Pointing to places in a deductive geospatial theory

Richard Waldinger
Artificial Intelligence Center
SRI International
waldinger@ai.sri.com

Peter Jarvis
Artificial Intelligence Center
SRI International
jarvis@ai.sri.com

Jennifer Dungan
Ecosystem Science and
Technology Branch
NASA Ames Research Center
jennifer@gaia.arc.nasa.gov

Abstract

Issues in the description of places are discussed in the context of a logical geospatial theory. This theory lies at the core of the system GeoLogica, which deduces answers to geographical questions based on knowledge provided by multiple agents.

1 Introduction

Many questions cannot be answered from information in a single geographical source; often the answer must be deduced from information provided by several sources. It may not be obvious which sources to consult. Because multiple sources seldom agree on conventions of nomenclature or notation, it becomes a problem to determine what place corresponds to a particular description. The same name may apply to many places, and the same place may have many names.

In the system GeoLogica, the coordination between multiple information sources is carried out by an automated deduction system, or theorem prover, that operates in a formal geospatial theory. GeoLogica differs from a search engine in that, instead of merely finding a list of documents with vocabulary that matches the question, it attempts to understand the question and provide an answer.

In developing this system, we have been forced to develop systematic ways for naming places, and for identifying places corresponding to given descriptions.

In this paper, we shall first describe the GeoLogica system. We'll present the representation of place names and discuss mechanisms for finding places corresponding to a given description. We'll discuss the solution of a sample problem, mention some related work, and describe proposed extensions.

2 Outline of GeoLogica

Questions are posed to GeoLogica in a subset of English and translated into logic by a natural language parser, the system Gemini (Dowding et al., 1993). The logical form of the question is rephrased as a theorem and presented to the theorem prover SNARK (Stickel et al., 2000). (A knowledgeable user of GeoLogica may prefer to bypass the parser and phrase the query directly in logical form.) The geospatial theory that SNARK uses for this application consists of a set of axioms, logical sentences that provide definitions and describe properties of important spatial constants, functions, and relations, including those in the logical form of the query. When SNARK proves a theorem, it shows that the theorem follows logically from the axioms in the theory. SNARK also has an answer-extraction mechanism; in addition to proving the validity of the theorem, it can extract from the proof an answer to the query encoded in the theorem, using mechanisms originally developed for automated program synthesis as well as question answering.(Manna and Waldinger, 1980)

Using the appropriate axioms, SNARK transforms and decomposes the query to simpler and simpler subqueries. If the right subqueries of the query are answered, SNARK can extract the answer to the main query from the proof. Answers may be logical terms or, when demanded, visualizations, such as maps or satellite images. There may be many proofs of a theorem, and each proof may yield a different answer; it is possible to induce SNARK to find more and more proofs of the same theorem, and hence more and more answers to the query.

SNARK has some geospatial knowledge that is built into its axioms, but it has access to a far larger body of knowledge through its procedural attachment mechanism. Procedural attachment allows subqueries to be answered by external knowledge sources, which may be programs or data bases and may reside on any machine accessible through the Web. We shall use the generic term

”agent” for an external program invoked by the procedural attachment mechanism.

The capabilities of agents are advertised by axioms in the theory, and the agents themselves are linked to symbols of the theory, so they may be invoked when they are appropriate. The procedural-attachment mechanism allows SNARK to behave as if the information possessed by the external agents were represented as axioms in the geospatial theory.

3 Agents

Among the external agents we have been accessing are

- the Alexandria Digital Library Gazetteer (Hill et al., 1999): a repository of information about some six million place names, including a latitude/longitude or bounding box, a geographical type, a list of alternative names, a small map, and a list of regions that include the place in question.

- the CIA World Factbook (Central Intelligence Agency, 2002): an almanac of most of the world's countries, including geographic, economic, and political information about each, such as its principal subdivisions, bordering countries, capital cities, religions, and principal exports.

- the ASCS (Pease et al., 2002) search engine, which searches the Web for pages that are encoded in DAML (the DARPA Agent Markup Language) and extracts their content; much of the Factbook has been encoded in DAML and is actually accessed through ASCS.

- the TextPro (Appelt and Martin, 1999) information-extraction engine, a system that can derive information from English text documents.

- a variety of procedures for performing numerical and geographical computations, such as those involving latitudes and longitudes and bounding boxes.

- a number of providers of maps and other geographical images, including TerraVision (Reddy et al., 1999), NIMA's Geospatial Engine, Generic Mapping Tools, the NASA Goddard Distributed Active Archive, and the NASA Landsat Project.

A new agent can be added to the system by introducing one or more axiom that advertise its capabilities and introducing a link between a symbol in the theory and the agent.

4 Answer extraction from proofs

A query to GeoLogica is translated by Gemini into a logical expression Q[?x], which contains a variable ?x. (Our convention is to prefix variables with question marks.) This is taken as a theorem to be proved. In this context, the variable in the theorem stands for an object to be found. The query asks us to find a value for ?x such that Q[?x] must be true. We may write the logical form of the query as

```
find ?x such that Q[?x].
```

During the proof, the variable ?x will be instantiated, that is, replaced by other terms, which may in turn contain other variables to be instantiated.

Let us give a simple example. Suppose our query is

```
find ?x such that mother(john, ?x).
```

In other words, our task is to find an entity ?x that is the mother of John. Assume that our theory contains the axiom

```
mother(john, sue),
```

i.e., the mother of John is Sue. Then the proof is immediately complete, and the variable ?x is instantiated to sue. That is, Sue is the answer to our query.

Of course, in general, the answer will not be provided by a single axiom; a sequence of inferences will be necessary, and components of the answer will only be discovered gradually.

5 The geospatial theory

The geospatial theory provides the meanings for the symbols in the query, describes the relationships between various geospatial concepts, advertises the capabilities of our agents, and serves as a repository of knowledge in its own right. One of the first problems we had to face in developing this theory was to provide a way of representing named geographical features. It is desirable for such a place to have a name that allows us to identify it uniquely. For instance, while Springfield, United States does not uniquely specify a town, Springfield, Illinois, United States does.

We have developed a hierarchical naming scheme that allows us to mimic the mechanism used in addressing mail to provide a logical term that designates a named place.

We distinguish between

- regions, which stand for areas on Earth, not necessarily contiguous.

- geographical feature types, such as countries, cities, lakes, or schools.

- subregion indicators, such as Illinois or Springfield, which name subregions of a given region.

Then our named regions are built up as follows:

- Earth is the entire Earth.

- ```
  feature(
    ?geographical-feature-type,
    ?subregion-indicator,
    ?region)'
  ```

  is the subregion of ?region whose name is ?subregion-indicator and whose type is ?geographical-feature-type.

Our convention is that ?region is a variable that stands for a region, and so forth.

For instance,

```
feature(country, Canada, Earth)
```

stands for the country Canada, and

```
feature(
  city,
  Paris,
  feature(country, France, Earth))
```

stands for the city Paris, France. GeoLogica abbreviates countries by their names; thus Canada is simply canada.

It is not intended that notations of the above form are to be used by people; the average user of GeoLogica will never see them. They are used for the internal representation of places as terms in a logical theory.

Nothing guarantees that terms of the above form stand for places that exist or are unique:

```
feature(
  city,
  springfield,
  united states)
```

is not unique, and

```
feature(city, new-york, japan)
```

does not exist.

Properties of these terms are specified by axioms in the geospatial theory.

It should be understood that a function symbol such as feature does not stand for a program that can be computed; it is a notation for speaking about a place.

Strings, Names, and Places

In formulating the geospatial theory we have found it necessary to distinguish between strings, names, and the entities they stand for. For instance, canada is a symbol that denotes the actual country Canada, a region on Earth. The string "Canada", on the other hand, is merely a string, that could be the name of a person rather than a country. Between them we have the name

```
regionq("Canada")
```

which denotes the name of the country Canada, not the country itself. Here regionq maps a string into the corresponding name, although a corresponding region may not exist. We do not assume that a country has a unique name, nor is it impossible for the same name to stand for two countries, although this is more common for cities than for countries. The relation between the name for a region and the region itself is called region-val. In other words,

```
region-val(
  regionq(?region-string),
  ?country)
```

is true if ?region-string is a name for ?country. Thus, for example,

```
region-val(
  regionq("Canada"), canada)
```

is true because "Canada" is a name for the country Canada.

Geographical Feature Types

It is necessary to deal explicitly with geographical feature types within a geospatial theory. We may be asked to find a feature of a certain type. The Alexandria Digital Library Gazetteer requires us to specify the type of a region before it will attempt to find it. We have incorporated the ADL's feature type classification scheme, which is hierarchical. For example, capital is a subtype of city, which is a subtype of populated place.

We also distinguish between types and their name strings, because different agents may have different conventions for specifying types. The ADL, for instance, uses plural strings to stand for types. Thus the string "countries" corresponds to the geographical feature type country.

6 Latitude/longitude pairs and bounding boxes

Another way of describing places on Earth is by coordinate systems, such as latitude and longitude. There are many representations for latitudes and longitudes, in terms of numbers or strings. For example, the 37th North latitude can be represented by the signed string "37" or the compass string "37N". We can also use decimal notation, or the notation based on degrees, minutes, and seconds. Different knowledge agents will produce different representation of latitude and longitude as outputs, and expect different representations as inputs. For instance, the Alexandria Digital Library Gazetteer agent accepts and produces latitudes and longitudes in signed string notation. The agent that computes the distance between latitude/longitude pairs requires latitudes and longitudes in compass notation. The axiom that advertises an agent must specify the notations expected and produced. The
```

geospatial theory, therefore, must discriminate between these notations. Also, some agents will be able to convert from one representation to another.

For latitudes and longitudes represented in compass notation, there is a function symbol

```
lat-long-compass(
  ?lat-compass,
  ?long-compass)
```

that yields the corresponding latitude/longitude pair. Similarly, for the signed notation, there is the function symbol

```
lat-long-sign-string(
  ?lat-sign-string,
  ?long-sign-string).
```

The bounding box of a region is the smallest rectangle that encloses the region, where the sides of the bounding box are made of latitude and longitude lines. Our notations for bounding boxes resembles those for latitude/longitude pairs. Thus

```
bounding-box-sign-string(
  ?lat-sign-string1,
  ?lat-sign-string2,
  ?long-sign-string1,
  ?long-sign-string2)
```

is the bounding box determined by the four numbers given in signed string notation. The numbers correspond to the north, south, east, and west extremes, respectively, of the region in question. Note that the bounding box of a region may contain a lot of terrain outside the region in question. The bounding box of the United States includes much of Canada and Mexico.

In the language of the geospatial theory, region-to-lat-long(?region) is the function that maps a region into its bounding box.

The geospatial theory can be extended to deal with representations of the boundaries of regions, such as vectors of latitude/longitude pairs.

The Procedural Attachment Mechanism

The procedural attachment mechanism allows an agent that is attached to a symbol in the theory to be executed while the proof is in progress. Let us consider a simple example. Suppose that our proof-in-progress contains somewhere the term

```
plus(?real, 2),
```

where ?real stands for a real number. Assume that the symbol plus is attached to an agent, an ordinary program, that performs numerical addition. Because ?real is a variable that has not yet been instantiated, or assigned a value, the agent cannot operate. But now assume that, at some step in the proof, the variable ?real is instantiated by constant 3. Then the resulting term

```
plus(3, 2)
```

is sent to the external addition program, which returns the constant 5.

The Alexandria Digital Library gazetteer is used for several purposes, and the gazetteer is procedurally attached to more than one symbol in the theory. For instance, the symbol place-to-lat-long invokes the gazetteer simply to find the bounding box corresponding to some place whose name is ?region-string and whose type is named ?geo-feature-type-string.

```
place-to-lat-long(
  ?region-string,
  ?geo-feature-type-string,
  ?lat-sign-string1,
  ?lat-sign-string2,
  ?long-sign-string1,
  ?long-sign-string2).
```

The above place-to-lat-long is used mainly to find countries, which are usually uniquely specified by their names.

For example, suppose our proof-in-progress contains the formula

```
place-to-lat-long(
  "Zimbabwe", "countries",
  ?lat-sign-string1,
  ?lat-sign-string2,
  ?long-sign-string1,
  ?long-sign-string2).
```

Then the ADL Gazetteer will be invoked to find the bounding box for Zimbabwe. The variables in the above formula will be instantiated appropriately. SNARK will behave exactly as if the axiom

```
place-to-lat-long(
  "Zimbabwe", "countries",
  "-15.22","-22.93","33.65","25.11").
```

were included in the geospatial theory. For this reason, we call the above sentence a ”virtual axiom.”

Other symbols are procedurally attached to more complex invocations of the ADL, which find a region that is a subregion of a given named region, bounding box, or both. These invocations are necessary when there are many places with the same name, and we need to tell the Gazetteer which one we mean.

Axioms that Advertise the Capabilities of an Agent

By advertising an agent with one or more axioms, we allow it to be invoked when it is appropriate. This makes it easy to add new agents without reprogramming the system. When a new query is presented, the agents that are appropriate for its subqueries stand forward, invoked not by name but as a by-product of the theorem-proving process.

Some of the simplest of these axioms are those that advertise agents that translate from one notation to another. For example, the agent that translates from the signed string notation to the compass notation for latitude and longitude is advertised by the axiom

```
lat-long-compass(
  ?lat-compass, ?long-compass) =
lat-long-sign-string(
  lat-compass-to-sign-string(
    ?lat-compass),
  long-compass-to-sign-string(
    ?long-compass)).
```

Here lat-compass-to-sign-string and long-compass-to-sign-string are function symbols with procedural attachments to programs that perform the conversion from compass notation to signed string notation, for latitudes and longitudes, respectively.

Applying this axiom to a lat-long pair in compass notation, such as

```
lat-long-compass("37N", "122E")
```

will yield the term

```
lat-long-sign-string(
  lat-compass-to-sign-string("37N"),
  long-compass-to-
    sign-string("122E")).
```

Because of the procedural attachments to the function symbols, the compass notation will be converted to signed string notation, yielding the term lat-long-sign-string("37", "-122").

Now let us look at one of the axioms that advertises the Alexandria Digital Library Gazetteer. The following axiom will invoke the gazetteer to find the bounding box for a country:

```
  (region-to-lat-long(?country) =
   bounding-box-sign-string(
     ?lat-sign-string1,
     ?lat-sign-string2,
     ?long-sign-string1,
     ?long-sign-string2))
<=
  (region-val(
     regionq(
       ?region-string),
       ?country) &
     place-to-lat-long(
       ?region-string, "countries",
       ?lat-sign-string1,
       ?lat-sign-string2,
       ?long-sign-string1,
       ?long-sign-string2)).
```

In other words, to find the bounding box (in signed string notation) for a country, find a string that can serve as a name for the country and submit that string, with geographical feature type string "countries", to the gazetteer. The resulting four number strings will correspond to the desired bounding box.

7 A Sample Problem.

Let us consider a problem solved by GeoLogica to illustrate the discovery of a place characterized by a logical combination of properties. The query is

> Show a petrified forest in Zimbabwe within 750 miles of the capital of South Africa.

Here, our convention is that "show" means to display a satellite image, using the TerraVision three-dimensional terrain viewer.

The logical form of the query is

```
find ?x such that
  show(?x) &
  patient(?x, ?y) &
  petrified-forest(?y) &
  in(?y, zimbabwe) &
  within-dist-of(?y,
    within-dist-of(?z, ?u)) &
  mile-unit(?z) &
  count(?z, 750) &
  capital(?u) &
  capital-of(?u, south-africa).
```

In other words, we want to find ?x which is a showing of ?y, where ?y is a petrified forest that is in Zimbabwe and within a distance ?z of ?u, where ?z is in units of miles and has a magnitude of 750, and ?u is the capital of South Africa.

We will not follow the proof in detail; it will be described informally, with indication of where the principal agents were invoked, and what virtual axioms were introduced.

To show a region in TerraVision, it is necessary to find the center of its bounding box, because that is the point we have TerraVision focus on. To find a petrified forest in Zimbabwe, we first find the bounding box of Zimbabwe; this is given by the virtual axiom

```
place-to-lat-long(
  "Zimbabwe", "countries",
  "-15.22","-22.93","33.65","25.11"),
```

as we have seen.

Then we search for petrified forests within that bounding box, checking to see that they are indeed in Zimbabwe. The subquery is

```
place-to-lat-long-part-of-type-bounds(
   ?region-string, "petrified forests",
   "Zimbabwe", "countries",
   "-15.22","-22.93","33.65","25.11"
```

```
?lat-sign-string1,
?lat-sign-string2,
?long-sign-string1,
?long-sign-string2).
```

This causes an invocation of the Alexandria Digital Library Gazetteer in a more complicated way than before. We only specify the type, petrified forest, of the region we are looking for, not its name; we search only within the bounding box of Zimbabwe; and we insist that the found region be a subregion of the country Zimbabwe.

The location of the single petrified forest in Zimbabwe is given by the virtual axiom

```
place-to-lat-long-part-of-type-bounds(
  "Makuku Fossil Forest",
   "petrified forests",
  "Zimbabwe", "countries",
  "-15.22","-22.93","33.65","25.11"
  "-15.65","-15.65","29.95","29.95").
```

Note that, though we have requested a bounding box, the ADL has actually given us a latitude/longitude pair -15.65, 29.95; the north and south latitudes, and the east and west longitudes, are respectively the same.

It is still necessary to ensure that the petrified forest we have found satisfies the additional constraint, that it be within 750 miles of the capital of South Africa. The name of the capital of South Africa, Pretoria, is discovered by invoking the ASCS search engine, which contains the DAML encoding of the CIA World Factbook, including the capitals of countries.

To ensure that the forest is within 750 miles of Pretoria, we first find the latitude and longitude of Pretoria, using the ADL again. We then use a geographical computational agent to ensure that the distance between the latitude/longitudes for the forest and for Pretoria is sufficiently small. The virtual axiom provided by the agent is

```
lat-long-dist("25.75S","28.16667E",
              "15.65S","29.95E",
              "708.0386").
```

Note that this agent requires compass notation—it will not accept signed strings. The agent determines that the distance between Pretoria and the forest is 708 miles, within the 750-mile limit that was specified in the query.

8 SNARK

Although many theorem provers can be used for question answering, SNARK is particularly well suited, for a number of reasons.

- It has strategic controls that allow it to exhibit high performance in a particular subject domain, finding proofs of hundreds of steps in theories with hundreds or thousands of axioms.
- It has a mechanism for extracting answers from proofs.
- It has a procedural attachment mechanism.
- It has special built-in procedures for reasoning efficiently about space and time.

For those who are concerned, it is a first-order logic theorem prover with resolution and paramodulation, implemented in Common Lisp. It has been used in NASA's Amphion system, for automatic software composition, and the Specware formal software development system of the Kestrel Institute, as well as several SRI systems. The current geospatial theory has about a thousand axioms.

9 Other query forms

Here are some other forms of queries that can be answered using the current implementation of GeoLogica:

> Is Zimbabwe north of South Africa?
>
> What is the capital of Zimbabwe.
>
> What is the distance from Arcturus, Zimbabwe to the capital of Cuba?
>
> Display the Generic Mapping Tools map for a beach in Thailand.
>
> Show a cave in Afghanistan within 100 miles of Kandahar, Afghanistan.
>
> Show another.
>
> Show a place in which Mohammed Atta met with an Iraq official.

A question such as "Show another" is treated by allowing SNARK to continue where it left off, finding another proof to the previous theorem, and hence another answer to the previous question. This can be done repeatedly until the set of answers found is depleted.

The last question uses TextPro to establish that Mohammed Atta visited the airport at Ruzyne, Czech Republic, on June 2, 2000. At the same time, the alleged Iraq secret service al Ani was with the Iraq embassy to the Czech republic. Embassies are in capital cities, and the ADL Gazetteer tells us that Ruzyne is less that 7 miles from Prague, the capital of the Czech Republic. TerraVision then displays a satellite image of Ruzyne. It might be mentioned that it is unclear whether such a meeting ever did take place.

10 Use of the gazetteer

The ADL Gazetteer is the agent we rely on most. We have found that we needed to exhibit care in two areas, the treatment of variant or alternative names and the treatment of parts or subregions.

Let us illustrate the alternative names problem. For each place name, the gazetteer maintains a set of alternative names. It generally prefers the local name for a place to the English name. Thus, ask for Prague, Czechoslovakia and it will give you Praha. Ask for Bangkok, Thailand, and it will give you Krung Thep.

On the other hand, the gazetteer also returns the names of places with names that contain the name we want. Search for Chicago, Illinois, and we will find Chicago Ridge, Chicago Heights, and East Chicago, as well as "Chicago (country seat)", which is the place we want. Thus we must be careful to accept true alternative names, such as Chicago (county seat) and Praha, while discarding false alternative names such as East Chicago.

We have also gone through some trouble to ensure that, in searching for a place name within a region, we do not capture places that are within the bounding box of the region but outside the region itself. For instance, in earlier implementations, when searching for Kansas City, Kansas, we would also find Kansas City, Missouri. The bounding box of Kansas contains both cities. Neither city is listed directly as a part of a state. Both cities were listed as parts of places that contained the string "Kansas" in their names. Kansas City, Missouri, is part of the Kansas City MO topographic map, for instance. So we would find both cities.

This problem has been solved by adopting a more precise test of subregion membership; we follow a chain of regions that contain Kansas City, until we get the actual state of Kansas. Thus, Kansas City, Kansas, is listed as a part of Wyandotte County, which in turn is part of Kansas.

11 Relation to other work

There is a large body of work, not surveyed here, in using theorem proving to answer questions or to synthesize programs from specifications. The approach of using procedural attachment to a logical theory to coordinate multiple agents is relatively new, but see Infomaster (Genesereth et al., 1993) and Ariadne (Knoblock and Minton, 1998). The system CYC (Lenat and Guha, 1998) includes a large theory of spatial reasoning and now also incorporates procedural attachment.

Fonseca et al. (Fonseca et al., 2002) are developing a geographical ontology, but it seems not to include axioms, only vocabulary. (A theory is an ontology in which the meaning of the symbols is pinned down by axioms.) Hobbs (Hobbs et al., 1999) is leading a team in building a more spatial ontology and theory in DAML. These should be valuable resources for us.

12 Plans for Future Work

Rather than merely finding and displaying individual places, we hope to introduce capabilities for dealing with and manipulating data, obtained from online agents, for collections of places. We could ask for differences, averages, and maximums, or compare figures from different years. Results could be displayed in charts and tables.

Although GeoLogica may answer a question appropriately, it is usually poor at explaining or defending its answer. Yet its user may not wish to accept the answer without question, particularly if the veracity of any of the sources is in doubt. The proof of the theorem contains a full explanation, but most people do not find logical proofs easy to understand. We plan to develop a more comprehensible explanation facility, in which an understandable explanation will be extracted from the proof, just as the answer itself is now.

SNARK has facilities for temporal reasoning, including date and time arithmetic and the relationships between temporal intervals. However, we have not yet introduced time into the geospatial theory. Taking this step will enable us to deal with changes, such as historical changes in the names and boundaries of countries, changes in the environment, weather, and the movement of populations and individuals.

So far we have been dealing with isolated questions, but it may be more realistic to imagine an information-seeking dialogue, in which the user of GeoLogica poses a series of questions, each refining or elaborating on the one before, and some providing new information to be used in answering future questions.

We have added facilities for extracting information from English text (TextPro), but we have not yet used this in the geographical domain. Care must be observed because information obtained through information extraction is not so reliable as that obtained from other sources.

We are currently applying similar techniques to provide tools of interest to intelligence analysts.

13 Conclusion

Automating question answering forces us to look closely at the relationship between descriptions and places, and a logical geospatial theory is an ideal arena for making this relationship explicit.

Acknowledgments This research has been supported by the NASA Intelligent Systems Program, the ARDA Aquaint Program, and the DARPA DAML Program.

We would like to thank Doug Appelt, Chris Culy, John Fry, Jerry Hobbs, David Martin, Martin Reddy, Susanne Riehemann, Mark Stickel and Mabry Tyson for contributions to the ideas behind this work.

References

Douglas E. Appelt and David L. Martin 1999. Named entity recognition in speech: approach and results using the TextPro system http://www.nist.gov/speech/publications/darpa99/html/-ie30/ie30.htm

Central Intelligence Agency 2002. The world factbook 2002 http://www.cia.gov/cia/publications/factbook/

John Dowding, J. Mark Gawron, Douglas Appelt, John Bear, Lynn Cherny, Robert Moore, and Douglas Moran. 1993. GEMINI: A natural language system for spoken-language understanding. *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 54–61. Ohio State University, Columbus, Ohio

F. Fonseca, M. Egenhofer, C. Davis. G. Camara, and N. Bletter 2002. Semantic granularity in ontology-driven geographic information systems *Annals of mathematics and artificial intelligence*, e6(1–2):121–151.

Michael R. Genesereth, Arthur M. Keller, and Oliver M. Duschka. 1997. Infomaster: an information integration system. *SIGMOD Record (ACM Special Interest Group on Management of Data*,26:539–542. Ohio State University, Columbus, Ohio

Linda L. Hill, J. Frew and Q. Zheng. 1999. Geographic names: The implementation of a gazetteer in a georeferenced digital library. *D-Lib*

Jerry Hobbs et al. 2003. A DAML spatial ontology. http://www.daml.org/listarchive/daml-spatial/0002.htm.

Craig A. Knoblock and Steven Minton. 1998. The Ariadne approach to web-based information integration. *IEEE Intelligent Systems*, 13(5):17–20.

Douglas B. Lenat and R. V. Guha. 1994. Enabling agents to work together *Communications of the ACM*, 37(7)

Zohar Manna and Richard Waldinger. 1980. A deductive approach to program synthesis. *ACM transactions on programming languages and systems*2:90–121.

Adam Pease, John Li, and Chris Barbee. 2002. DAML agent semantic communications service (ASCS) http://oak.teknowledge.com:8080/daml/damlquery.jsp

Martin Reddy, Yvan G. LeClerc, Lee Iverson, and Nat Bletter. 1999. TerraVision II: Visualizing massive terrain databases in VRML. *IEEE computer graphics and applications (special issue on VRML)*. 19(2)30–38.

Mark E. Stickel, Richard J. Waldinger, and Vinay K. Chaudhri. 2000. A guide to SNARK. SRI International, Menlo Park, CA.