Finally, it improves query success possibility in the sense that if the candidate sources are prioritized based on the size ranking under certain probability distribution assumptions, then it is more likely that a query request can be answered accordingly. As an example, if a user is looking for a foreign company, say 'Sumitomo Life Insurance Co.', then *Worldscope* should be given more precedence than *Company Capsules* because of the size distinction. It is therefore our observation that any practically useful system wanting to address the issue of source selection must take into accounts these scope and size differences and reflect them in the descriptions of source contents.

1.3 Research Strategy and Contribution

Our research strategy for addressing the issue of source selection is as follows:

- The content of a data source is described by a target concept that corresponds to the populated instances stored in that data source. For example, *Company Capsules* will be described as a data source about the concept *Company*.

- The target concept is characterized by a scope that is substantiated by a set of pre-defined characteristics. For example, *Company* is characterized by its geographical region (e.g., US or Canadian), its status (e.g., public or private), and its industry type (e.g., manufacturing or service), which all together define a scope for this particular concept *Company*.
- The target concept and associated characteristics are represented using an intensional assertion, which is accompanied by the size with respect to the scope (e.g., there are more than 500 of such companies).

Our research approach is to extend the use of *Description Logic* (DL) to propose a novel representation system to implement the above strategy. It will be pointed out that there are technical barriers within description logic limiting the types of data sources that can be represented. Specifically, we will show that (1) DL is awkward in representing sufficient conditions (Doyle and Patil, 1991), and (2) though DL can describe the properties of objects in a concept in versatile ways, it can describe the properties of a concept itself only in the case of existential quantification (Quantz, 1992). For the first point, whereas it is easy in DL to represent “All companies in data source R are US companies”, it is difficult to say “All US companies are in data source R”. For the second point, whereas it is easy in DL to represent the statement “There are some courses in the concept *Graduate_Course* such that each of which must be taken by at least 3 (and at most 30) students”, it is difficult to represent the statement “There are 5 courses in the concept *Graduate_Course* with such property (e.g., taken by at least 3 students)”. These barriers cause us to extend DL with the notion of *Generalized Quantifiers*.

This paper improves the previous results of generalized quantifiers to make them interoperable with traditional first-order logic. As a result, the proposed formalism, by integrating the nice features of generalized quantifiers into description logic, achieves more expressive power than various representation systems purely based on description logic. In a nutshell, the present work intends to forward the following research contributions.

- Existing representational approaches to source selection only provide functionality at the level of relevance, but bear little to the notions of complete source and finer-grained characterization of relevant sources.
- Current systems are theoretically limited because of the types of quantifiers that can be asserted in their source description languages.
- Description logic, underlying most current systems, can be enriched with the results from generalized quantifier research, to enhance its usability for representing various source contents.
- Our proposed representation scheme, while powerful enough to accommodate a wide variety of source contents, enjoys the same favorable mathematical properties that traditional logic-based languages are known to hold.

This research attempt is complementary to, not competing with, other resource discovery tools based upon the technique of keyword searching (e.g., search engines currently available on the net such as *www.google.com*) in two ways. Firstly, whereas this work focuses particularly on structured data sources, relational databases to be precise, most known search engines focus on unstructured or semi-structured web-page-like documents. Due to the structure of those focused data sources, construction of intensional assertions to describe data sources becomes possible. Secondly, current search engines usually cannot directly access the contents of those relational data sources, but have to go through the interface of CGI; namely, source contents are

dynamically generated as output of CGI programs. Our research approach, regarding this point, has the advantage of making those hidden contents visible to the subsequent source ordering purposes.

The rest of this paper is organized as follows. Section 2 provides two case studies that highlight the importance of the two source characteristics – scope and size. Two applications are considered; namely company information source and city weather information source. The information sources included in each case study are real and intend to illustrate the wide variety of source contents that can currently be found from the net. Section 3 provides a summary of related literature on source selection. Specifically, we focus on the data model and the language underlying each system, and its focused application areas. Section 4 paves the way for our proposed solution by laying out the basic elements that define the source selection problem. Section 5 begins with a review of some important tools required to build our representation language. It covers topics from description logic, quantification function, and generalized quantifiers. We then formally introduce the proposed description language by incorporating generalized quantifiers into DL to come up with a powerful, yet semantically consistent, language. The syntax and semantics of the assertion language are presented in this section, followed by discussions on the mathematical properties of the language. Section 6 describes the computational behaviors of the proposed language. Two types of reasoning central to source ordering are discussed. A general analysis on the complexity of the language is also given. Section 7 concludes the paper with a summary of our contributions by comparing the proposed formalism to other competing approaches and suggests a list of areas where the proposed language can be usefully applied.

2. Two Case Studies

Descriptions to characterize the content of a data source vary in several ways. Two characteristics however are of our particular interest for the issue of source selection, namely *entity domain (scope)* and *size*. In much of the literature on relational databases (Date, 1986), a domain is a set of possible values from which the actual values contained in a given column are drawn. An attribute domain represents the use of a domain within a source, and it refers to the attached information with respect to each of the populated instances. The domain associated with the key attribute is an entity domain, and it refers to the ontological concept of the data populated in a source. An entity domain can usually be classified into a set of sub-domains based on *inheritance* (set-inclusion) relationship. Finally, size refers to the cardinality of a data source indicating the number of instances of a source with respect to the entity domain or sub-domain. To illustrate how these two characteristics can function to capture the contents of data sources found from the net, we focus on two examples – company information sources and city weather information sources.

2.1 Company Information Source

Company information is one common area where a lot of data resources can be found from the net. With the widespread popularity of e-commerce, more and more company information is becoming available through the net. A huge number of companies are putting up their own web sites to publicize their information. Web-based directories and database sources are additional

resources pertaining to company information. Data sources of this category are used for many purposes: financial investment, marketing research, competition analysis, making a business deal, and many others. In spite of the abundance of data sources, there is still no one precise way when a user needs to search on a particular company. The search strategy that one takes and the resources that one uses usually are based on personal experiences and certain idiosyncratic methods. Three aspects of source characteristics are however relatively important when selecting which data source to use: the *location* of the company, the *status* of the company, and the type of *industry* that the company is in.

Many sources organize their information according to the geographical criteria. The geographical coverage may range in distinct degrees of details, from national or regional to multi- or international. It is generally true that the more focused one source is on one location, the more likely it is that it will contain more companies in that location. Knowing which location(s) of the companies that a data source covers and comparing it to the location of the interested company will usually be an important factor when searching a particular company.

The status about a company, being either private or public, is also very important. Publicly held companies are required to release financial information to their shareholders and to the Securities and Exchange Commission (SEC). It is therefore true that finding information on public companies is often much easier to than on privately held companies. Some sources may include information on public companies, some both public and private companies, and a few may deal only with private companies. Whichever the case may be, knowing the status of the companies a data source covers will help to find relevant sources.

The industry of business for a company is another important factor. Generally, a company is pretty much characterized by the type of industry that the company is involved in. Many data sources are organized in that way to cater the needs of people in the same field of interest. An industry can usually be broken down into sub-industries indicating the relationships of the products. For example, the Standard Industrial Classification (SIC) codes are such as a system developed by the United States government to classify businesses according to their primary type of activity.

The location, the status, and the type of industry are central factors in determining what information source actually contains. Using the terminology introduced earlier about entity domain and sub-domain, we will say that the concept “company” is an entity domain and those publicly held manufacturing companies located in the US are instances of the sub-domain of “company”. The following table summarizes the range of company information sources and their respective content descriptions. It is important to note that along with each entity domain description, there is corresponding size information usually appearing as part of the source descriptions documented on the web pages.

Data Source	Industry	Location	Status	Size
Thomas Reg. Online	Manufacturing, Food	US/ Canada	Public/ Private	194,000
Executive Grapevine	Consulting	World	Public/ Private	54,000
Kompass USA	Manufacturing	US		53,000
Moody's Corp. Profile	Banking, Utility, Transportation	US	Public	8,000 Top 1,3000
Hoover's Capsules		US	Public/ Private	All public >1000 private
Leonard's Guide Motor Freight Directory	Trucking, Freight	US	Private	1,100
Edgars/ SEC		US	Public	12,000
Moody's International Manual	Banking, Utility, Transportation	World/ Outside US	Private	Mostly private 120,500
Lasers and Optronics Buying Guide	Laser, Fiber Optics, Electro-optics	US	Private	1,600
Canadian Mines Handbook	Mining	Canada	Public/ Private	2,400
Chinese Ent./ Company	Manufacturing	China	Public/ Private	Largest 50,000
Tennessee High-Tech DB&KB	High-tech	US Tennessee	Private	3,200
DB Publishing Company	Manufacturing	US		335,000

2.2 City Weather Information Source

Nowadays, it is generally easy to acquire temperature, rainfall, and snowfall information for specific locations from the net. In addition to the basic information, it is sometimes possible to obtain other weather information including humidity, winds, and even color satellite imagery. Some Internet sites have hourly data available though most have daily and monthly only. Some have short-term weather information and some have long-term time-series statistical data. Applications for using such weather information range from research purposes (e.g., meteorology) to purely leisure purposes (e.g., traveling). Though the number of available sources about the weather information of cities is currently enormous, searching the weather information for a particular city can however be overwhelming. As a matter of fact, while it is quite straightforward to find weather information about a well-known city, it is relatively more difficult if the target of interest is a small town (e.g., a town Mazu located in the northern part of Taiwan). This is mostly due to the complex interactions of coverage regarding country regions, city granularity, and temporal points. Two aspects of source characteristics are therefore important in this case: the *legal status* of the city, and the *location* of the city. The following table summarizes the range of information sources about city weather and their respective content descriptions. Similar to the former case study on company information, it is important to note that along with each entity domain description there is also corresponding size information. Later in Section 5, we will come back to this case and use it as our running example while we present our research framework.

<i>Data Source</i>	<i>Legal status</i>	<i>Location</i>	<i>Size</i>
TNIT Weather Server	Major city	Taiwan	18
National Weather Service	Major city/town	World	Most
Orientation Taiwan	Major city	Taiwan	34
CNN Weather	Major city	Taiwan	45
Yahoo Weather	City	Taiwan	25
USA Today	Capital city	World	All
World Weather Almanac			
Taiwan Central Weather Bureau	City	Taiwan	All
Intellicast	Capital city	World	Almost all
Kaohsiung City Weather Service	Town	Taiwan/ Kaohsiung	All

3. Literature Review

3.1 Infomaster

Infomaster (Genesereth, Keller, and Duschka, 1997) is a virtual information system allowing users to draw on multiple views to access distributed information sources. It is particularly useful when users with different perspectives must access common information in an integrated environment. The major technological component underlying Infomaster is an Agent Communication Language (ACL), which is a knowledge-representation language combining KQML, Ontology, and KIF. A data source in Infomaster must represent its content knowledge using KIF and stores that knowledge in an agent before registering to Infomaster. There is a World Wide Web interface accessible to users who can enter queries using menu, SQL, or ACL. Currently, Infomaster is used by the CommerceNet Smart Catalog project for several applications, such as rental housing.

3.2 Information Manifold

Information Manifold (Levy, Rajarman, and Ordille, 1996b) provides services for automatic source identification and selection. The user specifies what he/she wants, and the system determines which information sources are relevant to the query using descriptions of the sources available to the system. The four major tasks of the query processor are: 1) find relevant sources, 2) combine feasible sources, 3) generate sound query plan, and finally 4) perform run-time query optimization. Central in the system is the concept of a world-view relation (i.e., an ontology) which is a virtual relation containing relation and attribute names for shared uses. Descriptions of data sources and queries are both represented by conjunctive logical statements. Integrity constraints which are part of the source descriptions are used to identify relevant sources and to enable query optimization at execution time. Applications that used Information Manifold as the underlying system can be found in the automobile market area.

3.2 SIMS

SIMS (Arens, Chee, Hsu, and Knoblock, 1993; Ambite, Arens, Hovy, Philpot, Gravno, Hatzivassiloglou, and Klavans, 2001) allows intelligent access to heterogeneous, distributed information sources by relieving users from having to know the locations of the sources. The concept of a domain model is central to the system. The domain model uses a knowledge representation system to store a fixed vocabulary describing objects in the domain, their attributes, and relationships among them. Queries in SIMS need not specify the name, location, and access method of sources. The system automatically determines how information is obtained from different sources, and how the sources should be joined, combined, and operated. SIMS processes queries in a manner hidden from the users, and uses a query planning system to efficiently execute a query. The domain model, source model, and queries are all represented using LOOM. Applications of SIMS are mainly in the airport transportation area and recently in the energy data collection area.

3.3 TSIMMIS

TRIMMIS (Garcia-Molina, Papakonstantinou, Quass, Rajaraman, Sagiv, Ullman, Vassalos, and Widom, 1997) enables access to multiple heterogeneous information sources by translating source information into a common self-describing object model called *Object Exchange Model* (OEM). Central in the system is a component called translators (wrappers), whose responsibility is to convert queries over information in the common model (OEM) into requests the sources can execute, and conversely convert the data returned by the source into the common model. The query language is called OEM-QL, which adapts existing SQL-like language for object-oriented models. TSIMMIS focuses on accessing sources exhibiting diverse and dynamic information contents, and it handles diverse forms of data sources – unstructured sources, semi-structured sources, or sources that traditional database schema cannot describe. A graphical interface tool called MOBIE via the Web browsers lets Internet users connect to the translators and form queries using QEM-QL. Currently, TSIMMIS are being used particularly for web-wrapping applications.

3.4 HERMES

HERMES (Karacapilidis and Papadias, 1998) is a system based on a hybrid knowledge base for integrating information from diverse data sources and reasoning systems. A mediator language is the central element of the system, which allows defining a domain, a source, and a query, and allows extracting information from multiple domains. The mediator language underlying HERMES is a logic-based and rule-based language. The language extends Prolog clauses with the reasoning capability over uncertainty and time. A yellow-page server, containing information about which domain is current accessible, is used to assist mediator in locating a data source. To register a new service (data source or reasoning system) to the system, a new entry into the yellow page must be added. Applications of HERMES are mainly seen in the semantic and domain integration areas.

3.5 Infosleuth

Infosleuth (Nodine, 1998) allows local autonomy of information sources in an environment where multiple sources have to be located, evaluated, retrieved, and merged for a task. Infosleuth was built on MCC Carnot technology to develop a three-layer architecture- frame layer, meta-model layer, and ontology layer. The ontology is used to capture database schema, conceptual models, and architectural mapping of resources known to the agents. The collaborating agents within Infosleuth exchange messages to each other through a high-level language called KQML. Queries are specified in the representation language KIF with respect to a common ontology. When a query is issued, it is usually routed by the mediation and brokerage agents to those specialized agents managing the specific distributed sources. Users can connect to the Infosleuth's web agent by Java applets available within a web browser. Applications of Infosleuth are mostly in the enterprise modeling and model integration areas.

3.6 Summary of Related Work

Primarily, the related work pointed to a common theme of "intelligent information integration". Each research and development community however has its own view of the integration strategy and its main topics of interests – syntactic, semantic and schematic heterogeneity of data sources. Unsurprisingly, different groups address the problem in different ways, and most importantly the differences tend to be reflected on the underlying data model and the language chosen for each system. The following summary captures the different aspects that each system positions itself within the spectrum of information integration research.

Approach	Data Model	Language	Application	Focus
INFOMASTER	Frame	ACL (KIF, KQML)	Rental housing Stanford Info. Net	Multiple views Information broker
INFORMATION MANIFOLD	Relation	Predicate logic Integrity constraint	Automobile market	Query re-writing Query optimization
SIMS	Object-oriented	LOOM (LISP-like)	Airport transportation Energy data collection	Domain/Source model Query rewriting Semantic query optimization
TSIMMIS	Object-oriented	OEM, OEM-QL	Wrapping	Retrieval/Translator for structural and un-structural data
HERMES	Functional call	Mediator language Domain function	Terrain maps Law enforcement	Domain integration Semantic integration
INFOSLEUTH	Object-oriented	KIF, KQML	Enterprise modeling	Inter-resource collaboration

4. Problem Formulation

Formally, the framework of source selection consists of five key elements. The problem can be formulated as a five-tuple given by $\langle S, Q, O, L, A \rangle$. S is a set of data sources in the relational form, Q is a query corresponding to a user's request, O is an ontology storing a set of shared terminology representing ontological concepts and the relationships among those concepts, L is a language describing the content of each data source in S , and finally A is a reasoning procedure deducing relevance and ordering of the source set in S with respect to Q . The five elements can

be expressed logically to give a more formal view of the system. We devote the following text to elaborate on each element.

4.1 Source Set and Query

The source set S consists of a collection of relations $\{R_1, R_2, R_3, \dots, R_n\}$ where each R_i is a subset of the Cartesian product of a list of domains characterized by a name. Each R_i has a set of attributes $\{A_1, A_2, A_3, \dots, A_m\}$ where each A_i is a column of a relation designated by a name. Logically, all of the tuples in a relation can be represented as predicates sharing the similar structure of $R_i(A_1:a_1, A_2:a_2, A_3:a_3, \dots, A_m:a_m)$ where each a_i is the value for the corresponding attribute A_i . A query over such a relation is a logical formula of the form $Q(C(k_1, k_2, \dots, k_t), X_1, X_2, \dots, X_n)$. For a system to answer this query is meant to say, in logical sense, to find a set of substitutions $Q(X_1/v_1, X_2/v_2, \dots, X_n/v_n)$ that satisfies that the condition C specified in the query Q with $k_1 \sim k_t$ being the ground terms mentioned in C . In other words, the query will succeed if the formula $R(A_1, A_2, \dots, A_m) \models Q(X_1/v_1, X_2/v_2, \dots, X_n/v_n), C(k_1, k_2, \dots, k_t, X_1/v_1, X_2/v_2, \dots, X_n/v_n)$ is logically true.

4.2 Ontology and Source Description

An ontology O is a world view of “things” containing a set of ontological concepts $\{\Sigma\}$ and the relationships among concepts $\{\Phi\}$. Concepts are organized into hierarchies via their *inheritance* relationship. For example, the concept “computer-company” is a sub-concept of “company”. Two concepts are related to each other through a relationship; for instance “company” is related to the other concept “Location” through the relationship “located_in”. An ontology provides a common system to map the different entity domains in the data sources to certain concepts which are related to each other within the ontology.

Each data source in the source set S can then be described using the constructs provided in the ontology to assert the content of that source. The source description language should consist of a set of logical operators such that each source can be described intensionally as a formula. The formula binds the instances of that source to a concept, along with its related source characteristics, to become a logical statement.

4.3 Reasoning and Query Answers

While there is only *scope* characteristics of sources contained in the description, reasoning of a source involves only determining relevance. The answer is either yes or no; namely the data source is separated into two disjoint sets S_{relevant} and $S_{\text{irrelevant}}$. On the other hand, if the source description also includes *size* characteristics, then the reasoning process involved additional ordering; namely there is a finer degree of “how relevant it is”. We use the operator $=_{\text{agent}}$ to denote the reasoning process where the left-hand side constitutes the input to the agent and the right hand side constitutes the output of the agent’s reasoning.

$$A(S \times Q \times O \times L_{S(scope)}) =_{\text{agent}} \{ \text{Unordered Set } (S_{\text{relevant}}, S_{\text{irrelevant}}) \}$$

$$A(S \times Q \times O \times L_{S(scope, Size)}) =_{\text{agent}} \{ \text{Ordered Set } (S_{\text{relevant}}, S_{\text{irrelevant}}) \}$$

The objective of source selection can then be stated formally as follows: Let *Ordered Set* (S_{relevant}) be the set after relevance is determined. Let $P_S: \{P_1, P_2, P_3, \dots, P_n\}$ be the permutation of (S_{relevant}). The goal of source order reasoning is to find a $P_i = \{\text{an ordered set } S_{\text{relevant}}: \{s_1, s_2, s_3, \dots, s_n\}\}$ such that $s_i(A_1/a_1, A_2/a_2, \dots, A_m/a_m) \models Q(X_1/v_1, X_2/v_2, \dots, X_n/v_n), C(k_1, k_2, \dots, k_t, X_1/v_1, X_2/v_2, \dots, X_n/v_n)$ is logical true and i is the smallest index compared to other permutations (i.e., $P_1, P_2, \dots, P_{i-1}, P_{i+1}, \dots, P_n$). The intuition of this formalization is illustrated in Diagram 1.

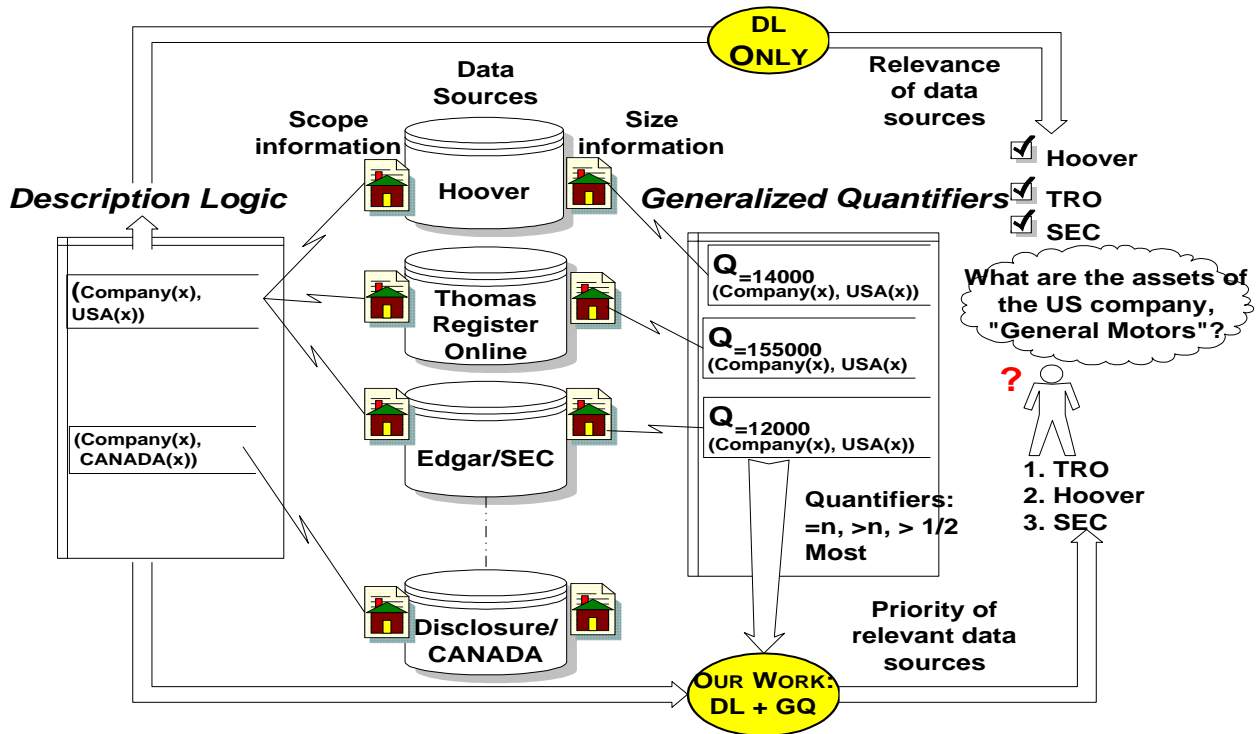


Diagram 1: An illustration of the source selection problem

The above diagram illustrates the research objective in the following sense – while the previous systems only partition the data sources into two sets (relevant and irrelevant), our proposed solution can further index the set of relevant data sources, reflecting the degree of coverage, to allow prioritizing of those relevant data sources. To be more specific, for the previous systems, there is essentially no distinction among *Hoover*, *Thomas Register Online*, and

Edgar/SEC because they all contain US companies. But our proposed framework makes a further step to suggest priority of those three data sources, thus enabling an access sequence.

5. An Expressive Logic-based Representation Language

5.1 A Running Example

Consider a scenario that will be used throughout the rest of this paper. A user is interested in finding weather information regarding Taiwan’s cities from the Internet. Specifically the user would like to have the request P_1 answered. Three data sources pertaining to city weather are listed below, where readers should note that Taipei, Koushung, Taichung, and Tainan are assumed to be all of Taiwan’s major cities (Table 1).

$\langle P_1 \rangle$: What is yesterday’s temperature for the city “Taipei”?

R_1 :		R_2 :		R_3 :	
<i>CityName</i>	<i>Yesterday Temp.</i>	<i>CityName</i>	<i>Yesterday Temp.</i>	<i>CityName</i>	<i>Yesterday Temp.</i>
Taipei	65 ‘F	Taipei	65 ‘F	Koushung	78 ‘F
Koushung	78 ‘F	Koushung	78 ‘F	Taichung	70 ‘F
Taichung	70 ‘F	Taichung	70 ‘F	Tokyo	58 ‘F
Tainan	67 ‘F				
New York	41 ‘F				

Table 1 A user request and three data sources.

The mathematical simplicity and uniformity of a relation, such as R_1 - R_3 , makes it possible to characterize its content intensionally. According to the Entity-Relationship model (Chen, 1976), a relation can be conceived as a collection of entities and attribute values where entities play the role of indicating existence of certain objects in the real world and attributes play a different role of providing descriptions for those entities. To concentrate discussions, we are only concerned about entity coverage but not attribute coverage in this paper. Every relation should have at least one key-identifying attribute, embodying the existence of certain objects, according to the unique entity integrity rule underlying the relational theory (Date, 1986). This property renders a representational alternative based upon the notion of intensionality, which is fundamentally different from the notion of extensionality; namely, enumerating the relation as a collection of extensional instances. For example, we can characterize R_1 intensionally below if that’s what R_1 ’s key attribute, *Cityname*, is supposed to be populated with. Likewise, R_2 and R_3 can also be characterized in the same way.

- R_1 contains {Major cities in Taiwan}, characterized intensionally.
- R_1 contains {Taipei, Koushung, Taichung, Tainan}, characterized extensionally.

Being that case, there is however one semantic gap pointed out by (Kent, 1978) regarding the two different notions of cities.

- The real set of existing major cities in Taiwan.
- The set of Taiwan’s major cities currently represented in each data source.

To bridge this gap, we note that though data sources R_1 - R_3 all contain information about Taiwan’s major cities, the degree with respect to the real existing set that each individual source

covers is unclear given the current descriptions. This point triggers the need to add coverage degree to the description of each individual data source, as the following literal statements demonstrate.

- R₁: Contains all of the major cities in Taiwan.
- R₂: Contains most of the major cities in Taiwan.
- R₃: Contains some of the major cities in Asia.

The diversity of source content doesn't end up there. Let's consider other literal statements describing a range of data sources found on the Internet, of which most are real data sources.

- R₄: Contains more than 3 major cities in Taiwan.
- R₅: Contains more than 1/2 of the major cities in Taiwan.
- R₆: Contains 2 major cities for every country located in Asia.

The above characterization for R₁-R₆ represent the case that the entities contained in each source are quantified toward the real set of existing major cities in Taiwan. They illustrate a sufficient condition for entities that might appear in each data source. Unsurprisingly, data sources that can only be asserted in the form of necessary condition have the equal chance to occur as those in sufficient conditions. R₇-R₁₂ illustrate this point. It is useful to note that, unlike sufficient conditions, necessary conditions are quantified toward the set of entities within the data source itself, not the existing entities (e.g., major cities in Taiwan) in the real world. Readers are invited to pay special attention to R₆ and R₁₂ because they constitute the two most challenging cases that highlight the superiority of our proposed language.

- R₇: All of the contained are major cities in Taiwan.
- R₈: Most of the contained are major cities in Taiwan.
- R₉: Some of the contained are major cities in Taiwan.
- R₁₀: More than 3 of the contained are major cities in Taiwan.
- R₁₁: More than 1/2 of the contained are major cities in Taiwan.
- R₁₂: Most of the contained are major cities in Asia, most of which are in Taiwan.

5.2 The Role of logical quantifiers

In the case that there is a universally quantified data source about major cities in Taiwan like R₁, answering P₁ can be performed in a very effective fashion. To see that, let R₁(x) be a predicate that is assigned true if x is in R₁, and U, the domain x ranges over, be the real set of Taiwan's major cities, then the following inference rule deduces that the particular city 'Taipei' can always be found in R₁. In other words, R₁ is guaranteed to be able to answer P₁. Such a data source is said to have completeness knowledge (Etzioni, Golden, and Weld, 1994; Motro, 1989) with respect to a scope.

$$\{U = \text{the existing set of Taiwan cities, } x \in U, \forall x R_1(x)\} \rightarrow R_1(\text{"Taipei"})$$

or $\{U = \text{the existing set of things, } x \in U, \forall x (\text{Taiwan_city}(x) \Rightarrow R_1(x))\} \rightarrow R_1(\text{"Taipei"})^2$

Data sources with completeness knowledge are undeniably effective for answering users' queries in the form of intensional reasoning. However, as other data sources suggest, establishing completeness knowledge with respect to the real world is not common. In most cases, data sources, as R₂ exemplifies, only contain incomplete knowledge. There are problems when asserting incomplete data sources with quantifiers other than the traditional universal and

² In this paper, \Rightarrow denotes logical conditional and \rightarrow denotes logical implication (deduction).

existential quantifiers if the rudimentary semantics of quantifiers are not well handled. Consider R_2 as an example.

$U =$ the existing set of things, $x \in U$, $\text{Most } x (\text{Taiwan_city}(x) \rightarrow R_2(x))$

The above characterization of R_2 is a wrong one. In fact as argued by (Sher, 1991), any generalized quantifiers (e.g., most, more-than-3, more-than-1/2, less-than-1/3 and etc.) will fail when they are intended to operate with logical conditional. The explanation in the context of this particular assertion is as follows. Suppose that the universe U is $\{\vartheta, \lambda, \mu, \nu, \rho, \sigma, \zeta, \psi\}$, and the corresponding truth table is as listed in Table 2. It is clear that $\text{Most } x (\text{Taiwan_city}(x) \rightarrow R_2(x))$ should be true because most elements out of the universe (e.g., six out of eight elements) turn out to be true for the whole assertion. But obviously it should be false because there are more elements (e.g., ρ and σ) that are in $\text{Taiwan_city}(x)$ but not in $R_2(x)$, than those are (i.e., ζ). There is a logical fallacy here. In other words, the whole assertion is true not because most of Taiwan’s major cities are in R_2 , but because most things in the universe are not Taiwan’s major cities. The impact is that although in reality only R_2 contains most of the Taiwan’s major cities, this assertion is, however, mistakenly true for R_3 - R_6 . Note however that the assertion for R_1 ; namely, $\forall x (\text{Taiwan_city}(x) \rightarrow R_1(x))$, doesn’t involve this logical problem, mainly because of the particularity of the universal quantifier. This logical fallacy being present, we have a situation where for those data sources that can only be asserted by generalized quantifiers, the semantics of logical conditional must be re-examined. We will address this issue in Section 5.5 where we follow a research line that has been rather successful in providing a coherent explanation of semantics of generalized quantifiers through the set-theoretical view.

	$Taiwan_city(x)$	$R_2(x)$	$\text{Most } x (\text{Taiwan_city}(x) \rightarrow R_2(x))$
ϑ	False	-	True
λ	False	-	True
μ	False	-	True
ν	False	-	True
ρ	True	False	False
σ	True	False	False
ζ	True	True	True
ψ	False	-	True

Table 2 A truth table involving a generalized quantifier and logical conditional.

5.3 Mechanism

5.3.1 Description Logic

Description Logic (DL), also known as terminological logic, is strongly related to frame-like languages and has been used as a formalism underlying various knowledge representation systems, such as ARGON (Patel-Schneider, Brachman, and Levesque, 1984), KL-ONE (Brachman and Schmolze, 1985), KRYPTON (Brachman, Fikes, and Levesque, 1983), and LOOM (MacGregor and Bates, 1987). Recently there have been efforts applying DL in the database field (Anwar, Beck, and Navathe, 1992; Borgida and Brachman, 1993; Beneventano, Bergamaschi, and Lordi, 1994; Borgida, Brachman, and McGuinness, 1989; Beck, Gala, and

Navathe, 1989; Borgida, 1995; Bresciani, 1995; Bresciani, 1996; Bergamaschi, Lordi, and Sartori, 1994; Devanbu, 1993; Kessel, Rousselot, Schlick, and Stern, 1995), mainly because of the capabilities that are considered fundamental in semantic data modeling such as structural description and taxonomic classification (Hull and King, 1987) are inherently equipped within DL. Structural constructors pivotal within DL are *concept*, *individual*, and *role* where an individual represents a single object, a concept is a collection of individuals, and a role is a relation associating a pair of concepts or individuals. A new concept can be defined by conjoining concepts that are already defined or by adding *role restriction* and *numeric quantifier restriction* to a defined concept (Calvanese, Lenzerini, and Nardi, 1992; Hollunder and Baader, 1994; MacGregor, 1994). DL's clean syntax (e.g., dot operator and set connectives) comprises of a set of syntactic constructors for complex DL expressions to be formulated, which basically represent classification and subsumption relations among concepts

Systems built on DL usually have two components called *T-box* (Terminological box) and *A-Box* (Assertion box) (Brachman et al., 1983; Giacomo and Lenzerini, 1996). The former box functions like noun-phrase terms that can be referenced while interacting with the system, and the latter functions like propositional sentences that relate the terminological objects in T-box. This separation enormously influences subsequent research on system integration and knowledge representation (Arens et al., 1993; Knoblock, Yigal, and Hsu, 1994; Levy, Srivastava, and Kirt, 1995; Levy, Rajarman, and Ordille, 1996a; Mays, Tyler, McCuire, and Schlossberg, 1987). In our approach, terms appearing in T-box, including individuals, concepts, and roles, correspond respectively to the instance names, target concepts, and concept characteristics. Assertions stored in A-box are to describe the content of each data source by referring to the terms in T-box. Each data source is linked to a target concept, which is characterized by a set of roles and corresponding concepts in T-Box (see Table 3).

The instance name 'Taipei' appearing in P_1 can be modeled as an individual of the concept, *City*. A concept schema for *City*, which echoes the notion of source characteristics, can be written as E_1 , where *THING*, denoted as \perp , is the top primitive concept built into the system. The target concept *City* is characterized by two primitive concepts, *Region* and *Status* through respectively the roles *Located* and *Legal*. The data source R_0 is linked to the concept schema as in E_2^3 where *Located:Taiwan* corresponds to the *Filler* operations in DL. Note that the two universal quantifiers appearing in E_1 represents role restriction, which in this case restricts the values related to the individuals in *City* through the roles *Located* and *Legal* be drawn from only *Region* and *Status* respectively.

T-box:		
Taiwan \in Region	Asia \in Region	World \in Region
Major \in Status	Capital \in Status	Non-capital \in Status
Minor \in Status		
Town \in Status	Legal_city \in Status	City_town \in Status
$E_1:$	City $\sqsubset \perp \sqcap \forall \text{Located.Region} \sqcap \forall \text{Legal.Status}$	
A-box:		
$E_2:$	$R_0 \sqsubset \text{City} \sqcap \text{Located:Taiwan} \sqcap \text{Legal:Major}$	

Table 3 An example of T-box and A-box.

³ In this paper, \sqcap denotes the concept conjunction operation conventionally used in DL, and \cap denotes the set-intersection connective.

From the viewpoint of first-order logic, a concept in DL is a unary predicate and a role is a binary predicate drawing domains from two concepts. Role restriction and subsumption relations are defined on the model-theoretical semantics where a concept C_1 subsumes another concept C_2 just in case the extension of C_2 is a subset, not necessarily proper, of the extension of C_1 for unary predicates (Borgida, 1995). Role subsumption is defined the same way for binary predicates. The following Table 4 (Patel-Schneider and Swartout, 1993) summarizes the syntax and model-theoretical semantics of DL's constructs relevant to this paper.

<i>Syntax input</i>	<i>Syntax abstract</i>	<i>Semantics</i>
TOP	\perp	Δ^I
(Define-primitive-concept CN Concept)	$CN \sqsubseteq C$	$CN^I \subseteq C^I$
(Define-concept)	$CN \sqsupseteq C$	$CN^I = C^I$
(And Concept ₁ Concept ₂ ...Concept _n)	$C_1 \sqcap C_2 \sqcap \dots \sqcap C_n$	$C_1^I \cap C_2^I \cap \dots \cap C_n^I$
(Instance Individual Concept)	$IN \in C$	$IN^I \in C^I$
(All Role Concept)	$\forall R.C$	$\{d \in \Delta^I \mid R^I(d) \subseteq C^I\}$
(Filler Role Individual)	$R:i$	$\{d \in \Delta^I \mid R^I(d) \subseteq i^I\}$
(Subset Role ₁ Role ₂)	$R_1 \sqsubseteq R_2$	$R_1^I \subseteq R_2^I$
(At-least n Role Concept)	$\geq n R.C$	$\{d \in \Delta^I \mid R^I(d) \cap C^I \geq n\}$
(Inverse Role)	R^{-1}	$R^I \cap (\Delta^I \times \Delta^I)$

Table 4 A summary of DL's syntax and semantics

Since the semantics of DL is based upon the model-theoretical view, subsumption in DL therefore has a strong synergy with the idea of logical conditional. In other words, the following characterizations between the two concepts should all be considered equivalent (Baader, 1996).

$$\begin{aligned}
\langle \text{DL perspective - subsumption} \rangle: & C_1 \sqsubseteq C_2 \\
\langle \text{Set-theoretical perspective - containment} \rangle: & C_1^I \subseteq C_2^I \\
\langle \text{First-order logic perspective - conditional} \rangle: & \forall x (C_1(x) \supset C_2(x))
\end{aligned}$$

Following the above analysis, we can actually re-write E_1 and E_2 in a more logic-like form (Baader, 1996). For instance, let U be the universe from which all individuals (i.e., Δ^I) are drawn, E_3 and E_4 rewrite E_1 and E_2 respectively into the form of first-order logic. Logically, E_3 and E_4 should be evaluated time-invariantly true in the system if they are meant to be equivalent with E_1 and E_2 .

$$E_3: \forall x \text{ City}(x) \quad (\text{THING}(x) \wedge (\forall y \text{ Located}(x, y) \quad \text{Region}(y)) \wedge (\forall z \text{ Legal}(x, z) \quad \text{Status}(z)))$$

$$E_4: \forall x R_0(x) \quad (\text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}))$$

Concerning the role `Located` in E_4 , it says that the city x is located in the region Taiwan. Supposing that every city located in Taiwan is also located in Asia, which is in actuality true, then it is valid to replace the predicate `Located(x,Taiwan)` with `Located(x,Asia)` and the whole assertion still remains true. This point actually has some impacts on the problem of answering queries because if a request is asking for cities located in Taiwan, then a data source containing Asian cities should also be considered as a plausible data source for answering that request. In light of this relationship, those individuals within a concept with this property are organized into a hierarchy where the constraint of *inverse role subsumption* (Calvanese et al., 1992) should hold among levels of individuals. For example, all cities in Taiwan are in Asia, and all cities in Asia are in the world all the way but not vice versa (see Figure 1). We call such relationship between two individuals a *role individual subsumption* relationship, which is denoted as \uparrow .

<Role individual subsumption> $\uparrow(\text{Taiwan, Asia}), \uparrow(\text{Asia, World})$
 $\uparrow(\text{Capital, Major}), \uparrow(\text{Non-Capital, Major}),$
 $\uparrow(\text{Major, Legal_City}),$
 $\uparrow(\text{Minor, Legal_City}), \uparrow(\text{Legal_City, City_town}),$
 $\uparrow(\text{Town, City_town})$

<DL > $\text{Located}(x, \text{Taiwan}) \sqsubset \text{Located}(x, \text{Asia}) \sqsubset \text{Located}(x, \text{World})$
 $\text{Legal}(x, \text{Capital}) \sqsubset \text{Legal}(x, \text{Major}) \sqsubset \text{Legal}(x, \text{Legal_city}) \sqsubset \text{Legal}(x, \text{City_town})$
 $\text{Legal}(x, \text{Non-capital}) \sqsubset \text{Legal}(x, \text{Major}), \text{Legal}(x, \text{Minor}) \sqsubset \text{Legal}(x, \text{Legal_city})$
 $\text{Legal}(x, \text{Town}) \sqsubset \text{Legal}(x, \text{City_town})$

<Set> $\text{Located}^{-1}(\text{Taiwan}) \subseteq \text{Located}^{-1}(\text{Asia}) \subseteq \text{Located}^{-1}(\text{World})$

<Logic > $\forall x \text{Located}(x, \text{Taiwan}) \Rightarrow \text{Located}(x, \text{Asia}) \Rightarrow \text{Located}(x, \text{World})$

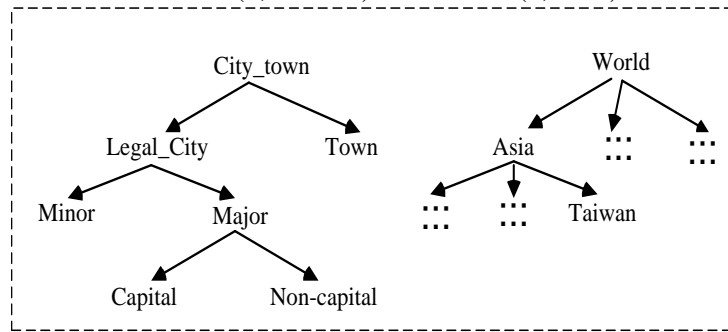


Figure 1 Levels of individuals in concepts Region and Status.⁴

Universal and existential quantifiers, though typical in traditional logic, are atypical in human language. A lot of knowledge that prevail in worldly usage can only be stated in an imprecise form (Westerstahl, 1989), as our previous source examples indicate. Accordingly, allowing generalized quantifiers to be used in source assertions will be of great practical value. To accomplish that goal begs for a fundamental question needed to be investigated as to the nature of quantifiers. In other words, what is a quantifier? Historical studies in mathematical logic provided a clue to unfold this question. According to an influential work about quantifiers (Frege, 1968), Frege contended that a quantifier Q_τ is a second-level object that takes the first-level logical predicate object as input and returns either truth or falsity, where the truth value depends only on the size of the extension satisfying the first-level predicate. Namely, every formula $Q_\tau(\Phi)$ is associated with a quantification function $f_\tau: P(U) \rightarrow \{\text{True}, \text{False}\}$, where $P(U)$ is the power set of U . The function f_τ is assigned true if the model Γ of the logical formula Φ satisfies the cardinality relation specified by Q_τ , and is assigned false otherwise. In other words, the logic formula $Q_\tau(\Phi)$ is true if and only if $f_\tau(\Gamma) \rightarrow \text{True}$. Let's see how the conventional universal and existential quantifiers can be defined on this basis. Again U denotes the universe of discourse.

$$f_\forall(\Gamma) = \quad \text{True} \quad \text{if} \quad U - \Gamma = 0 \quad ; \text{otherwise False}$$

⁴ In this figure, “Capital” means capital cities and “Non-capital” means non-capital major cities, which are disjoint from “Minor cities” and “Towns”. We assume here that there is a taxonomic scheme to classify cities based on their legal status.

$$f_{\exists}(\Gamma) = \begin{array}{ll} \text{True} & \text{if } \Gamma > 0 \\ \text{;otherwise False} & \end{array}$$

A quantification function can also be encoded as a pair (γ, α) such that $\gamma + \beta = \alpha$, where α is the size of the universe, γ is the size of the model Γ for Φ , and β is the size of the complement of Γ . The beauty of Frege's characterization of quantifiers is that it provides a uniform treatment of quantifiers using only cardinal numbers. Mostowski, following the same course, laid out a mathematical foundation for generalized quantifiers based on quantification functions in his seminal paper (Mostowski, 1957). Some generalized quantifiers that frequently come across in human's language follow, out of which we will assume that the literal term "Most" is interpreted as "more than 2/3" of the set in question.

$$\begin{array}{ll} f_{\text{Most}}(\Gamma) = & \text{True if } \gamma \geq 2/3 \alpha \quad \text{;otherwise False} \\ f_{\geq 3}(\Gamma) = & \text{True if } \gamma \geq 3 \quad \text{;otherwise False} \\ f_{=2}(\Gamma) = & \text{True if } \gamma = 2 \quad \text{;otherwise False} \\ f_{\text{more-than-1/2}}(\Gamma) = & \text{True if } \gamma > 1/2 \alpha \text{ or } \gamma > (\alpha - \gamma); \text{otherwise False} \end{array}$$

Given the above quantification functions available, we can describe the contents of R_2 - R_6 using quantification functions. We can see that those data sources such as R_2 - R_6 , which can only be existentially quantified in traditional logic, are now differentiated in a finer manner through their associated quantification functions.

- R_1 : quantifier Q_{\forall} with the quantification function f_{\forall}
- R_2 : generalized quantifier Q_{Most} with the quantification function f_{Most}
- R_3 : quantifier Q_{\exists} with the quantification function f_{\exists}
- R_4 : generalized quantifier $Q_{\geq 3}$ with the quantification function $f_{\geq 3}$
- R_5 : generalized quantifier $Q_{\text{More-than-1/2}}$ with the quantification function $f_{\text{More-than-1/2}}$
- R_6 : quantifier Q_{\forall} and generalized quantifier $Q_{=2}$ with the quantification functions f_{\forall} and $f_{=2}$

5.3.2 Generalized Quantifiers

Mostowski's original and rigorous work inspired immense intellectual interests on the logical and algebraic properties of generalized quantifiers, and since then has spawned a line of elegant theory for researching the nature of generalized quantifiers (Barwise and Cooper, 1981; Benthem, 1982; Sher, 1991). One important result that is of our great interest here is that generalized quantifiers fail for logical conditional (i.e., \Rightarrow) as we mentioned in Section 5.2 about the "Most" quantifier. Barwise (Barwise and Cooper, 1981), aiming at that problem, proposed a rudimentarily different perspective in studying generalized quantifiers. He dismissed the perspective that a quantifier is a logical symbol and switched to view quantifiers as noun phrases. In brevity, Barwise considered a traditional quantifier as a determiner, and it must be supplemented with a set expression to compose an operative quantifier (see Figure 2). Therefore, the meanings of determiners like 'Most' or 'More than 3' are open unless the sets to be modified by these determiners are given.

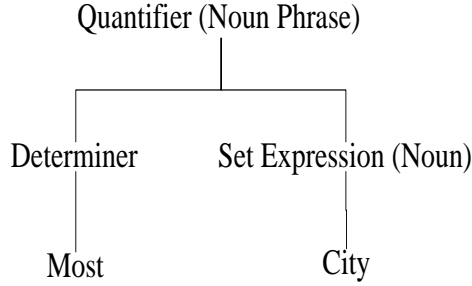


Figure 2 Barwise’s conception of quantifiers.

The following syntax and semantics illuminate Barwise’s stance, where again U denotes the universe of discourse.

All (A’s, B’s)	$\{A \subseteq U \mid A \cap B = A \}$
Most (A’s, B’s)	$\{A \subseteq U \mid A \cap B > A - B \}$
Some (A’s, B’s)	$\{A \subseteq U \mid A \cap B > 0\}$
$>n$ (A’s, B’s)	$\{A \subseteq U \mid A \cap B > n\}$

To better understand these notations using our example, let A denote the set (model) of instance satisfying the predicates expressing cities in Taiwan, and B denote the set (model) of the instances satisfying predicates expressing a data source R . Then $|A|$ and $|B|$ should denote respectively the number of Taiwanese cities in the real world and the number of instances in a data source R , and $|A \cap B|$ should denote the number of Taiwanese cities in the data source R . Now the above notation can be translated into English as follows: to say a data source has more than n Taiwanese cities is meant to say that the cardinality of $|A \cap B|$ is greater than n .

Generalized quantifiers, though potentially promising to the database management area, have been surprisingly under-explored. Recently, there are two works related to this area, (Hsu and Parker, 1995) and (Quantz, 1992). The former paper made an attempt to demonstrate the value of adding generalized quantifiers into traditional terminological logic in the domain of biology (i.e., bound anaphora), in particular on the issue of referential representation for object disambiguation (Quantz, Schmitz, and Kussner, 1994). The latter paper showed the usefulness of generalized quantifiers in query formulation and proposed an approach to extending SQL with the ability to process generalized quantifiers.

Barwise’s conception of generalized quantifiers’ for resolving the case of logical conditional allow us to write down the assertions of those data sources for which generalized quantifiers are needed. However, readers can note that the contents of R_6 and R_{12} still cannot be uttered within Barwise’s framework. The reason can be attributed to, as Sher criticized in (Sher, 1991), Barwise’s dismissal of quantifiers as logical symbols. According to Mostowski, a quantifier should be formula-building and enable us to construct propositions. Namely, syntactically quantifiers must allow us to bind free variables appearing in their attached formulas to generate more complex formulas. In that regard, Barwise’s forsaking generalized quantifiers as logical symbols makes it difficult to syntactically formulate propositions that involve first-order logic variable (e.g., R_6) or involve more than two formulas both containing generalized quantifiers (e.g., R_{12}). It is thus fair to say that Barwise’s work, though useful in forwarding a consistent approach to handling logical conditional with generalized quantifiers, actually loses some compositional expressiveness. This motivates us to seek a language that still maintains the spirit of Barwise’s conception of generalized quantifiers and yet is inter-operable with first-order logic formula so that source contents such as R_6 and R_{12} can be expressed.

- R₁: $Q_{\forall} (\{\text{Major cities in Taiwan}\}, R_1)$
- R₂: $Q_{\text{Most}} (\{\text{Major cities in Taiwan}\}, R_2)$
- R₃: $Q_{\exists} (\{\text{Major cities in Taiwan}\}, R_3)$
- R₄: $Q_{\geq 3} (\{\text{Major cities in Taiwan}\}, R_4)$
- R₅: $Q_{\text{More-than-1/2}} (\{\text{Major cities in Taiwan}\}, R_5)$
- R₆: ???
- R₇: $Q_{\forall} (R_7, \{\text{Major cities in Taiwan}\})$
- R₈: $Q_{\text{Most}} (R_8, \{\text{Major cities in Taiwan}\})$
- R₉: $Q_{\exists} (R_9, \{\text{Major cities in Taiwan}\})$
- R₁₀: $Q_{\geq 3} (R_{10}, \{\text{Major cities in Taiwan}\})$
- R₁₁: $Q_{\text{More-than-1/2}} (R_{11}, \{\text{Major cities in Taiwan}\})$
- R₁₂: ???

In spite of our heavy drawing upon description logic and Barwise’s program of generalized quantifiers, there are however difficulties in using them to represent certain types of data sources. For description logic, one difficulty rests upon the observation that DL is better suited for representing integrity knowledge, but not for representing completeness knowledge. The other difficulty is, as pointed out by (Quantz, 1992), description logic can only assert properties of objects in a set, but not properties of the set itself as an object. In our city example, whereas it is easy to say ‘All cities in the data source R are Taiwan’s major cities’, it is awkward to say ‘All (Most, More-than-n, More-than-1/2) of those Taiwan’s major cities are in R’ or ‘Most (More-than-n, More-than-1/2) of those that are in R are Taiwan’s major cities’. Barwise’s generalized quantifiers arises to remedy this deficiency. Yet his limited view excludes the possibility of having traditional first-order logic variables as part of the formulas that generalized quantifiers are associated with. Targeting at those difficulties, we accordingly propose our solution in more formal detail in this section.

5.4 Limit of Prior Representation Languages

One problem about DL concerns its awkwardness in representing sufficient conditions, and due to that reason completeness knowledge is difficult to be asserted. Let’s examine the problems when trying to use DL to assert R₁’s content, which has complete knowledge about major cities in Taiwan. Three possible alternatives are shown in E₅-E₇, which are all unfeasible due to the following explanations. E₅ is not describing what R₁ contains because it only says that all those in R₁ are Taiwan’s major cities, which is essentially a kind of integrity knowledge as opposed to completeness knowledge. It is the former kind of knowledge because the semantics of `Define-primitive-concept` enforce that the extension of the left-hand side is a subset of the extension of the right-hand side. It is therefore tempting to choose E₆ instead using the DL primitive `Define-concept`, since now the set of Taiwan’s major cities are contained in R₁. Yet E₆ is still an awkward characterization because it is an overstatement in the sense that the primitive “ \sqsupseteq ” also implies necessary condition, which is untrue because R₁ also covers some US cities. The final attempt may be to put `City` at the left-hand side as E₇. This alternative still doesn’t work because in this case R₁ must have been defined somewhere by `City`, which inevitable will cause a definitional cycle (Baader, 1990). In other words, while “ \sqsubset ” represents necessary condition and “ \sqsupseteq ” represents equivalency, it is unclear how sufficient condition alone can be stated in a natural way in DL. As a consequence, it is awkward, if not impossible, using

DL to express complete data sources. In that regard, a logical alternative, as E_8 shows, outperforms DL and allows completeness knowledge to be represented.

E_5 : $R_1 \sqsubset \text{City} \sqcap \text{Located: Taiwan} \sqcap \text{Legal: Major}$

E_6 : $R_1 \sqsupset \text{City} \sqcap \text{Located: Taiwan} \sqcap \text{Legal: Major}$

E_7 : $\text{City} \sqcap \text{Located: Taiwan} \sqcap \text{Legal: Major} \sqsubset R_1$

E_8 : $\forall x ((\text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major})) \Rightarrow R_0(x))$

The other problem about DL is its inability to represent other quantifiers besides the existential quantifier for describing properties at the set level. Although numeric quantifier restrictions are equipped within DL, which look like generalized quantifiers, they however can only be applied to the objects of a concept through *role restriction* (Baader, 1996). Consider the following E_9 as an example, which says ‘all graduate courses must be taken by at least 3 students’. E_{10} shows the logical representation for E_9 . That statement is essentially distinct from another statement ‘there are more than 3 of such courses’ as stated by E_{11} using generalized quantifiers. Accordingly, we can note that numeric quantifier restriction in DL can be used to specify properties only for the existential case at the set level, and only for the universal case at the object level. In other words, DL basically, in this paper’s context, allows us to say ‘There are some cities with respect to the real world in a data source R such that each of which holds the property P (e.g., cities located in Taiwan)’. Yet it is awkward, if not impossible, to say ‘All (Most, More-than-n, More-than-1/2) of those that hold the property P are in a data source R ’ or ‘Most (More-than-n, More-than-1/2) of those that are in a data source R hold the property P ’.

E_9 : $\text{Graduate_course} \sqsubset \text{Course} \sqcap \geq 3 \text{ Taken.Student}$

E_{10} : $\forall x \text{Graduate_course}(x) \Rightarrow (\text{Course}(x) \wedge \text{Student}(s) \wedge \geq 3 \text{ Taken}(x, s))$

E_{11} : $Q_{\geq 3}(\text{Graduate_course}(x), \text{Taken}(x, s))$

The discussions so far suggest that there are representational benefits to be gained if generalized quantifiers are incorporated into DL. Yet means must be sought to avoid Barwise’s syntactic limit such that R_6 involving first-order logic variable and R_{12} involving more than one applications of generalized quantifiers can also be expressed.

5.5 The Proposed Language

5.5.1 Language Definition

In general, Barwise considered that for any quantifier Q_τ , the formula $Q_\tau(A's, B's)$ is a set-theoretical relation between A and B , where τ explicates that relation through its quantification function f_τ . We can embed this notion into description logic by defining a new operator, \rightarrow_{Q_τ} , such that $A(\bar{x}; x_1 \sim x_i) \rightarrow_{Q_\tau} B(\bar{x}; y_1 \sim y_j)$ is true if and only if $Q_\tau(A(\bar{x}), B(\bar{x}))$ is true with the following restrictions.

- $A(\bar{x}; x_1 \sim x_i)$ is a conjunctive DL expression involving $i+1$ variables.
- $B(\bar{x}; y_1 \sim y_j)$ is a conjunctive DL expression involving $j+1$ variables.
- $x_1 \sim x_i, y_1 \sim y_j$ must be either constants or bound variables by either \exists or \forall quantifier.
- \bar{x} is a free variable.
- There cannot be generalized quantifier appearing within A or B .

The semantics of the new operator is defined following the semantics of the generalized quantifier Q_τ along with its quantification function f_τ encoded as the pair (γ, α) . Literally, we can translate the sentence, $A(\bar{x}; x_1 \sim x_i) \rightarrow_{Q_\tau} B(\bar{x}; y_1 \sim y_j)$, into the statement ‘If the premise

predicate $A(\bar{x}; x_1 \sim x_i)$ is true, then there are $\gamma \bar{x}$'s in $A(\bar{x}; x_1 \sim x_i)$ out of $\alpha \bar{x}$'s that also satisfy the consequence predicate $B(\bar{x}; y_1 \sim y_j)$, where again α is the cardinality of the model satisfying $A(\bar{x}; x_1 \sim x_i)$, and γ is the cardinality of the model satisfying both A and B'. We can note that the mapping between the operator \rightarrow_{Q_τ} and Barwise's form of generalized quantifiers is always valid because the only free variable in A and B is \bar{x} , which guarantees that the generalized quantifier Q_τ is applied to two sets of the same sort. In this case, the two sets are composed of only unary tuples. Given that semantic correspondence, our approach is advantageous compared with Barwise's approach in two ways. Firstly, first-order variables quantified by traditional universal and existential quantifiers are now allowed within the premise and consequence predicates that build up the formula involving generalized quantifiers. Secondly, we can now consider $A(\bar{x}; x_1 \sim x_i) \rightarrow_{Q_\tau} B(\bar{x}; y_1 \sim y_j)$ as a regular first-order atomic formula, which can be connected via traditional logical connectives, such as conjunction, disjunction, and negation⁵, to construct more complex formulas. The reason for the second advantage is that the formula $A(\bar{x}; x_1 \sim x_i) \rightarrow_{Q_\tau} B(\bar{x}; y_1 \sim y_j)$, although behaving in the sense of set relation from within the parentheses, is of pure logical sense from the view outside the parentheses (i.e., the interpretation space of the whole sentence is still {truth, falsity}). We now show how this approach can express the contents of R_1 - R_{12} . Again, readers can pay special attention to R_6 and R_{12} because they support respectively the first and second advantages.

- R_1 : $\text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_\forall} R_1(x)$
- R_2 : $\text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_{\text{Most}}} R_2(x)$
- R_3 : $\text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_\exists} R_3(x)$
- R_4 : $\text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_{\geq 3}} R_4(x)$
- R_5 : $\text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_{\text{More-than-1/2}}} R_5(x)$
- R_6 : $\forall y \text{Region}(y) \wedge \hat{\uparrow}(y, \text{Asia}) \Rightarrow (\text{City}(x) \wedge \text{Located}(x, y) \wedge (\text{Legal}(x, \text{Major}) \rightarrow_{Q_{=2}} R_6(x)))$
- R_7 : $R_7(x) \rightarrow_{Q_\forall} \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major})$
- R_8 : $R_8(x) \rightarrow_{Q_{\text{Most}}} \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major})$
- R_9 : $R_9(x) \rightarrow_{Q_\exists} \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major})$
- R_{10} : $R_{10}(x) \rightarrow_{Q_{\geq 3}} \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major})$
- R_{11} : $R_{11}(x) \rightarrow_{Q_{\text{More-than-1/2}}} \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major})$
- R_{12} : $(R_{12}(x) \rightarrow_{Q_{\text{Most}}} \text{City}(x) \wedge \text{Located}(x, \text{Asia})) \wedge (R_{12}(x) \wedge \text{City}(x) \wedge \text{Located}(x, \text{Asia})) \rightarrow_{Q_{\text{Most}}} \text{Located}(x, \text{Taiwan})$

We now define more formally, through the notions of well-formed formula and logical satisfaction, the syntax and semantics of our language combining both traditional universal and existential quantifiers and generalized quantifiers.

<Definition 1>: An assertion ϵ is a **well-formed formula (wff)** in our language Ω provided that:

- (a) If ϵ is in the set of first-order wffs expressed in DL form, then ϵ is in Ω .
- (b) If ϵ is $A(\bar{x}; x_1 \sim x_i) \rightarrow_{Q_\tau} B(\bar{x}; y_1 \sim y_j)$, then ϵ is in Ω .
- (c) If ϵ_1 and ϵ_2 are in Ω , then so are $\epsilon_1 \vee \epsilon_2$, $\epsilon_1 \wedge \epsilon_2$, $\epsilon_1 \Rightarrow \epsilon_2$, $\epsilon_1 \Leftrightarrow \epsilon_2$, and $\neg \epsilon_1$.
- (d) If ϵ doesn't satisfy any of the above, then ϵ is not in Ω .

⁵ In this paper, \wedge denotes logical conjunction, \vee denotes logical disjunction, and \neg denotes logical negation.

<Definition 2>: A wff ε in our language Ω is said to be true with the model Γ provided that:

- (a) All first-logic formulas are true under Γ .
- (b) All generalized-quantifier formulas of the form $A(\bar{x}; x_1 \sim x_i) \rightarrow_{Q_r} B(\bar{x}; y_1 \sim y_j)$ are true under Γ .
- (c) Every universal quantification function is true: $f_{\forall}(\Gamma) \rightarrow \text{true}$.
- (d) Every existential quantification function is true: $f_{\exists}(\Gamma) \rightarrow \text{true}$.
- (e) Every generalized quantification function is true: $f_{\tau}(\Gamma) \rightarrow \text{true}$.
- (f) ε is not true if any of the above is not true.

Definition 1 and 2 concern respectively the syntax and semantics of our proposed language. It is clear from the definitions that our language differs from other DL-based languages primarily in the addition of the term $A(\bar{x}; x_1 \sim x_i) \rightarrow_{Q_r} B(\bar{x}; y_1 \sim y_j)$, which essentially achieves the capability of generalized quantifiers we intend to capture in our language. The proposed language being defined as such, there are two questions that we need to investigate. The first question pertains to the syntactic transformation of formulas, and the second question pertains to the deductibility of a set of formulas. Both questions are fundamental to the studying of logical languages (Enderson, 1971), and are discussed in the format of theorems organized into three related groups in the next sub-section.

5.5.2 Mathematical Properties

The first group of theorems⁶ consider two aspects, one being the relations of equivalency and implication between traditional quantifiers and their corresponding representations using our formalism, and the other being the transformability of the formulas in our language into *Prenex Normal Form* (i.e., all first-order quantifier symbols are left to other symbols). The significance of the following theorems can be illustrated from Theorem 1.3 and 1.4, which say that between the ‘maximal’ (i.e., universal quantifier) and the ‘minimal’ quantifier (i.e., existential quantifier), there are actually other non-traditional (i.e., generalized) quantifiers between these two extreme cases. In the context of this paper, it means that we can have more diverse ways of describing data source contents besides only being allowed to say they are either universally or existentially quantified. For presentation simplicity, we abbreviate $A(\bar{x}; x_1 \sim x_i) \rightarrow_{Q_r} B(\bar{x}; y_1 \sim y_j)$ as $A(\bar{x}) \rightarrow_{Q_r} B(\bar{x})$ in the sequel wherever there is no notional confusion.

<Theorem 1: Equivalency, implication>: The following properties hold for the language Ω .

$$\langle 1.1 \rangle \quad \forall \bar{x} A(\bar{x}) \Rightarrow B(\bar{x}) \vdash \leftarrow \left| A(\bar{x}) \rightarrow_{Q_{\forall}} B(\bar{x}) \right.$$

$$\langle 1.2 \rangle \quad \exists \bar{x} A(\bar{x}) \Rightarrow B(\bar{x}) \vdash \leftarrow \left| A(\bar{x}) \rightarrow_{Q_{\exists}} B(\bar{x}) \right.$$

$$\langle 1.3 \rangle \quad (A(\bar{x}) \rightarrow_{Q_{\forall}} B(\bar{x})) \rightarrow (A(\bar{x}) \rightarrow_{Q_{\exists}} B(\bar{x}))$$

$$\langle 1.4 \rangle \quad (A(\bar{x}) \rightarrow_{Q_r} B(\bar{x})) \rightarrow (A(\bar{x}) \rightarrow_{Q_3} B(\bar{x})) \text{ if } \tau \text{ is not the generalized quantifier "None"}$$

<Theorem 2: Prenex Normal Form>: The following properties hold for the language Ω .

$$\langle 2.1 \rangle \quad \neg \forall \bar{x} A(\bar{x}) \rightarrow_{Q_r} B(\bar{x}) \vdash \leftarrow \left| \exists \bar{x} \neg (A(\bar{x}) \rightarrow_{Q_r} B(\bar{x})) \right.$$

$$\langle 2.2 \rangle \quad \neg \exists \bar{x} A(\bar{x}) \rightarrow_{Q_r} B(\bar{x}) \vdash \leftarrow \left| \forall \bar{x} \neg (A(\bar{x}) \rightarrow_{Q_r} B(\bar{x})) \right.$$

$$\langle 2.3 \rangle \quad (C(x) \Rightarrow (\forall \bar{x} A(\bar{x}) \rightarrow_{Q_r} B(\bar{x}))) \vdash \leftarrow \left| \forall \bar{x} (C(x) \Rightarrow (A(\bar{x}) \rightarrow_{Q_r} B(\bar{x}))) \right.$$

⁶ Proofs of this group of theorems are based on the set-theoretical semantics defined for generalized quantifiers.

- <2.4> $(C(x) \Rightarrow (\exists \bar{x} A(\bar{x}) \rightarrow_{Q_r} B(\bar{x}))) \vdash \leftarrow \exists \bar{x} (C(x) \Rightarrow (A(\bar{x}) \rightarrow_{Q_r} B(\bar{x})))$
 <2.5> $\forall x C(x) \Rightarrow (A(\bar{x}, x) \rightarrow_{Q_r} B(\bar{x}, x)) \vdash \leftarrow \exists x (C(x) \Rightarrow (A(\bar{x}, x) \rightarrow_{Q_r} B(\bar{x}, x)))$
 <2.6> $\exists x C(x) \Rightarrow (A(\bar{x}, x) \rightarrow_{Q_r} B(\bar{x}, x)) \vdash \leftarrow \forall x (C(x) \Rightarrow (A(\bar{x}, x) \rightarrow_{Q_r} B(\bar{x}, x)))$
 <2.7> $\forall x (C(x) \odot (A(\bar{x}, x) \rightarrow_{Q_r} B(\bar{x}, x))) \vdash \leftarrow \forall x C(x) \rightarrow \forall x (A(\bar{x}, x) \rightarrow_{Q_r} B(\bar{x}, x)) \odot \in \{\wedge, \vee\}$
 <2.8> $\exists x (C(x) \odot (A(\bar{x}, x) \rightarrow_{Q_r} B(\bar{x}, x))) \vdash \leftarrow \exists x C(x) \rightarrow \exists x (A(\bar{x}, x) \rightarrow_{Q_r} B(\bar{x}, x)) \odot \in \{\wedge, \vee\}$

The second group of theorems⁷ is related to Definition 2 as to reducing the notion of satisfaction in logic to arithmetic testing of a cardinal relation defined by the quantification functions in question. For example, to check if a wff bound by a universal quantifier is true, we check if the size of the model is equal to the size of the universe. As another example, to check if a wff bound by a generalized quantifier Q_{most} is true, we check if the size of the model satisfies the specified cardinal relation (e.g., $\gamma \geq 2/3 \alpha$). By taking advantage of this concept, we can actually further transform the source assertions, originally in necessary condition (e.g., R₇-R₁₂), into a standardized representation in the form of sufficient condition. The transformation is significant primarily because most subsumption-based reasoning services (Borgida and Brachman, 1993; Bergamaschi, Lordi, and Sartori, 1994; Buchheit, Jeusfeld, Nutt, and Staudt, 1994) that rely on the deduction rule *modus ponens* (McGuinness and Borgida, 1995); that is, $\{\eta \Rightarrow \iota, \eta\} \rightarrow \iota$, would require the data source predicate R_i's to be positioned at the right-hand side of \Rightarrow , and become the results to be deduced. To discuss the properties of transformation, we need some more understanding of the cardinality behaviors associated with various quantifiers.

<**Definition 3**>: The **cardinal relation** specified by a quantification function f_τ , is an arithmetic relation of the form $\gamma \theta \text{Exp}(\gamma, \alpha)$, where θ is an arithmetic operator out of $\{=, >, <, \geq, \leq\}$, and $\text{Exp}(\gamma, \alpha)$ is an arithmetic expression consisting of only two parameters γ and α .

<**Definition 4**>: A cardinal relation is called **model-independent** if its $\text{Exp}(\gamma, \alpha)$ is a constant.

<**Definition 5**>: A cardinal relation is called **model-dependent** if its $\text{Exp}(\gamma, \alpha)$ contains at least α .

Table 5 exemplifies the above definitions for those quantifiers that have appeared in this paper. With the notion of model dependence/independence, we can derive Theorem 3.

Quantifier	Exp(γ, α)	Model dependent/independent	Category
\forall	$\gamma = \alpha$	dependent	Universal quantifier
\exists	$\gamma > 0$	independent	Existential quantifier
Q_{Most}	$\gamma > 2/3 \alpha$	dependent	Portion quantifier
$Q_{\geq 3}$	$\gamma > 3$	independent	Numeric quantifier
$Q_{=2}$	$\gamma = 2$	independent	Numeric quantifier
$Q_{\text{more-than-1/2}}$	$\gamma > 1/2 \alpha$	dependent	Fraction quantifier

Table 5 Model dependent and independent quantifiers.

<**Theorem 3**>: Let $\varepsilon_1: A(\bar{x}) \rightarrow_{Q_r} B(\bar{x})$ be a wff assertion in our language Ω , where A is some data source predicate R. Let Q_τ be a model-independent quantifier, then we can always write ε_1 into $\varepsilon_2: B(\bar{x}) \rightarrow_{Q_r} A(\bar{x})$ such that $\varepsilon_1 \vdash \leftarrow \varepsilon_2$.

⁷ Proofs of this group of theorems are based on the symmetric properties of generalized quantifiers (Barwise and Cooper, 1981).

The above theorem does not apply to model-dependent quantifiers because their cardinal relations involve sizes of the universes, which are reversed after transformation. Now suppose we superscript each $\text{Exp}(\gamma, \alpha)$ to become $\text{Exp}^U(\gamma, \alpha)$ to designate precisely the universe of discourse in question for every particular occurrence of the \rightarrow_{Q_r} operator in the source assertions, then the following theorem holds.

<Theorem 4>: Let $\varepsilon_1: A(\bar{x}) \rightarrow_{Q_r} B(\bar{x})$ be a wff assertion in our language Ω , where A is a data source predicate R . Let Q_τ be a model-dependent quantifier, being only universal, existential, and fraction quantifier, then we can always write ε_1 into $\varepsilon_2: B(\bar{x}) \rightarrow_{Q_r} \text{Exp}^U(\gamma, \alpha) A(\bar{x})$ such that $\varepsilon_1 \mapsto \leftarrow \varepsilon_2$.

Portion quantifier is the only one left out of the above theorems mainly because it requires a further mapping of a qualitative term into a quantitative cardinal relation. In other words, whereas ‘Most’ can mean ‘more than 2/3’, it can also be interpreted as ‘more than 1/2’ or ‘more than 3/4’, each of which will generate a different cardinal relation. However, once the cardinal relation for a particular portion quantifier is given, then Theorem 2 can be applied. We therefore have the following result.

<Theorem 5>: Let $\varepsilon_1: A(\bar{x}) \rightarrow_{Q_r} \text{Exp}^U(\gamma, \alpha) B(\bar{x})$ be a wff assertion in our language Ω , where A is a data source predicate R . Let Q_τ be a portion quantifier and $\text{Exp}^U(\gamma, \alpha)$ is given, then we can always write ε_1 into $\varepsilon_2: B(\bar{x}) \rightarrow_{Q_r} \text{Exp}^U(\gamma, \alpha) A(\bar{x})$ such that $\varepsilon_1 \mapsto \leftarrow \varepsilon_2$.

Consider now a more complex case such that the premise predicate $A(\bar{x}; x_1 \sim x_i)$ is a conjunctive formula with more than one predicates, one of which is a data source predicate (e.g., R_{12}), then the following theorem is derived.

<Theorem 6>: Let $\varepsilon_1: (R(\bar{x}; x_1 \sim x_i) \wedge C(\bar{x}; z_1 \sim z_k)) \rightarrow_{Q_r} B(\bar{x}; y_1 \sim y_j)$ be a wff assertion in our language Ω , where C is a conjunctive formula, and R is a data source predicate. Let Q_τ be a quantifier, either universal, existential, portion or fraction quantifier, we can always write ε_1 into $\varepsilon_2: \varepsilon_1: (C(\bar{x}; z_1 \sim z_k) \wedge B(\bar{x}; y_1 \sim y_j)) \rightarrow_{Q_r} R(\bar{x}; x_1 \sim x_i)$ such that $\varepsilon_1 \mapsto \leftarrow \varepsilon_2$.

The third group of theorems⁸ is to establish the inferential relations of formulas involving generalized quantifiers and role individual subsumption (i.e., \uparrow) we discussed in Section 5.3.1. We only list a partial set of theorems in this group. The value of this group of theorems can be seen from the following inferences stated in English⁹.

‘If there are more than 5 Taiwan cities in R , then there are more than 5 Asian cities in R ’.

‘If there are less than 5 Asian cities in R , then there are less than 5 Taiwan cities in R ’.

‘If all of the Asian cities are in R , then all of the Taiwan cities are in R ’.

‘If some of the Taiwan cities are in R , then some of the Asian cities are in R ’.

<Theorem 7: Deduction with role individual subsumption>: The following properties hold for the language Ω , where θ is the arithmetic operator within the cardinal relation, $\gamma \theta \text{Exp}(\gamma, \alpha)$,

<7.1> $\{\forall x \uparrow(c, x), A(\bar{x}; c) \rightarrow_{Q_r} B(\bar{x}; c)\} \rightarrow A(\bar{x}; x) \rightarrow_{Q_r} B(\bar{x}; x)$ if θ is \geq or $>$

<7.2> $\{\forall x \uparrow(x, c), A(\bar{x}; c) \rightarrow_{Q_r} B(\bar{x}; c)\} \rightarrow A(\bar{x}; x) \rightarrow_{Q_r} B(\bar{x}; x)$ if θ is \leq or $<$

<7.3> $\{\forall x \uparrow(x, c), A(\bar{x}; c) \rightarrow_{Q_v} B(\bar{x}; c)\} \rightarrow A(\bar{x}; x) \rightarrow_{Q_v} B(\bar{x}; x)$

<7.4> $\{\forall x \uparrow(c, x), A(\bar{x}; c) \rightarrow_{Q_s} B(\bar{x}; c)\} \rightarrow A(\bar{x}; x) \rightarrow_{Q_s} B(\bar{x}; x)$

⁸ Proofs of this group of theorems are based on the monotonic properties of generalized quantifiers (Barwise and Cooper, 1981).

⁹ The relationship of *role individual subsumption*, \uparrow , enforces that all Taiwan cities are Asian cities.

The following assertions illustrate the applications of theorems covered in this sub-section to produce assertions of a standardized structure in the form of sufficient condition, which were formulated originally in necessary conditions.

- $R_7: \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_{\forall}} R_7(x) \text{ where } \text{Exp}^{|R_7|}(\gamma, \alpha)$
 $R_8: \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_{\text{Most}}} R_8(x) \text{ where } \text{Exp}^{|R_8|}(\gamma, \alpha)$
 $R_9: \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_{\exists}} R_9(x) \text{ where } \text{Exp}^{|R_9|}(\gamma, \alpha)$
 $R_{10}: \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_{\geq 3}} R_{10}(x) \text{ where } \text{Exp}^{|R_{10}|}(\gamma, \alpha)$
 $R_{11}: \text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \wedge \text{Legal}(x, \text{Major}) \rightarrow_{Q_{\text{More-than-1/2}}} R_{11}(x) \text{ where } \text{Exp}^{|R_{11}|}(\gamma, \alpha)$
 $R_{12}: (\text{City}(x) \wedge \text{Located}(x, \text{Asia}) \rightarrow_{Q_{\text{Most}}} R_{12}(x)) \text{ where } \text{Exp}^{|R_{12}|}(\gamma, \alpha) \wedge (\text{City}(x) \wedge \text{Located}(x, \text{Taiwan}) \rightarrow_{Q_{\text{Most}}} R_{12}(x) \text{ where } \text{Exp}^{|R_{12} \cap \text{City}(X) \cap \text{Located}(X, \text{Taiwan})|}(\gamma, \alpha)$

6. Source Ordering

The proposed representation formalism, in particular the size information, makes it possible for a *reasoning engine* to determine the order of relevant data sources. The order should allow relevant data sources with more possibility answering a request to be given more precedence for selection. In fact, the order among relevant sources can be understood as a generalized view of subsumption where not only the scope information encoded by DL formulas but also the size information embodied by generalized quantifiers is considered as well. In particular, the various generalized quantifiers associated with the satisfied source assertions with respect to a given query can facilitate establishing a further prioritization of the relevant sources, thus resulting in a finer-grained concept hierarchy. Indeed, considerations of both scope and size information in a logic-based language involve studying their interactions and demand a new subsumption algorithm consisting of cardinal-number comparisons. Such algorithm should be able to deal with different cases when the sizes of models regarding DL formulas are extensionally given (e.g., model-independent quantifiers) or intensionally related (e.g., model-dependent quantifiers). Theorems covered in Section 5.5.2 are meant to facilitate that reasoning purpose.

6.1 Reasoning Requirements

Determining source ordering involves both extensional and intensional comparisons. In the first case (e.g., 1000 vs. 2000 US companies), the reasoning engine requires comparison of numeric cardinality. In the second case (e.g., 1000 vs. 1/3 of the US companies), the reasoning engine needs to know the actual number of US companies (or at least the lower bound) before it can do any comparison. It is also interesting to note that it may not always be the case that if the ordering involves model-dependent quantifiers, then intensional comparison is always required. In fact, an intensional comparison is needed only when the ordering involves one model-dependent operator and one model-independent operator. For instance, we do not need to know the size of the US companies if one source contains 1/3 of the US companies and the other source contains 1/4 of the US companies. It is obvious that the former source should contain more companies than the latter source. Indeed, it can be proved that an intensional comparison is

necessary only when the comparison is performed across the two categories (i.e., model-dependent vs. model-independent quantifiers).

The above cases of reasoning are captured in the semantics of our language since our source assertions are interpreted as cardinal relations. To actually solve the intensional case, the reasoning engine can be programmed as a constraint solving system where two (or all) data sources are orderable iff the system is arithmetically solvable. In that regard, work on conjunctive query implication (Ullman, 1989), dealing with checking satisfiability of a set of arithmetic equations, offers a sound basis to approach that problem.

6.2 Computational Behaviors

The computational behavior of the proposed language is an important element determining the practicality of the proposed language. The computational behaviors are mainly dependent on the tractability of determining subsumption of DL formulas incurred in our language. A significant amount of past work in DL has been devoted to this area, and basically come to the conclusion that complexity of DL-based concept languages is essentially determined by the types of constructors that are allowed to appear. Inclusion of certain constructors; for example, *role composition*, will cause subsumption checking to fall beyond the tractability boundary, and lead to intractable computational cost (Brachman and Levesque, 1984), or even undecidability (Schmidt, 1989). Therefore, the main objective of the previous work has been to establish languages that are maximally expressive within the polynomial-time complexity class (Borgida and Patel-Schneider, 1994; Donini, Nutt, Lenzerini, and Nardi, 1991).

Adding generalized quantifiers to the DL-based language may cause an original tractable language intractable. However, there are also situations where addition of generalized quantifiers will not influence the complexity class of the underlying DL language. In fact, it is interesting to study what is the maximal subset of GQ that will not influence the complexity of DL if the data sources of interest are linear orderable. Further studying of this problem suggests tremendous theoretical and practical values because it not only can provide insights to the DL research community on the computational behaviors of a DL-enriched language, but also can make the proposed language more practically useful. The following diagram outlines the architecture for source ordering and the connections to computational issues.

The above architecture captures the possibility of utilizing a Prolog implementation to realize the constraint programming functionality. As mentioned in Section 6.1, the two types of source ordering, intensional and extensional, can be programmed using a constraint solver where a set of sources are linear orderable if a consistent set of substitutions can be assigned to the Prolog variables. A typical problem that a Prolog program based on *Constraint Logic Programming* (CLP) can solve is as follows: the program stores $C(X, Y) :- X = Y + 2$ and the user queries $C(5, Y)$. A program based on standard Program will fail for this type of problem because no explicit constants are given to the variables X and Y . But a CLP-based Prolog (e.g., *ECLiPSe*) will generate the answer $Y = 3$ if the default domain is integer. We can actually use this functionality to handle the case of intensional source ordering to test if two data sources are orderable for those source descriptions with intensional generalized quantifiers attached (e.g., compare $\frac{1}{2}$ of the US cities and 30 US cities). To enable efficient and correct comparisons, the

system must also store a set of cardinality equations invariant across all cases (e.g., $\frac{1}{2}$ US is always greater than $\frac{1}{3}$ US cities, and $\frac{1}{2}$ Asian cities is always greater than $\frac{1}{2}$ Taiwanese cities).

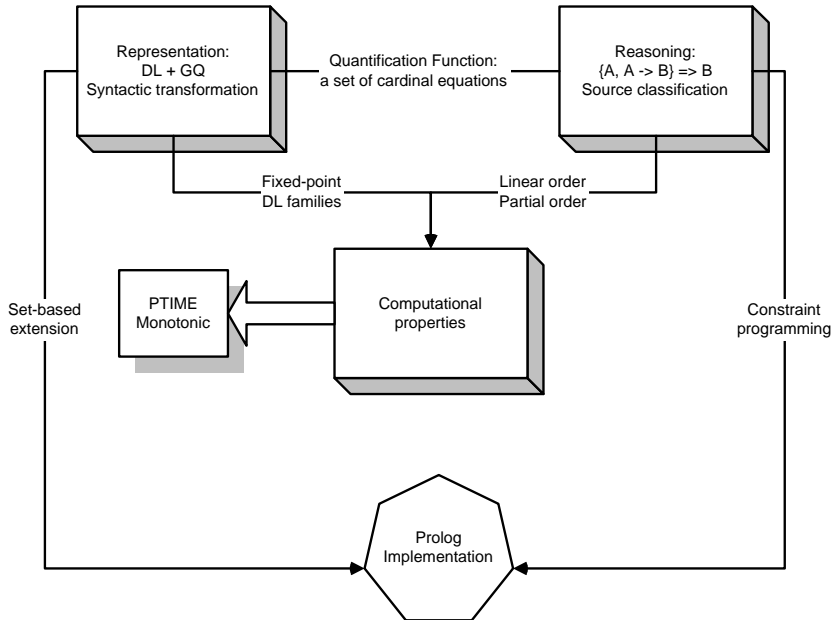


Diagram 2: A source-ordering architecture

7. Discussions and Conclusions

An information agent hoping to address the issue of source selection must deal with uncertainty to decide which source to use since most data sources contain only incomplete knowledge of the world. The information agent must suggest to the users which subset of data sources are most likely to answer a query before actually accessing the sources. Our proposal is motivated by the weakness of description logic in representing sufficient condition for capturing completeness knowledge, and its failure to describe the properties of a concept itself at the set level. We advance a language that exploits the features of generalized quantifiers, and show that this language is expressive in asserting source contents, that otherwise are unable to be stated in other approaches.

Our efforts, compared with other related work (Levy, Mendelzon, and Sagiv, 1995, Arens, Chee, Hsu, and Knoblock, 1993), outperforms in allowing data source contents to be characterized in a finer fashion. Indeed, there is a fundamental improvement in terms of the representational power when comparing our proposed language to the previous research work outlined in Section 3. The languages underlying each of those research projects, KIF (Knowledge Interchange Format), KQML (Knowledge Query and Manipulation Language), and LOOM are, like Classic, are a descendent of the KL-ONE family, which is essentially an implementation of the specification based on description logic or a subset of description logic. Description logic has been considered an important formalism to give a logical ground to those frame-based systems, object-oriented systems, and KL-ONE-like languages. Therefore, the

representation powers of those systems are limited by the maximal capability of description logic. In other words, those systems cannot transcend what the complete set of the DL specification can possibly express. Through our previous discussions, it can be concluded that only scope information can be described in those systems and size information is beyond the representational limit of those systems.

Adding GQ to the underlying formalism has the following impact – whereas other systems are eloquent in representing integrity knowledge and therefore provide an effective way to rule out irrelevant sources, our systems are further capable of differentiating those relevant data sources via size differences. In other words, while other competing systems only characterize data sources at the level of relevance, we are able to take a step further to characterize the coverage degree of each relevant source, and represent it using generalized quantifiers and quantification functions.

Our source selection approach heavily depends on the system's knowledge of the scope and size information of source contents. While such information is sometimes available in web-page format or other electronic forms and public to users' access, the information tends to be given in a very brief fashion; namely, their scope and size are described roughly. We believe that with the current growth in the number of data sources, not before long the pressure from the user community will force source owners to specify data source contents in terms of their scope and size information at a meaningful level of detail. This assumption is especially justified within a single organization with a recognized organizational need to establish a high degree of interoperability among sources, for example, through a data warehouse. Our language will be able to work with such source specifications and translate them into our proposed language with minimal manual interventions to provide automatic source selection services.

References

- Ambite, L., Arens, Y., Hovy, E., Philpot, A., Gravano, L., Hatzivassiloglou, V., and Klavans J. (2001): Simplifying Data Access: The Energy Data Collection Project. *IEEE Computer*, 34(2), 47-54.
- Anwar, T., Beck, H. and Navathe, S. (1992) Knowledge Mining by Imprecise Querying: A Classification-based Approach. *IEEE Intl. Conf. On Data Eng.*, 622-630.
- Arens, Y., Chee, C., Hsu, C. and Knoblock, C. (1993) Retrieving and Integrating Data from Multiple Information Sources. *Intl. Journal of Intelligent and Cooperative Information Systems*, 2(2), 127-158.
- Baader, F. (1990) Terminological Cycles in KL-ONE-based Knowledge Representation Languages. *AAAI-90*, 621-626.
- Baader, F. (1996) A Formal Definition for the Expressive Power of Terminological Knowledge Representation Languages. *Journal of Logic Computation*, 6(1), 33-54.
- Barwise, J. and Cooper, R. (1981) Generalized Quantifiers and Natural Language. *Linguistics and Philosophy*, 4, 159-219.
- Beck, H., Gala, S. and Navathe, S. (1989) Classification as a Query Processing Technique in the CANDIDE Semantic Data Model. *IEEE Intl. Conf. on Data Eng.*, 572-581.
- Beneventano, D., Bergamaschi, S. and Lordi, C. (1994) Terminological Logics for Schema Design and Querying Processing in OODBs. 1st *KRDB-94*.
- Benthem, J. (1982) Questions About Quantifiers. *The Journal of Symbolic Logic*, 4(2), 443-466.
- Bergamaschi, S. and Sartori, C. (1992) On Taxonomic Reasoning in Conceptual Design. *ACM TODS*, 17(3), 385-442.
- Bergamaschi, S., Lordi, S. and Sartori, C. (1994) The E/S Knowledge Representation System. *Data & Knowledge Engineering*, 14, 81-115.
- Borgida, A. (1995) Description Logics in Data Management. *IEEE TKDE*, 7(5), 671-682.
- Borgida, A. and Brachman, R. (1993) Loading Data into Description Reasoner. *SIGMOD Intl. Conf. on Mgnt. of Data*, 217-226.
- Borgida, A. and Patel-Schneider, P. (1994) A Semantics and Complete Algorithm for Subsumption in the CLASSIC Description Logic. *Journal of Artificial Intelligence Research*, Vol. 1, 277-308.
- Borgida, A., Brachman, J. and McGuinness, D. (1989) CLASSIC: A Structural Data Model for Objects. *SIGMOD Intl. Conf. on Mgnt. of Data*, 58-67.
- Brachman, R. and Levesque, H. (1984) The Tractability of Subsumption in Frame-based Description Languages. *AAAI'84*, 34-37.
- Brachman, R. and Schmolze, J. (1985) An Overview of the KL-One Knowledge Representation System. *Cognitive Science*, 9(2), 171-216.
- Brachman, R., Fikes, R. and Levesque, H. (1983) Krypton: A Functional Approach to Knowledge Representation. *IEEE Computer*, 16(10), 67-73.
- Bresciani, P. (1995) Querying Databases from Description Logics. 2nd *KRDB-95*.
- Bresciani, P. (1996) Some Research Trend in KR & DB. 3rd *KRDB-96*.
- Bright, H., Hurson, A., and Pakzad, S. (1994) Automated Resolution of Semantic Heterogeneity in Multidatabases. *ACM TODS*, 19(2), 212-253.
- Buchheit, M., Jeusfeld, M., Nutt, W. and Staudt, M. (1994) Subsumption between Queries to Object-oriented Databases. *Information Systems*, 33-54.

- Calvanese, D., Lenzerini, M. and Nardi, D. (1992) A Unified Framework for Class-based Representation Formalisms. *KR-92*, 109-120.
- Chen, P. (1976) The Entity-Relationship Model: Toward a Unified View of Data. *ACM TODS*, 1(1), 9-36.
- Date, C. J. (1986) *An Introduction to Database Systems*. Addison Wesley Publishing Company.
- Devanbu, P. (1993) Translating Description Logics into Information Server Queries. *2nd Intl. Conf. on Information and Knowledge Management*.
- Donini, F., Nutt, W., Lenzerini, M. and Nardi, D. (1991) Tractable Concept Languages. *IJCAI-91*, 458-468.
- Doyle, J. and Patil, R. (1991) Two Theses of Knowledge Representation: Language Restrictions, Taxonomic Classification, and the Utility of Representation Services. *Artificial Intelligence*, 48, 261-297.
- Enderson, H. (1971) *A Mathematical Introduction to Logic*. Academic Press Inc.
- Etzioni, O., Golden, K. and Weld, D. (1994) A Softbot-based Interface to The Internet. *CACM*, 37(7), 72-76.
- Etzioni, O., Golden, K. and Weld, D. (1994) Tractable Closed World Reasoning With Updates. *KR-94*, 178-189.
- Frege, G. (1968) *The Foundation of Arithmetic*. Tran. J. L. Evaston II: Northwestern U. Press.
- Garcia-Molina, H., Papakonstantinou, Y., Quass, D., Rajaraman, A., Sagiv, Y., Ullman, J., Vassalos, V., and Widom, J. (1997) The TSIMMIS Approach to Mediation: Data Models and Languages. *Journal of Intelligent Information System*, 8(2), 117-132.
- Genesereth, M., Keller, A., and Duschka, O. (1997) Infomaster: An Information Integration System. *SIGMOD Conference*, 539-542.
- Giacomo, D. and Lenzerini, G. (1996) Tbox and Abox Reasoning in Expressive Description Logics. *KR-96*, 316-327.
- Hollunder, B. and Baader, F. (1994) Qualifying Number Restriction in Concept Languages. *KR-94*, 335-346.
- Hsu, P. and Parker, D. (1995) Improving SQL with Generalized Quantifiers. *IEEE Intl. Conf. on Data Eng.*, 298-305.
- Hull, R. and King, R. (1987) Semantic Database Modeling: Survey, Applications, and Research Issues. *ACM Computing Survey*, 19(4), 201-260.
- Karacapilidis, N. and Papadias, D. (1998) Hermes: Supporting Argumentative Discourse in Multi-Agent Decision Making. *AAAI/IAAI*, 827-832
- Kent, W. (1978) *Data and Reality*. North-Holland Publishing Company.
- Kessel, T., Rousselot, F., Schlick, M. and Stern, O. (1995) Use of DL within the Framework of DBMS. *2nd KRDB-95*.
- Knoblock, C., Yigal, A. and Hsu, C. (1994) Cooperating Agents for Information Retrieval. *2nd Intl. Conf. on Cooperative Information Systems*.
- Konopnicki, D. and Shmueli, O. (1998) Information Gathering in the WWW: The W3QL Query Language and the W3QS system. *ACM TODS*, December.
- Levy, A., Mendelzon, Y. and Sagiv, A. (1995) Answering Queries Using Views. *PODS-95*, 95-105.
- Levy, A., Rajarman, A. and Ordille, J. (1996a) Query-Answering Algorithms for Information Agents. *AAAI-96*, 40-47.
- Levy, A., Rajarman, A. and Ordille, J. (1996b) Querying Heterogeneous Information Sources Using Source Descriptions. *VLDB-96*.

- Levy, A., Srivastava, D. and Kirt, T. (1995) Data Model and Query Evaluation in Global Information Systems. *Journal of Intelligent Information Systems*, 5(2), 121-143.
- MacGregor, R. (1994) A Description Classifier for the Predicate Calculus. *AAAI-94*, 213-220.
- MacGregor, R. and Bates, R. (1987) The LOOM Knowledge Representation Language. *Technical Report ISI/RS-87-188*, USC/ISI.
- Mays, W., Tyler, S., McCuire, J. and Schlossberg, J. (1987) Organizing Knowledge in a Complex Financial Domain. *IEEE Expert*, 2(3), 61-70.
- McGuinness, D. and Borgida, A. (1995) Explaining Subsumption in Description Logics. *IJCAI-95*, 816-821.
- Mostowski, A. (1957) On a Generalization of Quantifiers. *Fundamenta Mathematicae*, 44, 12-36.
- Motro, A. (1986) Completeness Information and Its Applications to Query Processing. *VLDB-86*, 170-178.
- Motro, A. (1989) Integrity = Validity + Completeness. *ACM TODS*, 14(4), 480-502.
- Nodine, M., (1998) The InfoSleuth Agent System. *CIA*, 19-20
- Patel-Schneider, P. and Swartout, B. (1993) Description Logic Specification from the KRSS Effort. *KR-93*.
- Patel-Schneider, P., Brachman, R. and Levesque, H. (1984) ARGON: Knowledge Representation Meets Information Retrieval. *Intl. Conf. on AI Application*, 280-286.
- Quantz, J. (1992) How to Fit Generalized Quantifiers into Terminological Logics. *ECAI-92*, 543-547.
- Quantz, J., Schmitz, B. and Kussner, B. (1994) Using Description Logics for Disambiguation in Natural Language Processing. *DL-94*.
- Scheuermann, P. et. al. (1990) Report on the Workshop in Heterogeneous Database Systems. *SIGMOD Record*, December.
- Schmidt, M. (1989) Subsumption in KL-ONE Is Undecidable. *KR-89*, 421-431.
- Sher, G. (1991) *The Bounds of Logic: A Generalized Viewpoint*. The MIT Press.
- Sheth, A. P., and Larson, J. (1990) Federated Database Systems for Managing Distributed, Heterogenous, and Autonomous Databases. *ACM Computing Surveys*, 22(3), 183-236.
- Ullman, J. (1989) *Principles of Database and Knowledge-Base Systems*. Vol. 1, Computer Science Press, New York.
- Westerstahl, D. (1989) Quantifiers in Formal and Natural Languages. *Handbook of Philosophical Logic*, 1-131.