

University of Redlands

InSPIRe @ Redlands

MS GIS Program Major Individual Projects

Theses, Dissertations, and Honors Projects

9-2004

GIS Processing for Geocoding Described Collection Locations

Robert Frank Johnson

University of Redlands

Follow this and additional works at: https://inspire.redlands.edu/gis_gradproj

 Part of the [Geographic Information Sciences Commons](#)

Recommended Citation

Johnson, R. F. (2004). *GIS Processing for Geocoding Described Collection Locations* (Master's thesis, University of Redlands). Retrieved from https://inspire.redlands.edu/gis_gradproj/62



This work is licensed under a [Creative Commons Attribution 4.0 License](#).

This material may be protected by copyright law (Title 17 U.S. Code).

This Thesis is brought to you for free and open access by the Theses, Dissertations, and Honors Projects at InSPIRe @ Redlands. It has been accepted for inclusion in MS GIS Program Major Individual Projects by an authorized administrator of InSPIRe @ Redlands. For more information, please contact inspire@redlands.edu.

UNIVERSITY OF REDLANDS

GIS Processing for Geocoding Described Collection Locations

A Major Individual Project Report
Submitted in partial fulfillment of the requirements for the degree of
Master of Science in Geographic Information Systems

By

Robert Frank Johnson

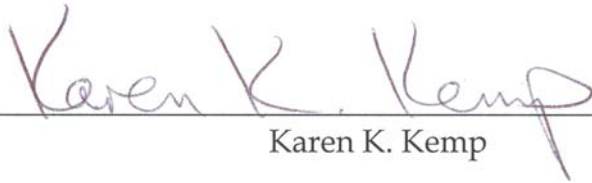
Committee in charge:
Dr. Kelly Chan, Chair
Kent R. Beaman, MS
Dr. Karen K. Kemp

September 2004

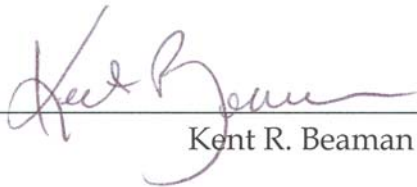
GIS Processing for Geocoding Described Collection Locations

Copyright © 2004
By
Robert Frank Johnson

The report of Robert Frank Johnson is approved.

A handwritten signature in dark ink, reading "Karen K. Kemp", written over a horizontal line.

Karen K. Kemp

A handwritten signature in dark ink, reading "Kent R. Beaman", written over a horizontal line.

Kent R. Beaman

A handwritten signature in dark ink, reading "Kelly Chan", written over a horizontal line.

Kelly Chan, Committee Chair

September 2004

ABSTRACT

GIS Processing for Geocoding Described Collection Locations

By Robert Frank Johnson

Much useful data is currently not available for use in contemporary geographic information systems because location is provided as descriptive text and not in a recognized coordinate system format. This is particularly true for datasets with significant temporal depth such as museum collections. Development is just beginning on applications that automate the conversion of descriptive text based locations to geographic coordinate values. These applications are a type of geocoding or locator service and require functionality in two domains: natural language processing and geometric calculation. Natural language processing identifies the spatial semantics of the text describing a location and tags the individual text elements according to their spatially descriptive role. This is referred to as geoparsing. Once identified, these tagged text elements can be either converted directly to numeric values or used as pointers to geometric objects that represent geographic features identified in the description. These values and geometries can be employed in a series of functions to determine coordinates for the described location. This is referred to as geoprocessing.

Selection of appropriate text elements from a location description and ancillary data as input is critical for successful geocoding. The traverse, one of many types of location description is selected for geocoding development. Specific text elements with spatial meaning are identified and incorporated into an XML format for use as geoprocessing input. Information associated with the location is added to the XML format to maintain database relations and geoprocessing error checking functionality. ESRI's ArcGIS 8.3 is used as a development environment where geoprocessing functionality is tested for XML elements using ArcObjects and VBA forms.

ACKNOWLEDGEMENTS

I acknowledge and thank my committee, Dr. Karen Kemp, and faculty of the MS-GIS program for guidance during the course of class work and for patience through completion of project development. I also express thanks for the support of fellow students, MS-GIS program and Redlands Institute staff, and ESRI instructors. I am indebted most of all to my family and friends for their support and encouragement.

TABLE OF CONTENTS

| | | |
|-------|---|----|
| 1 | INTRODUCTION | 1 |
| 1.1 | Project Background..... | 2 |
| 1.1.1 | Area of Study | 2 |
| 1.1.2 | Client..... | 3 |
| 1.1.3 | Prior Development and Start Up | 3 |
| 1.1.4 | Internet Mapping Service | 3 |
| 1.1.5 | Spatial Semantics..... | 4 |
| 1.1.6 | Natural Language Processing | 4 |
| 1.1.7 | Geoprocessing | 4 |
| 1.2 | Project Objectives | 5 |
| 1.2.1 | Spatial Semantics..... | 5 |
| 1.2.2 | Geoprocessing Tools..... | 6 |
| 1.2.3 | Data Requirements for Geoprocessing | 6 |
| 1.2.4 | Parsing | 6 |
| 1.2.5 | Data Schema | 6 |
| 1.2.6 | Demonstration..... | 6 |
| 1.3 | Report Structure | 6 |
| 2 | PROBLEM DEFINITION | 9 |
| 2.1 | Museum Collection Data | 9 |
| 2.1.1 | Contemporary Context..... | 10 |
| 2.1.2 | Needs Analysis..... | 10 |
| 2.1.3 | Location Source Data..... | 12 |
| 2.1.4 | Descriptive Elements and Methods..... | 16 |
| 2.1.5 | Uncertainty in Data..... | 21 |
| 2.1.6 | Summary | 24 |
| 2.2 | Geoparsing..... | 24 |
| 2.2.1 | Geoparsing Software Requirements..... | 24 |
| 2.2.2 | Geoparsing Data Requirements..... | 26 |
| 2.2.3 | Gazetteer as Geographic Lexicon | 27 |
| 2.2.4 | Spatial Semantics..... | 28 |
| 2.2.5 | Products..... | 29 |
| 2.3 | Geoprocessing | 30 |
| 2.3.1 | Products..... | 30 |
| 2.3.2 | Software Requirements | 30 |
| 2.3.3 | Data Requirements | 31 |
| 2.3.4 | Precision | 31 |
| 2.4 | Data Transfer and Storage | 32 |
| 2.5 | Software Solutions | 35 |
| 2.5.1 | Current Commercial Software | 35 |
| 2.5.2 | Inventory of Required Software | 38 |
| 2.6 | Summary | 39 |

| | | |
|-------|---|----|
| 3 | GEOPROCESSING DATABASE..... | 41 |
| 3.1 | Schema..... | 41 |
| 3.1.1 | Modeling Locations for Geoprocessing..... | 41 |
| 3.1.2 | Implementation..... | 41 |
| 3.2 | Spatial Reference..... | 42 |
| 3.2.1 | Area of Interest..... | 42 |
| 3.2.2 | Coordinate System..... | 42 |
| 3.2.3 | Units of Measure..... | 43 |
| 3.2.4 | Precision..... | 43 |
| 3.3 | Datasets..... | 43 |
| 3.3.1 | Sources of Data..... | 44 |
| 3.3.2 | Data Preparation..... | 44 |
| 4 | GEOPROCESSING SOLUTIONS AND TESTING..... | 47 |
| 4.1 | Geoprocessing Research..... | 47 |
| 4.1.1 | Figure/Ground Geometry..... | 48 |
| 4.1.2 | Geometric Functions..... | 48 |
| 4.1.3 | Spatial Relation Variable Functions..... | 50 |
| 4.1.4 | ArcObjects Development..... | 52 |
| 4.2 | Completed User Interfaces..... | 55 |
| 4.2.1 | Euclidean Location Calculator..... | 55 |
| 4.2.2 | Route Location Calculator..... | 57 |
| 4.3 | Tests of Euclidean Traverse Geoprocessing Functions..... | 58 |
| 4.3.1 | Place Name POB..... | 58 |
| 4.3.2 | Multiple Traverse Segments..... | 61 |
| 4.4 | Tests of Linear Referencing Geoprocessing Functions..... | 64 |
| 4.4.1 | Route Intersecting Route POB..... | 65 |
| 4.4.2 | Route Intersecting Place Name POB..... | 67 |
| 4.5 | Mixed Euclidean and Route Traverse Geoprocessing Functions..... | 69 |
| 5 | SUMMARY AND CONCLUSIONS..... | 73 |
| 5.1 | Lessons learned..... | 74 |
| 5.2 | Recommendations for Further Geocoding Development..... | 74 |
| 5.2.1 | Collection Database Administration..... | 75 |
| 5.2.2 | Geoprocessing Methods..... | 75 |
| 6 | WORKS CITED..... | 77 |

APPENDICES

| | |
|---|-----|
| APPENDIX 1: Spatial Semantics in Descriptions of Location | 79 |
| APPENDIX 2: Geoparser Output File Specifications..... | 111 |
| APPENDIX 3: GNIS Feature Types..... | 125 |
| APPENDIX 4: ArcMap VBA Code | 127 |
| APPENDIX 5: Glossary of Terms | 145 |

LIST OF TABLES

| | |
|--|----|
| Table 1 - Biological Collection Data Functional Use Cases..... | 11 |
| Table 2 - Collection Database Fields With Spatially Relevant Content | 13 |
| Table 3 - Types of Locality Descriptions..... | 20 |
| Table 4 - Phrase Fragment Repairs..... | 27 |
| Table 5 - Geoprocessing Spatial Reference Datasets | 44 |
| Table 6 - Distance Unit Normalization Constants..... | 51 |
| Table 7 - Cardinal Directions in Radian Measure | 52 |
| Table 8 - Euclidean Parser Output Elements and VBA Test Inputs | 59 |
| Table 9 - Orthogonal Euclidean Traverse Parser Output Elements | 62 |
| Table 10 - Linear Referencing Parser Output Elements and VBA Test Inputs | 65 |
| Table 11 - Route Location Parser Output Elements and VBA Test Inputs | 68 |
| Table 12 - Parser Output Elements for Mixed Route and Euclidean Traverse | 70 |

LIST OF FIGURES

| | |
|--|----|
| Figure 1 - Figure/Ground Object Schemas | 17 |
| Figure 2 - Figure/Ground Relation of Junction Points on Road Lines | 18 |
| Figure 3 - Phrase Structure Tree | 25 |
| Figure 4 - MetaCarta GTS Basic Interface..... | 36 |
| Figure 5 - GEOLocate User Interface..... | 37 |
| Figure 6 - MaNIS User Interface | 38 |
| Figure 7 - Point Object and Interfaces | 53 |
| Figure 8 - IMSegmentation Interface Model | 55 |
| Figure 9 - User Interface for Euclidean Geoprocessing..... | 56 |
| Figure 10 - User Interface for Route Geoprocessing | 57 |
| Figure 11 - Euclidean Location Calculator Form Sample..... | 60 |
| Figure 12 - Euclidean Traverse from Place Name to Locality | 61 |
| Figure 13 - Euclidean Orthogonal Traverse | 64 |
| Figure 14 - Route Location Calculator Form Sample..... | 66 |
| Figure 15 - Route Traverse Location Calculation Geometry | 67 |
| Figure 16 - Route Traverse Showing Direction Determinants | 69 |
| Figure 17 - Mixed Euclidean and Route Traverse Legs..... | 72 |

LIST OF ABBREVIATIONS

| | |
|--------------|--|
| COGO | Coordinate Geometry |
| DEM | Digital Elevation Model |
| DTD | Document Type Definition |
| ESRI | Environmental Systems Research Institute |
| GIS | Geographic Information System |
| GNIS | Geographic Name Information System |
| IMS | Internet Mapping Service |
| LACM | Los Angeles County Museum |
| MaNIS | Mammal Networked Information System |
| NLP | Natural Language Processing |
| PLSS | Public Land Survey System |
| POB | Point of Beginning |
| USGS | United States Geological Survey |
| XML | Extensible Markup Language |

1 INTRODUCTION

Museum specimen collections have substantial potential as a source of useful data for spatial analysis and research but they are frequently poorly suited for study using current geographic information systems (GIS). A major problem inhibiting GIS studies of museum collection data arises when specimen provenance is recorded in the catalogue as a text description of the collection locality. Use of geographic analytical techniques and graphic display is substantially impaired or impossible even when catalogues are in a digital format (e.g. a relational database). Inclusion of geographic coordinate data with catalogue entries facilitates incorporation of collection data into GIS and provides improved opportunities for visualization, research and management. The following report presents background, organization, and functional requirements for desktop GIS solutions that address part of this problem. A portion of the Natural History Museum of Los Angeles County (LACM 2003) herpetology collection serves as source material.

The LACM is recognized internationally for its natural and cultural materials collections and for its research programs. Collections include nearly 35 million items and span 4.5 billion years of earth history. The Museum's specimen and document collections provide an educational showcase to the public as part of their mission "to inspire wonder, discovery and responsibility for our natural and cultural worlds" (LACM 2003). These collections also serve as primary source material supporting research in numerous disciplines including herpetology, the study of reptiles and amphibians.

As of early 2003, the Museum herpetology database listed over 143,000 collected herpetological voucher specimens and/or observations taken from around the world. A selection of over 33,000 records from Riverside and San Bernardino Counties in California was taken from the collection database for use with GIS development. Publication of these data through an internet mapping service (IMS) was desired and proposed as an alternative to printed media. The IMS would provide an outreach service to professionals with data and basic geographic analysis tools for use in resource management and research. This form of web publication can provide a user organization with: visualization, facilitated updates, increased dissemination, improved access by a definable user base, text and interactive graphic content, and display of user constructed spatial and non-spatial queries. None of this functionality is possible without first developing methods to expediently calculate and store coordinate data of useable and known precision for all collection localities within the extent of a study area. The Museum's herpetology collection manager requested support in this effort, from the University of Redlands through the Master of Science in Geographic Information Systems (MS-GIS) program.

Herpetology collection records are currently entered in an Access (Microsoft) Database. The majority of records have no spatial coordinate values associated with collection localities but other location information is present. Latitude and longitude are entered for less than 1 percent of the records and these to a precision of one arc second (roughly equivalent to 90 meters). Nearly all records include a terse text description of the collection locality and other useful geographic data: country, state, county and elevation. Ancillary location information is sometimes found in remarks field; this can include additional useful information (e.g. place names) but frequently is usually overly specific (under rock, under board, etc.).

The following chapter presents an overview of this project's background and objectives. It introduces principal parties involved, identifies the origins of the project and relates the history of its development. Project objectives are stated with requirements needed to meet them. Finally, the structure and presentation of the balance of this report are explained.

1.1 Project Background

The project presented here has a thorny development history. Some basic components of the project have persisted but development has followed a meandering course as the result of changes in client relations and proposed products.

1.1.1 Area of Study

The problem domain of this project concerns the automated conversion of text descriptions of location to spatial coordinate values. The automated methods used to accomplish this end are a type of geocoding process or *locator* (Zeiler 1999). Geocoding is a general descriptive term for converting any text based spatial reference to quantified spatial coordinates. Some common types of text-based locations include street addresses and legal descriptions using the public land survey system. *Geocoding services* (GIS conversion tools) are well established for street address to coordinate conversion and supported by a substantial spatial database of street grid reference.

The conversion process of location descriptions to coordinate values requires two domains of study; these are natural (human) language processing (NLP) and geometric calculation. Natural language processing is required to identify key words and phrases in the text; these convey spatial meaning. These geographic character strings (keywords and phrases) are tagged with their linguistic (spatial in this case) function in a process referred to in this report as geoparsing. Output of tagged strings from language processing provides the variables (or pointers to them) required to calculate geographic coordinate values for the locations

described in the text. The geometric calculation phase of the geocoding process is called geoprocessing in this report. Geoprocessing functions include assigning values to the text variables identified by geoparsing, performing geometric calculations, and storing the results. The goal of this study is the specification of a NLP output format and content most suitable for use in geoprocessing.

1.1.2 Client

Project work was originally started with the Los Angeles County Museum (LACM) as the client. Mr. Kent Beaman was the Herpetology Collections Manager and served as point of contact. LACM subsequently eliminated both the Herpetology Collection Manager and Curator of Herpetology positions in June 2003. LACM was not considered the client thereafter but Mr. Beaman continued to serve in a client capacity without direct support from LACM.

1.1.3 Prior Development and Start Up

Related project development was originally undertaken by several of the first group of MS-GIS students working under the direction of Dr. Glen Hyman (P. Carbajales and K. Johnson 2002) as part of class work for GIS-624 (Customizing GIS for the Web). Contact between members of the MS-GIS program and Kent Beaman of LACM was established and a prototype internet mapping service was developed. Project development was hampered by the initial lack of geographic coordinates; a grab sample of approximately 200 collection localities was manually geocoded. A lapse in continuity ensued with the departure of both Dr. Hyman from the faculty and the students from the program. Solutions provided by this effort were not provided to the client and neither the geocoding methodology nor internet functionality was available for later implementation and continued development.

The phase of project development reported here, started underway when the author was studying in the MS-GIS program under resident faculty member, Tarek Rashed. A telephone introduction and conversation in February 2003 was followed by a meeting with museum representatives Kent Beaman and Michael D. Wilcox. Priority user (business) needs were identified and included in a concept paper and proposal for development of an internet mapping service featuring the museum collection data.

1.1.4 Internet Mapping Service

Project development was originally directed towards creation of an internet mapping service for the LACM herpetology collection. The concept paper and the proposal were prepared followed by a use case analysis. The concept paper identified the problem of limited access and usability of the herpetological collection then presented web-based solutions to address the problem. A proposal to publish the herpetology collection data via an internet mapping

service was made. This was to provide professionals with database access augmented with basic geographic analysis and display functionality. A use case analysis study resulted in the identification of actors and use cases from both the system and user sides. Identified users included museum staff as well as outside researchers. Use cases were developed for both these groups at an intermediate level including: cataloguing functions including geoparsing and geoprocessing, reporting updates, spatial and non-spatial queries, and graphic and tabular display of results. Mapping service development halted when it became apparent that there was no expedient solution providing spatial coordinates for the 10,000+ unique collection localities.

1.1.5 Spatial Semantics

A course of independent study (see Appendix 1) was undertaken to develop more fully understanding in human cognition in spatial relationships and in methods for representing those relations as descriptive text. This study provided a fundamental understanding of semantics essential for successful parsing of spatial objects, entities, and relations. This understanding is necessary for reliable interpretation of meaning from locations' descriptions and is required prior to calculation of reliable numeric spatial values.

1.1.6 Natural Language Processing

Project development shifted from establishing methods for a user-side based web service to establishing methods for geocoding the basic data (spatial coordinates) required for any GIS implementation. Use case analysis during internet mapping service development identified two basic components of the geocoding process; these are geoparsing and geoprocessing. Both are required for conversion of text descriptions to geographic coordinates. Geoparsing is an automated process where spatial meaning is extracted from text descriptions of location. Geoprocessing uses the output of the geoparsing process as input and calculates the coordinate data for each location described.

Project development continued with a preliminary study in natural language processing of location descriptions with the processed (parsed) dataset as the primary project goal and deliverable. A natural language "toolkit" software module running in Python was identified as adequate to the task. Background study in software operations was undertaken but there was minimal progress before activity was halted. It was determined there was inadequate use of GIS technology involved with the proposed geoparsing application for consideration as an MS-GIS major individual project.

1.1.7 Geoprocessing

The final phase of project development began with the intention of creating a geoprocessing application. The application would use an input dataset that

simulates text parsed from location descriptions. The goal of the application was production of a new specimen collection spatial data table for the herpetology collection database. A software requirements specification was prepared for the application and development started. Additional training in coordinate geometry (COGO) and linear referencing functionality in ArcMap was undertaken and reported as an independent study. It became apparent however, that the original scope of application development was overly ambitious to allow a reasonable time for completion. Software development for geocoding was started and achieved functionality adequate to provide a basis by which geoprocessing input (parser output) requirements could be assessed.

1.2 Project Objectives

The objective of this project is identification of a geocoding data schema that specifies format and data content for a text file containing spatial data. The file is an intermediate data product in the geocoding process; it comprises the parsed output from natural language processing of location descriptions and serves as input for a geoprocessing application that calculates spatial coordinates. In defining this schema, the project seeks to identify spatial elements and relations that are present in natural language descriptions of location for a specific set of descriptive types. The project also seeks to specify a useful data structure that is well suited to convey spatial meaning for interpretation by geometric calculation application software. It is critical for efficiency and reliable precision that data content and structures are well formed. This will facilitate identification and generation of the appropriate geometric variables required by geoprocessing.

Several intermediate objectives were realized in order to acquire the primary goal; these were to:

- Research spatial semantics,
- Research and develop geoprocessing tools,
- Assess data requirements for geoprocessing,
- Gain an understanding of parsing,
- Develop the schema, and finally
- Demonstrate how geoprocessing would work with the schema in an XML format.

These intermediate objectives are explained below.

1.2.1 Spatial Semantics

Study of linguistic representation of spatial concepts was pursued to provide reliable basis for interpretation of meaning from locations' descriptions. The goal of the study was development of a framework to identify and extract the contributing elements of a spatial scene conveyed by descriptive text.

Accomplishment of the goal allowed for identification of the types and structures of spatial scenes in the collection database. A limited set of description types,

collectively referred to as a *traverse* (described in Section 2.1.4.2), was subsequently selected for development.

1.2.2 Geoprocessing Tools

Despite the fact that much collection data are not useable in a geographic information system, GIS is quite useful in the spatial enabling of collection data. Basic requirements needed for geocoding support in GIS were explored. Geometric representation of the conceptually generalized spatial objects, entities, and spatial relationships used in traverse location descriptions were identified. Section 4.1 provides a discussion of geometric representation. Subsequently, GIS functionality appropriate for use in calculations with that geometry was identified. This functionality was combined with methods that provided the desired geographic coordinate values.

1.2.3 Data Requirements for Geoprocessing

GIS functionality and methods require specific data for geoprocessing. One (or more) source of data for each geoprocessing function is identified. Some data are available from the collection database. Ancillary sources provide geometry, geographic references, and sources of coordinates.

1.2.4 Parsing

A fundamental knowledge of parsing technology is necessary. It provides a basis to assess the type of interpretation that NLP can make of a description of location and the type of output NLP can produce.

1.2.5 Data Schema

Geoprocessing data requirements are identified and presented. These guide development of the data schema (format, structure, and content) that predicates parser functionality. The schema is embodied in an extensible markup language (XML) format with a document type definition (DTD). Content is established to meet the geoprocessing requirements for traverse type descriptions.

1.2.6 Demonstration

The lack of a completed application or actual input data precludes a true demonstration of concepts. A virtual step-through demonstration is in order where all data content is explained in terms of how it supports and articulates with geoprocessing functions.

1.3 Report Structure

The balance of the report is divided into four chapters: problem definition, geoprocessing database, geoprocessing solutions and testing, and conclusions. Problem definition introduces geocoding locations based on text and explains its

major components. This is followed by presentation of software requirements for solving the problem and a recommended intermediate geocoding data product. The geoprocessing database chapter addresses data requirements for geocoding in terms of data types, sources, preparation, and schema. The solutions testing chapter provides an explanation of programming ArcObjects and Visual Basic for Applications-based functionality developed for the project. It relates code and form development for testing the viability of select descriptive data for calculation of coordinate values. The summary provides a recap of project successes and pitfalls with recommendations for future development.

2 PROBLEM DEFINITION

Spatial data stored in text formats is not readily available for use with mainstream commercial geographic information technology. Substantial human effort is required to quantify locations recorded as text and *spatially enable* the associated data. Inadequate effort has been applied to deal with the enormous volume of spatial information stored in text form. The result is that the majority of meaningful historic spatial data is unusable for GIS. This is unfortunate because much historic data is pertinent to current research needs; museum collections' potential for contributing to biodiversity studies provides one example.

Museum collection data serve as the focus of development in this report and represent substantial information. Museum biological collection holdings are estimated in excess of one billion records worldwide (Beaman et al. 2004). The descriptions of location (or provenance) provided with museum collections provide the basis for a determination of geographic coordinate values.

The following chapter identifies and explains:

- The nature of museum collection data,
- Methods for extracting spatial information from text,
- Methods for performing geometric calculations in GIS, and
- Methods for converting text information to spatial coordinates.

2.1 Museum Collection Data

Natural history museums curate and exhibit cultural, botanical, and zoological specimens derived from contemporary and paleontological contexts. Collections of biological voucher specimens and observations serve to provide data for research as well as educate the museum-going public. Most collected specimens rely on primary documentation that specifies the location where the specimen was taken. Primary documentation is typically text hand entered on paper media; types include catalogues of voucher specimens, field observations, field collection notes, logs, and maps. Some of the spatially relevant data in these records is in text format and may be directly transcribed to digital format. These spatial data frequently include descriptive text of the collection locality. Use of a descriptive location is an expedient method that allows identification of sites that are frequently in rural settings and well away from established street address grids.

The location descriptions vary significantly in style and content. This variation and other factors lead to uncertainty in levels of precision. The following section examines:

- contemporary needs for museum collection spatial data,

- where spatial data resides in the collection
- the nature of the descriptions of location,
- how descriptions can be classified, and
- uncertainty inherent in descriptions of location.

2.1.1 Contemporary Context

Spatial analysis afforded through GIS is a recognized and growing research tool for many disciplines. The study of biological diversity is among them and the demand for spatially enabled biological data is increasing. Beaman, Wieczorek, and Blum (2004) recently asserted the situation as this:

Managing finite natural resources is among the greatest challenges in this century facing biologists, information scientists, managers, economists, and policy-makers. Meeting this challenge is critical for sustaining human growth and prosperity, maintaining economic stability, and improving the quality of life for all species. Knowledge of biological diversity is fundamental to natural resource management, and the urgency for this knowledge ever increases as we convert the final tracts of the natural landscape into human-managed systems. Much of this knowledge remains locked in physical archives and on library shelves. In order for biodiversity management to keep pace with the rate of resource exploitation, digital access to biological diversity information is imperative in this decade.

Museum collections hold great promise as a source of data but the majority of spatial information from museum collections is in text format. The enormous volume of this information calls for a collaborative effort to enable it spatially. The Internet provides a venue for participants to share common resources and contribute knowledge, data, and effort. GIS may serve not only for display and for analysis but also contribute to a data conversion effort.

2.1.2 Needs Analysis

Ultimately, the needs of a geocoding service are based on end users' data needs but some geocoding needs are based more on operational needs. Accordingly, if the geocoding service is to be successful, it must meet needs for users, data, and software. Users include the GIS specialists operating the geocoding service and users of the data produced by the geocoding service. User needs are presented as use cases for a GIS for biological collection data in Table 1.

When the geocoding service is partitioned into its primary functional components, the geoprocessing software can be shown as a user of the geoparsing output. Regarding software needs, both the geoparsing and

geoprocessing software certainly have specific needs in terms of both input and output data requirements.

Table 1 - Biological Collection Data Functional Use Cases

| Actor | Use Case | Description |
|-------------------------------|---------------------------------------|--|
| Data User | Query by Taxonomy | GIS is queried for and returns all instances for specific taxa. |
| Data User | Query by Geography | GIS is queried for and returns records of collections taken within (or outside) a specified area. |
| Data User | Query by Collection Date | GIS is queried for and returns records for specimens collected during a specified time interval. |
| Data User | Query by Collector | Herpetology GIS is queried for and reports all species' records collected by specified individual/group. |
| Data User | Query by Catalogue number | Herpetology GIS is queried for individual or specific range of collection numbers and returns those records. |
| Data User | Refine Query | Query performed on results of previous query. |
| Data User | Display Photograph of Queried Record | Representative photograph(s) of species selected by a locality query are displayed. |
| Data User | Tabular Display of Query Results | GIS query result is displayed in tabular form. |
| Data User | Cartographic Display of Query Results | GIS query result is displayed as a map. |
| Data User | Query by Precision of Location | Precision of collection location's coordinate values are classified as (increasing precision) regional, sub-regional, locality, or site. |
| Data User, Collection Manager | Database Error Reporting | Errors are identified by users and reported to the collection manager. |
| Collection Manager | Database Updates | Existing records are edited in the herpetology collection and spatial database |

| Actor | Use Case | Description |
|------------------------------------|------------------------------------|--|
| Collection Manager, GIS Specialist | Database Location Error Correction | Reported location errors are reported by the Collection Manager to GIS Specialist for verification. |
| Collection Manager | Appending New Records to Database | The museum will acquire additional herpetology collections. Records for these additions will be added to database. |
| Data User | Show all collection localities | System returns table or shows map of all localities listed in database. |
| GIS Specialist | Geocoding Collection Records | Text comprising descriptions of collection locations, is automatically interpreted and converted to numeric geographic coordinate point data |

2.1.3 Location Source Data

Primary spatial data for museum collections typically cannot be used with the technology employed with contemporary geographic information systems. This is because primary source materials for collection locations have been predominately hand written until recent times. These sources include but are not limited to: specimen labels, field notes, map plots, collection logs, other manually recorded notations, sketches, and photographs. Most of these data sources are available for conversion to digital format. Many of these sources include spatial information appropriate for interpretation and entry into a relational database. Entry of primary source information into database format may require some interpretation and in any event, this transcription predicates that the database serve as a secondary source. This is not necessarily deleterious with adequate safeguards and is a necessary step towards display and analysis.

The LACM herpetology collection database serves as an example of the dimensions and scale of spatio-temporal data included with a museum's biological specimen collection. The collection database includes many related tables but the majority of spatially usable data are available from a single table (HerpColl). This table lists voucher specimens and observations from only two California Counties but includes over 33,000 records. One third of the forty-four fields that comprise the table include spatial or temporal references; the vast majority of these are relevant to spatial display and analyses. The most relevant fields are summarized below in Table 2 and include: an enumeration of collection localities, government administrative units where the collection was

made (country, state, and county), a text description of the collection locality, latitude, longitude, and elevation of the collection locality, date the collection was made, the collector, and collection specific remarks.

Table 2 - Collection Database Fields With Spatially Relevant Content

| Field Name | Description | Utility |
|-----------------------------------|--|--|
| Localitykey | An enumeration of specimen collection localities. | Listing of non-unique collection localities. Helps identify collection events. |
| County | Name of county where specimen was collected. | Provide medium scale horizontal constraint for collection location. |
| State | Name of state where specimen was collected. | Provide medium-small scale horizontal constraint for collection location. |
| Country | Name of nation where specimen was collected. | Provide small scale horizontal constraint for collection location. |
| Locality | Text description of the collection locality. | Provides [variable] relatively fine grain written portrayal of spatial scene for collection locality. |
| Latitude (Deg, Min, Sec, N/S) | Geographic coordinate value of angular departure from equator at locality. (Provided in separate fields). | Combined with Longitude to provide relatively fine grain metric values representing a collection locality. |
| Longitude (Deg, Min, Sec, E/W) | Geographic coordinate value of angular departure from prime meridian at locality. (Provided in separate fields). | Combined with Latitude to provide relatively fine grain metric values representing a collection locality. |
| Elevation | Elevation above mean sea level at collection locality. | Provides medium-large horizontal constraint for collection location. |

| Field Name | Description | Utility |
|--------------|--|--|
| Date | Specific date of collection for specimen listed including month, day, and year. | Provides consistent daily temporal reference for any set of collection events and geographic reference appropriate at time of collection. |
| Partial Date | Approximate date of collection; varies from range of two days to a year. | Provides variable precision, temporal reference for any set of collection events and geographic reference appropriate at time of collection. |
| Collector | Name of individual or organization that prepared primary specimen collection record. | Can be linked to any existing event gazetteer to specify geographic constraints. |
| Remarks | Various spatial and non-spatial attributes of collected specimen. | Occasional ultra fine grain location information. |

The collection locality's enumeration field is named *Localitykey*. This would be useful reference but the field's name is misleading and the locality descriptions listed are not spatially unique. The enumeration lacks definition; it is impossible to discern a uniform method for assigning numbers to localities. This is apparent by comparing localities 7779 and 7924. Both are described identically in the table as "Mill Canyon, Headwaters of San Gorgorio R. above Banning" and there are greater problems than the misspelling in the river's (Gorgonio) name. Inspection of the records reveals that two different elevations are listed for the fifteen collections made by two individuals on three separate dates over a period of six years.

The administrative-based spatial units where the collection was taken provide useful reference. Three levels of increasing scale are country, state, and county. Consistently completed, these fields already permit spatial analysis of the collection at local, state, regional, or national levels without additional analysis and data entry. Additionally, geometric representations for these areas that is suitable for use in a GIS, is already available. The extent of a county can be used as a topology check for calculation results representing a collection locality's coordinates. The resulting point's coordinates must be within the extent of the county of collection.

The text description of the collection locality serves as the focus of the project. It provides most of the fine grain spatial resolution in defining the collection's location. The location is typically conveyed in a figure/ground relationship where the [collection] figure is a point on more complex ground geometry (multi-point, line, or area). The ground geometries comprise familiar named features such as populated places, roads, and stream courses and other natural landmark features. Location descriptions are explored more fully in the section 2.1.4 below.

Latitude and longitude fields in the collection table provide the most useful spatial location information. Precision is at the one arc second level and represents east-west and north-south distance ranges of 85 and 101 feet respectively, at the latitude of the study area. It is unfortunate that geographic coordinates have been entered for fewer than two percent of the 10,668 enumerated localities recorded within two counties. No metadata exist for the recorded spatial coordinates so their method of determination, accuracy, and suitability for use in any particular analysis is not available or is unknown. The utility of existing coordinate data is limited by the relatively few records that include values and by reliability that cannot be easily determined.

The recorded elevation of the collection locality provides additional numeric spatial data that may be relatively fine grained. When used with an identifiable extent described in the text, the elevation effectively reduces the extent of a collection location to the previously documented hypsographic alignment(s) for that contour. Elevation can be useful in error checking. Elevation values recorded for the collection locality are matched against previously documented values for the calculated location. Ancillary elevation data of documented accuracy is readily available from the USGS in digital elevation model (DEM) format. Significant disparity between values recorded with the collection and the documented source identify a questionable collection location. The collection's provenance classification may be re-assigned to an appropriate level.

The date a collection was made is useful in considering change or lack thereof, in a given locality's spatial attributes over time. Increasing change in time may increase uncertainty of spatial attributes associated with a locality. See Section 2.1.5 for a discussion of uncertainty due to temporal change.

The recorded collector or observer at a collection locality may provide useful horizontal and temporal constraints for a specimen. Ancillary records such as field notes and maps may define the extent of fieldwork and thereby the collections made during that work. Computer files that capture these geometries could provide error checking of geographic coordinates calculated for the

collections. Error checking could take the form of a topology check like that suggested for political administrative units (above).

The remarks field frequently holds information providing additional spatial detail or constraint. It frequently holds redundant information or data that would be more appropriate in a separate field (e.g. elevation). Corrective actions are required to improve consistency of spatial content. Recommendations for improvements in the collection database are proposed in Section 5.

2.1.4 Descriptive Elements and Methods

Museum collections require adequate spatial resolution in their recorded provenance because meaning of a collected item and its utility for use in spatial analysis depreciate significantly without it. National, state, and county scale resolutions do not typically provide this and collection records require the improved definition provided by text describing the collection locality. Text achieves finer resolution by relating the locality to identifiable geographic features as in “near Metcalf Bay, S side Big Bear Lake”. The following discussion identifies and organizes types of location descriptions and the elements of descriptive text that can be used for locating collection specimens in their geographic context.

Records in the museum collection database display no consistent manner of describing collection localities. This diversity results from numerous individuals recording documentation for more than a century. Their descriptions vary significantly in geographic content and methods of relating the locality to geographic features. For example, it has been noted that collection localities have become more frequently related to roadways over the course of the past century as travel by automobile increased (Bryant 2004). An understanding of these descriptive variables is necessary to develop ways to extract useful meaning from the text. Consideration of the description in the context of a cognitive *spatial scene* (described immediately below) provides a useful framework for examining them.

2.1.4.1 Spatial Components of Descriptive Text

It is critical to convey effectively, a spatial scene that specifies the relationship between a collection locality and its surrounding geography. The spatial scene is a figure/ground relation with the collection locality as the figure (primary object) and some part(s) of the surrounding geography as the ground (secondary object). Methods of representing this relationship have been discussed for decades (Talmy 1983, Fillmore 1968) and are illustrated in Figure 1. The locality figure is perceived as relatively new within a geographic ground framework that is

already established in memory. The figure can be mobile as compared with relatively fixed secondary (and sometimes tertiary) ground objects of known spatial characteristics. The figure is perceived as a simple geometry (usually a point) against the larger ground object represented by geometry that is more complex.

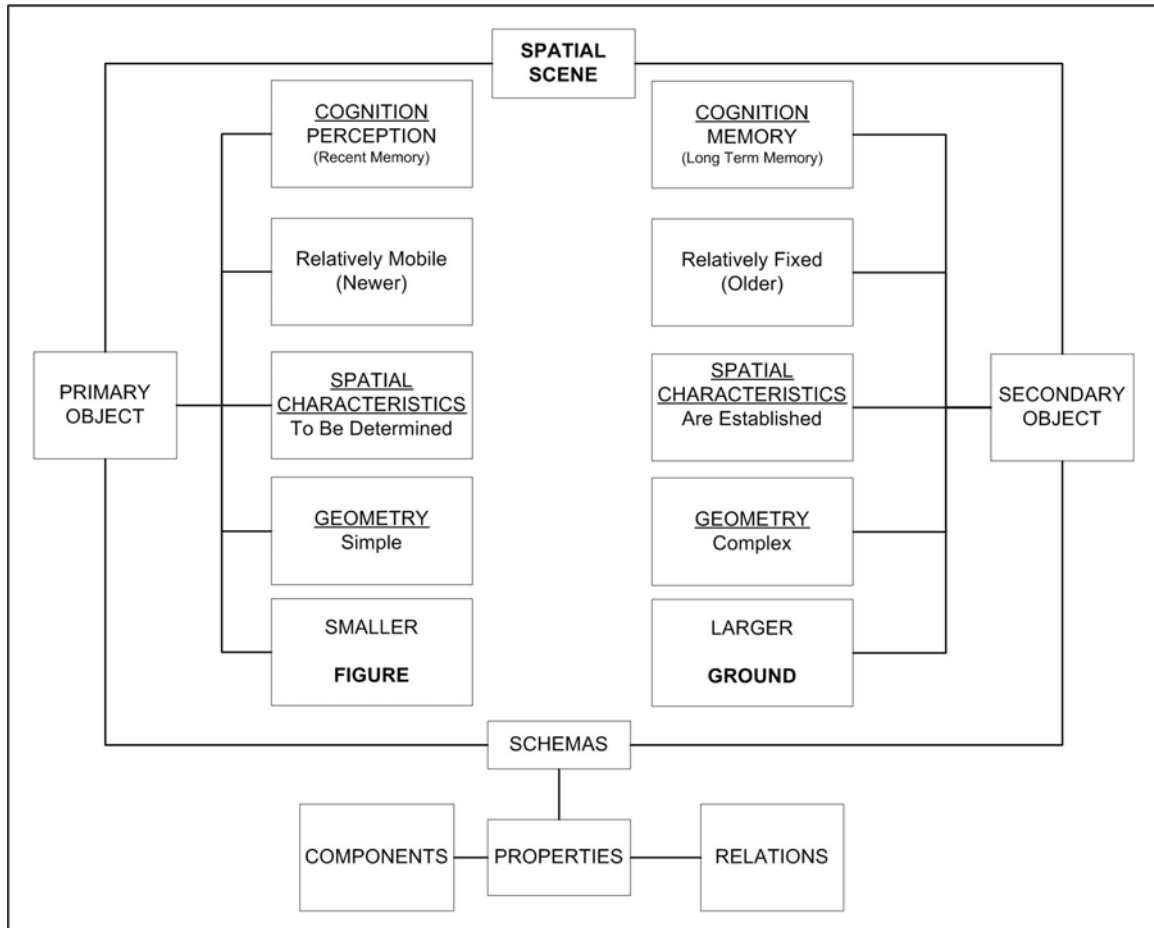


Figure 1 - Figure/Ground Object Schemas

The spatial relationships and object geometries in figure/ground relations are useful in interpreting location descriptions. Figure 2 provides an example of the figure/ground relationship between points representing junctions and lines representing highways.

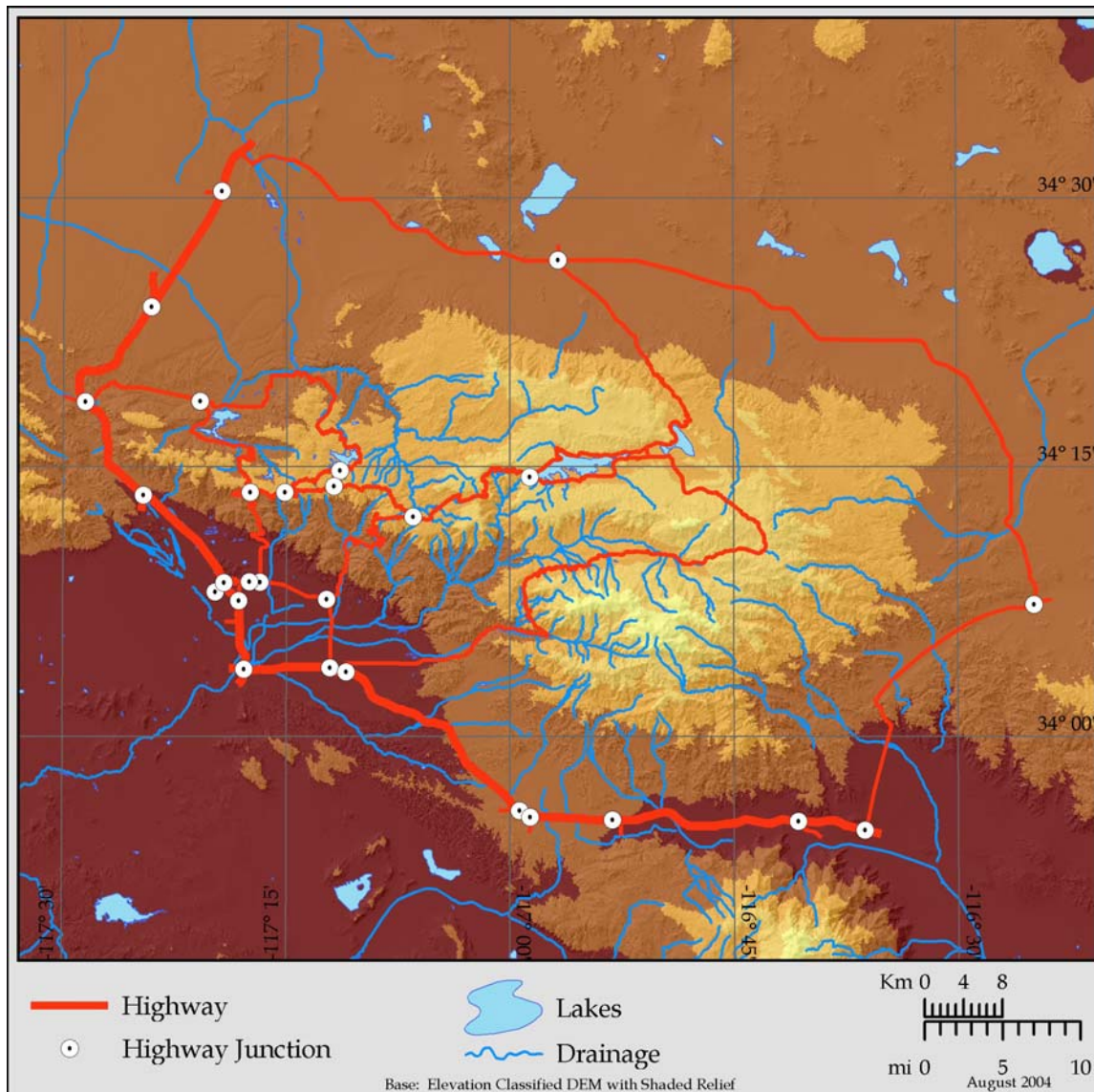


Figure 2 - Figure/Ground Relation of Junction Points on Road Lines

Figure geometry is variable but it is simple relative to the ground geometry against which it is presented. It can be perceived as a point or a line and for either case, fixed or moving. A figure perceived as a moving point is somewhat akin to a static figure perceived as a fixed line. Likewise, a line moving perpendicular to its axis simulates an area. The significant difference between static and moving figures is the temporal element. The simpler moving geometry simulates the static complex geometry as a set but requires the passing of time while the static complex geometry may comprise the same set at a given moment in time. *Ground geometry* is always more complex than the figure geometry presented with it. It can portray a collection of points, lines, bounded planes, and discrete volumes.

Spatial relations between figure and ground can indicate *relative position* and *relative orientation* within a spatial scene. The description “1 mi. outside Palm Spr. In Panorama Land Project” indicates two topological relationships for the collection locality; it is outside Palm Springs and inside the Panorama Land Project. The description “Colorado River, across from Topock, Arizona” specifies a collection locality by the Colorado River. It is already established by other collection data fields that it is in San Bernardino County, California but provides its position relative (across the river) to a tertiary feature (Topock). Spatial relations are also accomplished with descriptive text that identifies a *biased reference* between the figure and ground objects. Biased referencing specifies attributes of the ground object and relates the position of the figure to these. Attributes include *non-symmetrical directedness* and *earth-based directedness*. Non-symmetrical directedness specifies unique or identifiable elements of spatial direction such as head, foot, front, and back. An example of this is “Little San Bernardino Mtns, head of Berdoo Canyon” where head specifies the uppermost part of Berdoo Canyon. The collection locality is an area that will be generalized to a point. Earth based directedness is used to identify directional positioning between figure and ground based on vertical and/or horizontal directions relative to the earth. The figure can be described as above, below, or at some cardinal direction relative to the ground figure. This method of spatial description is very common and examples of earth-based directedness such as “2 mi above fish hatchery in whitewater canyon” are found throughout the herpetology locality descriptions. Note that horizontal directedness is inferred by the [upstream] vertical directedness specified in the description.

2.1.4.2 Classification of Descriptions

Descriptions of location vary significantly in many respects. Most are sentence fragments and grammatically incorrect yet they still convey meaning to the reader. The clarity of spatial definition they provide varies over a gradient from very fuzzy to crisp. The gradient has been classified (Bryant 2004) into four useful categories: regional, sub-regional, locality, and site (from fuzzy to crisp). The herpetology collection table comprises locality descriptions that rely on street addresses, the public land survey system (PLSS), geographic place names, roads, distances, and directions. These objects and entities are generalized in appropriate figure/ground geometries and presented in various relations. A street address places a figure point on a ground network of linear features. Linear features are essential in many descriptions; roads and named drainage features (e.g. rivers) are the most common. Geographic (or place) names are locations that can serve as either figure or ground reference depending on usage. A named place included in the description of a specimen collection locality can be conceptually generalized as a point or may retain conceptualization as an area. Both these possibilities are realized in the terse description, “Covington

Flat, Joshua Tree Nat'l Monument". They may also be generalized to one or more points that serve as ground reference for the figure as in "Camp O-Ongo, Btw. Running Springs & Sky Forest, San Bernadino Mts."

The focus of this study is a method of location description referred to here as a *traverse*. The name traverse is adapted from an instrument survey procedure where the coordinates for an unknown point or a sequence of unknown points are calculated using direction and distance measurements taken from a point with known coordinates. The known point is referred to as the *Point of Beginning* (POB). The surveyor measures distance and direction to an unknown point from the POB, calculates the new coordinates, moves to the newly identified point, and measures the next unknown point continuing along the series. This method is similar to many descriptions of location with spatial content that includes a starting point with one or more iterations of distance and direction. Samples of locality descriptions from the herpetology collection table are presented below in Table 3; these include several varieties of the traverse type (in bold).

Table 3 - Types of Locality Descriptions

| Description Type | Descriptive Text (Figure/Ground Relation) |
|---|---|
| Place Name | <i>Desert Center</i> (Point on Bounded Plane) |
| Place Names | <i>Camp O-ongo, between Skyforest and Running Springs</i> (Point on Multi-Point) |
| Place Name with Buffer | <i>0.1 mi. from Lake Filmore</i> (Line on Bounded Plane) |
| Place Name, Orthogonal Traverse | <i>10 mi E, 7.5 mi S Cima</i> (Point on Bounded Plane) |
| Place Name and Route, Traverse | <i>Lucerne Valley, 6 mi S on Crystal Cr. Rd.</i> (Point on Line) |
| Place Names and Route | <i>Cajon Campground, Hwy 395 in Cajon Pass</i> (Point on Line) |
| Intersecting Routes, Traverse | <i>3 mi. N. Hwy 60 on Whitewater Cyn Rd.</i> (Point on Line) |
| Street Address, Place Name and Route, Traverse | <i>01099-Varner Road, 5 mi. W. Thousand Palms</i> (Point on Line) |
| Water Course at Place Name | <i>Cajon Wash at Devore</i> (Line within Area) |
| Water Course at Route | <i>Cajon-Hesperia Rd. Bridge, W. fork of Mojave River</i> (Point on Line) |

| Description Type | Descriptive Text (Figure/Ground Relation) |
|--|---|
| Place Names, Public Land Survey System | <i>Agua Caliente Indian Reservation (T5S, R4E, Sec 3, SE corner), Andreas Canyon (Point on Bounded Plane)</i> |
| Combinations of Above | <i>4.4 mi. N on Whitewater Cyn Rd. from Interstate Hwy 10, 100 yds. E of rd at edge of stream (Point near Line)</i> |

2.1.5 Uncertainty in Data

Descriptive data seldom specify a location with absolute precision and certainty. This notion introduces variable levels of uncertainty for calculated results that are based on spatial information extracted from the descriptions. Uncertainty must be identified, classified, and quantified in order that results may be used appropriately for spatial analysis or display purposes. The following section introduces and discusses types and sources of uncertainty inherent in the descriptions themselves and in geographic reference data for described places.

2.1.5.1 Descriptive Data

As discussed in Section 2.1.4.2 above, location descriptions convey the location of the collection locality relative to known geographic features. An analogy to a journey is used here for considering traverse type location descriptions. Named places or intersecting geographic features frequently designate the beginning of the journey with the collection locality at the end of the journey. The place where one begins the journey is specified in the description along with the distance, direction, and sometimes, the path (or route) to follow. The locality is at the end of the journey but the certainty of its spatial coordinates relies on the geometric and relational components that are provided in the spatial scene's description. The definition of the start point and the precision of direction and distance values are most important.

Problems encountered with location descriptions include: the amount of detail that is provided with the description, the physical extent of location(s) referenced by the description, variable and unstated levels of precision for measurements, and use of secondary source material. The level of detail provided in the description is not proportional to the quantity of text but rather the horizontal extent of identifiable reference features and relative precision provided by the stated unit(s) of measure. The information contained in database descriptions of location is secondary source material because it is manually transcribed from primary (or other secondary) sources. This process can introduce error not present in the primary source. Additionally, multiple instances of the same place

name used at different locations may be returned from queries of reference sources.

2.1.5.2 Geographic Names

Geographic feature names may prove problematic when multiple features share a common name; the *homonym problem*. For example, there are 975 places named *Beaver Creek* or some variation, in United States. *Site precision* or *locality precision* levels of spatial resolution are desired from the description. The locality description, "*Pachalka Spr., Clark Mt.*", uses two place names to specify the location. Clark Mountain covers most of the extent of one township (36 mi.²) and provides *sub-regional precision* level spatial resolution. Clark Mountain does however, provide good reference for the spring as it is inferred that the smaller spring is *near* the larger mountain. This would be useful for differentiating between springs if several are named Pachalka within the same county. Pachalka Spring provides *site quality* spatial resolution; it specifies a named natural feature (and perceived surrounding environment) that is unlikely to exceed more than a hectare or two. This extent approaches the one arc second precision of standard reference sources like the USGS's GNIS gazetteer. The uncertainty in the description, *Slate Range Mtns, between Death Valley and Borax Lake.*, is substantial; one feature of regional level spatial resolution is located by placing it between two adjacent features of regional extent. This location description requires that a point represent a region that may exceed 100 square miles.

2.1.5.3 Spatial Relation Variables

Descriptions include variable and unstated levels of precision for measurements of distance and direction. Distance measures are specified through a mix of English and metric units ranging in length from a foot to a mile and meter to kilometer. Measure quantities used with distances also vary widely in precision. Section 4.1.3.1 provides a discussion of normalization procedures for precision in variable distance units and quantities.

Distances described in the herpetology collection range from 2.5 yards to 63 miles. These are seldom recorded with precision greater than one place to the right of the decimal for any unit of measure. It is difficult to assess consistently, the level of precision for distance measures. It appears that distance measures are a mix of measured and estimated values. Distances measured with automotive odometers are common and distinct; their descriptions refer to traveled routes and similar to most vehicle odometers, distances are recorded in miles with one place to the right of the decimal. Frequent estimation is apparent in other values listed. Distances in units of yards are illustrative of this observation. Distance values less than 100 are evenly divisible by 5 (usually 10)

with few exceptions. Those values between 100 and 1000 are evenly divisible by 50 and values greater than 1000 are divisible by 100. It is apparent these values are not measured but estimates of perceived distance or roughly scaled from a base map.

Direction measure is described almost entirely by use of cardinal directions of unspecified declination. Horizontal direction is seldom recorded in units of angular measure and is typically specified in cardinal directions with common usage of first and second inter-cardinal directions. This translates to angular measure with a precision no greater than 22.5 degrees. There is no mention as to whether directions are true or magnetic; it is likely both are present. This element of uncertainty by itself has the potential to skew second intercardinal directions one full interval (e.g. East Northeast to East) as the difference between magnetic and true north in the project study area (13 degrees – 20 minutes) is greater than the half the difference (11 degrees – 15 minutes) between the second intercardinal directions.

2.1.5.4 Location Variation with Time

Uncertainty due to temporal variation is significant for those collections with considerable time depth. The herpetology database includes collection records dating from the late 19th Century through the early 21st Century. Significant change can occur in the name given to a place, the extent of the place, or its location. The greatest temporally induced spatial problem relates to changes in named places. Named places referenced by location descriptions can change in name or location over time. Consider the number of streets named for Dr. Martin Luther King, Jr. before 1968 and since that time. More pertinent to the collection database is the change of Joshua Tree National Monument to Joshua Tree National Park. Significant changes in the extent of named places introduce uncertainty as well. The locality description, *0.25 mi W Palm Springs*, written in 1918 almost certainly refers to a location within the current city limits given population growth of that community over the past 86 years. Natural landmarks typically move on a geological time frame but cultural features can and do move in short order. The physical town site of Ruth, Nevada (including buildings) has been moved twice in the past century to accommodate expansion of mining areas. It is currently more than two miles from its previous location. Some places and routes cease to exist. Several neighboring communities of Ruth, Nevada are now within the operations area of the mine and no longer exist. In similar manner, the alignment of portions of the Interstate highway system covered or eliminated previously constructed and designated travel routes.

2.1.6 Summary

The preceding series of discussions has introduced museum collections as a source of spatial data in descriptive text format. Contemporary research issues such as biodiversity studies drive user needs that cannot be met until the collection data are spatially enabled. It has been shown how the uncertain nature of historic location descriptions provides a challenge to the derivation of reliable geographic coordinates. The next section introduces the methods required to extract the spatially descriptive text elements that provide meaning in location descriptions. These methods break down descriptions into discrete strings of text that constitute recognizable spatial objects, entities, and relations.

2.2 Geoparsing

Geoparsing is a type of parsing operation that is specialized for spatial and geographic information. The geoparsing process is an automated linguistic analysis method that uses text for input. It identifies the spatial semantics in descriptions of location, segregates individual semantic components into strings of text, and marks them according to their spatial function. The identified spatial components are tagged according to their semantic function and stored in a standardized output file structure. Computer interpretation of natural history collection localities has been a GIS related study interest for some time (McGranaghan 1989).

Parsing descriptive text is an essential component of the geocoding process. It interprets meaning in the location description provided in the database and converts it to a format that can be used as input for spatial processing. Geoparsing requires *natural language processing* (NLP) software that can identify and interpret grammatical elements of natural language. It also requires the natural language is recorded as text in digital format. NLP requires a reference dictionary to provide a basis for identification of grammar and syntax. Software and data requirements for geoparsing are discussed below.

2.2.1 Geoparsing Software Requirements

Interpretation of written human language that is stored in digital text format is provided by NLP software. Geoparsing is an NLP configuration specifically for interpretation of key words and phrases in text that are identifiable as geographic objects or entities or that comprise spatial relationships between those objects and entities. In this process, series of text characters (strings of letters, words, and phrases) that represent spatial information are identified and the relationships between them are interpreted. An example of the linguistic relationships that a parser must recognize and operate on is diagrammatically presented in Figure 3 as a phrase structure tree. The sentence “S” is divided into a noun phrase “NP” and a verb phrase “VP”. The noun phrase comprises a

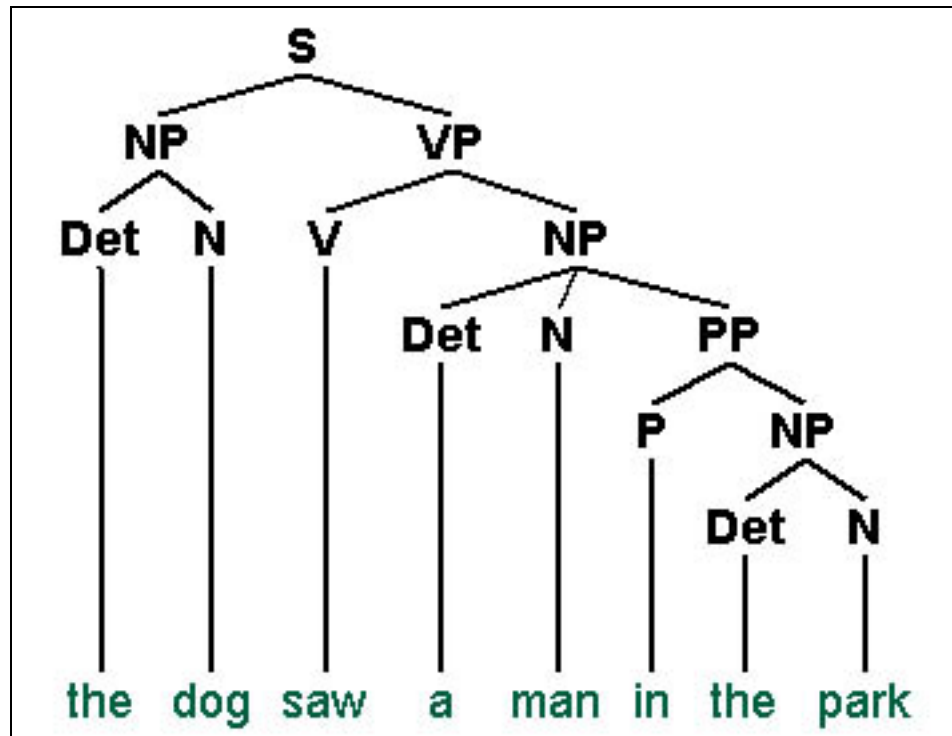


Figure 3 - Phrase Structure Tree

(Loper 2003)

determinate (article) “**Det**” and a noun “**N**”. The verb phrase comprises a verb “**V**” and another noun phrase. This noun phrase has a determinate, a noun, and a prepositional phrase “**PP**”. The prepositional phrase describes the location of “a man” and his topological relationship (“in”) with “the park”. Note that for most terse location descriptions in the collection database, the description is linguistically poorly formed (not a complete sentence). The phrase “The collection was taken at” could be attached to many descriptions and provide the missing noun phrase and portions of the verb phrase. The location description typically provides the elements included in the prepositional phrase of a well-formed sentence.

The geoparser produces a list that includes spatially descriptive strings and their associated spatial function. Additional information particular to each collection is copied directly from the collection record in the database and is included with the strings and functions. This list is structured according to an established schema that provides a basis for interpretation of the spatial semantics of the original location description. Appendix 2 presents the XML format developed for this project.

An operational geoparsing application may be considered a type of *expert system* because it is a domain specific application of NLP. Expert systems are artificial intelligence systems that provide for modeling at higher levels of abstraction than are available through traditional procedural programming languages. These systems emulate human reasoning but are typically limited to a single domain of knowledge; geography in this instance. They operate on a “rule based” method of interpreting information where certain output is specified for certain input conditions; an if/then conditional operation. Patterns are identified in the input data and are matched against rules to find which rule(s) apply. Rules for a geoparsing application are provided in a *grammar* that specifies the set of phrase structures that are considered well-formed (syntax). The list of strings that include these phrase structures for geographic places and relations can be assembled from sources such as gazetteers and spatial lexicons. Terms of geography provided by gazetteer data are discussed immediately below. Terms of spatial relations provided in the English language are explored in Section 2.2.4.

2.2.2 Geoparsing Data Requirements

Different types of data are required for basic parsing; these include a source of text that provides input and reference data against which the input is compared. Collection locality descriptions from the museum collection database provide the source of text input. A *grammar* is used by parsing software to provide the reference against which strings from the locality descriptions are compared. The grammar includes a lexicon which identifies the collection of lexical (i.e. noun, adjective, verb, conjunction, etc.) and other usage categories that are permissible for words in a language. Other categories can include geographic feature information (e.g. populated place). The lexicon specifies permissible terms (strings) of geography and spatial relations. Terms of geography are provided by gazetteer entries that list the features or entities that are physically situated within an area of interest. The terms of spatial relations are identifiable as the sets of strings comprising distance, direction, and relative position.

As noted above, the descriptions of location entered in collection database records are fragmentary and grammatically incorrect. Input of these descriptions into a NLP application could generate more errors than usable output if measures are not taken to accommodate this particular type of input. The fragmentary condition issue can be dealt with relatively easily by concatenating (adding) the missing text string (phrase) to the description prior to processing. The missing phrase is either “The collection was made” or “The collection was made at” in nearly every instance. Several examples from the herpetology collection are provided in Table 4 to illustrate this point.

Table 4 - Phrase Fragment Repairs

| Repair Phrase | Location Description |
|----------------------------|--|
| The collection was made | 6.5 mi S Banning on Hwy. 243, |
| | ca. 2 mi. S Kelso on Kelbaker Rd., |
| The collection was made at | Allured Mine, 9 mi N Cima, |
| | mouth of Palm Canyon, San Jacinto Mts. |

The examples provided here in Table 4 and elsewhere are terse descriptions with poorly formed grammar yet they do not represent the most contorted natural language usage likely to be encountered. Descriptions commonly place the POB placed first and follow it with a comma and then the Euclidean or route traverse directions as is apparent in the descriptions, *Desert Center, 5 miles east of* and *Elsinore, 2 mi. W of town*. Any NLP application used to parse spatial semantics from these requires development of a custom grammar in order to recognize these and other formations as syntactically valid and interpret them according to form.

2.2.3 Gazetteer as Geographic Lexicon

Gazetteers list, locate, classify, and describe named geographic features for a specified extent or area of interest. These features' names must be included in the grammar of geoparsing software when working with descriptions of location within the area of interest. Typically, the geographic features included in gazetteers, are named places for many categories of natural and cultural objects and entities. Appendix 3 lists feature types provided by the USGS in their digital gazetteer, the *Geographic Name Information System* (GNIS). Gazetteer information provides support for geoparsing operations; it specifies valid geographic names with their associated spatial and non-spatial attributes. Digital gazetteers include spatial information (location) as part of listed features' content. This spatial information is not necessary for generating a domain specific grammar that supports geoparsing. It will however, be required during geoprocessing when a feature's geometry contributes to a description of location.

Custom gazetteers can be developed to reflect geographic interests for specific domains of knowledge such as the geography of biological collections. Gazetteers of domain specific knowledge should include specialized geography that may not appear in a standard gazetteer. Some of these specialized geographic listings refer to entities that may lack physical demarcation like project-specific fieldwork collection areas; these are particularly germane to the event gazetteers discussed below. Other gazetteers may include salient features that provide field workers with convenient physical anchors for descriptive text. Transportation features like roads, railroads, and pipelines are substantial but are unlikely to be listed in standard gazetteers.

Development of *event gazetteers* that relate sets of geography by temporal attribute has been suggested (Beaman et al. 2004). An *event*-based gazetteer is well suited for use with existing biological collection databases. These databases typically include fields that can be related to attributes identified as useful in specifying historical events. Those identified attributes include: names, owners, descriptions, location, time, type, and actors (Allen 2004). This will assist spatial enabling of collection data by linking collectors and collection dates with delimited areas. Any design for an event-based gazetteer must reflect the way user thinks about related collections. It may also prove useful to design these gazetteers with interleaved event and location hierarchies.

Construction of event gazetteers can be accomplished relatively quickly for coarse grained geography such as the administrative unit (county, National Forest) where collections were made. Significantly greater effort is required for development of extents for specific collection projects. Once accomplished, the inventory of allowable geographic extents for temporal constraints associated with specimen collection events can help with geoprocessing. Error checking of results can be matched against appropriate extents. It can also act as a spatial filter and reduce or eliminate multiple instances of matching geographic names (the *homonym problem*) selected from general gazetteers.

Descriptions of named geographic features provide spatial location but typically as a point. Some digital gazetteers provide “footprint” or minimum bounding box corner coordinates. There are inherent problems with spatially summarizing geographic features in this manner. A point may be appropriate geometry to represent features such as springs or road junctions. It is more difficult to represent extensive areas such as mountain ranges or cities with a single point. Significant uncertainty is introduced in the spatial location; this varies as a function of feature area and scale of analysis. The use of the rectangular area provided by the footprint may initially appear attractive but may over represent the extent of some features. The Colorado River provides an example; it occupies a relatively small area along an alignment compared to its minimum bounding box that includes much of the states of Arizona, Colorado, New Mexico, and Utah.

2.2.4 Spatial Semantics

An ontology relating the linguistic realm of spatial semantics is a critical consideration for geoparsing; it provides the inventory of spatial objects and concepts with their relationships. These provide the foundation for interpretation of descriptions. An ontology developed within the geographic domain promotes a “better understanding of the geographic world” (Smith and Mark 1998). A geographic ontology must specify basics such as the definition

and nature of space itself in addition to types of spatial entities (or objects) and their boundaries. The ontology must address topological (theory of boundaries, interiors, connectedness and separation), mereological (theory of part and whole), and physical spatial (dimensionality and geometry) properties. Geographic representations must discriminate between those objects or entities that are delimited by physical characteristics and those contrived by human (cultural/artificial) conceptualization. Represented borders' origins must be identified as *bona fide* (physically based) or *fiat* (contrived). All of these specifications must be established in order to identify the relevant linguistic elements that represent each spatial property in a textual description of location.

The most relevant application of spatial semantics is in the development of a lexicon and grammar for the geoparser. The grammar forms the basis for automated analysis and interpretation of linguistic and spatial relationships in location descriptions. Appendix 1 provides a more thorough discussion of spatial semantics and its relation to descriptions of location.

2.2.5 Products

The product of geoparsing must meet the input needs of geoprocessing. Parser output must include several categories of information: database key values, the parsed text from location descriptions, and error checking data. All information should be contained in one file and in a text format that can be easily read and clearly understood by humans. Database key values must accompany each parsed description of location so geocoded spatial coordinates and geoprocessing attributes can be related to individual museum collection records in the database.

Parsed text for each described location must include the components that comprise a traverse description: the POB and a traverse segment (distance and direction). Each of these components has additional levels of detail. The POB is designated by either a place name (e.g. Yucaipa) or an intersection that can be between two roads (e.g. Interstate-10 and California Route 30) or between a road and a place name (I-10 and Redlands). Roads are identified as to whether they are the travel route or the intersecting route. The segment includes distance, direction, and segment type. Distance includes both a quantity and a unit of measure (e.g. 12.5 Km). Direction includes a cardinal (e.g. West), inter-cardinal (e.g. Northwest), or second inter-cardinal direction (e.g. West Northwest). The segment type indicates whether the segment follows the alignment of a travel route in a general direction from the POB or proceeds directly along the line of a cardinal direction without regard to intervening features.

Error checking information incorporated in the parser output includes the elevation of the locality recorded for a specific record and pointers to related geometry. The recorded elevation can be compared for significant discrepancies

against the elevation of the location calculated by geoprocessing. Topology checks can also be used for error checking when results are compared with commonly referenced geometry (e.g. county) or specialized domain specific geometry (e.g. collection project area).

2.3 Geoprocessing

Geoprocessing is an essential component of the geocoding process. It provides many types of geographic functions including data conversion. It provides an automated process by which functionally identified and tagged text elements of descriptions of location are converted to geometric representations. These representations are then used in calculation of quantified spatial coordinates. Geoprocessing requires software capable of reading parsed location description input, converting that text input to geometric representation, using those representations to perform geometric calculations, and storing the results. The results must include spatial coordinates and a unique value that serves to relate each calculated location with a museum database collection. Once this is successfully accomplished, the collection database is spatially enabled for GIS analysis and display.

2.3.1 Products

There are several important products of geoprocessing. The primary goal of geoprocessing software in this project is production of geographic coordinates from parsed descriptions of museum collection locations. Results must be stored in a format suitable for import to and use in an existing relational database. By-products of geoprocessing include ancillary data that provide for assessment of the reliability of the calculated results. These data include error tracking, level of precision, and process documentation. Examples include multiple instances of locations and abnormal geoprocessing termination. Uncertainty in location descriptions can be quantified through analysis and the results assigned to meaningful categories. Tracking of basic processing information should be related to each determination of location for future reference.

2.3.2 Software Requirements

Geoprocessing requires functionality for a variety of tasks including:

- read and write text input,
- interpret tag file (XML) input as to type and sequence of calculations to perform,
- perform geometric functions for traverse calculations,
- perform spatial queries for value at a location, and
- perform spatial queries for topology relationships.

Some of this functionality is commercially available but some requires custom development. The basic geometric and spatial query functionality is available

through commercial off the shelf GIS products. Customization is required for reading and writing files. Extensible markup language parsers (readers) are available but must be customized for reading specific input files. Development of input interpretation and command generation functions require customization of existing software. ESRI ArcMap provides all required functionality with VBA providing custom development of interface and access to the needed geometric, query, and I/O functions.

2.3.3 Data Requirements

Geoprocessing requires a variety of data to support functionality in geometric calculations and error checking of results. Several types of spatial data are required to support basic geoprocessing functionality for COGO-like and linear referencing calculations. The spatial data includes points and lines represented in tabular and geometric formats; both require certain attributes. Additional reference data may be useful to calculate level of precision or to check for error.

The POB for any traverse is by definition, a point. Points can be either a place name or an intersection. The source of point coordinates for place names can be a gazetteer service returning coordinates. It can also be a shape file including all place names possible for the parsed input. Points for intersections can be either stored in a file or calculated from geometry in other files. There are two types of intersections; between two roads and between a road and place. When two roads intersect, the intersections can be stored as point shapes with attributes. Intersections can also be derived from a table of properly attributed text coordinate values. When the intersection's location is calculated, geoprocessing functionality will require that the roads are stored as line geometry with attributes adequate to identify individual routes. When the intersection is between a road and place, it is necessary to calculate the point on the road nearest the place. Once again, roads are lines and places are represented by points that can be stored as geometry or tabular coordinates.

Error checking requires two different types of data: raster and closed polygons. Rasters in DEM format provide elevation reference for continuous 900 square meter areas. The elevation at calculated locations can be extracted from the raster and compared against the elevation listed when the collection was first recorded. A variety of different closed polygon areas can provide the geometry for assessing topology relations at a calculated location. This can be used to confirm that the calculated location of collected specimens is within a specified area such as a county.

2.3.4 Precision

Precision requirements for geoprocessing are not stringent in most respects. The uncertain nature of the much of the input data used in geometric calculations

precludes any possibility of highly precise results. Generalization of beginning points for traverse calculations significantly reduces precision. Areas large and small are reduced to points. Reference data for point locations seldom improve over a one arc second resolution (Section 2.1.5 discusses uncertainty in descriptions of location). This has effects on geoprocessing. It creates a need for identification of valid levels of precision provided to and returned from calculations. It also allows use of (declaration of) less precise numeric variables during programming.

2.4 Data Transfer and Storage

Information required for geoprocessing comes from one table (HerpColl) in the collection database. The location description field's content is input for the NLP parser. Parser output provides the spatial information identified for each location description. Additional fields' contents are ready for use in geoprocessing with no further preparation; these provide location-related attributes. Information from parser output and unaltered field content are assembled into a structure that is suitable for use as geoprocessing input. In considering suitable input formats, it is reasonable to take into account that the parsing/assembling procedures and the geoprocessing operations may not be run by the same individual or group. Additionally, operations may not occur on the same machine or the same computing platform. Accordingly, a data storage format is needed that is suitable for a collaborative and cross platform environment.

The extensible markup language (XML) format is selected for data storage of this intermediate geocoding data. As such, it will convey geoparser output to geoprocessor input in a format that is compatible with a variety of computing platforms and environments. XML is well suited for this role for several reasons. It is designed to store data (rather than display it) and can incorporate a document type definition that provides a schema with data type definitions. Further, it is *self-documenting* in tagged text format; data elements and their assigned functional category (tag) can easily be read and understood by users. The hierarchical structure reflects the organization of the basic spatial information inherent in traverse type descriptions and that is required for geoprocessing data input requirements.

Data stored in well-formed XML provides improved opportunity for collaborative efforts between those working in geoparsing and geocoding. The XML schema developed for this project is presented immediately below as a structure only. Examples follow that populate elements with content. The XML schema is fully presented and explained in Appendix 2.

```

<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE Location_Descriptions [
  <!ELEMENT Location_Descriptions (Location_Description)>
  <!ELEMENT Location_Description (Record_Number,Locality_Text,County_Name,
    Elevation,Points_Of_Beginning,Segments)>

  <!ELEMENT Record_Number (#PCDATA)>
  <!ELEMENT Locality_Text (#PCDATA)>
  <!ELEMENT County_Name (#PCDATA)>
  <!ELEMENT Elevation (Elevation_Value,Elevation_Unit)>
  <!ELEMENT Elevation_Value (#PCDATA)>
  <!ELEMENT Elevation_Unit (#PCDATA)>
  <!ELEMENT Points_Of_Beginning (Place,Place_and_Route,Route_and_Route)>
  <!ELEMENT Place (PlaceName)>
  <!ELEMENT PlaceName (#PCDATA)>
  <!ELEMENT Place_and_Route (PlaceName,TravelRoute)>
  <!ELEMENT TravelRoute (#PCDATA)>
  <!ELEMENT Route_and_Route (TravelRoute,CrossRoute)>
  <!ELEMENT CrossRoute (#PCDATA)>
  <!ELEMENT Segments (Segment)>
  <!ELEMENT Segment (Segment_Type,Direction,Distance,Remarks)>
  <!ELEMENT Sequence_Number (#PCDATA)>
  <!ELEMENT Segment_Type (#PCDATA)>
  <!ELEMENT Direction (#PCDATA)>
  <!ELEMENT Distance (Distance_Value,Distance_Unit)>
  <!ELEMENT Distance_Value (#PCDATA)>
  <!ELEMENT Distance_Unit (#PCDATA)>
  <!ELEMENT Remarks (#PCDATA)>
]>
<Location_Descriptions>
  <Location_Description>
    <Record_Number></Record_Number>
    <Locality_Text></Locality_Text>
    <County_Name></County_Name>
    <Elevation>
      <Elevation_Value></Elevation_Value>
      <Elevation_Unit></Elevation_Unit>
    </Elevation>
    <Points_Of_Beginning>
      <Place>
        <PlaceName></PlaceName>
      </Place>
      <Place_and_Route>
        <PlaceName></PlaceName>
        <TravelRoute></TravelRoute>
      </Place_and_Route>
      <Route_and_Route>
        <TravelRoute></TravelRoute>
        <CrossRoute></CrossRoute>
      </Route_and_Route>
    </Points_Of_Beginning>
    <Segments>
      <Segment>
        <Segment_Type></Segment_Type>
        <Direction></Direction>
        <Distance>
          <Distance_Value></Distance_Value>
          <Distance_Unit></Distance_Unit>
        </Distance>
        <Remarks></Remarks>
      </Segment>
    </Segments>
  </Location_Description>
</Location_Descriptions>

```

Examples of specimen data from the LACM herpetology collection encoded in this XML format are provided below. Collection table fields contributing the data are identified (italicized in parentheses) in the leading descriptive texts. All other elements are parser output derived from the descriptive text. The first example is record (*IDKey*) number 809 with a locality description (*Locality*) of “3 mi S. Victorville, San Bernardino Nat'l Forest.” This is a basic Euclidean traverse style of description with a place name for a point of beginning. *County* and *Elevation* fields from the HerpColl table provide “San Bernardino” and “1829 ft” respectively.

```
<Location_Description>
<Record_Number>809</Record_Number>
<Locality_Text>3 mi S. Victorville, San Bernardino Nat'l
Forest</Locality_Text>
<County_Name>San Bernardino</County_Name>
<Elevation>
  <Elevation_Value>1829</Elevation_Value>
  <Elevation_Unit>ft</Elevation_Unit>
</Elevation>
<Points_Of_Beginning>
  <Place>
    <PlaceName>Victorville</PlaceName>
  </Place>
</Points_Of_Beginning>
  <Segments>
    <Segment>
      <Segment_Type>EuclideanTraverse</Segment_Type>
      <Direction>S</Direction>
      <Distance>
        <Distance_Value>3</Distance_Value>
        <Distance_Unit>mi</Distance_Unit>
      </Distance>
    </Segment>
  </Segments>
</Location_Description>
```

The second is record number 2043 with a locality description of “9 mi SE Mecca on Hwy 195.” This is a route traverse style of description with the intersection of a place name and a highway for a point of beginning. No value is listed for elevation in the database.

```

<Location_Description>
  <Record_Number>2043</Record_Number>
  <Locality_Text>9 mi SE Mecca on Hwy 95</Locality_Text>
  <County_Name>Riverside</County_Name>
  <Points_Of_Beginning>
    <Place_and_Route>
      <PlaceName>Mecca</PlaceName>
      <TravelRoute>Hwy 95</TravelRoute>
    </Place_and_Route>
  </Points_Of_Beginning>
  <Segments>
    <Segment>
      <Segment_Type>RouteTraverse</Segment_Type>
      <Direction>SE</Direction>
      <Distance>
        <Distance_Value>9</Distance_Value>
        <Distance_Unit>mi</Distance_Unit>
      </Distance>
    </Segment>
  </Segments>
</Location_Description>

```

2.5 Software Solutions

Given that it is preferable to incorporate as much automation as possible into the geocoding process, software solutions that assist coordinate generation from text are sought. This section presents some of the current application software that can be used for finding coordinate values that represent the geography of spatially descriptive text. Existing software products are presented to provide a comparative basis for functionality. Project specific development considerations are presented to finish.

2.5.1 Current Commercial Software

Some progress has been achieved towards the goal of automating the generation of useful coordinate data from described locations. Two applications have been developed that both parse and geoprocess; these are MetaCarta® Version 2.2.1 (MetaCarta 2004), GEOLocate® Version 2.04 (GEOLocate 2004). They accommodate different domains of geographic interest but neither application allows user access to or control of the intermediate product; the parsing results. A related geocoding application is also presented; the MaNIS Georeferencing Calculator Version 031013 (Wieczorek 2004). It provides no parsing capability but produces geometric calculations (*offsets*) based on user inputs and an indicator of uncertainty for calculations.

METACARTA

MetaCarta is developed as a stand-alone PC application and held by a private firm; it is primarily concerned with identification of place names, geocoding

them, and cartographically displaying them with their document related attributes. A street address locator service is provided as well. It can search a specified region or the entire world for geographic names. It is designed to accept text input from a wide variety of digital document sources and formats and is directed towards military, commercial, and intelligence applications. This application disregards spatial modifiers in the text. All locations are generalized to a point representing the area of the entire place name. The amount of generalization varies with the physical extent of the named place. The demonstration system interface for MetaCarta GTS Basic is shown in Figure 4.

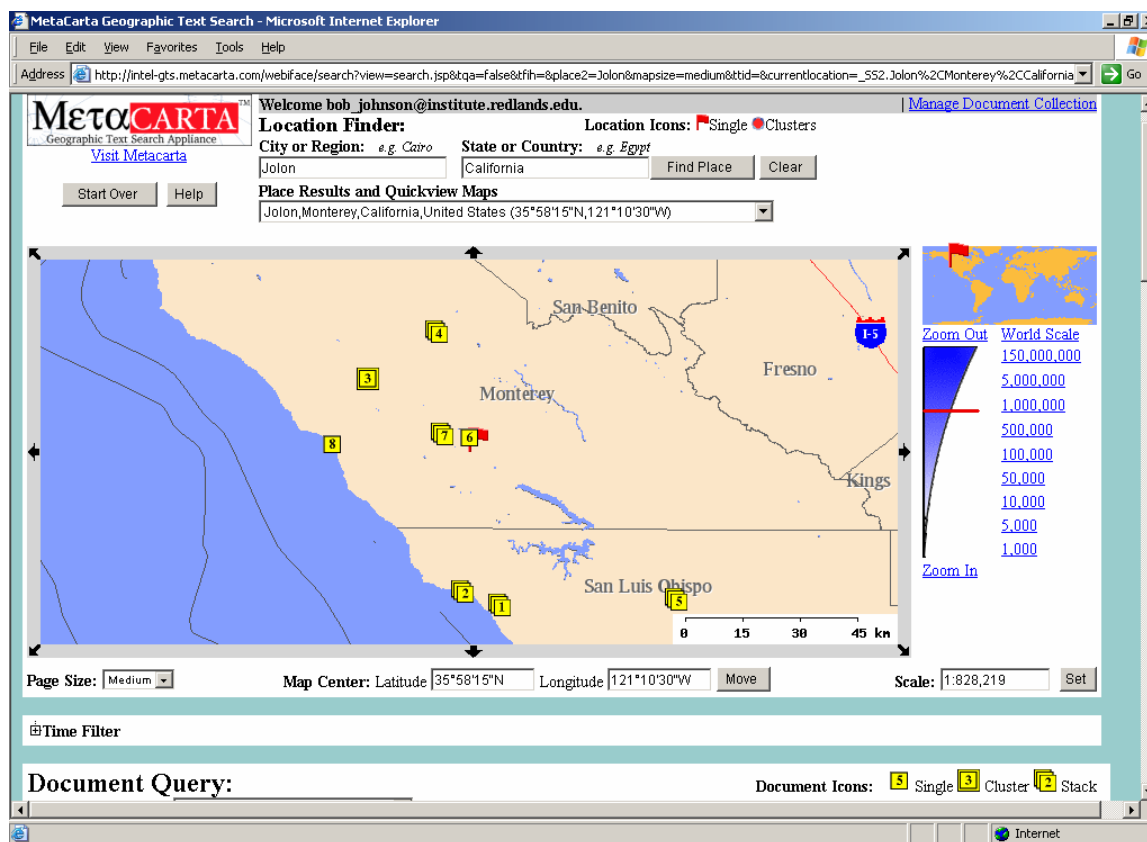


Figure 4 - MetaCarta GTS Basic Interface

GEOLocate

GEOLocate was developed as a stand-alone application by a collaboration of public and private interests. It is distributed by the University of Tulane Natural History Museum. The design of GEOLocate is concerned with identification of a greater variety of geographic features including place names and linear features such as roads and rivers. It is designed specifically to work with descriptions of location, not text documents in general. Additionally, it identifies direction and distance components in descriptive text and applies them to offset the resulting geographic coordinates from a named geographic feature. Calculation of

coordinates uses Euclidean geometry and direction and distance offsets are limited to a single set of distance and direction. The application calculates neither distance along linear features like roads or drainages nor multiple sets of Euclidean calculations (e.g. 2 miles east, 1 mile north of ...). Its search area is currently limited to North America. It accepts input via several standard file formats or manually in a custom interface. Results can be exported to standard file formats. It provides good geocoding functionality, assesses a level of precision for results, and displays results in graphic and tabular formats. The user is permitted to edit the displayed results. The manual entry interface for GEOLocate is shown below in Figure 5.

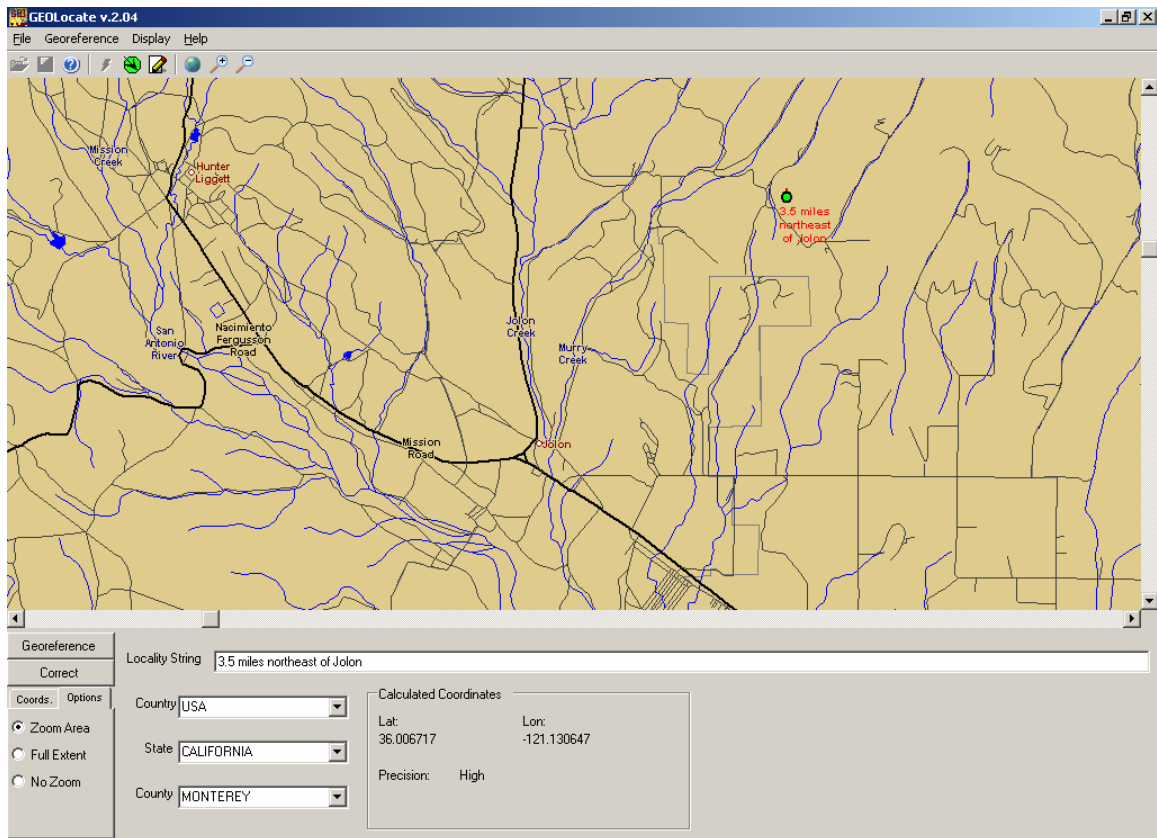


Figure 5 - GEOLocate User Interface

MaNIS

The MaNIS Georeferencing Calculator was developed as an internet java applet and as part of the Mammal Networked Information System Project. MaNIS is a collaboration of institutions developing a distributed database of mammal specimen data (<http://dlp.cs.berkeley.edu/manis/ProjectSummary.html>). The Georeferencing Calculator provides basic Euclidean geometry calculation functionality with up to two sets of direction and distance measures. It also provides an assessment of maximum possible error based on user data entered

on the interface form. Error calculation methods appear well thought out and provide consistent results and a model for related development. The user must provide all input data. Results must be cut from the web form and pasted in a separate application's file for storage. See Figure 6 below for an illustration of the MaNIS coordinate and error calculator interface.

Georeferencing Calculator - Microsoft Internet Explorer

Address: <http://dip.cs.berkeley.edu/manis/gc.html>

Version 031013

Georeferencing Calculator

Calculation Type:

Locality Type:

Step 3) Enter all of the parameters for the locality.

Coordinate Source:

Coordinate System:

Latitude:

Longitude:

Datum:

Coordinate Precision:

Offset Distance:

Extent of Named Place:

Distance Units:

Distance Precision:

Direction:

Decimal Latitude:

Decimal Longitude:

Maximum Error Distance:

[Georeferencing Calculator Manual](#) [Georeferencing Guidelines](#)

This application was originally written by John Wiecezorek. Later versions benefitted from contributions from Qinghua Guo, Carmen Boureau, and Craig Wiecezorek.

John Wiecezorek 3 Nov 2001 *Rev. 22 Aug 2003, JRW*
University of California, Berkeley, CA 94720, Copyright © 2003, The Regents of the University of California.

Applet GC started

Figure 6 - MaNIS User Interface

2.5.2 Inventory of Required Software

The demonstration of partial solutions described in this report requires functionality not available from commercial off the shelf technology. Some custom application development is required. GIS functionality is provided through the Environmental Systems Research Institute (ESRI) ArcMap interface running under either ArcInfo or ArcEditor licenses. Customization of the system functionality is provided by Visual Basic for Applications (VBA) code. This code is developed to access and utilize the properties and methods of geometric objects that are accessible via ArcObjects.

2.6 Summary

The preceding chapter has introduced location descriptions as a form of useful if imperfect spatial data. Uncertainty is apparent in the data as the result of variation of descriptive style, spatial content, and limitations of external geographic reference. The data are not fit for use in GIS without geocoding; a two-part process requiring NLP and geoprocessing. Development has been undertaken by public and private interests with limited success. Natural language processing is an automated process that identifies, classifies, and tags spatially meaningful text elements of descriptions. Geoprocessing converts those text elements identified by NLP, to geometric representations, performs geometric calculations, and returns geographic coordinate values for the described location. These two disparate processes are linked by the NLP output that serves as geoprocessing input. This linking data requires a format that is compatible with a variety of computing platforms and environments and XML is recommended.

Geoprocessing development can be discussed now that a background establishing the problem domain has been presented. The next chapter presents a discussion of the data. These data support the geometric calculations required by geoprocessing.

3 GEOPROCESSING DATABASE

This chapter explains data required for testing and display of the solutions provided with the project. The database design required by geoprocessing needs is simple. Relatively few datasets are required to provide geometry and non-spatial attributes needed for geoprocessing.

3.1 Schema

The data schema for geoprocessing supports the requirements of geometric functions for different geometry including point, polyline, polygon, and raster data. The role each of these plays and the attributes it supplies to geoprocessing is explained below.

3.1.1 Modeling Locations for Geoprocessing

The figure/ground relationships discussed in Section 2.1.4.1 provide the basis for modeling location calculation and error checking. Known locations used for traverse starting points are generalized to zero-dimensional points and highways are generalized to one-dimensional lines for use in calculating new locations. Locations used for error checking are represented in two separate, two-dimensional geometric forms; these are polygons and fields (or rasters). Polygons serve error checking of calculated positions by providing a reference for topology relations while fields provide location attribute data for the calculated position that can be checked against recorded observations.

3.1.2 Implementation

The geoprocessing functions used with this project require specific sets of data to calculate locations. The location calculation functions are developed to traverse either a Euclidean vector or a route. Both these functions require a start point (POB), a distance, and a direction as input. Using these inputs, they calculate and return the described location situated at the end of the traverse. Both functions require specific geometries with adequate attribute information to complete the determination. The Euclidean traverse requires one type of geometry: a point with attributes including the name of the feature it represents and the coordinate values expressed as latitude and longitude. The route traverse requires two types of geometry: a point and a polyline with measure (m) values assigned at each vertex. The point geometry is a POB reference with attributes including the names of both routes that intersect at that point. Coordinate values are required for these points but are derived from the geometry itself. Specialized polyline geometry is used to represent travel routes (highways). This is referred to as a polyline-m and has attribute data including the name of the route. It also has an “m” (measure) value at each vertex that acts like a milepost marker and corresponds directly to the travel distance along the route. Measure values are expressed in miles. Note that m values are not an

attribute but are integral with the geometry. They provide for interpolation of distance (mileage) at any point between vertices and irrespective of the spatial reference used in the GIS. M values can be measured and recorded at closely spaced intervals along the actual routes. These recorded measurements can serve to calibrate GIS representation by correcting measure values to reflect accurately the true travel distance along routes that traverse terrain with significant slope. Note that calibration of m values was not implemented for this project. A set of calibration points was provided with the original route geometry. That route geometry was simplified (edited as described in Section 3.3.2.1) and considered unfit for use with the original calibration point set.

Additional data are required for functions that provide error checking of calculated locations. Error checking requires two types of geometric reference, polygon and raster. Polygons provide an area for use in establishing a topology relation. The calculated location's point must be properly inside the polygon of the collection area (e.g. county) described for that location. As discussed in Section 2.3.3, the elevation value extracted for that location provides a comparison against the value described for that location.

3.2 Spatial Reference

The following presents formal parameters used in defining space in the GIS. This definition affects geometric calculations and display of data.

3.2.1 Area of Interest

An area of interest (AOI) was identified for project development. Delimiting an AOI was appropriate given the considerable (27,469 square mile) extent of and 33,000+ collection record count for the original location description dataset. It is appropriate to specify an AOI extent by physiographic region or sub-region. Constraint of a large area represented by a dataset to an AOI provides more manageable reference datasets (routes, named places, etc.). It also reduces conflicts arising from duplicate feature (place and road) names. The AOI for this project is the extent of herpetofauna associated with the San Bernardino Mountains. This area can be considered regional in extent for biological field collection provenance. The boundary of the study area was delineated by professional biologists based on current land use, land management, elevation, and vegetation.

3.2.2 Coordinate System

The Lambert Conformal Conic projection was selected for use in the project because its conformal characteristics provide minimal angle and distance distortion. These characteristics are considered essential when determining location using calculation methods that derive variables directly from feature geometry. The projection was adapted for optimal geometric representation of

the AOI and the surrounding area. The characteristics of the projection employ a prime meridian of 117 degrees west longitude, standard parallels at 34.0 and 34.4 degrees north latitude, and the latitude of origin mid-way between them at 34.2 degrees. The projection is based on the 1983 North American Datum.

3.2.3 Units of Measure

Many different units of distance measure are present in both the descriptions of location and in the feature classes that support geoprocessing. As a result, unit conversion operations are required regardless of which unit of measure is selected as the map distance unit for the GIS. The meter is specified as the unit of measure for the map document. This selection is based on the software's default unit associated with the map projection.

3.2.4 Precision

Feature geometry, objects, and their attributes participate in a GIS as a geodatabase, dataset, or feature class. Those data that include geometry must be transformed to a common datum and re-projected to a common projection. There is no real need to specify precision any greater than the unit of measure (meter) given the uncertainty present in the descriptions.

3.3 Datasets

Data for geocoding are required in digital file formats and several types of data are required for different purposes in geocoding described locations. Specific data types are required by each of the geoprocessing functions used by geocoding solutions tested for this project. Datasets provide the coordinates for places named in descriptions, geometry for calculations and error checking, and imagery for display. Functions, data types, and specific files used for geoprocessing are listed below in Table 5. Note that no file of spatial elements parsed from described location text is included with this project.

Table 5 – Geoprocessing Spatial Reference Datasets

| Function | Data Type | File Name | Provides |
|--------------------|-------------------------------------|------------------|--|
| Euclidean Traverse | Point Shape File | Place Names | List of Valid Names Collection of Points |
| Route Traverse | Polyline Shape File with “m” values | SBM_Routes | List of Valid Routes Movement Along Route |
| Route Traverse | Point Shape File | AOI_Junctions | List of Valid Route Junctions |
| Error Checking | Polygon Shape File | Counties | Verify by Topology |
| Error Checking | DEM Grid File | NEDgrid | Verify by Comparison of Described/Calculated Elevations |

Since no XML input file was prepared, identification and entry of the elements of any described spatial scene is performed manually. For the purposes of this demonstration, the required data are identified and selected from descriptive text by the user and entered into the form for geometric calculation. These data provide relationships between geometric features and provide pointers to representative feature geometry contained in other files. Manual entries provide all “descriptive text” input data required for testing the solution.

3.3.1 Sources of Data

Data required to test project solutions is available from public sources but requires preparation. The USGS GNIS digital gazetteer is the source of place name points and their attributes. Line geometry for highway routes is provided by the United States Forest Service but is available from other state and federal agencies as well. The DEM data is available in “seamless” format on-line through the USGS. All of the data used for geoprocessing required preparation prior to use.

3.3.2 Data Preparation

None of the required data was obtained in a format ready for immediate use. Preparation of individual datasets for implementation varied. Minimally, transformation and projection to a common spatial reference was completed. All data including geometry were transformed to NAD 83 and projected to a modified version of Lambert Conformal Conic. Other preparatory efforts vary according to each dataset’s required attributes or geometry. Each supporting data set is discussed below and these are organized by basic categories of vector and raster types.

3.3.2.1 Vector Data

Area of Geographic Interest

An area was contrived for delimiting the area of geographic interest (AOGI). It was noted that the herpetological AOI would not be appropriate for this use. Geographic references used in describing locations within the AOI could easily be situated outside of it. It was decided that the area of geographic reference extended beyond the herpetological AOI to at least the first major bounding highway. As a hedge to compensate for place names and road intersections that may be situated near the opposite side of the road, a buffer of 1 mile beyond the first major highway was added to the area of geographic reference.

Place Names

Place Name collections for each state are available for download from the USGS. They are available as delimited text files that include spatial and non-spatial attributes for each listed feature. Spatial attributes include geographic coordinates and elevation. Non-spatial attributes include for example: feature name, feature type, county, and state.

The data are in tabular format and added to a map document as *x,y events*. A point is assigned to each record in the source table. This provided all the place names in the state and far more than was necessary for this project. A selection of only those records from San Bernardino and Riverside counties was made and the selection was exported as an attributed point shape file. This two-county collection of place names was further reduced by using the polygon from the AOI (see above) to make a spatial selection of features. This final selection provided the "Place Names" data used in the project map document. A copy of the place names point shape file was named "Populated Places" and used in the map document for display purposes. Only those features attributed as "populated places" with any specified population value (towns and cities) were included based on the query:

```
"featype" = 'ppl' AND "estpop" <> ''.
```

Routes

Route geometry was provided as ESRI shape files. These files included polylines with measure values. Any of the lines that crossed or continued beyond the area of geographic interest (AOGI) were truncated and only those portions within the AOI were retained for further modification. Route line geometry was then simplified from multi-part lines to a single continuous line segment as necessary. Strictly increasing measure values (in mile units) were then re-assigned to each polyline based on its total length and irrespective of slope.

Junctions

Junctions were created from the completed route geometry. A copy of the routes was converted to an ESRI coverage format. Coverage tables were joined to provide access to all attributes required by a modified version of the US Streets (file-based) geocoding service. This service produced a table with coordinate and attribute data suitable for use as a point event file of highway junctions. The resulting points and their attributes were then exported (saved) as a shape file for use in geoprocessing.

Hydrography

Surface water bodies polygons and drainage flow polylines were added to the map document for display purposes. Data for these was available on-line from the USGS National Hydrography Dataset (NHD) as ESRI shape format files. The files were transformed from NAD27 to NAD83 and projected to match the project. Editing was completed for one major drainage (Mill Creek) within the AOI. Segments comprising the drainage were unnamed in the NHD data and would not display until they were named.

3.3.2.2 Raster Data

DEM

A digital elevation model was available on-line from the USGS as one seamless ESRI grid format file. This file's horizontal coordinate system was geographic coordinates in decimal degrees based on the North American Datum 1983 geodetic model. It required re-projection to match that of the project. The DEM was subsequently broken out into five separate classes based on equal intervals (~707 meters) of elevation values. The classes were assigned display colors grading from dark brown for the lowest elevations to very pale brown for the highest elevations. This was done solely for display purposes and does not affect geoprocessing functionality.

Shaded Relief

The DEM was also used as an input source for a spatial analysis. The spatial analysis was a surface analysis that returned a raster file of shaded relief with characteristic attributes. The sun azimuth is set to 315 degrees (northwest) and sun elevation at 45 degrees above the horizon. The shaded relief layer was set to display as 65% transparent and overlaid on the classified DEM. This raster layer is used for display purposes and does not contribute to geoprocessing functionality.

4 GEOPROCESSING SOLUTIONS AND TESTING

This chapter presents software application development undertaken to produce geoprocessing functions. Tasks for development included the basic geoprocessing requirements for geocoding locations described in the traverse format. Application development is explained in this section as to how each developed component serves to test the adequacy of information parsed from location descriptions. This information is included in the XML file format of parsed location descriptions and ancillary data discussed in Section 2.4. First for discussion is how geographic representation in the GIS matches tagged elements of XML input data representing geographic features and their relations. Functions are then identified for dealing directly with geometry of features and spatial relations between them. Next, specific programming objects are identified for use in development of geoprocessing tests. The user interface forms are then presented and discussed. Finally, the tests are presented with illustrations for both developed and proposed geoprocessing functionality required for location descriptions in traverse form.

4.1 Geoprocessing Research

Selection of the traverse description type for prototype development determined the scope of initial investigation into geoprocessing methods. The traverse type includes two sub-types: the Euclidean traverse and the route traverse. Both types include a POB with a direction and distance to the collection locality (end point). Calculation of the Euclidean traverse uses standard Euclidean geometry. Getting from the POB to a point representing the collection locality is along a straight line, for the distance and direction described, and irrespective of intervening terrain or features. Calculation of the route traverse is more complex. It requires that:

- the POB and collection (end) point are on travel route geometry,
- the direction is that direction most closely matching the alignment relative to the POB, and
- the described distance is measured from the POB along the route.

Several topical areas of interest required definition prior to user interface development and coding. These include identification of:

- geometries appropriate for representing traverse figure and ground objects,
- geometric functions that meet needed geoprocessing requirements for the objects,
- geometric relation parameters, and
- methods to normalize all distance measure to a common unit.

4.1.1 Figure/Ground Geometry

The figure/ground geometric relationships required for the POB and the collection locality are presented here with their representative geometry. Three types of POBs are modeled for traverse type descriptions and are designated *place*, *place and route*, and *route and route*. *Place* is a place name which is a geographic entity or object that occupies some area. This area must be generalized to a representative point and is conceptualized as a point on a bounded plane. This affects the level of precision for the point and precision will vary proportionately to the generalized area. That area can be stored for use in classifying the level of precision of the calculated point (collection location). The figure/ground relationship is the representative point on the bounded plane described by the area of the place name.

The remaining POBs represent types of intersecting features. The conceptual figure/ground relation for both is point on line. *Place and route* is an intersection between a place (an area as above) and a route (line). Travel routes are actually elongate areas but are generalized to a line. The point of intersection is the point on the line closest to the place area's representative point. *Route and route* is an intersection of two routes (lines). Again, the elongate route areas are generalized to lines and the POB is the intersection of the two lines.

Collection locality figure/ground relations are less complex and there are only two figure/ground relations. Geometry for all collection locality figure objects is a point in traverse type descriptions. This geometry is required as the purpose of geocoding museum collection locations, is to obtain coordinate values of a single point that is representative of the location. Ground objects include a bounded plane and lines. The bounded plane ground object is frequently more a conceptual entity than an object; it is the coordinate system on which the collection figure point resides. Lines are used for ground objects when the location is calculated as a distance and direction from a point on a line to another point on the same line.

Useful error checking related ground entities may be applied with either type of collection location figure/ground relationship. Bounded planes can be used in a topology check when the plane represents the extent of a place name like a county or city or project.

4.1.2 Geometric Functions

ArcGIS was reviewed for geometric calculation functionality that would support geoprocessing development of Euclidean and linearly referenced geometries. Initially, formal software extensions were studied. One extension, Survey Analyst, provides COGO functions that could provide for Euclidean functional needs but these were not accessible for development using VBA. Subsequently,

functions suitable for use in development were found in ArcGIS through use of ArcObjects. Functions required for both the Euclidean and the linear referencing calculations were identified.

Several functions are required for coordinate calculation. Functions are required for determination of POBs and collection localities. As discussed above in Section 4.1, there are three types of POB: place, route and place, and route and route. Current development for the place POB requires no geometric calculation to obtain the coordinates of the point. The point's coordinates are available through query of its geometry in the shape file. The route and place is an intersection type of POB and requires identification of the point on the line representing the route that is closest to the point representing the place. Project development did not directly construct a solution to this but the required methods are used for the remaining type of intersection POB, route and route. Geometric calculation of the route and route POB is accomplished by making a copy of the route and using the copy's proximity operator method with the junction point as the argument. This method returns a point on the route line closest to the junction point. This effort may appear moot given that the route lines were used as input to the geocoding service that produced the junctions. Scrutiny of junction points and route lines revealed slight displacement of the junction points from the route lines. The proximity operator method was effectively used to confirm that a POB is actually on the associated route's representative geometry and not just very close. The proximity operator method can also be useful with routes and junctions in situations where the geometry representing a route is more complex than a line (e.g. polygon area or multiple line feature). An alternate method for determining the intersection of routes is use of a method that returns the intersection of two lines. Care must be taken selecting the correct results because some roads intersect more than once.

Calculation of locality coordinates assumes that the POB for a described location is already determined. The required Euclidean methods are simple; a new point is "constructed" at the locality by the POB using its "ConstructAngleDistance" method (on IConstructPoint interface) with the distance and direction from the POB to the locality. Linear referencing methods require a sequence of functions that:

- get the distance from beginning of route polyline to the POB,
- get the measure value at the distance from beginning of line to POB,
- sum the normalized travel distance (to mile units) with the measure value at POB, then
- get the coordinate value of line at location of summed measure value.

Additionally, code for a determination of travel direction along the route was developed. Methods for this require a sequence of functions:

- identify points that are at stated travel distance in both directions along route from POB,
- construct a reference point relative to POB using stated route distance and directions but using Euclidean method,
- calculate distance between two points on route and Euclidean reference point, and
- select closest point (smaller distance) to Euclidean point as locality.

4.1.3 Spatial Relation Variable Functions

Two important geometric relation variables are required for geoprocessing; these are distance and direction. Conversion to a useable format is required for both. Described directions in the herpetology collection database are given almost exclusively in cardinal and intercardinal terms. Direction conversion is straightforward from each case of cardinal reference to radian measure. Distance units in descriptions in the database vary and must be normalized. The normalization process is discussed immediately below in Section 4.1.3.1.

4.1.3.1 Distance Unit Normalization

The mile is the most frequently used unit of distance measure for described locations in the collection database. Multiplication by appropriate constants is required to normalize all distances described with other units, to miles and maintain relative levels of precision. This is accomplished with respect to levels of precision observed in the data (see Section 2.1.5). Simple constants are used (without ArcObjects) in VBA as algebraic operators and the constants change according to the level of precision apparent in the data. Meters, for example, appear in location descriptions with distances of less than 100 meters specified at 5-meter intervals and with distances greater than 100 meters at 100-meter intervals. Accordingly, meter distances are calculated to a 5 meter (0.00311 mile) precision for distances less than or equal to 100 meters with results rounded to 3 places to the right of the decimal. Distance values greater than 100 meters (0.062 mile) are multiplied by 0.062 and rounded to two places to the right of the decimal. All distance measure units present in the collection descriptions are dealt with in a similar manner. Table 6 lists distance units, precision intervals, multipliers, and precision to the right of the decimal for each interval.

Table 6 - Distance Unit Normalization Constants

| Unit | Range | Distance Interval | Multiplier | Precision (in mile units) |
|-----------|------------|-------------------|------------|---------------------------|
| meter | 0 - 100 | 5 | 0.00311 | 0.001 |
| | x > 100 | 100 | 0.06200 | 0.010 |
| yard | x < 100 | 5 | 0.00284 | 0.001 |
| | 100 - 1000 | 50 | 0.02840 | 0.010 |
| | x > 1000 | 100 | 0.05700 | 0.010 |
| feet | x < 5 | 5 | 0.00090 | 0.001 |
| | 5 - 95 | 5 | 0.00090 | 0.001 |
| | 100 - 400 | 100 | 0.01900 | 0.010 |
| kilometer | x < 5 | 0.1 | 0.06200 | 0.010 |
| | x => 5 | 1 | 0.62100 | 0.010 |
| mile | all | all | 1.00000 | 1.000 |

A selection from the herpetology collection, *Snow Crk Rd., 100 yds. S St. Hwy 111*, can serve an example for distance conversion. The distance of 100 yards is within the range of 100 – 1000 yards where distances are typically recorded at a level of precision no greater than 50 yards. The distance 100 is divided by 50 for a quotient of 2. This is used as the multiplicand with the multiplier constant of 0.0284 [mile] (from Table 3). The product of 0.0568 mile is rounded to 0.06 mile. This procedure is handled by the second case of the VBA code presented below.

```
Case "yards", "yds"
  Select Case sgl_DistanceQuantityIn
    Case Is < 100 'precision to 5 yards
      sgl_DistanceQuantityOut = sgl_DistanceQuantityIn / 5 * 0.00284
      sgl_DistanceQuantityOut = Round(sgl_DistanceQuantityOut, 3)
    Case 100 To 1000 'precision to 50 yards
      sgl_DistanceQuantityOut = sgl_DistanceQuantityIn / 50 * 0.0284
      sgl_DistanceQuantityOut = Round(sgl_DistanceQuantityOut, 2)
    Case Is > 1000 'precision to 100 yards
      sgl_DistanceQuantityOut = sgl_DistanceQuantityIn / 100 * 0.057
      sgl_DistanceQuantityOut = Round(sgl_DistanceQuantityOut, 2)
    Case Else
      EuclideanMove = 2
  End Select
```

4.1.3.2 Direction Unit Conversion

Direction is also converted using simple VBA code for each case of the 16 different cardinal and inter-cardinal (first and second) directions shown in Table 7. No calculation is required; each direction is assigned its rotational value in radians. Rotation begins (is zero) at east and increases in a counter-clockwise

direction. Two places to the right of the decimal provide more than adequate precision (~ 0.5 degree) given the high degree of uncertainty associated with these directions.

Table 7 - Cardinal Directions in Radian Measure

| Cardinal Direction | Radian Measure | Cardinal Direction | Radian Measure |
|--------------------|----------------|--------------------|----------------|
| East | 0.00 | West | 3.14 |
| East Northeast | 0.39 | West Southwest | 3.53 |
| Northeast | 0.78 | Southwest | 3.93 |
| North Northeast | 1.18 | South Southwest | 4.32 |
| North | 1.57 | South | 4.71 |
| North Northwest | 1.96 | South Southeast | 5.10 |
| Northwest | 2.36 | Southeast | 5.50 |
| West Northwest | 2.75 | East Southeast | 5.89 |

A selection from the herpetology collection, *Snow Crk Rd., 100 yds. S St. Hwy 111*, can also serve as an example for direction conversion. The direction value “S” (south) from the example is directly converted to a radian measure of 4.71. This procedure is handled by the case of the VBA code presented below.

```
'South
Case "S", "s"
    sgl_RadianValue = 4.71
```

4.1.4 ArcObjects Development

Geoprocessing functions were developed using ArcGIS running under an ArcInfo license although an ArcEditor license would have sufficed. Geometry is calculated and its attributes are queried using methods and properties of ArcObjects. User interface, control, and conversion functions are handled by Visual Basic for Applications (VBA). Two VBA “forms” (user interfaces) were developed for use within ArcMap; one for calculation of Euclidean geometric functions and one for calculation of linear referencing geometric functions. At this time, the form for Euclidean geometry only supports place name POBs and the form for linear referencing only supports route intersection POBs. Error checking geometric functions are not included on the forms or in the code but are discussed and the required error checking reference geometry is present. All data supporting geoprocessing functionality are included as individual feature layers in the ESRI map document.

4.1.4.1 ArcObjects for Euclidean Geometry

Euclidean geoprocessing is accomplished through ArcObjects methods using only points. Two ArcObjects Interfaces are required: IPoint and IConstructPoint. Query interface of point geometry provides the “ConstructAngleDistance” method via the IConstructPoint interface shown in Figure 7. This method requires variables with values representing the POB, distance, and direction.

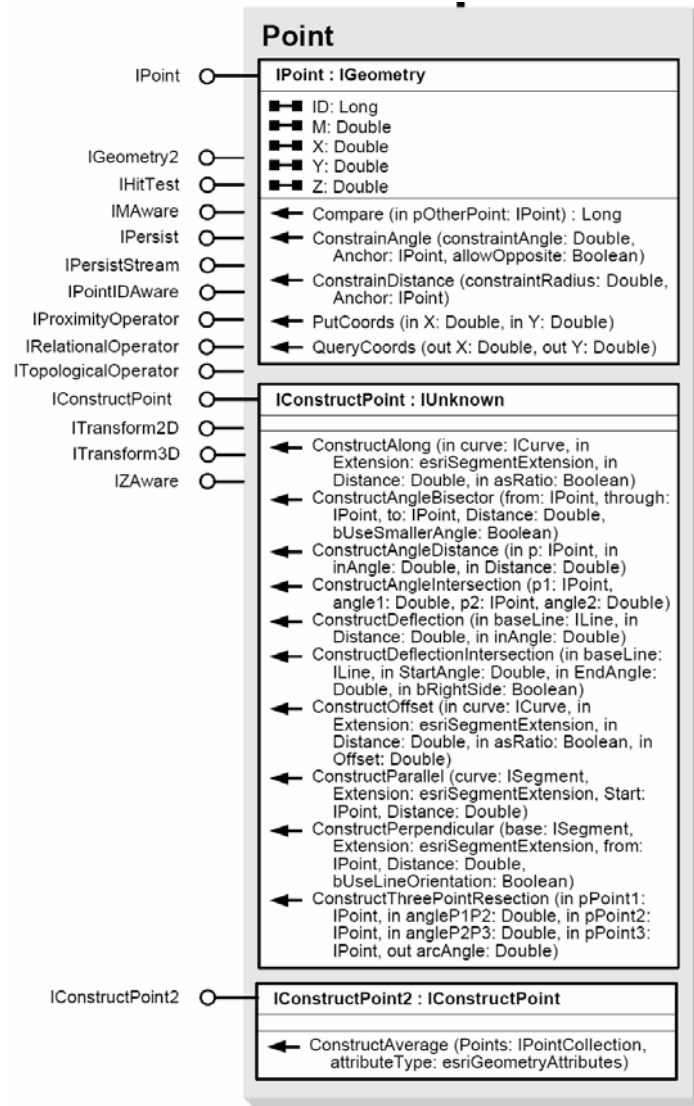


Figure 7 - Point Object and Interfaces

ESRI (2001)

The following code fragment from the form for Euclidean traverse portrays this use of geometry objects.

```
'equatorial circumference value used for miles per degree "constant"
m_DDAngularDistance = sgl_DistanceQuantityOut / 69.172

Set m_pToPoint = New esriCore.Point 'set the new point
Set m_pConstPt = m_pToPoint 'geometry operator is used

m_pConstPt.ConstructAngleDistance m_pInPoint, sgl_RadianValue,
m_DDAngularDistance 'Calculate the ToPoint
```

4.1.4.2 ArcObjects for Linear Referencing

Processing requirements for linear referencing functionality are met using ArcObjects' methods using points and polyline-m geometries. Considerably more objects' interfaces and methods are required to calculate linearly referenced locations than for Euclidean geoprocessing. The POB must be firmly established as actually on the route. This is accomplished through query interface of the route geometry using the IProximityOperator interface. This method can also be applied in future development for finding the intersection of a place name and route. The nearest point on the route to the listed junction coordinates is determined as POB. The measure value of the POB is then determined in a two step process using the "QueryPointAndDistance" method of the ICurve interface followed by the "GetMsAtDistance" method of the IMSegmentation interface shown in Figure 8. The measure value of the destination is calculated by adding/subtracting the travel distance as appropriate and extracting the coordinates at that value's location on the polyline using IMSegmentation's "GetPointsAtM" method.

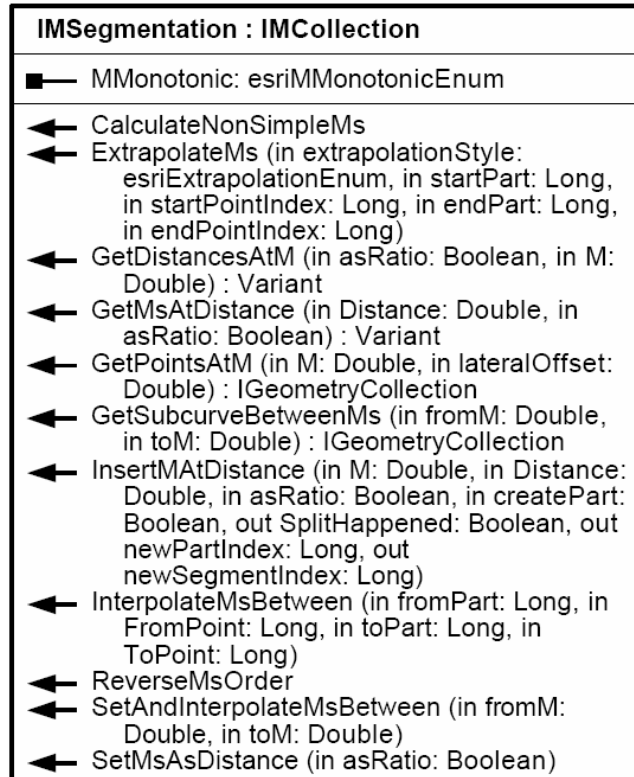


Figure 8 - IMSegmentation Interface Model
(ESRI 2001)

The following fragments of code written for the Route Location calculator form illustrate the measure value extraction method of the interface for measure segmentation along a curve.

```
Dim pMSeg As IMSegmentation
Set pMSeg = pCurve
'Assign value(s) for M (at distance) to Ms (possible array)
Ms = pMSeg.GetMsAtDistance(dAlong, False)
```

4.2 Completed User Interfaces

Completed VBA forms used to test geoprocessing functions required for geocoding described locations are presented below. Form presentation and functionality are discussed. See Appendix 4 for the ArcMap VBA forms' code.

4.2.1 Euclidean Location Calculator

The Euclidean location calculator interface gives the user the ability to calculate latitude and longitude of a point at a given distance and direction from another

point with known geographic coordinates. It is shown in Figure 9 with a sample of input and results.

The screenshot shows a software window titled "EUCLIDEAN LOCATION CALCULATOR". Inside the window, there is a text input field at the top containing "Yucaipa", with the label "Start Point (Place Name)" below it. Below this, there are two input fields: "Travel Distance" containing "12.5" and a unit dropdown menu set to "Kilometers". Below these is a "Travel Direction" dropdown menu set to "NW". A "CALCULATE" button is centered below the direction menu. At the bottom, there are two output fields: "Longitude" containing "-117.122" and "Latitude" containing "34.113", with the label "End Point" centered between them.

Figure 9 - User Interface for Euclidean Geoprocessing

The interface displays language that is familiar to users. The user may enter information in a logical manner: a city or other place name for the starting point, the distance from the start point to the collection locality, and the direction from the starting point to the collection locality. Data entry error is minimized by use of pull-down selection menus. No extraneous information is displayed that would confuse the user. The form provides the following capabilities:

- allows user entry of any GNIS listed place name within the AOGI (see Section 3.3.2.1) for a start point,
- allows user input of any distance quantity,
- allows user selection of distance units currently listed in collection database,
- allows user selection of direction of displacement from start point, and
- displays latitude and longitude of described point.

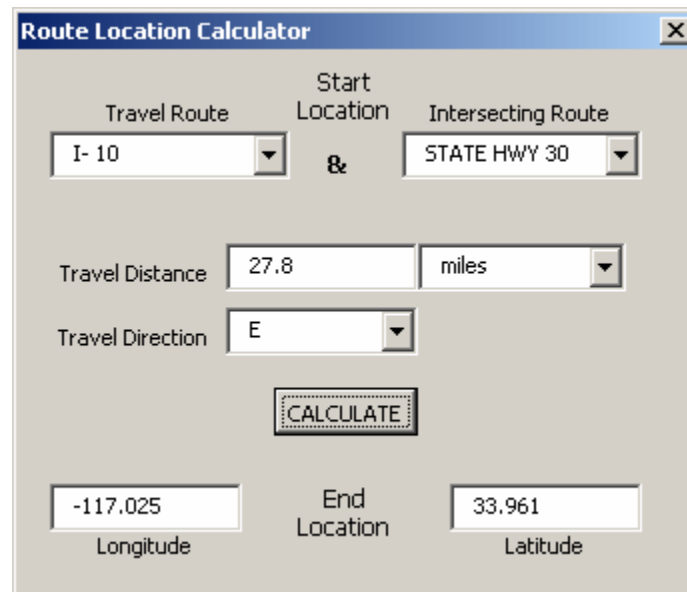
The interface has several limitations:

- "Travel Distance" accepts only distance units currently included in collection database,
- "Travel Direction" accepts horizontal directions only as cardinal direction up to second intercardinal level, and
- the form's processes reduce extent of populated places to one point.

Error messages from the Euclidean Location Calculator are in plain language that refers the user to input values on the form or advises user of the reason for process termination. Neither internal nor external error documentation is currently developed for the interface.

4.2.2 Route Location Calculator

The Route location calculator interface gives the user the ability to calculate latitude and longitude of a point on a travel route at a given distance and direction from an intersection of two routes. It is shown in Figure 10 with a sample of input and results.



The screenshot shows a window titled "Route Location Calculator". It contains the following fields and controls:

- Travel Route:** A pull-down menu with "I- 10" selected.
- Start Location:** A label with an ampersand "&" next to it.
- Intersecting Route:** A pull-down menu with "STATE HWY 30" selected.
- Travel Distance:** A text input field containing "27.8" and a unit pull-down menu set to "miles".
- Travel Direction:** A pull-down menu with "E" selected.
- CALCULATE:** A button with a dashed border.
- End Location:** Two output fields: "Longitude" with the value "-117.025" and "Latitude" with the value "33.961".

Figure 10 - User Interface for Route Geoprocessing

This interface provides users the ability to calculate latitude and longitude of a point on a specified highway, at a given distance along that highway from an intersecting route. The interface uses language that is familiar to users. The user may enter information in a logical manner: the route traveled, the intersecting route, the distance from the intersecting route, and the direction traveled from the intersecting route. As with the Euclidean calculator, data entry error is minimized by use of pull-down selection menus. No extraneous information is displayed that would confuse the user. Error messages are provided in language tailored to a software development aware user that potential problems were encountered during processing. Neither internal nor external help documentation has been developed.

The form provides the following capabilities:

- Allows user selection of routes for both “Traveled” and “Intersecting” highways,
- Allows manual input of any distance quantity,
- Allows user selection of only distance units currently listed in collection database,
- Allows user selection of one of 16 listed directions as the travel direction,
- Computes the best fit direction of travel on a route for any specified direction from a junction, and
- Displays Latitude and Longitude of result of calculation.

The interface has several limitations:

- It is currently constrained to state and federal highways in a specific area of interest,
- “Travel Distance” Accepts only distance units currently included in collection database,
- “Travel Direction” Accepts horizontal directions only as cardinal direction up to second intercardinal level. Entry of other units of angular measure is not permitted, and
- it currently lists only one point of intersection for routes that intersect more than once.

4.3 Tests of Euclidean Traverse Geoprocessing Functions

Examples of location calculations are presented here for two Euclidean traverses:

- a traverse with one leg that has a place name for a POB, and
- a traverse with multiple segments (or legs) using a place name for a POB.

The first example is supported by the Euclidean Location Calculator’s functionality. The second example demonstrates how functionality developed for the calculator can be extended. Both examples are presented with a listing of XML elements and that data’s spatial function. The correlating form label is listed as well for the Location Calculator example.

4.3.1 Place Name POB

The “Euclidean Location Calculator” VBA form verifies that geoprocessing functions are successful using a sub-set of elements (spatial features and variables) included with the parser output [XML] model. These elements include the POB, and distance and direction from the POB to the collection point; all are required for calculating location with a Euclidian traverse type of description. Place names are represented in the map document by the point shape file of the same name (“PlaceNames”). During calculation, the file is queried for a feature named that is an exact match with the input “Start Point” string from the form. When the feature is identified, a copy of the feature’s point geometry is made and the geometry’s coordinate values are assigned to variables.

Direction is a text value selected by the user from a [pull-down] list of cardinal directions on the form. The text input value is compared against all valid possibilities for horizontal direction. A numeric radian value is associated with each and the value for the matching direction is assigned to a variable.

The distance quantity value is entered into the form as text. This is converted to a numeric value when it is assigned to a numeric variable.

The distance unit is a text value selected by the user from a [pull-down] list of valid distance units on the form. The text input value is compared against all valid possibilities for distance units. Distance measure values are converted to an equal representation in miles and rounded according to precision that reflects the unit's precision relative to miles.

The following example provides an illustration of Euclidean geoprocessing using elements specified for parser output. A hypothetical example of a collection locality description is used here, "3.3 kilometers ESE of Forest Falls." The spatial variables identified in this traverse description are listed in Table 8 as XML elements with their associated VBA form input field label. The complete collection description is presented in XML format immediately after the table.

Table 8 - Euclidean Parser Output Elements and VBA Test Inputs

| XML Elements | VBA Form Inputs |
|--|-------------------------------|
| <code><PlaceName>Forest Falls</PlaceName></code> | Start Point |
| <code><Direction>ESE</Direction></code> | Travel Direction |
| <code><Distance_Value>3.3</Distance_Value></code> | Travel Distance <Quantity> |
| <code><Distance_Unit>Kilometers</Distance_Unit></code> | Travel Distance <Units> |

```

<Location_Description>
  <Record_Number>45678</Record_Number>
  <Locality_Text>3.3 kilometers ESE of Forest Falls</Locality_Text>
  <County_Name>San Bernardino</County_Name>
  <Points_Of_Beginning>
    <Place>
      <PlaceName>Forest Falls</PlaceName>
    </Place>
  </Points_Of_Beginning>
  <Segments>
    <Segment>
      <Segment_Type>Euclidean</Segment_Type>
      <Direction>ESE</Direction>
      <Distance>
        <Distance_Value>3.3</Distance_Value>
        <Distance_Unit>Kilometers</Distance_Unit>
      </Distance>
    </Segment>
  </Segments>
</Location_Description>

```

Selected XML element content is entered into the form (see Figure 11) where the locality coordinates are calculated and the results are displayed. Figure 12 illustrates these features and variables in their geographic setting.

Figure 11 - Euclidean Location Calculator Form Sample

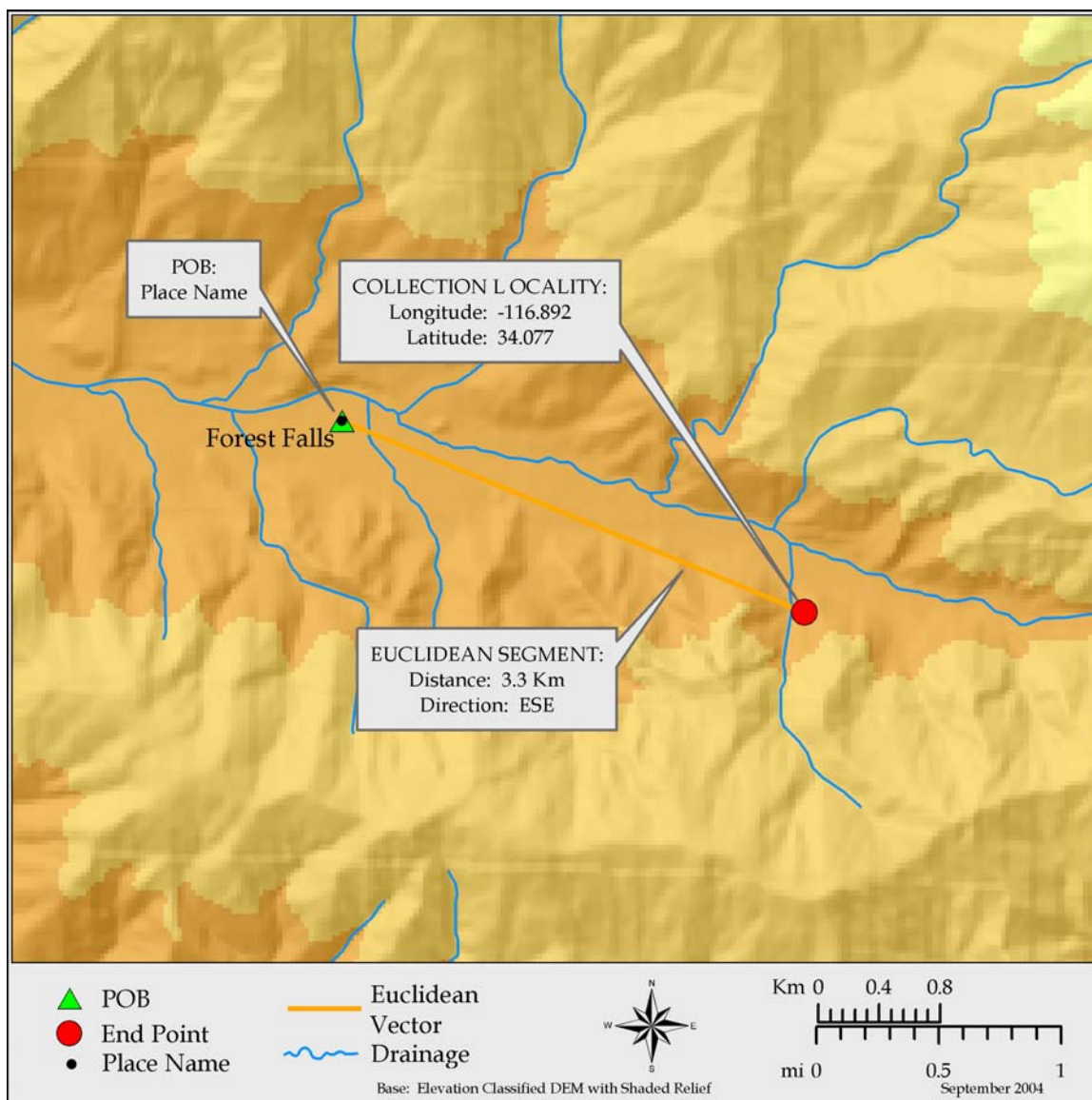


Figure 12 - Euclidean Traverse from Place Name to Locality

4.3.2 Multiple Traverse Segments

Some locations are described in Euclidean traverse format with descriptions that comprise more than one iteration (leg or line segment) of the required Euclidean descriptive elements. This type of traverse has been termed orthogonal. It includes an identifiable feature as a POB for the initial leg of the traverse but subsequent legs use the previous leg's end point for their POB.

The "Euclidean Location Calculator" VBA form does not currently provide for user input of multiple Start Point, Travel Direction, Travel Distance<Quantity>, and Travel Distance<Units> input sets. The following example does however, provide an illustration of geoprocessing considerations for this type of Euclidean traverse. A hypothetical example of a collection locality description is used here,

“900m east, 1200m north of Cushenberry Springs.” The spatial variables identified in this traverse description are listed in Table 9 as XML elements with a comment on the element content’s spatial function. The complete collection description is presented in XML format immediately after the table.

Table 9 - Orthogonal Euclidean Traverse Parser Output Elements

| XML Elements | Comment |
|---|-------------------------|
| <code><PlaceName>Cushenberry Springs</PlaceName></code> | POB |
| <code><Direction>east</Direction></code> | Leg 1 Direction |
| <code><Distance_Value>900</Distance_Value></code> | Leg 1 Distance Value |
| <code><Distance_Unit>m</Distance_Unit></code> | Leg 1 Distance Unit |
| <code><Direction>north</Direction></code> | Leg 2 Direction |
| <code><Distance_Value>1200</Distance_Value></code> | Leg 2 Distance Value |
| <code><Distance_Unit>m</Distance_Unit></code> | Leg 2 Distance Unit |

```

<Location_Description>
  <Record_Number>1234</Record_Number>
  <Locality_Text>900m east, 1200m north of Cushenberry Springs
</Locality_Text>
  <County_Name>San Bernardino</County_Name>
  <Points_Of_Beginning>
    <Place>
      <PlaceName>Cushenberry Springs</PlaceName>
    </Place>
  </Points_Of_Beginning>
  <Segments>
    <Segment>
      <Segment_Type>Euclidean</Segment_Type>
      <Direction>east</Direction>
      <Distance>
        <Distance_Value>900</Distance_Value>
        <Distance_Unit>m</Distance_Unit>
      </Distance>
    </Segment>
    <Segment>
      <Segment_Type>Euclidean</Segment_Type>
      <Direction>north</Direction>
      <Distance>
        <Distance_Value>1200</Distance_Value>
        <Distance_Unit>m</Distance_Unit>
      </Distance>
    </Segment>
  </Segments>
</Location_Description>

```

Figure 13 illustrates these features and variables in their geographic setting.

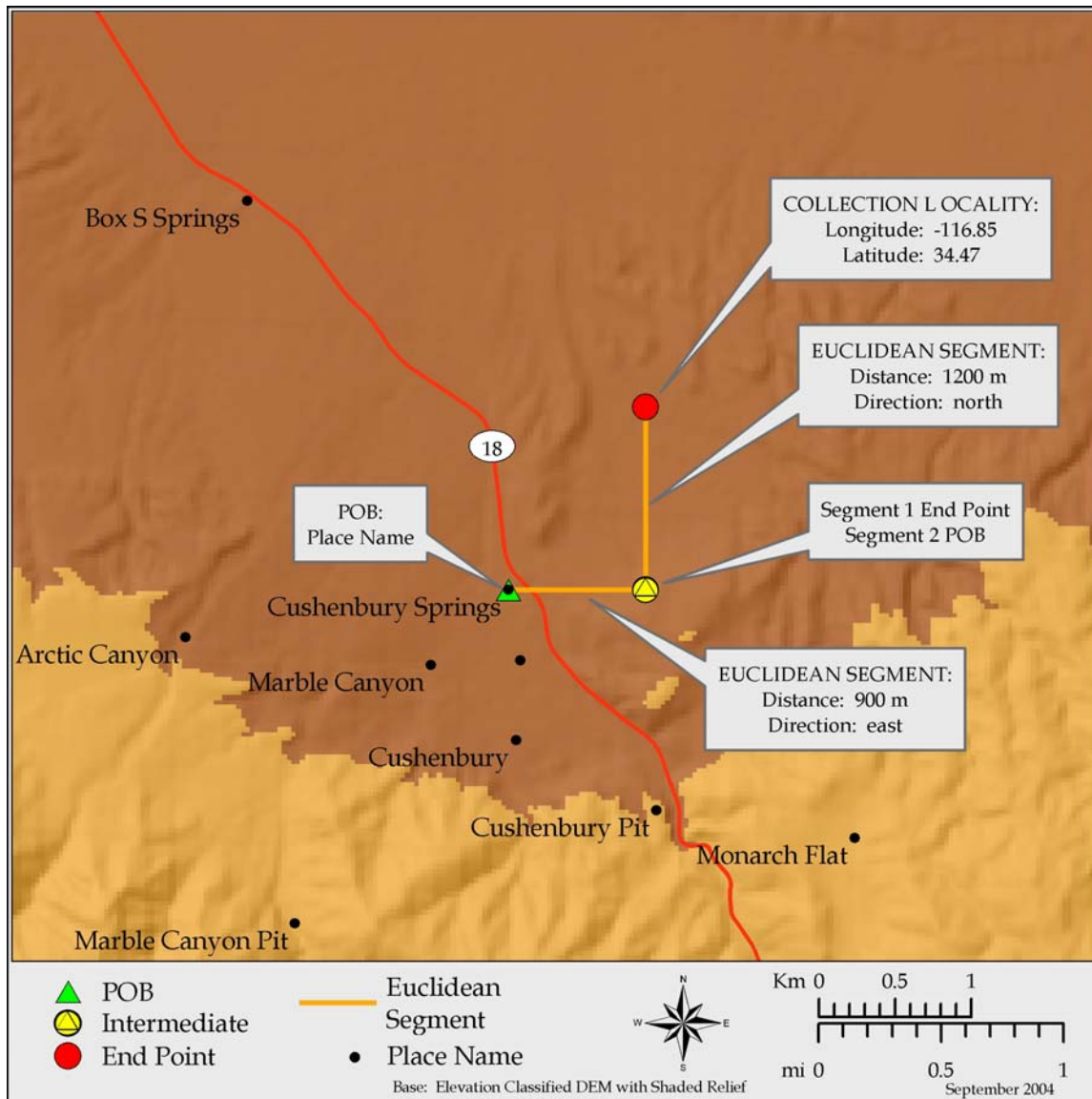


Figure 13 - Euclidean Orthogonal Traverse

4.4 Tests of Linear Referencing Geoprocessing Functions

Examples of location calculations are presented here for two single-leg Route traverses. The first uses the intersection of two routes for its POB; it is supported by the Route Location Calculator's functionality. The second uses the intersection of a place name and a route as a POB. This type of POB is not supported by the calculator form but the example does illustrate how route travel direction is determined by the calculator. Both are presented with a listing of XML elements and content.

4.4.1 Route Intersecting Route POB

The “Route Location Calculator” VBA form verifies that essential geoprocessing functions are viable when using a sub-set of the elements (spatial features and variables) included with the XML parser output model. The elements in this sub-set are required for linear referencing functions used in route traverse location calculations. The following example provides an illustration of the route traverse geoprocessing functions. An example drawn from the herpetology collection is used here, “27 mi W jct St Hwys 62 and 247 on edge of Hwy 247 near Old Woman Springs, ~10 mi E Lucerne Valley.” The spatial variables identified in this traverse description are listed in Table 10 as XML elements with their associated VBA form input field label. The complete collection description is presented in XML format immediately after the table.

Table 10 - Linear Referencing Parser Output Elements and VBA Test Inputs

| XML Elements | VBA Form Inputs |
|--|-------------------------------|
| <TravelRoute>State Hwy 247</TravelRoute> | Travel Route |
| <CrossRoute>State Hwy 62</CrossRoute> | Intersecting Route |
| <Direction>W</Direction> | Travel Direction |
| <Distance_Value>27</Distance_Value> | Travel Distance <Quantity> |
| <Distance_Unit>mi</Distance_Unit> | Travel Distance <Units> |

```

<Location_Description>
  <Record_Number>21767</Record_Number>
  <Locality_Text>27 mi W jct St Hwys 62 and 247 on edge of Hwy 247
  near Old Woman Springs, ~10 mi E Lucerne Valley.</Locality_Text>
  <County_Name>San Bernardino</County_Name>
  <Points_Of_Beginning>
    <Route_and_Route>
      <TravelRoute>State Hwy 247</TravelRoute>
      <CrossRoute>State Hwy 62</CrossRoute>
    </Route_and_Route>
  </Points_Of_Beginning>
  <Segments>
    <Segment>
      <Segment_Type>Route</Segment_Type>
      <Direction>W</Direction>
      <Distance>
        <Distance_Value>27</Distance_Value>
        <Distance_Unit>mi</Distance_Unit>
      </Distance>
    </Segment>
  </Segments>
</Location_Description>

```

These variables are entered into the form (see Figure 14) where the locality coordinates are calculated and the results are displayed. Figure 15 illustrates these features and variables in their geographic setting.

The screenshot shows a software window titled "ROUTE LOCATION CALCULATOR". Inside the window, there are several input fields and a calculation button. At the top, there are two dropdown menus: "STATE HWY 247" and "STATE HWY 62". Below the first dropdown is the label "Travel Route". Between the two dropdowns is the text "Start Point (Junction)" and an ampersand "&". Below the second dropdown is the label "Intersecting Route". Below these are two more input fields: "Travel Distance" with the value "27" and a unit dropdown menu set to "miles". Below that is a "Travel Direction" dropdown menu set to "W". In the center is a button labeled "CALCULATE". At the bottom, there are two output fields: "Longitude" with the value "-116.661" and "Latitude" with the value "34.377". The label "End Point" is centered between these two output fields.

Figure 14 - Route Location Calculator Form Sample

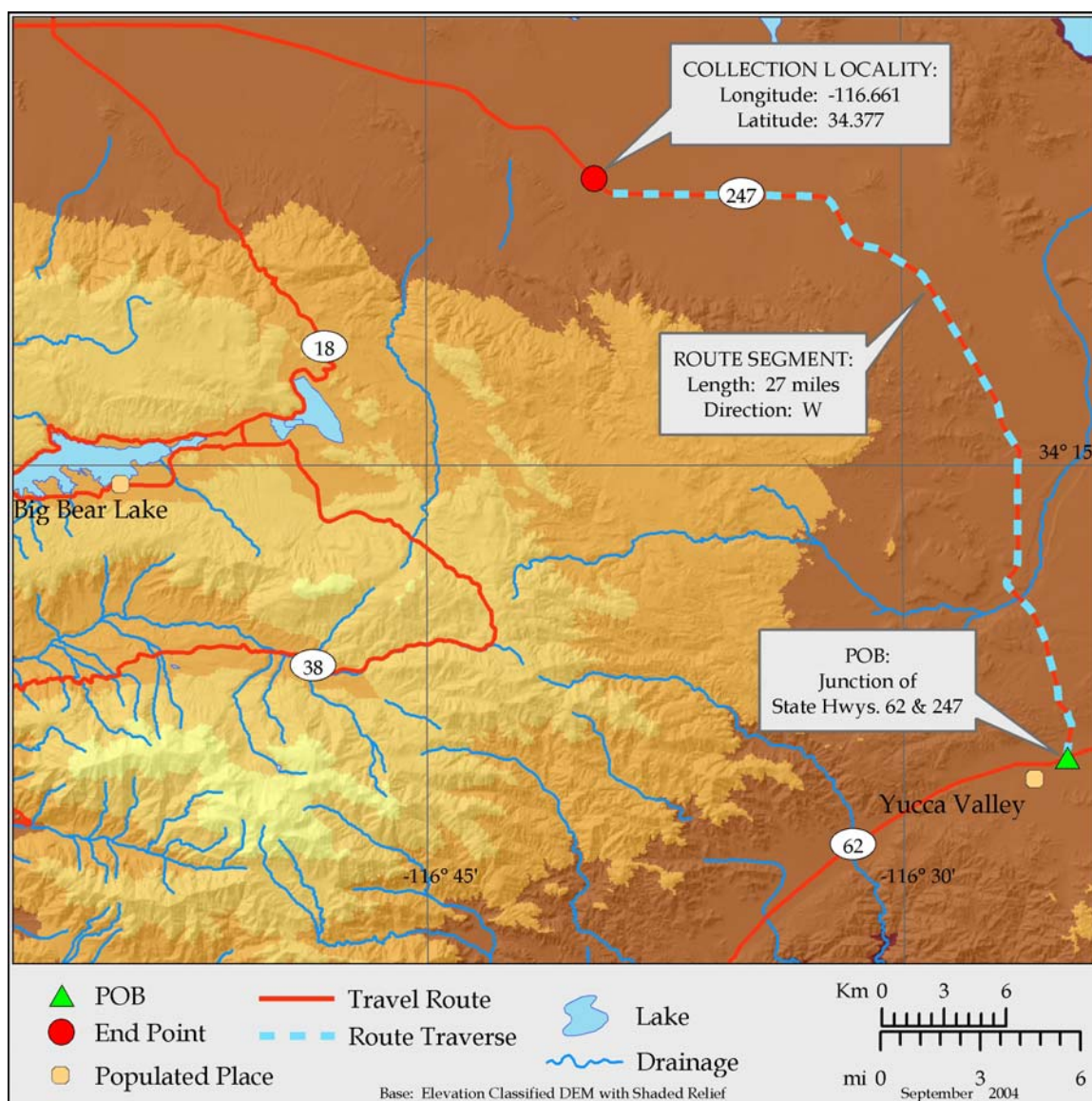


Figure 15 - Route Traverse Location Calculation Geometry

4.4.2 Route Intersecting Place Name POB

The “Route Location Calculator” VBA form does not currently support user input of the place name features required for calculating location on a route when the POB is a route intersecting a place name. The following example does provide an illustration of geoprocessing considerations for this type of route traverse POB. An example of a collection locality description from the herpetology collection is used here, “12.5 mi NE Big Bear City, Hwy 18.” The spatial variables identified in this traverse description are listed in Table 11 as XML elements with a comment on the element’s spatial function. The complete location description is presented in XML format immediately after the table.

Table 11 - Route Location Parser Output Elements and VBA Test Inputs

| XML Elements | Comment |
|--|---------------------------------|
| <code><TravelRoute>Hwy 18</TravelRoute></code> | Intersection POB Component 1 |
| <code><PlaceName>Big Bear City</PlaceName></code> | Intersection POB Component 2 |
| <code><Direction>NE</Direction></code> | Travel Direction |
| <code><Distance_Value>12.5</Distance_Value></code> | Distance Quantity |
| <code><Distance_Unit>mi</Distance_Unit></code> | Distance Unit |

```

<Location_Description>
  <Record_Number>20346</Record_Number>
  <Locality_Text>12.5 mi NE Big Bear City, Hwy 18.</Locality_Text>
  <County_Name>San Bernardino</County_Name>
  <Points_Of_Beginning>
    <Place_and_Route>
      <PlaceName>Big Bear City</PlaceName>
      <TravelRoute>Hwy 18</TravelRoute>
    </Place_and_Route>
  </Points_Of_Beginning>
  <Segments>
    <Segment>
      <Segment_Type>Route</Segment_Type>
      <Direction>NE</Direction>
      <Distance>
        <Distance_Value>12.5</Distance_Value>
        <Distance_Unit>mi</Distance_Unit>
      </Distance>
    </Segment>
  </Segments>
</Location_Description>

```

Figure 16 illustrates route traverse features and relations in a geographic setting. It also illustrates distances compared between the Euclidean end point and the two possible route end points in order to determine travel direction on a route.

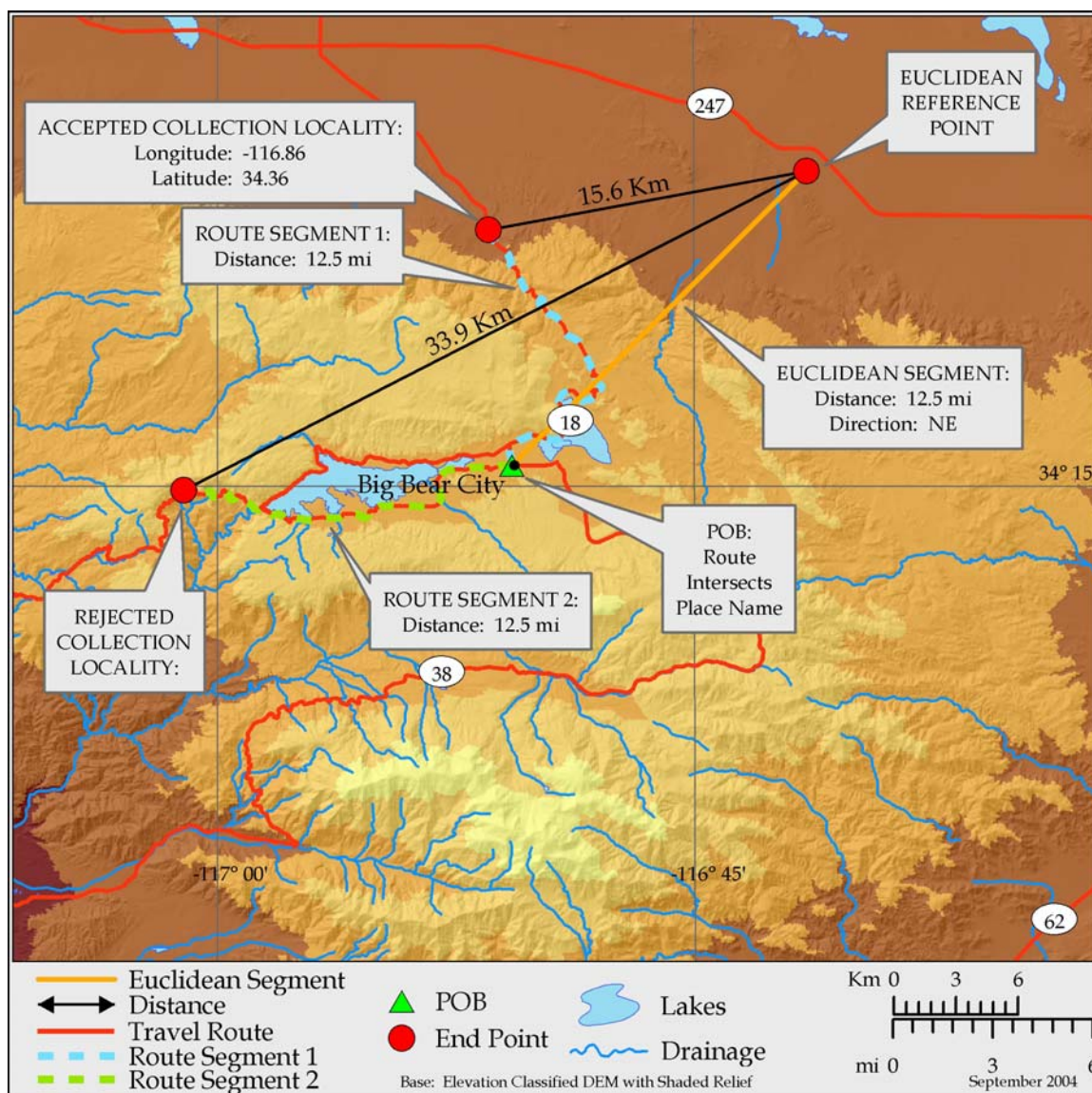


Figure 16 - Route Traverse Showing Direction Determinants

4.5 Mixed Euclidean and Route Traverse Geoprocessing Functions

The final example of relevant geoprocessing functions for described collection locations concerns those descriptions that include multiple segments of differing traverse types. No VBA form is currently developed for the project that provides geoprocessing functionality for mixed Euclidean and route traverse input data. The following example does provide an illustration of geoprocessing considerations for a location description that comprises different traverse types. The example used here is a hypothetical collection locality description, "1 Km NNE of a point at 7.9 mi E of I-10 along Hwy 38." The spatial variables identified in this traverse description are listed in Table 12 as XML elements with

a comment on each element's spatial function. The complete location description is presented in XML format immediately after the table. Figure 17 illustrates these features and relations in their geographic setting.

Table 12 - Parser Output Elements for Mixed Route and Euclidean Traverse

| XML Elements | Comment |
|--|---------------------------------|
| <code><TravelRoute>State Hwy 38</TravelRoute></code> | Intersection POB Component 1 |
| <code><CrossRoute>I-10</CrossRoute></code> | Intersection POB Component 2 |
| <code><Direction>east</Direction></code> | Leg 1 Direction |
| <code><Distance_Value>7.9</Distance_Value></code> | Leg 1 Distance Value |
| <code><Distance_Unit>mi</Distance_Unit></code> | Leg 1 Distance Unit |
| <code><Direction>NNE</Direction></code> | Leg 2 Direction |
| <code><Distance_Value>1</Distance_Value></code> | Leg 2 Distance Value |
| <code><Distance_Unit>Km</Distance_Unit></code> | Leg 2 Distance Unit |

```

<Location_Description>
  <Locality_Text>1 Km NNE of a point at 7.9 mi E of I-10 along Hwy
  38.</Locality_Text>
  <County_Name>San Bernardino</County_Name>
  <Points_Of_Beginning>
    <Route_and_Route>
      <TravelRoute>State Hwy 38</TravelRoute>
      <CrossRoute>I-10</CrossRoute>
    </Route_and_Route>
  </Points_Of_Beginning>
  <Segments>
    <Segment>
      <Segment_Type>Route</Segment_Type>
      <Direction>east</Direction>
      <Distance>
        <Distance_Value>7.9</Distance_Value>
        <Distance_Unit>mi</Distance_Unit>
      </Distance>
    </Segment>
    <Segment>
      <Segment_Type>Route</Segment_Type>
      <Direction>NNE</Direction>
      <Distance>
        <Distance_Value>1</Distance_Value>
        <Distance_Unit>Km</Distance_Unit>
      </Distance>
    </Segment>
  </Segments>
</Location_Description>

```

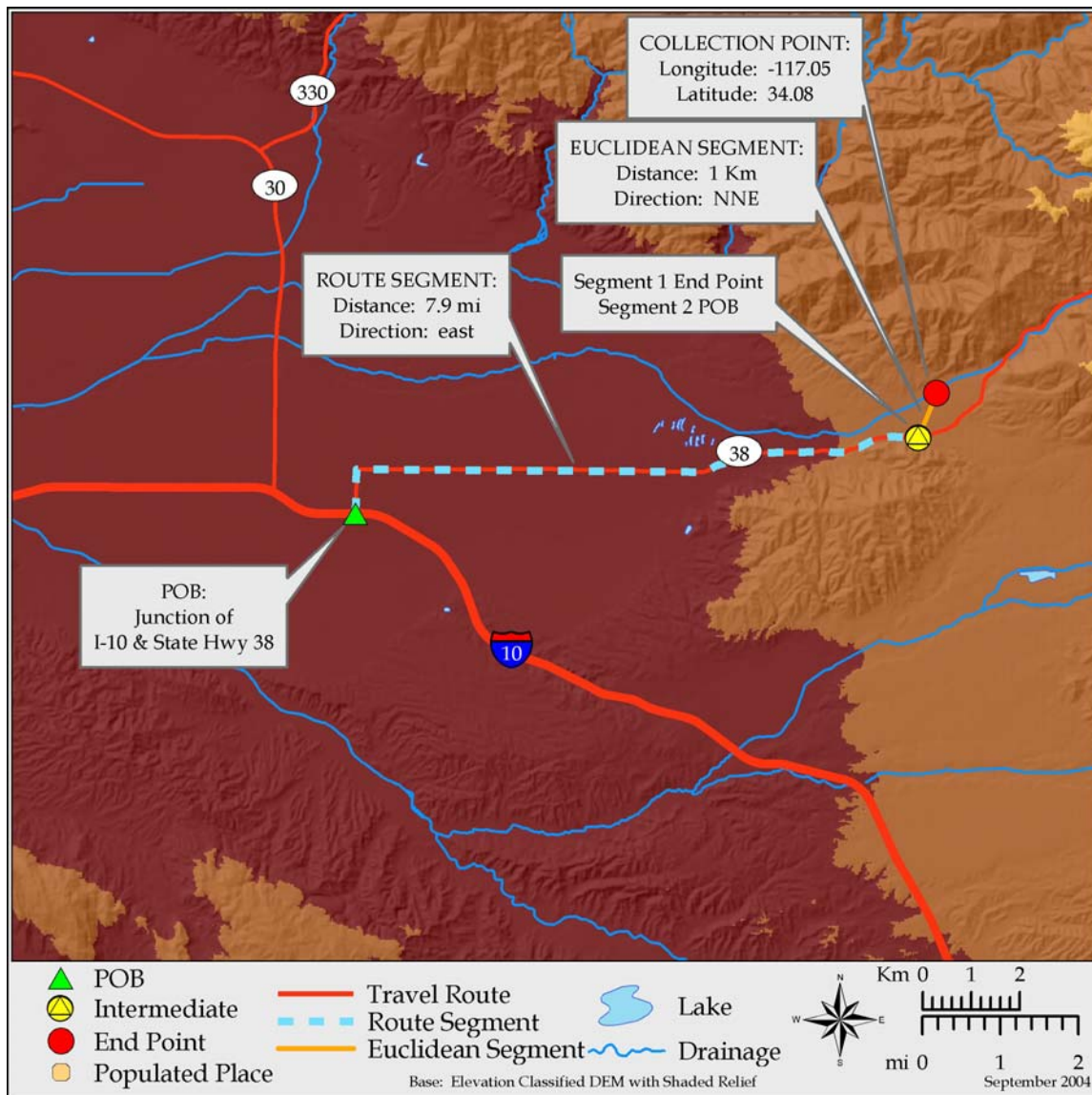


Figure 17 - Mixed Euclidean and Route Traverse Legs

5 SUMMARY AND CONCLUSIONS

Research and development efforts expended during the course of this project result from the fact that much of the useful data represented by museum specimen collections is not spatially enabled and cannot be used by contemporary GIS. The data represented by these collections is vital to current studies such as biodiversity and require development of geocoding or locator services that calculate coordinate values from text based location data. Historically, museum collection provenance is provided only in hand written descriptive format. This provenance information may be distributed between several related source documents. Significant expenditure of effort in research and data entry is required to prepare these data for a geocoding service. The vast quantity of these data necessitates use of geocoding services that provide both natural language processing and geometric calculation.

These two processes require very different types of software; one dissects text for its components and meaning, the other performs geometric calculation. Once spatially meaningful text elements of location descriptions are identified then calculation of geographic coordinates is possible using only those functionally identified elements of text as input. As part of this, the geoparsing and geoprocessing functions must share the same set of geographic names (gazetteer). The parser requires them for its list of acceptable terms (lexicon) and the processor requires them for their associated location data. These two processes may be run on disparate systems by different personnel. Accordingly, they require a method for interchange of data that is modular, web enabled, and well suited to store formatted data; XML is recommended.

No geoparsing was undertaken for the project. Instead, one specific method of location description was identified for use in project development. This method is called a traverse and refers to descriptions that provide an identifiable starting place with distance and direction of departure from the starting point. Specific elements of text were identified for the traverse and incorporated into an XML schema. Additional elements of the XML were added from the museum collection database to provide geographic reference for error checking of calculation results.

Geoprocessing development was undertaken for the project. Two functional VBA forms were constructed within ArcMap to calculate locations from text elements present in two types of traverse descriptions (along line segments and along established routes). One form uses Euclidean geometry and the other uses linear referencing to calculate coordinates. Each form requires input of a subset of the same data specified in the XML schema for that particular type of

description. Successful development of the forms provided a means to test and confirm the viability of XML element content for use in geoprocessing.

Project development has attained a distinct measure of success. This occurred despite difficulties that were encountered. Significant portions of project development have been identified but not attained. Some of the difficulties and recommendations for future continuation of this development are provided below.

5.1 Lessons learned

Impediments to progress in the project were encountered throughout the course of development. The biggest lesson here serves as the driving force behind this project. Geographic information systems require spatial data and attempts at development should not be undertaken until the required spatial data are available or proven methods to produce the spatial data are identified.

Linearly Referenced Data

Development of route data suitable for use in geoprocessing was accomplished with much work. Available shape files required much simplification and regeneration of measure values.

Raster Data

Raster data should be acquired in seamless sets for study areas whenever possible. Tiled raster data is poorly suited for display or analysis with projected coordinate systems due to overlap at margins. Significant time was expended by attempts to compensate for overlapping raster tile margins.

Software Documentation

Modification efforts occasionally required an in depth understanding of file structures; highway routes in coverage format serve as an example here. Documentation is poor for coverage files' structures and content. The methods for discovering and joining attributes associated with geometry in coverage format would have been impossible without advice from experienced users. It almost seems like this information is institutional.

5.2 Recommendations for Further Geocoding Development

Development effort for this project provides only a fraction of required functionality for the geoprocessing side of geocoding. Suggestions for additional work are provided below. The section on database administration is drawn primarily from experiences with the herpetology collection but recommendations are generally applicable to similar collections.

5.2.1 Collection Database Administration

Geoparsing that produces consistent and useful results requires uniform quality input data. The collection database table must be thoroughly edited and proofed prior to use as input for geoparsing. Data content must be consistent within and between fields. The primary digital source of information for geocoding must be in optimal condition for use as input data. Review and use of the LACM herpetology collection database has provided insights on appropriate format for collection location information. These insights apply to this particular collection but may be extended to similar collections as well. A set of actions considered essential for entry of historic data or correcting existing descriptive data are listed here:

- Discontinue entry of fractions into text descriptions. Fractions should be converted to a decimal equivalent and marked as a converted fraction (e.g. $\frac{1}{2}$ = f0.5). Marking converted fractions allows for compensating for their (according to MaNIS) inherent levels of precision (e.g. $\frac{1}{2}$ = 0.5 ± 0.25).
- Develop method to generalize range of distance ($\frac{1}{2}$ – $\frac{3}{4}$ mile) to a point with appropriate level of precision assigned. If multiple specimens were collected over the range, a method should be developed to distribute the collection locations over that range.
- Institute expert proof reading and spell check of manually entered data.
- Eliminate duplicate (redundant) material (in multiple columns).
- Sort data to appropriate field during data entry (e.g. delete elevation values from Locality or Remarks fields and enter in Elevation field)
- Identify all spatially unique collection localities, create a new table from them, and relate them to HerpColl with a foreign key.
- Develop a list of commonplace name abbreviations for the natural language processor's grammar and dictionary of valid place names.
- Identify and retain qualified data entry personnel and provide them with consistent guidelines for data entry procedures.
- Include a field that specifies the unit of measure for elevation.
- Establish and maintain description of each field and acceptable content.

5.2.2 Geoprocessing Methods

Geoprocessing development for this project constitutes a small fraction of requirements for calculating coordinates from descriptions. Future development should implement methods to handle multiple segments along a traverse. Methods to calculate location for additional types of location descriptions are needed as well. Suggestions for further development are provided below.

Description Types

- Locations described as place names require little additional development. They can be considered as a special case for the currently developed form where there is no distance.

- Locations described as place names with buffers can be treated as above but the new area introduced by the buffer must be considered for a level of precision determination.
- Location descriptions including intersecting natural and/or cultural features as POBs require functions that identify the appropriate type of intersection (or intersections) between feature types and represented by different geometry.
- Traverse descriptions that include multiple legs in the traverse require a method to retain the end point of one leg as the POB for the next consecutive leg.
- Software is already developed to generate coordinates for descriptions in Public Land Survey System format. This should be used if possible, as a module in further development.

Precision Determination

- Standardization of methods used to establish precision of results is essential to the reliability and consistency of data. MaNIS currently provides a standard that could be utilized as an internet service.
- A determination of the actual horizontal extent of a place name is essential to establish precision of calculated locations.

6 WORKS CITED

Allen, Robert B.

- 2004 "A Query Interface for an Event Gazetteer." *Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries (JCDL'04)*. 7-11 June 2004. Association for Computing Machinery and Institute of Electrical and Electronics Engineers, Inc. Tucson, Arizona. pp 1 - 2.

Beaman, Reed, John Wieczorek, and Stan Blum

- 2004 "Determining Space from Place for Natural History Collections in a Distributed Digital Library Environment." *D-Lib Magazine* 10.5: 28 April 2004
<http://www.dlib.org/dlib/may04/beaman/05beaman.html>

Bryant, James M.

- 2004 "Background" presentation. Accessing Historic Collections with GIS for Effective Land Management and Stewardship. Project Development Workshop hosted by Environmental Systems Research Institute. Redlands, California. 29 - 30 April 2004.

Carbajales, Patricia and Kevin Johnson

- 2002 *Creating an Internet Mapping Service to Communicate Amphibian and Reptile Species Records in a Natural History Museum Collection*. Project report prepared by University of Redlands, MS-GIS program for the Natural History Museum of Los Angeles County.

ESRI (Environmental Systems Research Institute)

- 2001 *Exploring ArcObjects Volume II - Geographic Data Management*. Redlands: ESRI.

GEOLocate

- 2004 University of Tulane. *GEOLocate Home Page*. Museum of Natural History, Tulane University. 18 August 2004.
<http://www.museum.tulane.edu/geolocate/default.aspx>

LACM

- 2003 Natural History Museum of Los Angeles County Foundation. *Research and Collections*. 2002. Natural History Museum of Los Angeles County. 18 August 2004
<http://www.nhm.org/research/index.html>

Loper, Edward

- 2003 Natural Language Tool Kit (version 1.2) Recursive Descent Parser Demonstration from NLTK Tutorial: Parsing. SourceForge.Net 2003
http://sourceforge.net/project/showfiles.php?group_id=30982
5 November 2003

MetaCarta

- 2004 MetaCarta, Inc. "MetaCarta Products." 2004
18 August 2004
<http://www.metacarta.com/site/products>

McGranaghan, Matthew

- 1989 "Context-Free Recursive -Descent Parsing of Location-Descriptive Text." *Proceedings of the Ninth International Symposium on Computer-Assisted Cartography*. 2 - 7 April 1989. American Society for Photogrammetry and Remote Sensing and American Congress on Surveying and Mapping. Baltimore, Maryland. pp 580- 587.

Smith, Barry and David M. Mark

- 1998 "Ontology and Geographic Kinds." *Proceedings of 8th International Symposium on Spatial Data Handling (SDH'98)*. 12-15 July 1998. Vancouver, Canada (International Geographical Union). pp 308-320.

Talmy, Leonard

- 1983 "How Language Structures Space." *Spatial Orientation, Theory, Research, and Application*. Pick, Herbert L., Jr. and Linda P. Acredolo (Eds.). Plenum. 225-282

Wieczorek, John

- 2004 *MaNIS/HerpNet/ORNIS, Georeferencing Guidelines*. 22 Jun 2004. University of California, Berkeley. 18 August 2004
<http://dlp.cs.berkeley.edu/manis/GeorefGuide.html>

Zeiler, Michael

- 1999 *Modeling Our World*. Redlands: ESRI.

APPENDIX 1: Spatial Semantics in Descriptions of Location

An Independent Study prepared by:

Robert F. Johnson

For

University of Redlands

MS-GIS Program

GIS-763.02

17 December 2003

1. INTRODUCTION

The following study relates spatial semantics of location descriptions and discusses their relevance to geographic information systems (GIS). This study was implemented as part of a larger project, an internet mapping service for natural history museum collections. Such a mapping service requires that data sets include location in graticule or coordinate format. One relevant problem this study seeks to address stems from the situation that provenance (collection location information) for many museum specimens is recorded only as a terse text description. The description, *1 mile N Del Rosa Ave. in Quail Canyon*, exemplifies the style and quality of spatial record for many thousands of collection localities (see selected samples in Table 3). These descriptive locations must be spatially referenced (interpreted and assigned useable values) prior to use in the GIS that underpins the mapping service. A clear understanding of spatial semantics is an essential element for success of the mapping service. It is necessary for reliable interpretation of meaning from locations' descriptions and is required prior to generation of accurate numeric spatial values.

The study's overall goal is development of a framework to identify and extract spatial meaning from location descriptions. Several related goals necessary to this end are to:

- define spatial semantics,
- gain understanding in how spatial semantics is relevant to GIS,
- identify and describe the larger context of which spatial semantics is part,
- identify and describe other related elements of the larger context,
- inventory and relate the elements pertinent to semantics for description of locations,
- inventory types of spatial knowledge represented by location descriptions, and
- identify computer languages and/or software related to the geographic interpretation of descriptions of location.

2. METHODS

Different research and development methods were used for identification and assembly of the myriad concepts from this study, into coherent relationships. Information was drawn primarily from library texts and a variety of internet resources. Concept mapping aided organization of subjects' material.

Research consisted of a literature review and was conducted using a variety of sources. Physical and digital holdings were consulted at the University of Redlands' Armacost Library, the University of California – Riverside's Rivera and Science Libraries, and the Environmental Systems Research Institute's Library. Generally, the most up to date and relevant material was available through the UC-Riverside Science Library. A major asset there was access to *Lecture Notes in Computer Science* (Springer-Verlag *various*). The internet also provided access to additional resources including: published journal articles and conference proceedings in addition to unpublished conference presentations (Torres 2002) and material from relevant courses at University at Buffalo (Donnelly 2003).

Concept mapping was undertaken at the suggestion of Dr. Karen Kemp. Several completed sessions were useful in relating spatial semantics with ontology for geographical kinds of concepts, cognitive processes, and schemas for spatial scenes. See Appendix 1.3 for an illustration of concept mapping for components and schemas of the spatial scene.

3. FINDINGS

Study results are presented below. The results present and discuss elements of:

- forms of semantic representation,
- specifications for spatial representation,
- human perception of spatial properties,
- components and portrayal of spatial relations, and
- computer software needs.

Examples of location descriptions follow presentation of the findings.

3.1 DEFINITION OF TERMS

Many of the semantic concepts and spatial relations presented below reference each other and can prove somewhat problematic during the first reading of the text. A few definitions are essential and are listed below for basic relevant terms.

3.1.1 SEMANTICS

The term semantics refers to *the meaning of a string in some language* (WordNet 1.6) where string is a word or phrase comprising a sequence (or string) of characters. The critical emphasis here is on the meaning communicated. Note that semantics should not be confused with the term syntax which describes how symbols (the strings of characters) may be combined independent of their meaning.

3.1.2 SPATIAL

The term spatial is: *of, pertaining, or relating to space which is defined as denoting area or location*. This includes all scale (microscopic to cosmic) and contrasts in scope with the notion of geographical scale which is limited to space at or near the earth's surface (Mark 1993). The use of spatial in this study refers exclusively to conceptual entities and physical objects in geographical space.

3.1.3 ONTOLOGY

This term originated in philosophical usage as a systematic account of Existence. It has been adopted for use in the world of artificial intelligence as an explicit formal specification of how to represent the objects, concepts and other entities that are assumed to exist in some area of interest and the relationships that hold among them (WordNet 1.6).

3.2 SEMANTIC REPRESENTATION

Spatial semantics applies to meaning derived from any method for communicating spatial characteristics of geographic information. The spatial relation between objects or entities in space forms a sub-set of characteristics. Linguistic, graphical, and mathematical methods are available to record spatial relations in forms such as text, Euclidean geometry, imagery, and set theory. Its potential for use with GIS is an appropriate area of study as significant geographic (spatial) data was originally recorded and continues to be stored only

as descriptive text. This study reports basic considerations for the role of spatial semantics in computer interpretation of textual descriptions of location.

3.2.1 LINGUISTIC EXPRESSION

The spatial semantics of interest for this study are based in natural languages and represent spatial relationships between objects of geographic utility. Written and spoken forms of natural language (e.g. English) provide people the means to communicate physical relationships between objects or entities. Differences within and between individual languages and the cultures using them, may promote disparity between interpreted meanings. Multiple meanings for common phrases are indicative of this problem. The phrase “over the hill” is somewhat illustrative of this point. Meaning is highly dependent on the subject, context of use, and culture. Temporal change may also result in disparity not only within the language but also in objects’ and entities’ properties. Administrative boundaries change, populated places can grow in size with time, or either can vanish completely.

The focus of the current study is on written descriptions of natural history collection localities. Computer interpretation of these has been a GIS related study interest for some time (McGranaghan 1989). These terse descriptions are seldom grammatically correct sentences (e.g. “1 mile N Del Rosa Ave. in Quail Canyon”). See Table 3 for additional examples.

3.2.2 ARTIFICIAL INTELLIGENCE AND MATHEMATICAL REPRESENTATION

Artificial (or formal) languages represent real world concepts and relationships directly in or easily translatable to mathematical formats that can be interpreted by computers. Development of artificial language with the functionality to correctly interpret meaning (semantics) from natural language (text or audio) is a decades long challenge. Computer systems with abilities in artificial intelligence for interpretation of natural language are referred to as “expert systems”. These systems are however, typically limited to one domain of knowledge.

There are several methods for mathematical representation of spatial relationships. Geographic objects can be represented as geometric objects in three dimensional space. Relationships between objects can be quantified and expressed through arithmetic, geometric, set, or trigonometric functions.

3.2.3 GEOGRAPHIC INFORMATION SYSTEMS AND GRAPHIC REPRESENTATION

Geographic information systems have relied exclusively on location data that is stored in or references to coordinate or graticule based formats. Much spatial information is recorded only in a text based format. Any advance in computing

systems' abilities to identify, interpret, and translate natural language spatial relations to a format that can be interpreted by formal languages is highly desirable.

Two dimensional graphics can portray and store spatial relationships between three-dimensional objects in space. *Haptic* and *transperceptual* space are converted to *pictorial* space (see below) and planar (two-dimensional) spatial relationships can be portrayed in plan or section views.

3.3 ONTOLOGY OF AND COGNITIVE CONTEXTS FOR SPATIAL SEMANTICS

Effective spatial semantics require consistent definition of nomenclature, concepts, understanding, and interrelationships. This is attained through identification of ontology and realization of cognition. Ontology specifies their conceptualization and cognitive science explains the foundation of human spatial understanding.

3.3.1 ONTOLOGY

Spatial semantics reside within a larger context of spatial understanding referred to as ontology. Originally coined for use with philosophy, ontology has been adapted for various disciplines including artificial intelligence and computer science. An ontology developed within the geographic domain promotes a "better understanding of the geographic world" (Smith and Mark 1998). Effective consistent communication is possible only when all involved share the same ontology.

Clarity of semantics is ensured when ontology addresses all aspects of spatial entities and their relations. A geographic ontology must specify basics such as the definition and nature of space itself in addition to types of spatial entities (or objects) and their boundaries. The ontology must address topological (theory of boundaries, interiors, connectedness and separation), mereological (theory of part and whole), and physical spatial (dimensionality and geometry) properties. Geographic representations must discriminate between those objects or entities that are delimited by physical characteristics and those contrived by human (cultural/artificial) conceptualization. Represented borders' origins must be identified as *bona fide* (physically based) or *fiat* (contrived). All of these specifications must be set in order to identify the relevant linguistic elements that represent each property in a textual description of location. See Appendix 1.2 for an example of a geographic ontology.

3.3.2 COGNITIVE SCIENCE

Cognitive Science is a multi-disciplinary endeavor comprising Artificial Intelligence, Linguistics, Anthropology, Psychology, Neuroscience, Philosophy, and Education (Cognitive Science Society 1998). Its benefit to this study is the foundation of human spatial cognition it provides. Basic components of spatial relations discussed below include: cognitive clarity, spatial processes, types of knowledge, sources of information, scale, frame of reference, and relative perception.

3.3.2.1 Relative Spatial Perception and Cognitive Clarity

There are differences in perceptual clarity for spatial relations between objects. The results of controlled tests indicate variation in response time for human recognition of the spatial relationships between viewed objects (Bryant et al. 1992). The time between sensory perception through vision and full cognition of the viewed spatial relationship, increases through a series of three types of relationships. These relationships may suggest degrees of reliability for manually observed and recorded spatial observations. The three situations are presented below.

Environmental (gravity)

People can rapidly discern vertical relationships between viewed objects; these are above and below. They clearly recognize this relationship more quickly than any other spatial relation. The relationship is based on gravity, a primary environmental variable. It is interesting to note that speed of cognition is not impaired by the orientation of the observer's body (standing, sitting, or reclining) at the time of the observation.

Egocentric view shed

Some horizontal relationships can be quickly ascertained. The next most easily discerned spatial relationship involves relating viewed objects to a person's physical point of view. Objects are easily recognized as being in front of or behind the observer; apparently after a complete panorama of a scene has been viewed.

Egocentric direction

Other horizontal relationships are not so quickly recognized with visual means. The least salient of the three relationships does not have the clear binary (logical) basis (up/down, in front, behind) apparent in the first two. Horizontal direction relative to oneself is dependent on discriminating between right and left. This is not a binary relationship but rather occurs across somewhat of a gradient of "degrees", so to speak, of leftness or rightness similar to degrees of azimuth.

3.3.2.2 Spatial Cognitive Processes

People learn about spatial relations by various mental processes. These include: experiencing it in real time (perception), recalling experiences and other sources of spatial facts (memory), and inferring new concepts from perception and memory (reasoning). Note that sources of information and frames of reference both contribute to these processes but are discussed separately and afterwards.

Perception

Perception refers to [re]cognition of spatial knowledge and relations through an individual's own sensory experience. The individual is in a position where direct contact with objects is possible. Their relationships in haptic space are learned through direct sensory inputs (sight, touch, sound, etc.). The individual becomes the frame of reference for observation of spatial relations during direct interaction with, and possible manipulation of objects.

Memory

Memory serves individuals' cognition of spatial relations by providing "declarative" and stored pictorial spatial knowledge. Attributes of spatial objects and relationships between them have been previously considered by the individual, stored for future use, and are recalled for use as needed from memory.

Reasoning

Reasoning allows individuals to infer valid spatial relations between given objects. This is based on knowledge of relations already in memory or remembered relations in conjunction with perception.

3.3.2.3 Types of Spatial Knowledge

Location descriptions may relate meaning through different types of spatial knowledge. The descriptions include information from different sources; these probably include the recorder's observations mixed with personal records and those of others. Four types of spatial knowledge are identified (Mark 1993) and explained below.

Declarative Knowledge ("Geographic Facts")

Declarative spatial knowledge consists of rote knowledge like important facts memorized in a course of study. These facts include attributes of and relations between geographic objects at a given time. Redlands has a population of 65,000 people, an elevation of 1,350 feet, and is upstream from Riverside are all examples of declarative spatial knowledge. It is apparent not all declarative knowledge conveys spatial relationships and in that sense can be non-dimensional. It is also apparent that declarative knowledge frequently incorporates and/or conveys spatial semantics.

Procedural Knowledge ("Navigation")

Procedural spatial knowledge consists of "navigational" knowledge required to make the traverse from a starting point to an ending point along a specified path. This involves both spatial and temporal relationships. A sequence of points are reasoned or remembered and consist of beginning, intermediate way points, and an ending point. These points are salient enough to recognize as are the spatial relationships between them and the temporal (procedural) sequence they must be visited.

Configurational Knowledge ("Euclidean")

Configurational knowledge involves a map like knowledge of relationships that is or approximates Euclidean geometry. It need not be complete or perfect and can serve even at a simple connectivity (or topological) level of understanding (Mark 1993).

Transformational ("Reasoning")

This is knowledge that is converted to one type from another type of the three knowledge types listed above. Looking at a map (configurational) and deciding on a route of travel (procedural) is one example of transformation. Not every transform direction is accepted as normally inferred for all combinations of knowledge types (e.g. configuration from procedure).

3.3.2.4 Sources of Spatial Information

Spatial information is organized and identified by its increasingly complex levels of reasoning. Understanding of each level of increasing complexity depends partly on metaphorical translation of underlying knowledge.

Haptic space

We recognize haptic space through direct perception. Haptic space is the tangible, three-dimensional personal space we live in and recognize through physical interaction and sensory input. Spatial relations are constantly subject to change through time. Direct sensing of haptic space provides much detail for geographically small objects and at very large geographic scale. It provides opportunity to learn spatial relations from direct contact and close observation from different points of view.

Pictorial space

We perceive pictorial space primarily but not exclusively through vision. Pictorial space includes three-dimensional sensed views of haptic space. Two-dimensional pictures may be translated by the mind into a three-dimensional view. Any image of space is at one point in time and from one observation point. Spatial relations are static and cognition of spatial relations is limited to those

visible at that time and point in space. It provides opportunity to view spatial relations of objects more geographic in nature and on a smaller scale. This type of indirect and limited sensing of haptic space does provide opportunities not afforded by direct sensing. Examples are single vistas taken from points not available to humans and easy view of multiple vistas from multiple view points.

Transperceptual space

People do not directly perceive transperceptual space but rather mentally assemble (reason) it from memory of multiple instances of reality taken from pictorial space. This assembly may include memory and perceived space. It allows human cognition of spatial relations that exist beyond a single instance of human perception. For most of human experience, it related to substantial “mesoscopic” space assembled from memories of multiple “vistas”. Given current technology, it apparently includes both microscopic spaces too small to be directly perceived and geographic objects so large that humans could not be cognizant of them even through multiple perceived “vistas”.

Metaphorical translation

It is proposed that perceived (haptic) space is the most basic form of knowledge. Pictorial is more complex in nature and transperceptual space is most complex. They are possible in part through metaphorical translation (reasoning) of knowledge derived from the simpler (more primitive) underlying sources of spatial information (Mark 1993).

3.3.2.5 Spatial Scale

The scale, at which spatial information is presented, provides an indication as to how it was likely obtained. From scale one can infer source, type of spatial knowledge, and cognitive processes required to render it.

Geographic

The range of information present in geographic scale is transperceptual; beyond human perceptual capability. Cognition is possible only by reasoning through pictorial space, memory, and metaphor translation.

Mesoscopic

This comprises geographic entities ranging in extent from more than one to an extent available only through many perceptual acts; it too relates transperceptual space. Cognition is possible through perception and/or memory of pictorial space and requires reasoning.

Macroscopic

This comprises small entities of geographic nature, large and “table top” objects. Cognition is possible for all of these within a single perceptual act in haptic space.

Microscopic

This comprises entities and objects invisible to the unaided eye. Range is smaller than human perceptual capability. Cognition is possible for this scale only through pictorial space and by reasoning with metaphor translation. This scale is not germane to this study.

3.3.2.6 Frame of Reference

The frame of reference is directly related to the type of spatial knowledge available to the observer.

Direct Perception

Observer is physically adjacent to or within a spatial scene. Haptic space is perceived at a macroscopic scale and provides information.

Indirect Perception

Observer recalls information from memory or uses recorded form of knowledge such as viewing a picture/rendition of scene or reading declarative information. This requires the observer provide considerable reasoning, interpretation, and metaphorical translation from lower to higher information complexity levels.

3.4 ELEMENTS PERTINENT TO SEMANTICS OF LOCATION

Breaking out components and relations from the scenario presented in a location description is necessary in finding its meaning. The following section presents the components of spatial scenes, linguistic representation, and topological relationships as significant elements for identification of spatial semantics.

3.4.1 SPATIAL SCENE

Spatial relations between objects can be effectively detailed in a “spatial scene” (Talmy 1983) and the essence of these scenes is contained in the schemas that portray them. Spatial scenes portray spatial relations by presenting geographic objects and their context as primary and secondary objects. Primary and secondary objects are in a figure/ground relationship respectively; the scene however, may incorporate additional reference objects.

Both primary and secondary objects include cognitive, spatial, temporal, and geometric characteristics that contrast between figure and ground. Various schemas are used to relate the elements of the spatial scene. These schemas form

relations amongst themselves, contain properties, and convey figure/ground relationships and geometries of spatial scene objects.

Components of the spatial scene should relate closely to linguistic structures used to describe them. Considerations presented below include: representative geometry, biased referencing, relative position, and absolute orientation.

3.4.1.1 Object Traits

Object properties are most germane to this study. Figure and ground objects' characteristics and spatial relations between objects are identified and listed in Table 1 below.

Table 1: Traits of Primary and Secondary Objects within a Spatial Scene

| | | C H A R A C T E R I S T I C S | | | | |
|----------------------------|--------------------|-------------------------------|---------------------------|------------------|----------|---------------|
| | | COGNITION | SPATIO-TEMPORAL | SPATIAL | GEOMETRY | RELATIVE SIZE |
| O B J E C T | PRIMARY (Figure) | Perception | Relatively New or Mobile | To Be Determined | Simple | Small |
| | SECONDARY (Ground) | Memory | Relatively Older or Fixed | Known | Complex | Large |

3.4.1.2 Scene Schemas

Relational and Geometric Components

Both spatial relations between objects and their geometries within the spatial scene are considered components of the schema. Components include spatial relationships between objects, figure and object geometries, and biased referencing for individual objects. Spatial relations between objects within a scene convey relative position and relative orientation. This can be between primary, secondary or tertiary (other reference) objects. Figure geometry is usually simpler than ground geometry and is typically conveyed as a point or line that is fixed or moving. The ground object serving as reference for the figure object is frequently the more complex of the two. Geometric simplicity for objects is determined by degree of dimensionality; fewer dimensions equal a simpler object. A point object with zero degree of dimensionality is considered simpler than an area object with two degrees of dimensionality. Ground object(s) can be: one or more point objects, a line or linear enclosure, a bounded plane, or a single volume. Geometry of figure and ground objects can facilitate

reference to the object based on differential physical “biased reference” attributes. Objects may possess distinguishable parts, non-symmetrical directedness, or earth based directedness. Distinguishable parts are named elements of an object that have a morphological basis for differentiating them from other parts of the object. The crest of a ridge or the toe of a slope can serve as examples. Non-symmetrical directedness refers to characteristics of objects with uniform morphology that serve to differentiate parts of the object based on their direction. The start and end of either a queue of people or a flowing stream provide examples. Earth based directedness derives from basic environmental factors related to the planet. Relative positions can be based on the gravitation field (vertical position) or cardinal directions (horizontal position). Consideration of personal frame of reference is involved when tertiary reference objects are present.

Schema Properties

Schemas require linguistic representation of real objects in space. Three key properties for schemas representing spatial scenes include: idealization, abstraction, and topology. Idealization is somewhat akin to cartographic generalization; making a simple yet effective representation of a complex object. This is done so many types of objects can be represented with only a few simple geometric shapes. Abstraction follows this in a somewhat complementary fashion. Idealization has taken the geometric essence of the object and abstraction ignores the rest of the object’s specific physical characteristics, making them irrelevant for use in the schema. Once a spatial scene has been idealized and abstracted, shape and magnitude are irrelevant. Topology remains as the only adequate method for relating spatial relations for these minimalist geometries.

Schema Relations

There are alternate schemas for representing objects in space. Properties idealized or abstracted away in one schema may be pertinent in another. It is suggested that [different] language use may filter or bias alternate schematization for the same spatial scene. Alternate schemas probably do not follow a continuum; rather they form a disjunct set. A number of considerations contribute to disjunct sets. One example is the over and under specificity of features within a schema. Corrections may be effected to minimize deleterious effects of disjunct schemas.

3.4.2 SPATIAL LEXICON

The set of words from a natural language, adequate for description of spatial relations is referred to as the spatial lexicon. It is essentially a vocabulary that is dependent on the language user’s knowledge. The spatial lexicon is frequently portrayed as a “closed set” (very limited in membership) and composed

primarily of prepositions. See Table 2 for a summary of the spatial lexicon. Note that while prepositions serve in a substantial capacity and primarily to describe spatio-temporal relations; other types of words also convey spatial relations. Adjectives can serve to supply biased references for objects. Spatial objects and earth based directions are nouns. Numerous verbs frequently indicate spatially directed action relative between an object and the earth (descend/rise) or between two objects (cross) or from an egocentric point of view (proceed/retreat).

Table 2: Closed Class Spatial Lexicon

| PARTS OF SPEECH | SPATIAL ROLE |
|------------------------|---|
| ADJECTIVES | Modifiers relating biased reference such as placement (upper) or directedness (e.g. northerly) |
| PREPOSITIONS | Some refer simply to spatial relations but some include mixed spatio-temporal usage (See Appendix 1.1) |
| NOUNS | Horizontal and vertical direction (e.g. north, top) |
| VERBS | Connote spatial displacement or direction associated with action (climb, descend, proceed, retreat, cross, traverse). |
| CONJUNCTIONS | Multiply or Exclude |

3.4.3 TOPOLOGY

Topology serves well in natural language to expediently express spatial relationships in non-Euclidean terms yet convey significant meaning. A set of specific prepositions serve this function and can be seen in many location descriptions. One example is “0.05 mi E Pisgah Crater Rd. on Nat'l Trails Hwy” indicating a collection point on a roadway generalized to a line in space.

3.5 SOFTWARE RELEVANT TO SPATIAL SEMANTICS

3.5.1 NATURAL LANGUAGE PROCESSING

Most computer languages have integral functionality for handling text strings but few of them are specifically designed for processing and interpreting meaning from text. Some form of Natural Language Processing (NLP) software must be used to identify pertinent elements of the descriptive text and how they are related to each other. It must also provide structured output useable for generating instructions required to calculate a coordinate or graticule referenced location.

3.5.2 SPATIAL AND NON-SPATIAL RELATIONAL DATABASES

A relational database developed for storing museum collection data includes a field exclusively for a text description of the collection locality. This type of software can produce files useful for storing both text description and geographic coordinate data that can be directly utilized by GIS.

3.5.3 TEXT/WORD PROCESSING

Journals and accounts of travelers or explorers can be stored as text files. File from this software are useful only as a storage medium for text. Geographic coordinate data could also be stored but would require conversion to another format prior to use by GIS.

3.5.4 DIGITAL GAZETTEERS

Web-based digital gazetteers such as the Alexandria Digital Library (ADL) or the USGS Geographic Name Information System (GNIS) provide locations for place names. Locations are provided as geographic graticule coordinates (latitude/longitude) with a one arc-second precision.

3.5.5 GEOGRAPHIC INFORMATION SYSTEMS

Information available from geographic information systems is a consideration in this study. Road network information is a significant component of study of location descriptions when many of the locations are referenced and adjacent to features of the roads.

3.5.5.1 Transportation Network

Geographic information systems contain and manage all arc/node elements that constitute a road network. They can also return coordinate data for those elements for user queries. This provides a significant source of information relevant to the many descriptions of locations that are based on roadway names and features (e.g. intersections).

3.6 QUASI-NATURAL LANGUAGE LOCATION DESCRIPTIONS

Samples of location descriptions taken from the Los Angeles County Museum's herpetology collection are provided below in Table 3. They illustrate considerations for working with descriptions that are not grammatically correct.

Table 3: Examples of Collection Location Descriptions

| | Location Description | Type | Crisp/ Fuzzy | Figure/ Ground Type |
|----|---|--------------------------------|-----------------|--|
| 1 | 0.05 mi E Pisgah Crater Rd. on Nat'l Trails Hwy. | Navigation | C | Point on Road Network |
| 2 | 0.1 mi E Lucerne Valley | Euclidean | F | Point Relative to Place Name |
| 3 | 01099 Varner Road, 5 mi. W. Thousand Palms | Geocode & Euclidean | C | Point on Street Address Network or on Place Name |
| 4 | 1 mile N Del Rosa Ave. in Quail Canyon | Navigation | C | Point on Road Network & Within Place Name Area |
| 5 | 1 mi NW Camp Angelus, HWY 38 at old Cold Creek | Euclidean and Navigation | C | Point Relative to Place Name at Road Network Node |
| 6 | Base of San Jacinto Mts., near Banning | Euclidean | F | Point Relative to Place Names |
| 7 | Between Niland and Blythe | Euclidean | F | Point Relative to Place Names |
| 8 | Big Bear Lake Metcalf Bay | Euclidean | C | Point Within Place Name Areas |
| 9 | near Palm Springs | Euclidean | F | Point Relative to Place Names |
| 10 | Palms to Pines Hwy, Coachella Valley | Navigation | F | Line Within Place Name Areas |

WORKS CITED:

Bryant, David J., Barbara Tversky, and Nancy Franklin

- 1992 Internal and External Spatial Frameworks for Representing Described Scenes, *Journal of Memory and Language*, 31(1):74-98.

Cognitive Science Society, Incorporated

- 1998 About the Society,
URL: <http://www.cognitivesciencesociety.org/contact.html>, Cognitive Science Society, Inc., Cincinnati, Ohio (last date accessed 16 December 2003)

Donnelly, Maureen

- 2003 Ontology and Cognition: Spatial Entities and Spatial Relations. State University of New York at Buffalo, Department of Philosophy class syllabus,
URL: <http://ontology.buffalo.edu/smith/courses03/md/SpSyll.htm>, (last date accessed: 15 December 2003).

Freeman, J.

- 1975 The modeling of spatial relations in *Computer graphics and image processing* 4:156-71 Academic Press

Mark, D. M.

- 1993 Human Spatial Cognition. In Medyckyj-Scott, D., and Hearnshaw, H. M., editors, *Human Factors in Geographical Information Systems*, Belhaven Press, 51-60.

McGranaghan, Matthew

- 1989 Context-Free Recursive -Descent Parsing of Location-Descriptive Text, Proceedings of the Ninth International Symposium on Computer-Assisted Cartography, American Society for Photogrammetry and Remote Sensing and American Congress on Surveying and Mapping, Baltimore, Maryland (University of Hawaii, Department of Geography presented paper), PP. 580- 587.

Smith, Barry and David M. Mark

- 1998 Ontology and Geographic Kinds, Proceedings of 8th International Symposium on Spatial Data Handling (SDH'98), 12-15 July, 1998, Vancouver, Canada (International Geographical Union), pp 308-320.

Springer Verlag

Varies *Lecture Notes in Computer Science*, restricted access web site, for content listing only see URL:

<http://www.springerlink.com/app/home/journal.asp?wasp=9hmwxlmuwruhe1lp8fvk&referrer=parent&backto=subject,30,49;,>

Springer-Verlag, Heidelberg (last date accessed: 15 December 2003).

Talmy, Leonard

1983 How Language Structures Space in *Spatial Orientation, Theory, Research, and Application*. Pick, Herbert L., Jr. and Linda P. Acredolo (Eds.). Plenum

Torres, Miguel

2002 Semantics Definition to Represent Spatial Data, International Workshop on Semantic Processing of Spatial Data, URL:

<http://geopro.cic.ipn.mx/Ptorres.pdf>

Centre for Computing Research, México, D.F., (last date accessed: 15 December 2003).

WordNet 2.0

2003 A Lexical Database for the English Language, Cognitive Science Laboratory, Princeton University, URL:

<http://www.cogsci.princeton.edu/~wn/index.shtml>.

APPENDIX 1.1 : Spatial Prepositions

SOURCE: English On Line

URL: <http://www.4-esl.com/Grammar/Intermediate/prepositions2.htm>

| PREPOSITION | USE | | | | | MEANING |
|-------------|---------|-----------------|-----------|--------------|-----------|---|
| | SPATIAL | SPATIO-TEMPORAL | EUCLIDEAN | NAVIGATIONAL | TOPOLOGIC | |
| Aboard | * | | * | | | within bounded plane |
| About | * | * | * | * | | near |
| Above | * | | * | | * | within area of bounding plane but higher |
| Across | * | | | * | | perpendicular to long axis |
| Across from | * | | * | | | diametrically opposed |
| Adjacent | * | | * | | * | bounding on one side |
| After | | * | | * | | further than secondary reference from moving perspective point |
| Against | * | | * | | * | touching but not within |
| ahead of | | * | | * | | closer than secondary reference |
| Along | * | * | * | * | | collinear |
| alongside | * | | * | | * | parallel and collinear |
| Amid | * | | * | | * | surrounded by a mass of points |
| Amidst | * | | * | | * | surrounded by a mass of points |
| Among | * | | * | | * | surrounded by a several points |
| Around | * | * | * | * | | approximately or within area of bounded plane |
| At | * | * | * | * | | a relation of proximity to, or of presence in or on |
| Before | | * | | * | | closer than secondary reference |
| Behind | * | | * | | | further than secondary reference and its perceived bounding plane |
| Below | * | | * | | * | within area of bounding plane but lower |
| beneath | * | | * | | * | within area of bounding plane but lower |
| Beside | * | | * | | * | see adjacent |
| between | * | | * | | | bounded by two points |

| PREPOSITION | USE | | | | | MEANING |
|-------------|---------|---------------------|-----------|--------------|-----------|---|
| | SPATIAL | SPATIO- TEMPORAL | EUCLIDEAN | NAVIGATIONAL | TOPOLOGIC | |
| beyond | * | | * | | | see after |
| By | * | * | * | * | | see adjacent |
| Close to | * | | * | | | at a short distance but not adjacent (see near) |
| Down | * | | * | | | at an increasingly lower level |
| Following | | * | | * | | like after but from fixed perspective point |
| For | * | * | * | * | | over a span of time or distance |
| Forward of | * | * | * | | | relatively closer to destination and between |
| From | * | | * | | * | indicates the original or starting position |
| In | * | | * | | * | completely surrounded by |
| In between | * | | * | | | bounded by two points |
| In front of | * | | * | | | same as forward of |
| Inside | * | | * | | * | Interior |
| Into | * | | * | | | towards interior |
| Near | * | | * | | | at a short distance but not adjacent |
| Next to | * | | | | | directly adjacent to and touching |
| Of | * | | * | | | indicates a separation in space |
| Off | * | | * | | * | near but not occupying same space |
| On | * | | * | | * | in contact with (usually upper surface) |
| Onto | * | | * | | | transition from not on to on |
| On top of | * | | * | | * | higher in elevation like above but must be touching |
| Opposite | * | | * | | | diametrically opposed |
| Out of | * | | * | * | | exterior to or not in |
| Outside | * | | * | | * | exterior to or not in |
| Over | * | | * | | * | like "above" but may be in contact |
| Past | * | * | * | * | | a point more distant than another from start point along a line or travel route |
| Prior to | | * | | * | | a point closer than another from a start point along a line or route |

| PREPOSITION | USE | | | | | MEANING |
|-------------|---------|---------------------|-----------|--------------|-----------|--|
| | SPATIAL | SPATIO- TEMPORAL | EUCLIDEAN | NAVIGATIONAL | TOPOLOGIC | |
| Through | * | * | * | * | * | starting or going in one side passing through the interior and continuing out the other side |
| To | * | | * | | * | marks ending point or destination |
| Toward | * | | | * | | approaching, coming near |
| Towards | * | | | * | | indirection or vicinity of |
| Under | * | | * | | * | within area of bounding plane but at lower elevation |
| Up | * | | * | | | from lower to higher |
| Upon | * | | * | | | same as on |
| Up to | * | | * | | | from lower point to and stopping at higher point |
| Via | * | | | * | | same as through |
| With | * | | * | | | indicative of distance separating two points |
| Within | * | | * | | * | same as in |

APPENDIX 1.2:

A Draft Geographic Ontology from the NCGIA Initiative 21 Workshop

(URL: <http://www.geog.buffalo.edu/ncgia/i21/i21ontology.html>)

- I. conduit
 - A. bi-directional linear
 - 1. route
 - a) path
 - (1) linear
 - (a) pass
 - (2) circular
 - (a) roundabout
 - (b) circular tourist route
 - B. unidirectional
 - 1. linear
 - a) outlet
 - b) inlet
 - c) one way streets
 - 2. circular
 - a) roundabout
- II. intersection
 - A. confluence
 - B. T junction
- III. landmark
 - A. mountain
 - B. clump of trees
- IV. space (always empty)
 - A. inside a room
 - B. inside the grand canyon
- V. place
 - A. [complete enclosure]
 - B. territory
 - C. home
 - D. neighborhood
 - E. region
- VI. topological features
 - A. surface

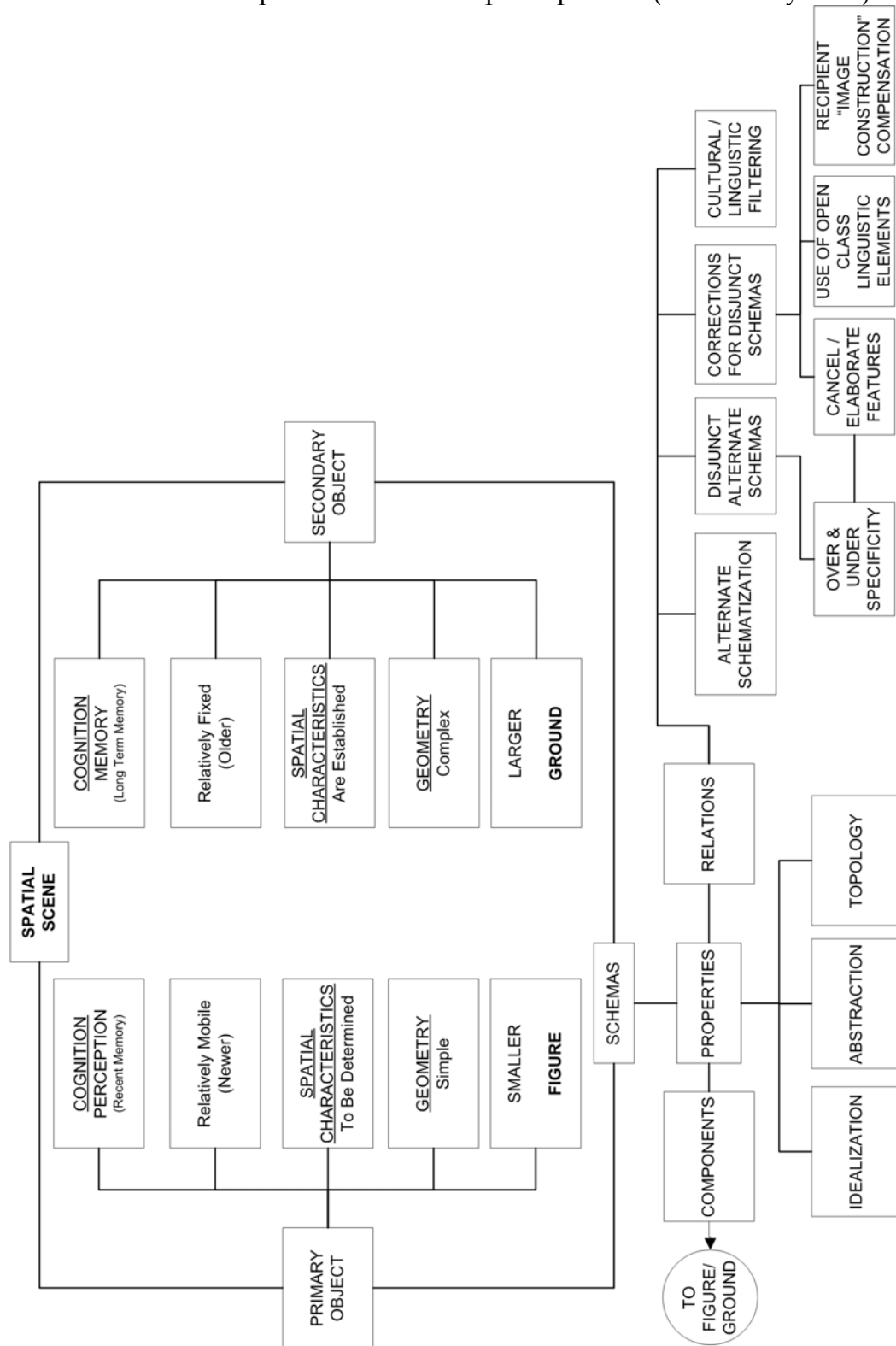
- 1. lake surface
 - 2. top soil
 - B. interior
 - 1. underneath lake surface
 - C. edge
 - 1. frontier
 - 2. barrier
 - 3. dam
 - 4. cliff
 - 5. shoreline
 - D. side/end
- VII. change
 - A. motion
 - 1. seasonal migration
 - 2. searching
 - B. [processes]
 - 1. natural processes
 - a) flooding
 - b) erosion
 - c) deposition
 - d) disease spread
 - e) periodic change
 - (1) tides
 - (2) seasons
 - 2. animate processes
 - a) producing
 - b) using resources
 - c) consuming resources
 - C. property change
 - 1. temperature change
 - 2. land use change
 - 3. color change
- VIII. egocentric/perceptual
 - A. horizon
 - B. vista
 - C. center
- IX. [partitions of the world]
 - A. body of water
 - 1. river
 - 2. lagoon
 - B. sky
 - C. land
- X. geometric features

- A. geometric features of land
 - 1. slope
 - 2. cliff
 - 3. flat, plateau, plain
 - B. geometric features of other entities
- XI. geographic features
 - A. positive features
 - 1. shadow
 - 2. forest
 - 3. meadow
 - 4. marsh
 - B. negative features
 - 1. chasm
 - 2. crater
 - 3. gap?
 - a) fissure
- XII. properties of geographical features
 - A. metric properties
 - 1. absolute
 - a) width
 - b) breadth
 - c) distance
 - 2. relative
 - a) nearness
 - B. non metric properties
 - 1. density
 - 2. color
 - 3.....
- XIII. location
 - A. relative
 - 1. here
 - B. absolute
 - 1. place
 - a) home
 - 2. region
- XIV. [spatial relations]
 - A. containment (in)
 - B. coincidence (at)
 - C. contact
 - 1. on, support
 - D. direction
 - 1. n,s,e,w
 - 2. towards <landmark>

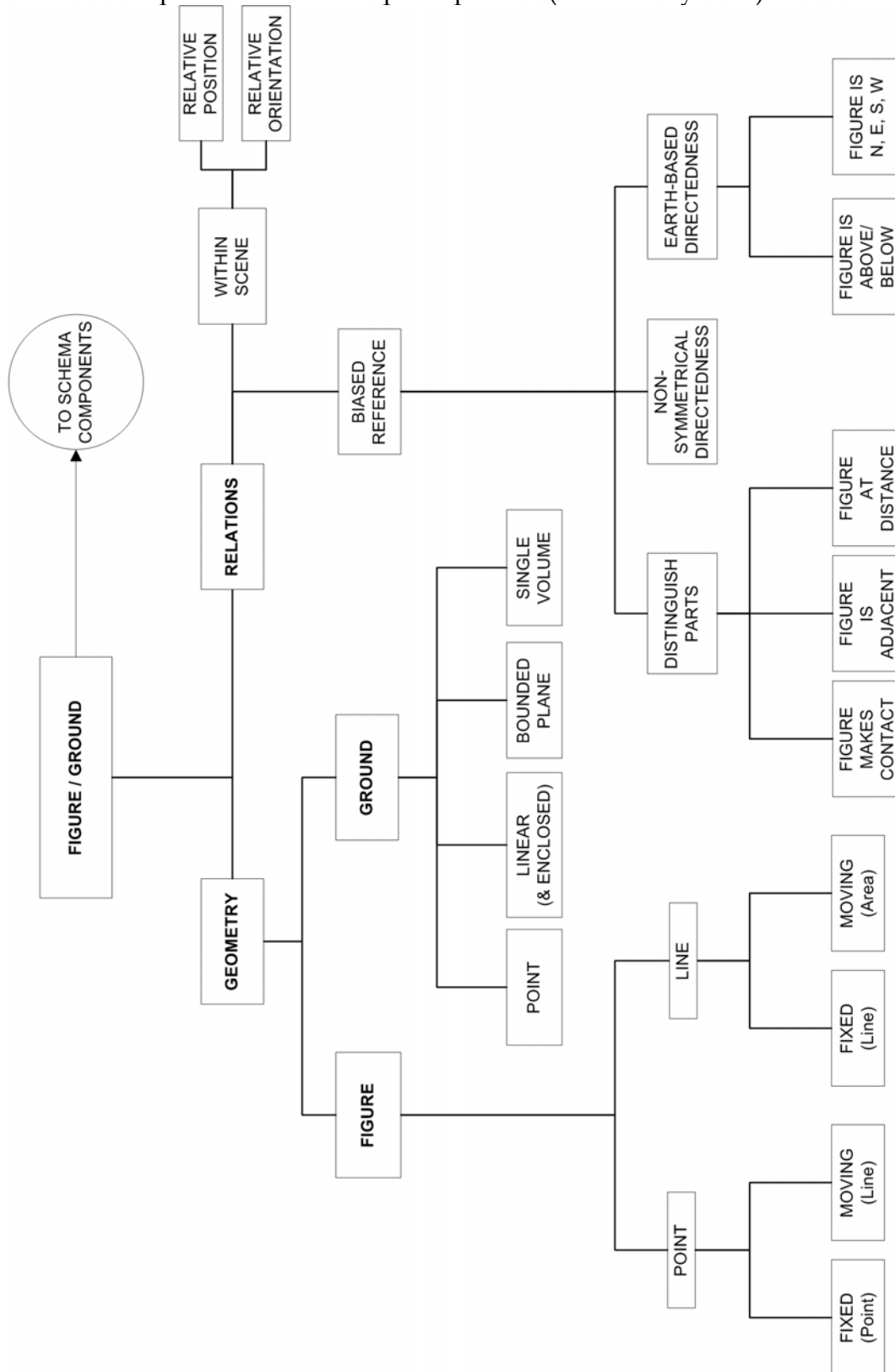
- 3. top/bottom
 - E. between
 - F. center/periphery
 - G. along
 - H. nearness
 - I. egocentric
 - 1. left, right
- XV. shape
 - A. straight
 - B. curved
 - C. corner
 - D. bent
 - E. nearly closed
 - 1. lagoon shaped
- XVI. [meteorological etc]
 - A. fires
 - B. wind
 - C. temperature
 - D. precipitation
 - 1. snow
 - 2. rain
 - 3. hail
- XVII. [institutions (constructed human world)]
 - A. ownership
 - 1. of property
 - 2. of rights
 - B. tribe, group
 - C. jurisdiction
 - D. freedom of movement
 - E. buildings
 - F. [roads]
 - 1. roads
 - 2. road networks
 - 3. road signs

G. bridge

APPENDIX 1.3: Spatial Scene Concept Map 1 of 2 (after Talmy 1983)



Spatial Scene Concept Map 2 of 2 (after Talmy 1983)



APPENDIX 2: Geoparser Output File Specifications

Introduction

The following document is prepared for guidance in system design and specifies the format, structural characteristics, and content of a geospatial data file. The file is intended for use with a proposed geocoding application that calculates latitude and longitude values for specific descriptive types of location. File contents are primarily the output of a natural language parsing process but also include ancillary spatial and non-spatial attributes. The parsing process uses as input, descriptions of herpetology specimen collection locations from the same museum database table named "HerpColl". Pertinent spatial elements of each collection locality description are interpreted and extracted from the text by the parser. These are combined with ancillary data associated with each collection and entered into the file. The file is in extensible markup language (XML) format and includes an embedded document type definition (DTD). The file structure with the DTD schema is illustrated on the following page. No element content is included with the illustration.

```

<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE Location_Descriptions [
  <ELEMENT Location_Descriptions (Location_Description)>
  <ELEMENT Location_Description (Record_Number,Locality_Text,County_Name,
    Elevation,Points_Of_Beginning,Segments)>

  <ELEMENT Record_Number (#PCDATA)>
  <ELEMENT Locality_Text (#PCDATA)>
  <ELEMENT County_Name (#PCDATA)>
  <ELEMENT Elevation (Elevation_Value,Elevation_Unit)>
  <ELEMENT Elevation_Value (#PCDATA)>
  <ELEMENT Elevation_Unit (#PCDATA)>
  <ELEMENT Points_Of_Beginning (Place,Place_and_Route,Route_and_Route)>
  <ELEMENT Place (PlaceName)>
  <ELEMENT PlaceName (#PCDATA)>
  <ELEMENT Place_and_Route (PlaceName,TravelRoute)>
  <ELEMENT TravelRoute (#PCDATA)>
  <ELEMENT Route_and_Route (TravelRoute,CrossRoute)>
  <ELEMENT CrossRoute (#PCDATA)>
  <ELEMENT Segments (Segment)>
  <ELEMENT Segment (Segment_Type,Direction,Distance,Remarks)>
  <ELEMENT Segment_Type (#PCDATA)>
  <ELEMENT Direction (#PCDATA)>
  <ELEMENT Distance (Distance_Value,Distance_Unit)>
  <ELEMENT Distance_Value (#PCDATA)>
  <ELEMENT Distance_Unit (#PCDATA)>
  <ELEMENT Remarks (#PCDATA)>
]>
<Location_Descriptions>
  <Location_Description>
    <Record_Number></Record_Number>
    <Locality_Text></Locality_Text>
    <County_Name></County_Name>
    <Elevation>
      <Elevation_Value></Elevation_Value>
      <Elevation_Unit></Elevation_Unit>
    </Elevation>
    <Points_Of_Beginning>
      <Place>
        <PlaceName></PlaceName>
      </Place>
      <Place_and_Route>
        <PlaceName></PlaceName>
        <TravelRoute></TravelRoute>
      </Place_and_Route>
      <Route_and_Route>
        <TravelRoute></TravelRoute>
        <CrossRoute></CrossRoute>
      </Route_and_Route>
    </Points_Of_Beginning>
    <Segments>
      <Segment>
        <Segment_Type></Segment_Type>
        <Direction></Direction>
        <Distance>
          <Distance_Value></Distance_Value>
          <Distance_Unit></Distance_Unit>
        </Distance>
        <Remarks></Remarks>
      </Segment>
    </Segments>
  </Location_Description>
</Location_Descriptions>

```

File Structure:

The file format is XML and comprises 23 elements in six levels. These are listed here by level and element:

1. "Location_Descriptions"
2. "Location_Description"
3. "Record_Number", "Locality_Text", "County_Name", "Elevation", "Points_Of_Beginning", and "Segments"
4. "Elevation_Value", "Elevation_Unit", "Place", "Place_and_Route", "Route_and_Route", and "Segment"
5. "PlaceName", "TravelRoute", "CrossRoute", "Segment_Type", "Direction", "Distance" and "Remarks"
6. "Distance_Value" and "Distance_Unit"

Elements and their content type are presented below by level.

Level 1: The root element is "Location_Descriptions".

"Location_Descriptions" contains no parsed content data (PCDATA hereafter) at Level 1 and has only the child element: "Location_Description"

Level 2: "Location_Description" is the only element at this level.

It contains no PCDATA at Level 2 but has six child elements:

"Record_Number", "Locality_Text", "County_Name", "Elevation", "Points_Of_Beginning", and "Segments". All geospatial elements parsed from "HerpColl" are stored as PCDATA in Level 3 and below.

Level 3: There are six elements at this level.

The elements "Record_Number", "Locality_Text", and "County_Name" contain PCDATA at this level and no children. The elements "Elevation", "Points_Of_Beginning", and "Segments" contain child elements but no PCDATA at Level 3. PCDATA at this level is a direct copy of three fields of the database source table "HerpColl". Element content with spatial meaning at this level is limited to the county name where the collection was taken.

Level 4: There are six elements at this level.

The elements "Elevation_Value" and "Elevation_Unit" have PCDATA at this level but no children. The elements "Place", "Place_and_Route", "Route_and_Route", and "Segment" have child elements but no PCDATA at this level. Element content with spatial meaning at this level is limited to the elevation (quantity and unit of measure) of the collection.

Level 5: There are seven elements at this level.

The elements “PlaceName”, “TravelRoute”, “CrossRoute”, “Segment_Type”, “Direction”, and “Remarks” include PCDATA at this level but no children. The element “Distance” has child elements but no PCDATA at this level. Element content with spatial meaning at this level includes named geography (places and travel routes) and one direction.

Level 6: There are two elements at this level.

The elements “Distance_Value” and “Distance_Unit” both contain PCDATA at this level and have no children. Element content with spatial meaning at this level is limited to one distance quantity and its unit of measure.

Error Checking and Relational Database Content:

Ancillary spatial and database information is associated with each collection record. This information includes a locality description, one vertical geographic reference, one horizontal geographic reference, and the primary key of the collection table. Sources for this information are as follows. A copy of several database fields’ content is stored as element content in the element tags: “Record_Number”, “Locality_Text”, and “County_Name”. Database elevation information is distributed as content between Elevation_Value and Elevation_Unit. The record number is available from the HerpColl table as a unique value and stored in the “IDKey” column. “Locality_Text” is a verbatim copy of the locality description in the “Locality” field of the HerpColl table. Likewise, “County_Name” is a duplicate of the table’s “County” field. Elevation and county name data provide reference for error checking of calculated coordinates following geoprocessing.

Spatial Content for Geoprocessing:

Each instance of “Location_Description” contains geometric features and relations in the spatial scene portrayed by a described location. Element content for each collection record must include at a minimum, valid entries for:

- PCDATA for Record_Number,
- PCDATA for Locality_Text,
- PCDATA for County_Name, and
- PCDATA for only one of the three child elements (cases) of Points_Of_Beginning.

Note that as suggested by the above, all elements listed below Location_Description cannot be used in every instance of Location_Description. PCDATA for child elements of Elevation and Segment are optional because these data are simply not always entered for every collection record. A source of elevation data may not have been available during field work. Some collections

are referenced only to a named geographic feature or entity and no direction or distance (segment content) is provided.

Most spatial content is stored below “Points_Of_Beginning” (POBs) and “Segments”. POBs include all possible starting points for parsed locations described in the traverse method. It is possible that several possibilities for a POB will be encountered by the geoparser. This occurs with collection record instances where the collection locality description includes:

- a place or route name that occurs more than once for the extent of a queried geography, or
- named routes that intersect more than once within the extent of queried geography.

Each POB in a traverse description can include content for only one of three types of location as a point:

- a place (or geographic) name,
- the intersection of a route and a place name, or
- the intersection of two routes.

“Segments” has content for all legs of a traverse. “Segment” provides content for individual legs of the traverse including:

- the method of calculating the endpoint of each leg,
- the length of each leg, and
- the direction of each leg.

When more than one “Segment” is listed under “Segments”, each “Segment” is listed in a sequence starting with the POB and ending with the “Segment” whose endpoint is at the collection locality.

Element Details (tree order):

“Location_Descriptions”

Root Element

Element Content: 1 child element – Location_Description

Purpose: Container for geoparser processing event’s output data.

“Location_Description”

Parent Element: Location_Descriptions

Element Content: 6 child elements – “Record_Number”, “Locality_Text”, “County_Name”, “Elevation”, “Points_Of_Beginning”, and “Segments”

Purpose: Container for all geoparsed collection records data.

“Record_Number”

Parent Element: Location_Description
Element Content: PCDATA – numeric, primary key for collection record.
Integer value is taken directly from HerpColl Table’s IDKey field.
Purpose: Provides foreign key relation of geoprocessing results to museum collection database.

“Locality_Text”

Parent Element: Location_Description
Element Content: PCDATA – text, location description from HerpColl table’s Locality field.
Purpose: Provides copy of locality description prior to parsing.
Minimum requirement for this field is an identifiable POB.

“County_Name”

Parent Element: Location_Description
Element Content: PCDATA – text, name of county of collection locality from HerpColl table’s County field.
Purpose: Name of county can serve as reference to polygon geometry that serves as topology reference in verification that geoprocessing result is within horizontal constraint.

“Elevation”

Parent Element: Location_Description
Element Content: 2 child elements – Elevation_Value and Elevation_Unit
Purpose: Container for recorded collection locality elevation data.
Elevation is not recorded for every collection but when present, provides a reference for comparison against elevation at calculated location.

“Points_Of_Beginning”

Parent Element: Location_Description
Element Content: 3 child elements – “Place”, “Place_and_Route”, and “Route_and_Route”
Purpose: Container for point of beginning information parsed from location description. Only one of three possible child elements will be present after parsing.

“Segments”

Parent Element: Location_Description
Element Content: 1 child element – “Segment”
Purpose: Container for all traverse type, distance, and direction data geoparsed from collection locality description. This element may be absent if location description ties collection to a geographic place. When present, all descendant child elements with PCDATA must have valid content.

“Elevation_Value”

Parent Element: Elevation
Element Content: PCDATA – numeric, elevation measure quantity. Quantity geoparsed from “Elevation” field is stored as numeric value.
Purpose: Provides the numeric portion of an elevation recorded for a collection locality (e.g. 890 from 890 feet).

“Elevation_Unit”

Parent Element: Elevation
Element Content: PCDATA – text, linear unit of measure used for elevation geoparsed from elevation field of HerpColl table.
Purpose: Identifies unit of measure used for recording elevation of collection locality (e.g. feet from 890 feet).

“Place”

Parent Element: Points_Of_Beginning
Element Content: 1 child element – PlaceName
Purpose: Container for geographic (place) name data parsed from collection locality description.

“Place_and_Route”

Parent Element: Points_Of_Beginning
Element Content: 2 child elements – PlaceName and TravelRoute
Purpose: Container for geographic (place) name data and linear travel way feature parsed from collection locality description. This element is used only when the POB is a route intersecting a named place.

“Route_and_Route”

Parent Element: Points_Of_Beginning
Element Content: 2 child elements – TravelRoute and CrossRoute
Purpose: Container for two linear travel way features parsed from collection locality description. This element is used only when the POB is the intersection of two travel routes.

“Segment”

Parent Element: Segments
Element Content: 4 child elements – Segment_Type, Direction, Distance, and Remarks
Purpose: Container for non-POB geoprocessing data. One or more instances of this element must be present whenever the element “Segments” is present below “Location_Description”

“PlaceName”

Parent Element: Place **OR** Place_and_Route
Element Content: PCDATA – text, named geographic place geoparsed from locality description.
Purpose: Provides name of place with known coordinates for use as or in calculation of POB. Note that only one of the two possible parent elements can appear under any instance of Points_Of_Beginning.

“TravelRoute”

Parent Element: Place_and_Route **OR** Route_and_Route
Element Content: PCDATA – text, named linear travel route geoparsed from locality description.
Purpose: Provides name of route being traveled away from POB. This is required when POB is identified as intersection of route with either another route or a place name. Note that only one of the two possible parent elements can appear under any instance of Points_Of_Beginning.

“CrossRoute”

Parent Element: Route_and_Route
Element Content: PCDATA – text, named linear travel route geoparsed from locality description.
Purpose: Provides name of route intersecting the route being traveled. The intersection forms the POB. This is required

when POB is identified as intersection of route with another route.

“Segment_Type”

Parent Element: Segment

Element Content: PCDATA – text, method of traverse from a segment’s start point (POB or end point of previous segment) to collection locality. Content can be considered a coded domain limited to “Euclidean” or “Route”.

Purpose: Identifies method of calculating a traverse for a given segment: either by Euclidean geometry or linear referencing.

“Direction”

Parent Element: Segment

Element Content: PCDATA – text, cardinal direction geoparsed from description of collection locality.

Purpose: Provides direction component of a traverse segment.

“Remarks”

Parent Element: Segment

Element Content: PCDATA – text, segment processing summary.

Purpose: Provides summary of any data processing related errors, events, or other errata related to geoparsing of segment data (e.g. distance given but no direction given from POB)

“Distance”

Parent Element: Segment

Element Content: 2 child elements – Distance_Value and Distance_Unit

Purpose: Container of elements that comprise distance component of a segment in a traverse.

“Distance_Value”

Parent Element: Distance

Element Content: PCDATA – numeric, quantity geoparsed from locality description, is stored as a double value.

Purpose: Provides the numeric portion of linear distance/length of a segment.

“Distance_Unit”

Parent Element: Distance
Element Content: PCDATA – text, linear unit of measure used for length of a segment and geoparsed from locality description.
Purpose: Identifies unit of measure used for recording linear component of a traverse segment.

Element Details (alphabetical order):

“County_Name”

Parent Element: Location_Description
Element Content: PCDATA – text, name of county of collection locality from HerpColl table’s County field.
Purpose: Name of county can serve as reference to polygon geometry that serves as topology reference in verification that geoprocessing result is within horizontal constraint.

“CrossRoute”

Parent Element: Route_and_Route
Element Content: PCDATA – text, named linear travel route geoparsed from locality description.
Purpose: Provides name of route intersecting the route being traveled. The intersection forms the POB. This is required when POB is identified as intersection of route with another route.

“Direction”

Parent Element: Segment
Element Content: PCDATA – text, cardinal direction geoparsed from description of collection locality.
Purpose: Provides direction component of a traverse segment.

“Distance”

Parent Element: Segment
Element Content: 2 child elements – Distance_Value and Distance_Unit
Purpose: Container of elements that comprise distance component of a segment in a traverse.

“Distance_Unit”

Parent Element: Distance
Element Content: PCDATA – text, linear unit of measure used for length of a segment and geoparsed from locality description.
Purpose: Identifies unit of measure used for recording linear component of a traverse segment.

“Distance_Value”

Parent Element: Distance
Element Content: PCDATA – numeric, quantity geoparsed from locality description, is stored as a double value.
Purpose: Provides the numeric portion of linear distance/length of a segment.

“Elevation”

Parent Element: Location_Description
Element Content: 2 child elements – Elevation_Value and Elevation_Unit
Purpose: Container for recorded collection locality elevation data. Elevation is not recorded for every collection but when present, provides a reference for comparison against elevation at calculated location.

“Elevation_Unit”

Parent Element: Elevation
Element Content: PCDATA – text, linear unit of measure used for elevation geoparsed from elevation field of HerpColl table.
Purpose: Identifies unit of measure used for recording elevation of collection locality (e.g. feet from 890 feet).

“Elevation_Value”

Parent Element: Elevation
Element Content: PCDATA – numeric, elevation measure quantity. Quantity geoparsed from “Elevation” field is stored as numeric value.
Purpose: Provides the numeric portion of an elevation recorded for a collection locality (e.g. 890 from 890 feet).

“Locality_Text”

Parent Element: Location_Description
Element Content: PCDATA – text, location description from HerpColl table’s Locality field.
Purpose: Provides copy of locality description prior to parsing. Minimum requirement for this field is an identifiable POB.

“Location_Description”

Parent Element: Location_Descriptions
Element Content: 6 child elements – “Record_Number”, “Locality_Text”, “County_Name”, “Elevation”, “Points_Of_Beginning”, and “Segments”
Purpose: Container for all geoparsed collection records data.

“Location_Descriptions”

Root Element
Element Content: 1 child element – Location_Description
Purpose: Container for geoparser processing event’s output data.

“Place”

Parent Element: Points_Of_Beginning
Element Content: 1 child element – PlaceName
Purpose: Container for geographic (place) name data parsed from collection locality description.

“PlaceName”

Parent Element: Place **OR** Place_and_Route
Element Content: PCDATA – text, named geographic place geoparsed from locality description.
Purpose: Provides name of place with known coordinates for use as or in calculation of POB. Note that only one of the two possible parent elements can appear under any instance of Points_Of_Beginning.

“Place_and_Route”

Parent Element: Points_Of_Beginning
Element Content: 2 child elements – PlaceName and TravelRoute
Purpose: Container for geographic (place) name data and linear travel way feature parsed from collection locality description. This element is used only when the POB is a route intersecting a named place.

“Record_Number”

Parent Element: Location_Description
Element Content: PCDATA – numeric, primary key for collection record. Integer value is taken directly from HerpColl Table’s IDKey field.
Purpose: Provides foreign key relation of geoprocessing results to museum collection database.

“Remarks”

Parent Element: Segment
Element Content: PCDATA – text, segment processing summary.
Purpose: Provides summary of any data processing related errors, events, or other errata related to geoparsing of segment data (e.g. distance given but no direction given from POB)

“Route_and_Route”

Parent Element: Points_Of_Beginning
Element Content: 2 child elements – TravelRoute and CrossRoute
Purpose: Container for two linear travel way features parsed from collection locality description. This element is used only when the POB is the intersection of two travel routes.

“Segment”

Parent Element: Segments
Element Content: 4 child elements – Segment_Type, Direction, Distance, and Remarks
Purpose: Container for non-POB geoprocessing data. One or more instances of this element must be present whenever the element “Segments” is present below “Location_Description”

“Segments”

Parent Element: Location_Description
Element Content: 1 child element – “Segment”
Purpose: Container for all traverse type, distance, and direction data geoparsed from collection locality description. This element may be absent if location description ties collection to a geographic place. When present, all descendant child elements with PCDATA must have valid content.

“Segment_Type”

Parent Element: Segment
Element Content: PCDATA – text, method of traverse from a segment’s start point (POB or end point of previous segment) to collection locality. Content can be considered a coded domain limited to “Euclidean” or “Route”.
Purpose: Identifies method of calculating a traverse for a given segment: either by Euclidean geometry or linear referencing.

“TravelRoute”

Parent Element: Place_and_Route **OR** Route_and_Route

Element Content: PCDATA – text, named linear travel route geoparsed from locality description.

Purpose: Provides name of route being traveled away from POB. This is required when POB is identified as intersection of route with either another route or a place name. Note that only one of the two possible parent elements can appear under any instance of Points_Of_Beginning.

APPENDIX 3: GNIS Feature Types

| | Feature | Conceptual Geometry at Medium Scale |
|----|----------|-------------------------------------|
| 1 | Airport | Area |
| 2 | Arch | Line |
| 3 | Area | Area |
| 4 | Arroyo | Line |
| 5 | Bar | Line |
| 6 | Basin | Area |
| 7 | Bay | Area |
| 8 | Beach | Area |
| 9 | Bench | Line |
| 10 | Bend | Line |
| 11 | Bridge | Line |
| 12 | Building | Point |
| 13 | Canal | Line |
| 14 | Cape | Area |
| 15 | Cemetery | Area |
| 16 | Channel | Line |
| 17 | Church | Point |
| 18 | Civil | Area |
| 19 | Cliff | Line |
| 20 | Crater | Area |
| 21 | Crossing | Line |
| 22 | Dam | Line |
| 23 | Falls | Line |
| 24 | Flat | Area |
| 25 | Forest | Area |
| 26 | Gap | Point |
| 27 | Geyser | Point |
| 28 | Glacier | Area |
| 29 | Gut | Line |
| 30 | Harbor | Area |
| 31 | Hospital | Point |
| 32 | Island | Area |

| | Feature | Conceptual Geometry at Medium Scale |
|----|-----------------|-------------------------------------|
| 33 | Isthmus | Area |
| 34 | Lake | Area |
| 35 | Lava | Area |
| 36 | Levee | Line |
| 37 | Locale | Area |
| 38 | Military | Area |
| 39 | Mine | Area |
| 40 | Oilfield | Area |
| 41 | Other | <Varies> |
| 42 | Park | Area |
| 43 | Pillar | Point |
| 44 | Plain | Area |
| 45 | Populated Place | Area |
| 46 | Post Office | Point |
| 47 | Range | Area |
| 48 | Rapids | Area |
| 49 | Reserve | Area |
| 50 | Reservoir | Area |
| 51 | Ridge | Area |
| 52 | School | Point |
| 53 | Sea | Area |
| 54 | Slope | Area |
| 55 | Spring | Point |
| 56 | Stream | Line |
| 57 | Summit | Point |
| 58 | Swamp | Area |
| 59 | Tower | Point |
| 60 | Trail | Line |
| 61 | Tunnel | Line |
| 62 | Valley | Area |
| 63 | Well | Point |
| 64 | Woods | Area |

APPENDIX 4: ArcMap VBA Code

```
'Robert Johnson
'University of Redlands
'Prepared for GIS-695B, Major Individual Project
'19 August 2004

'frmEuclideanMove

Option Explicit

'Geoprocessing variables
Private m_pMxDoc As IMxDocument
Private m_pFLayer As IFeatureLayer
Private m_pFClass As IFeatureClass
Private m_pFeature As IFeature
Private m_pInPoint As IPoint
Private m_pToPoint As IPoint
Private m_pConstPt As IConstructPoint
Private m_DDAngularDistance As Double
Private m_LatitudeIn As Double
Private m_LongitudeIn As Double
Private varEucMove As Variant

'Project Coordinate System Inverse Function variables
Private m_pSpatialReference As ISpatialReference
Private m_pProjCoorSys As IProjectedCoordinateSystem
Private m_pPcsInPt As WKSPoint
Private IpcsPtCt As Long

'Distance conversion & normalization variables
Private str_DistanceUnit As String
Private sgl_DistanceQuantityIn As Single
Private sgl_DistanceQuantityOut As Single

'Direction Conversion Module Level variables
Private sgl_RadianValue As Single
Private str_CardinalDirection As String

Private Sub cmdCalculateEndPoint_Click()

'The following section produces a Point Of Beginning for the
Euclidean Location Calculator functionality. Determination of POB shall
be a separate function in future development.

Set m_pMxDoc = ThisDocument
If Not TypeOf m_pMxDoc.ActiveView Is IMap Then
    MsgBox "A Data Frame Must Be Active!", , "Euclidean Calculator"
    Exit Sub
End If

'Find feature layer containing place names, assign to variable
Dim iLoop As Integer 'keep track of which layer you are on
Dim pCheckforLayer As ILayer 'a temporary variable used to check
all layers
For iLoop = 0 To m_pMxDoc.FocusMap.LayerCount - 1
```

```

        If TypeOf m_pMxDoc.FocusMap.Layer(iLoop) Is IFeatureLayer Then
        Set pCheckforLayer = m_pMxDoc.FocusMap.Layer(iLoop)
        If pCheckforLayer.Name = "Place Names" Then
            Set m_pFLayer = pCheckforLayer
        End If
    End If
Next iLoop

'error trap for invalid layer name
If m_pFLayer Is Nothing Then
    MsgBox "Invalid Layer Name in code"
End If

Set m_pFClass = m_pFLayer.FeatureClass

'DETERMINE POINT OF BEGINNING .. FROM A Place Name ENTERED ON INPUT
FORM
'set up query filter for ESRI [point]shape file of place names in
area of interest.
Dim pQFilter As IQueryFilter
Set pQFilter = New QueryFilter
pQFilter.WhereClause = "featname = '" & txtPlaceName.Text & "'"

'query Place Names feature class and return search cursor
Dim pFCursor As IFeatureCursor
Set pFCursor = m_pFClass.Search(pQFilter, True)

'access record
Set m_pFeature = pFCursor.NextFeature

'error trap for feature name not included in layer "Place Names"
If m_pFeature Is Nothing Then
    MsgBox "Invalid Place Name", , "Process Terminated"
End If
End If

'POB set equal to named place geometry
Set m_pInPoint = m_pFeature.Shape

'START INVERSE PROJECTION
'set WKS (x,y)values equal to POB (x,y)values
m_pPcsInPt.X = m_pInPoint.X
m_pPcsInPt.Y = m_pInPoint.Y

'get spatial reference from POB
Set m_pSpatialReference = m_pInPoint.SpatialReference

Set m_pProjCoorSys = m_pSpatialReference 'QI

'Inverse Projected Coordinate to Geographic Coordinate
IpcsPtCt = 1 'Point count of one
m_pProjCoorSys.Inverse IpcsPtCt, m_pPcsInPt
'END INVERSE PROJECTION

'assign POB coordinate to variables
m_LongitudeIn = m_pPcsInPt.X
m_LatitudeIn = m_pPcsInPt.Y

```

```

'Assign lat/lon values to InPoint geometry
m_pInPoint.X = m_pPcsInPt.X
m_pInPoint.Y = m_pPcsInPt.Y
'END OF POB SECTION

'assign distance and direction variables from form input:
'error trap for no data entered
    Select Case txtDistanceQuantityIn.Text
        Case "", "<Quantity>"
            MsgBox "Enter A Distance Quantity", , "INPUT ERROR"
            Exit Sub
    End Select
sgl_DistanceQuantityIn = txtDistanceQuantityIn.Text
str_DistanceUnit = cboDistanceUnit.Text
str_CardinalDirection = cboCardinalDirection.Text

'error trap for no direction selection
If str_CardinalDirection = "    Select !" Then
    MsgBox "Select A Direction", , "INPUT ERROR"
End If

'Call Euclidean Calculator function
varEucMove = EuclideanMove(m_LongitudeIn, m_LatitudeIn,
sgl_DistanceQuantityIn, str_DistanceUnit, str_CardinalDirection)

'No Processing Errors/Normal Termination
If varEucMove = 0 Then
    'Round POB value to 3 places right of decimal (~111m)
        m_pInPoint.X = Round(m_pInPoint.X, 3)
        m_pInPoint.Y = Round(m_pInPoint.Y, 3)

    'Round calculated value to 3 places right of decimal (~111m)
        m_pToPoint.X = Round(m_pToPoint.X, 3)
        m_pToPoint.Y = Round(m_pToPoint.Y, 3)

    'display calculated lat/lon values of segment endpoint on form
        txtXvalueout.Text = m_pToPoint.X
        txtYvalueout.Text = m_pToPoint.Y

    'Failure in cardinal to radian direction conversion procedure
    ElseIf varEucMove = 1 Then MsgBox "Direction Conversion Error", ,
"Process Failure" 'Display error message

    'Failure in input distance to normalized mile conversion
procedure
    ElseIf varEucMove = 2 Then MsgBox "Distance Normalization Error", ,
"Process Failure" 'Display error message

End If
End Sub
Private Function EuclideanMove(m_LongitudeIn, m_LatitudeIn,
sgl_DistanceQuantityIn, str_DistanceUnit, str_CardinalDirection) As
Integer

    'CONVERT CARDINAL DIRECTION FROM POINT OF BEGINNING TO DESTINATION
TO RADIAN MEASURE

```

```

Select Case str_CardinalDirection
    'East
    Case "E", "e"
        sgl_RadianValue = 0
    'East Northeast
    Case "ENE", "eNE", "EnE", "enE", "ENe", "eNe", "Ene", "ene"
        sgl_RadianValue = 0.39
    'Northeast
    Case "NE", "Ne", "nE", "ne"
        sgl_RadianValue = 0.78
    'Nor Northeast
    Case "NNE", "nNE", "NnE", "nnE", "NNe", "nNe", "Nne", "nne"
        sgl_RadianValue = 1.18
    'North
    Case "N", "n"
        sgl_RadianValue = 1.57
    'Nor Northwest
    Case "NNW", "nNW", "NnW", "nnW", "NNw", "nNw", "Nnw", "nnw"
        sgl_RadianValue = 1.96
    'Northwest
    Case "NW", "nW", "Nw", "nw"
        sgl_RadianValue = 2.36
    'West Northwest
    Case "WNW", "wNW", "WnW", "wnW", "WNw", "wNw", "Wnw", "wnw"
        sgl_RadianValue = 2.75
    'West
    Case "W", "w"
        sgl_RadianValue = 3.14
    'West Southwest
    Case "WSW", "wSW", "WsW", "wsW", "WSw", "wSw", "Wsw", "wsw"
        sgl_RadianValue = 3.53
    'Southwest
    Case "SW", "sW", "Sw", "sw"
        sgl_RadianValue = 3.93
    'Sou Southwest
    Case "SSW", "sSW", "SsW", "ssW", "SSw", "sSw", "Ssw", "ssw"
        sgl_RadianValue = 4.32
    'South
    Case "S", "s"
        sgl_RadianValue = 4.71
    'Sou Southeast
    Case "SSE", "sSe", "SsE", "ssE", "SSe", "sSe", "Sse", "sse"
        sgl_RadianValue = 5.1
    'Southeast
    Case "SE", "sE", "Se", "se"
        sgl_RadianValue = 5.5
    'East Southeast
    Case "ESE", "eSE", "Ese", "ese", "ESE", "eSe", "Ese", "ese"
        sgl_RadianValue = 5.89
    'All other cases not valid cardinal direction
    Case Else
        EuclideanMove = 1
End Select

'CONVERT INPUT DISTANCE TO NORMALIZED REPRESENTATION IN MILES
str_DistanceUnit = cboDistanceUnit.Text 'get input distance
unit

```

```

        sgl_DistanceQuantityIn = txtDistanceQuantityIn.Text ' get input
distance quantity

        'acceptable input cases
        Select Case str_DistanceUnit
            Case "meters", "m"
                Select Case sgl_DistanceQuantityIn
                    Case Is < 100 'precision to 5 meters
                        sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn / 5 * 0.00311
                        sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 3)
                    Case Is >= 100 'precision to 100m level
                        sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn / 100 * 0.062
                        sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
                    Case Else
                        EuclideanMove = 2
                End Select
            Case "yards", "yds"
                Select Case sgl_DistanceQuantityIn
                    Case Is < 100 'precision to 5 yards
                        sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn / 5 * 0.00284
                        sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 3)
                    Case 100 To 1000 'precision to 50 yards
                        sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn / 50 * 0.0284
                        sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
                    Case Is > 1000 'precision to 100 yards
                        sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn / 100 * 0.057
                        sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
                    Case Else
                        EuclideanMove = 2
                End Select
            Case "ft", "feet"
                Select Case sgl_DistanceQuantityIn
                    Case Is < 5 'avoid zero values from rounding with
precision to 5 feet
                        sgl_DistanceQuantityIn = 5
                        sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn / 5 * 0.0009 'for 5 - 95 feet range
                        sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 3)
                    Case Is < 100 'precision to 5 feet
                        sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn / 5 * 0.0009 'for 5 - 95 feet range
                        sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 3)
                    Case Is >= 100 'precision to 100 feet
                        sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn / 100 * 0.019 'for 100 - 400 feet range

```

```

        sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
        Case Else
            EuclideanMove = 2
        End Select
    Case "km", "Km", "Kilometers"
        Select Case sgl_DistanceQuantityIn
            Case Is < 5 'precision to 100 meters (0.1 Km)
                sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn / 0.1 * 0.062 ' for range of 0.8 - 5 Km
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
            Case Is >= 5 'precision to .2 Kilometer
                sgl_DistanceQuantityOut =
sgl_DistanceQuantityIn * 0.621 ' for range 5 - 22 Km
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 1)
            Case Else
                EuclideanMove = 2
        End Select
    Case "mi", "miles", "mile"
        sgl_DistanceQuantityOut = sgl_DistanceQuantityIn
    Case "<Units>"
        MsgBox "Select Distance Units!", , "INPUT ERROR"
        EuclideanMove = 2
    Case Else
        EuclideanMove = 2
End Select

'equatorial circumference value used for miles per degree
"constant"
m_DDAngularDistance = sgl_DistanceQuantityOut / 69.172

Set m_pToPoint = New esriCore.Point 'set the new point
Set m_pConstPt = m_pToPoint 'geometry operator construct
point is used
m_pConstPt.ConstructAngleDistance m_pInPoint, sgl_RadianValue,
m_DDAngularDistance 'Calculate the ToPoint

End Function

Private Sub UserForm_Initialize()
    'load distance unit values into Travel Distance unit combo box
    cboDistanceUnit.AddItem ("<Units>")
    cboDistanceUnit.AddItem ("Kilometers")
    cboDistanceUnit.AddItem ("meters")
    cboDistanceUnit.AddItem ("feet")
    cboDistanceUnit.AddItem ("yards")
    cboDistanceUnit.AddItem ("miles")
    cboDistanceUnit.Text = "<Units>"

    'load direction values into Travel Direction combo box
    cboCardinalDirection.AddItem (" Select !")
    cboCardinalDirection.AddItem ("N")
    cboCardinalDirection.AddItem ("NNE")
    cboCardinalDirection.AddItem ("NE")
    cboCardinalDirection.AddItem ("ENE")

```

```
cboCardinalDirection.AddItem ("E")
cboCardinalDirection.AddItem ("ESE")
cboCardinalDirection.AddItem ("SE")
cboCardinalDirection.AddItem ("SSE")
cboCardinalDirection.AddItem ("S")
cboCardinalDirection.AddItem ("SSW")
cboCardinalDirection.AddItem ("SW")
cboCardinalDirection.AddItem ("WSW")
cboCardinalDirection.AddItem ("W")
cboCardinalDirection.AddItem ("WNW")
cboCardinalDirection.AddItem ("NW")
cboCardinalDirection.AddItem ("NNW")
cboCardinalDirection.Text = "    Select !"
```

```
End Sub
```



```
'Robert Johnson
'University of Redlands
'Prepared for GIS-695B, Major Individual Project
'19 August 2004
```

```
'frmRouteGeocoder
```

```
Option Explicit
```

```
'Geoprocessing module level variables
```

```
Private m_pMxDoc As IMxDocument
Private m_pFLayer As IFeatureLayer
Private m_pFLayer2 As IFeatureLayer
Private m_pFClass As IFeatureClass
Private m_pFClass2 As IFeatureClass
Private m_pFeature As IFeature
Private m_pFeature2 As IFeature
Private m_JctPoint As IPoint
Private pPob As IPoint
Private m_XValueIn As Double
Private m_YValueIn As Double
Private dblPobMValue As Double
Private sgl_DistanceQuantityIn As Single
Private str_DistanceUnit As String
Private str_CardinalDirection As String
Private sgl_RadianValue As Single
Private InputErrorCode As Integer
Private sgl_DistanceQuantityOut As Single
Private mHi As Double
Private mLo As Double
```

```
Private Sub cmdFindPointRouteLocation_Click()
```

```
'Confirm map document is active before proceeding.
```

```
Set m_pMxDoc = ThisDocument
If Not TypeOf m_pMxDoc.ActiveView Is IMap Then
MsgBox "There Is No Active Map!"
Exit Sub
End If
```

```
'FIND LAYER: POINTS REPRESENTING ALL ROUTE INTERSECTIONS IN AREA OF
INTEREST.
```

```
Dim iLoop As Integer 'keep track of which layer you are on
Dim pCheckforLayer As ILayer 'a temporary variable used to check
all layers
For iLoop = 0 To m_pMxDoc.FocusMap.LayerCount - 1
If TypeOf m_pMxDoc.FocusMap.Layer(iLoop) Is IFeatureLayer Then
Set pCheckforLayer = m_pMxDoc.FocusMap.Layer(iLoop)
If pCheckforLayer.Name = "AOI_Junctions" Then
Set m_pFLayer = pCheckforLayer
End If
End If
Next iLoop

Set m_pFClass = m_pFLayer.FeatureClass
Set pCheckforLayer = Nothing 'House Cleaning
```

```

'DETERMINE POINT OF BEGINNING .. per TWO ROUTES LISTED ON INPUT FORM
'set up query filter for [point] shape file of Route Junctions in
AOI_Junctions
Dim pQFilter As IQueryFilter
Set pQFilter = New QueryFilter
pQFilter.WhereClause = _
"STAN_ADDR1 = '" & cboTravelRoute.Text & "' AND STAN_ADDR2 = '" &
cboXsctRoute.Text & "' OR STAN_ADDR1 = '" & cboXsctRoute.Text & "' AND
STAN_ADDR2 = '" & cboTravelRoute.Text & "'"

'query AOI_Junctions feature class and return search cursor
Dim pFCursor As IFeatureCursor
Set pFCursor = m_pFClass.Search(pQFilter, True)

'ACCESS POB RECORD
Set m_pFeature = pFCursor.NextFeature

'error trap for invalid and/or undocumented route junctions
If m_pFeature Is Nothing Then
    MsgBox "Intersection Not Recognized for Selected Roads", ,
"Route Calculator Error"
End
End If

'Set point variable equal to junction of routes
Set m_JctPoint = m_pFeature.Shape

'assign route junction coordinate values to variables
m_XValueIn = m_JctPoint.X
m_YValueIn = m_JctPoint.Y
'END OF POB SECTION

'FIND ROUTE FEATURE LAYER ... NAMED "SBM_Routes"
Dim pCheckforLayer2 As ILayer
For iLoop = 0 To m_pMxDoc.FocusMap.LayerCount - 1
    If TypeOf m_pMxDoc.FocusMap.Layer(iLoop) Is IFeatureLayer Then
        Set pCheckforLayer2 = m_pMxDoc.FocusMap.Layer(iLoop)
        If pCheckforLayer2.Name = "SBM_Routes" Then
            Set m_pFLayer2 = pCheckforLayer2
        End If
    End If
Next iLoop

Set m_pFClass2 = m_pFLayer2.FeatureClass
Set pCheckforLayer2 = Nothing 'House Cleaning

'set up query filter for [polyline] Travelled Route in shape file
"SBM_Routes"
Dim pQFilter2 As IQueryFilter
Set pQFilter2 = New QueryFilter
pQFilter2.WhereClause = "STREETNAME = '" & cboTravelRoute.Text &
""

'query "SBM_Routes" feature class and return search cursor
Dim pFCursor2 As IFeatureCursor
Set pFCursor2 = m_pFClass2.Search(pQFilter2, False)

```

```

'access record
Set m_pFeature2 = pFCursor2.NextFeature

'ERROR TRAP FOR INVALID ROUTE NAME
If m_pFeature2 Is Nothing Then
    MsgBox "Route Not Recognized!"
Else
    Dim Ipo As IPximityOperator
    Set Ipo = m_pFeature2.Shape
    Set pPob = Ipo.ReturnNearestPoint(m_JctPoint, 0)
End If

'EXTRACT MEASURE VALUE OF ROUTE AT POB
'Following code from ArcObjects Component Help (Core):
GetMsAtDistance Example

Dim pMA As IMAware
Dim pCurve As ICurve
Dim pTmpPt As IPoint, dAlong As Double, dFrom As Double, bRight As
Boolean
Dim pMSeg As IMSegmentation
Dim Ms 'variant to hold M value(s)
Dim i As Long 'Hold count of M values
Set pMA = m_pFeature2.Shape 'travel route shape file

If pMA Is Nothing Then
    MsgBox "Travel Route Not Recognized!"
Else
    If pMA.MAware Then 'Find distance from start point of polyline
to point at/by Jct.
        Set pCurve = m_pFeature2.Shape
        pCurve.QueryPointAndDistance esriNoExtension, pPob, False,
pTmpPt, dAlong, dFrom, bRight
        Set pMSeg = pCurve
        Ms = pMSeg.GetMsAtDistance(dAlong, False) 'Assign value(s)
for M (at distance) to Ms (possible array)
    Else
        Err.Raise vbObjectError + 11282000, "GetMsAtPoint", "Not M
Aware"
    End If
End If

If UBound(Ms) > 0 Then
    MsgBox "Routes Intersect More Than Once!"
Else
    i = LBound(Ms)
    dblPobMValue = Ms(i)
End If

'ASSIGN DISTANCE AND DIRECTION VARIABLES FROM INPUT FORM:
sgl_DistanceQuantityIn = txtDistanceQuantityIn.Text
str_DistanceUnit = cboDistanceUnit.Text
str_CardinalDirection = cboCardinalDirection.Text

'CONVERT CARDINAL DIRECTION FROM POINT OF BEGINNING TO DESTINATION TO
RADIAN MEASURE
Select Case str_CardinalDirection

```

```

'East
Case "E", "e"
    sgl_RadianValue = 0
'East Northeast
Case "ENE", "eNE", "EnE", "enE", "ENe", "eNe", "Ene", "ene"
    sgl_RadianValue = 0.39
'Northeast
Case "NE", "Ne", "nE", "ne"
    sgl_RadianValue = 0.78
'Nor Northeast
Case "NNE", "nNE", "NnE", "nnE", "NNe", "nNe", "Nne", "nne"
    sgl_RadianValue = 1.18
'North
Case "N", "n"
    sgl_RadianValue = 1.57
'Nor Northwest
Case "NNW", "nNW", "NnW", "nnW", "NNw", "nNw", "Nnw", "nnw"
    sgl_RadianValue = 1.96
'Northwest
Case "NW", "nW", "Nw", "nw"
    sgl_RadianValue = 2.36
'West Northwest
Case "WNW", "wNW", "WnW", "wnW", "WNw", "wNw", "Wnw", "wnw"
    sgl_RadianValue = 2.75
'West
Case "W", "w"
    sgl_RadianValue = 3.14
'West Southwest
Case "WSW", "wSW", "Wsw", "wsW", "WSw", "wSw", "Wsw", "wsW"
    sgl_RadianValue = 3.53
'Southwest
Case "SW", "sW", "Sw", "sw"
    sgl_RadianValue = 3.93
'Sou Southwest
Case "SSW", "sSW", "SsW", "ssW", "SSw", "sSw", "Ssw", "ssw"
    sgl_RadianValue = 4.32
'South
Case "S", "s"
    sgl_RadianValue = 4.71
'Sou Southeast
Case "SSE", "sSe", "SsE", "ssE", "SSe", "sSe", "Sse", "sse"
    sgl_RadianValue = 5.1
'Southeast
Case "SE", "sE", "Se", "se"
    sgl_RadianValue = 5.5
'East Southeast
Case "ESE", "eSE", "Ese", "ese", "ESE", "eSe", "Ese", "ese"
    sgl_RadianValue = 5.89
'All other cases not valid cardinal direction
Case Else
    InputErrorCode = 1
End Select

```

```

'END ... CONVERT CARDINAL DIRECTION FROM POINT OF BEGINNING TO
DESTINATION TO RADIAN MEASURE

```

```

'START ... CONVERT INPUT DISTANCE TO NORMALIZED REPRESENTATION IN MILES

```

```

Select Case str_DistanceUnit
    Case "meters", "m"
        Select Case sgl_DistanceQuantityIn
            Case Is < 100 'precision to 5 meters
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn /
5 * 0.00311
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 3)
            Case Is >= 100 'precision to 100m level
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn /
100 * 0.062
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
            Case Else
                InputErrorCode = 2
        End Select
    Case "yards", "yds"
        Select Case sgl_DistanceQuantityIn
            Case Is < 100 'precision to 5 yards
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn /
5 * 0.00284
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 3)
            Case 100 To 1000 'precision to 50 yards
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn /
50 * 0.0284
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
            Case Is > 1000 'precision to 100 yards
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn /
100 * 0.057
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
            Case Else
                InputErrorCode = 2
        End Select
    Case "ft", "feet"
        Select Case sgl_DistanceQuantityIn
            Case Is < 5 'avoid zero values from rounding with
precision to 5 feet
                sgl_DistanceQuantityIn = 5
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn /
5 * 0.0009 'for 5 - 95 feet range
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 3)
            Case Is < 100 'precision to 5 feet
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn /
5 * 0.0009 'for 5 - 95 feet range
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 3)
            Case Is >= 100 'precision to 100 feet
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn /
100 * 0.019 'for 100 - 400 feet range
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
            Case Else

```

```

        InputErrorCode = 2
    End Select
    Case "km", "Km", "Kilometers"
        Select Case sgl_DistanceQuantityIn
            Case Is < 5 'precision to 100 meters (0.1 Km)
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn /
0.1 * 0.062 ' for range of 0.8 - 5 Km
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 2)
            Case Is >= 5 'precision to .2 Kilometer
                sgl_DistanceQuantityOut = sgl_DistanceQuantityIn *
0.621 ' for range 5 - 22 Km
                sgl_DistanceQuantityOut =
Round(sgl_DistanceQuantityOut, 1)
            Case Else
                InputErrorCode = 2
        End Select
    Case "mi", "miles", "mile"
        sgl_DistanceQuantityOut = sgl_DistanceQuantityIn
    Case Else
        MsgBox "Select Distance Units!"
        Exit Sub
    End Select
'END... CONVERT INPUT DISTANCE TO NORMALIZED REPRESENTATION IN MILES

'BEGIN... DETERMINE ACTUAL TRAVEL DIRECTION ON ROUTE
'1st find point coordinates for Euclidean move and both directions on
route

'Error handling for travel distance values that exceed modelled extent
of route
    'Check for high m end of route
    Dim RtExtentErr As Integer 'Holds Route Extent Error Type
    If Ms(i) + sgl_DistanceQuantityOut > pMSeg.MMax Then
        RtExtentErr = 1
    End If
    'Check for low m end of route
    If sgl_DistanceQuantityOut > Ms(i) Then
        RtExtentErr = 0
    End If
    'Check for both ends of route
    If sgl_DistanceQuantityOut > Ms(i) And Ms(i) +
sgl_DistanceQuantityOut > pMSeg.MMax Then
        RtExtentErr = 2
    End If

    Select Case RtExtentErr
        Case 0
            mLo = pMSeg.MMin 'set low m value to route's lowest m value
            mHi = Ms(i) + sgl_DistanceQuantityOut 'Calculate Route High
m value
            MsgBox "CAUTION Travel Distance Exceeds Low m-value End of
Modeled Route", , "Route Calculator"
        Case 1
            mLo = Ms(i) - sgl_DistanceQuantityOut 'Calculate Route Low
m values

```

```

        mHi = pMSeg.MMax 'set high m value to route's highest m
value
        MsgBox "CAUTION Travel Distance Exceeds High m-value End of
Modeled Route", , "Route Calculator"
        Case 2
            MsgBox "Travel Distance Exceeds Modeled Extent of Route on
Both Ends!", , "Route Calculator"
            MsgBox "Procedure Terminated"
            txt_X.Text = ""
            txt_Y.Text = ""
            Exit Sub
        End Select

Dim pHighPoint As IPoint
Dim pLowPoint As IPoint
Dim pEuclideanPoint As IPoint
Dim pGeometry As IGeometry
Dim pPointColl As IGeometryCollection
Dim Offset As Double
Dim PtCount As Long
Dim m_pConstPt As IConstructPoint
Dim DistLow As Double
Dim DistHigh As Double
Dim pPtOnRoute As IPoint
Dim pToPoint As WKSPoint
Dim m_pSpatialReference As ISpatialReference
Dim m_pProjCoorSys As IProjectedCoordinateSystem
Dim IpcsPtCt As Long

Offset = 0 'Set offset to zero.

'CALCULATE POINTS ON ROUTE

'Calculate Point at Route High m Value:
Set pHighPoint = New esriCore.Point
Set pHighPoint = pGeometry 'QI
Set pPointColl = New Multipoint
Set pPointColl = pMSeg.GetPointsAtM(mHi, Offset)

PtCount = pPointColl.GeometryCount

Set pHighPoint = pPointColl.Geometry(PtCount - 1)
Set pPointColl = Nothing 'Housekeeping

'Calculate Point at Route Low m Value:
Set pLowPoint = New esriCore.Point
Set pLowPoint = pGeometry 'QI
Set pPointColl = New Multipoint
Set pPointColl = pMSeg.GetPointsAtM(mLo, Offset)

PtCount = pPointColl.GeometryCount

Set pLowPoint = pPointColl.Geometry(PtCount - 1)
Set pPointColl = Nothing 'Housekeeping

'CALCULATE EUCLIDEAN REFERENCE POINT:

```

```

    Set pEuclideanPoint = New esriCore.Point 'Instantiate Euclidean
reference point
    Set m_pConstPt = pEuclideanPoint 'QI (geometry operator construct
point is used)
    sgl_DistanceQuantityOut = sgl_DistanceQuantityOut * 1609.344
'Convert miles to meters
    m_pConstPt.ConstructAngleDistance pPob, sgl_RadianValue,
sgl_DistanceQuantityOut 'Calculate the ToPoint

'CALCULATE DISTANCES
    DistLow = Sqr((pEuclideanPoint.X - pLowPoint.X) ^ 2 +
(pEuclideanPoint.Y - pLowPoint.Y))
    DistHigh = Sqr((pEuclideanPoint.X - pHighPoint.X) ^ 2 +
(pEuclideanPoint.Y - pHighPoint.Y))

    Set pPtOnRoute = New esriCore.Point

    'Select closest point on route to Euclidean Point as correct
direction of travel.
    If DistLow < DistHigh Then
        Set pPtOnRoute = pLowPoint
    Else
        Set pPtOnRoute = pHighPoint
    End If

'START INVERSE PROJECTION SECTION(prepare POB copy for Euclidean)
    'set Lat/Lon-value-holding WKSPoint's (x,y)values equal to Point on
Route (x,y)DD values
    pToPoint.X = pPtOnRoute.X
    pToPoint.Y = pPtOnRoute.Y

    'get spatial reference from POB
    Set m_pSpatialReference = pPtOnRoute.SpatialReference

    Set m_pProjCoorSys = m_pSpatialReference 'QI

    'Inverse Projected to Geographic Coordinates
    IpcsPtCt = 1 'Initialize Point count variable to value of one
    m_pProjCoorSys.Inverse IpcsPtCt, pToPoint
'END INVERSE PROJECTION SECTION

'DISPLAY RESULTS ON FORM
    'Round values to 3 places to right of decimal (~111m)
    pToPoint.X = Round(pToPoint.X, 3)
    pToPoint.Y = Round(pToPoint.Y, 3)
    'Assign to display variable
    txt_X = pToPoint.X
    txt_Y = pToPoint.Y

End Sub

Private Sub UserForm_Initialize()

    'load Intersecting Road Names in Travel Route Box
    cboTravelRoute.AddItem ("I- 10")

```



```

cboTravelRoute.AddItem ("I- 15")
cboTravelRoute.AddItem ("I- 210")
cboTravelRoute.AddItem ("I- 215")
cboTravelRoute.AddItem ("STATE HWY 18")
cboTravelRoute.AddItem ("STATE HWY 30")
cboTravelRoute.AddItem ("STATE HWY 38")
cboTravelRoute.AddItem ("STATE HWY 60")
cboTravelRoute.AddItem ("STATE HWY 62")
cboTravelRoute.AddItem ("STATE HWY 66")
cboTravelRoute.AddItem ("STATE HWY 79")
cboTravelRoute.AddItem ("STATE HWY 111")
cboTravelRoute.AddItem ("STATE HWY 138")
cboTravelRoute.AddItem ("STATE HWY 173")
cboTravelRoute.AddItem ("STATE HWY 189")
cboTravelRoute.AddItem ("STATE HWY 243")
cboTravelRoute.AddItem ("STATE HWY 247")
cboTravelRoute.AddItem ("STATE HWY 259")
cboTravelRoute.AddItem ("STATE HWY 330")
cboTravelRoute.AddItem ("US 395")
cboTravelRoute.Text = "I- 10"

```

'load Intersecting Road Names in Intersecting Route Box

```

cboXsctRoute.AddItem ("I- 10")
cboXsctRoute.AddItem ("I- 15")
cboXsctRoute.AddItem ("I- 210")
cboXsctRoute.AddItem ("I- 215")
cboXsctRoute.AddItem ("STATE HWY 18")
cboXsctRoute.AddItem ("STATE HWY 30")
cboXsctRoute.AddItem ("STATE HWY 38")
cboXsctRoute.AddItem ("STATE HWY 60")
cboXsctRoute.AddItem ("STATE HWY 62")
cboXsctRoute.AddItem ("STATE HWY 66")
cboXsctRoute.AddItem ("STATE HWY 79")
cboXsctRoute.AddItem ("STATE HWY 111")
cboXsctRoute.AddItem ("STATE HWY 138")
cboXsctRoute.AddItem ("STATE HWY 173")
cboXsctRoute.AddItem ("STATE HWY 189")
cboXsctRoute.AddItem ("STATE HWY 243")
cboXsctRoute.AddItem ("STATE HWY 247")
cboXsctRoute.AddItem ("STATE HWY 259")
cboXsctRoute.AddItem ("STATE HWY 330")
cboXsctRoute.AddItem ("US 395")
cboXsctRoute.Text = "I- 215"

```

'load distance unit values into Travel Distance unit combo box

```

cboDistanceUnit.AddItem ("<Units>")
cboDistanceUnit.AddItem ("Kilometers")
cboDistanceUnit.AddItem ("meters")
cboDistanceUnit.AddItem ("feet")
cboDistanceUnit.AddItem ("yards")
cboDistanceUnit.AddItem ("miles")
cboDistanceUnit.Text = "<Units>"

```

'load direction values into Travel Direction combo box

```

cboCardinalDirection.AddItem ("N")
cboCardinalDirection.AddItem ("NNE")
cboCardinalDirection.AddItem ("NE")

```

```
cboCardinalDirection.AddItem ("ENE")
cboCardinalDirection.AddItem ("E")
cboCardinalDirection.AddItem ("ESE")
cboCardinalDirection.AddItem ("SE")
cboCardinalDirection.AddItem ("SSE")
cboCardinalDirection.AddItem ("S")
cboCardinalDirection.AddItem ("SSW")
cboCardinalDirection.AddItem ("SW")
cboCardinalDirection.AddItem ("WSW")
cboCardinalDirection.AddItem ("W")
cboCardinalDirection.AddItem ("WNW")
cboCardinalDirection.AddItem ("NW")
cboCardinalDirection.AddItem ("NNW")
cboCardinalDirection.Text = "N"
```

End Sub

APPENDIX 5: Glossary of Terms

event

1. Geometry that is stored in tabular form. It can be stored as coordinate values. It can also be stored as measure value(s) and the specific geometry the measure value(s) is associated with.
2. Phenomenon associated with a defined point or period in time.

expert system

A highly domain specific program that uses available information, heuristics, and inference to query and/or extend a knowledge base.

geoparsing

A software process that identifies and records spatial entities and their relationships in natural language.

geoprocessing

Geoprocessing embodies GIS operations, which include data conversion, geographic feature overlays, topology processing, coverage selection and analysis.

grammar

Studies concerning the formation of basic linguistic units.

herpetofauna

Amphibians and Reptiles

homonym problem

The return of multiple unique locations when a gazetteer is queried for a single place name (i.e. several locals share the same name).

lexicon

A reference list of words with information about them, specifying how they may be used.

locality precision

The level of precision when recorded provenance is within 500-2000m of the true location of a biological specimen collection.

parsing

A syntactical analysis made by assigning a constituent structure to a sentence or phrase.

regional precision

A location that is within the extent of the same major geographic feature (valley, mountain range, desert) as the true location of a biological specimen collection

spatial scene

A portrayal of spatial relations achieved by presenting geographic objects and their context as primary and secondary objects in a figure/ground relation. Both primary and secondary objects include cognitive, spatial, temporal, and geometric characteristics.

site precision

The level of precision when recorded provenance is within 500m of the true location of a biological specimen collection.

string

A linear sequence of alpha-numeric characters in digital text format.

sub-regional precision

The level of precision when recorded provenance is within the extent of the same minor geographic feature (small city, canyon, ridge) as the true location of a biological specimen collection

traverse

A method of relating spatial relationships similar to an instrument survey procedure of the same name. Coordinates of an unknown point or series of unknown points are calculated using direction and distance measurements taken from a point with known coordinates.