

University of Redlands

InSPIRe @ Redlands

MS GIS Program Major Individual Projects

Theses, Dissertations, and Honors Projects

12-2009

Using GIS to Manage Surfrider Foundation Membership and Promote Growth

Luke M. Moffat

University of Redlands

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



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

Recommended Citation

Moffat, L. M. (2009). *Using GIS to Manage Surfrider Foundation Membership and Promote Growth* (Master's thesis, University of Redlands). Retrieved from https://inspire.redlands.edu/gis_gradproj/140



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

**Using GIS to Manage Surfrider Foundation Membership
and Promote Growth**

A Major Individual Project submitted in partial satisfaction of the requirements
for the degree of Master of Science in Geographic Information Systems

by

Luke M. Moffat

Fang Ren, Ph.D., Committee Chair

Diana Sinton, Ph.D.

December 2009

Using GIS to Manage Surfrider Foundation Membership
and Promote Growth

Copyright © 2009

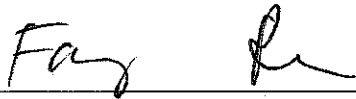
by

Luke M. Moffat

The report of Luke M. Moffat is approved.

A handwritten signature in cursive script, reading "Diana Sinton".

Diana Sinton, Ph.D.

A handwritten signature in cursive script, reading "Fang Ren".

Fang Ren, Ph.D., Committee Chair

December 2009

Acknowledgements

I would like to give thanks to my wife, Ashley, for her patience and support during this project. I would also like to thank my academic advisors, Fang Ren and Dianna Sinton, for their guidance, and Steven Daugherty and John Laska for their technical advice. Lastly, I would like to thank Mark Rauscher and the Surfrider Foundation for their support and cooperation.

Abstract

Using GIS to Manage Surfrider Foundation Membership

And Promote Growth

by

Luke M. Moffat

The Surfrider Foundation, a non-profit environmental organization, has an extensive membership distributed throughout the United States. It has become difficult for the organization to visualize the extent and support of its members and how spatial trends may affect the continued success of its programs. To discover the geospatial pattern and distribution of its membership and their donation activity, the Surfrider Foundation required a means of visualizing this data in order to perform further analysis and gain a better understanding of these datasets. For this purpose, an online GIS application was created to display all the datasets included in this project. This application allows the Surfrider Foundation to visualize, explore, and analyze membership distribution for themselves with no prior knowledge or skill with GIS software and theory. Simple GIS tools were also developed to allow the Surfrider Foundation to analyze its membership data.

Table of Contents

Chapter 1 – Introduction	1
1.1 Client	1
1.2 Problem Statement	2
1.3 Proposed Solution	2
1.3.1 Goals and Objectives	3
1.3.2 Scope	3
1.3.3 Methods	4
1.4 Audience	5
1.5 Overview of the Rest of this Report	5
Chapter 2 – Background and Literature Review	7
2.1 Analyzing Membership and Fundraising in Space and Time	7
2.1.1 Spatial and Temporal Aspects of Membership Distribution	7
2.1.2 Socio-demographic Characteristics of Members	9
2.2 Web GIS	10
2.3 Summary	11
Chapter 3 – Systems Analysis and Design	13
3.1 Problem Statement	13
3.2 Requirements Analysis	13
3.3 System Design	15
3.4 Project Plan	17
3.4.1 Problem Analysis	18
3.4.2 Requirements Analysis	19
3.4.3 Logical Design	19
3.4.4 Data Acquisition	19
3.4.5 Physical Design	19
3.4.6 Analysis	20
3.4.7 Documentaion	20
3.4.8 Presentaion and Deliverables	20

3.5	Summary	20
Chapter 4 – Database Design	21	
4.1	Conceptual Data Model.....	21
4.2	Logical Data Model.....	24
4.3	Data Sources	27
4.4	Summary	28
Chapter 5 – Implementation	29	
5.1	Data Manipulation and Processing	29
5.2	Geodatabase Loading.....	29
5.3	Map Document.....	30
5.4	Building the Web GIS Application	38
5.5	Introduction to the World Wide Web	46
Chapter 6 – Results and Analysis.....	49	
6.1	Web GIS Interface	49
6.2	The Table of Contents	51
6.3	Search Tools.....	53
6.4	Find an Address Tool	59
6.5	Analyzing Member Donations	60
6.6	Benefits of a Web GIS Application.....	63
Chapter 7 – Conclusions and Future Work	64	
7.1	Conclusion	64
7.2	Future Work	64
Works Cited	67	
Appendix A. Web GIS Application Code	69	
Appendix B. Data Table Export Code.....	91	

Table of Figures

Figure 3-1:	System design diagram	16
Figure 3-2:	Project workflow.....	18
Figure 4-1:	Surfrider foundation UML diagram.....	22
Figure 4-2:	Membership growth UML diagram	23
Figure 4-3:	Project database layers and tables	25
Figure 5-1:	Geodatabase file import interface	30
Figure 5-2:	Address Locator properties	31
Figure 5-3:	Membership records geocoded by zip code and symbolized by number of members at each point	32
Figure 5-4:	Join Data tool	33
Figure 5-5:	Donation and membership join.....	34
Figure 5-6:	Map document layers	35
Figure 5-7:	ESRI StrettMap World 2D Basemap	36
Figure 5-8:	ArcGIS Services Directory for the Surfrider MapServer	37
Figure 5-9:	Adobe Flex Builder interface.....	38
Figure 5-10:	GIS Task from the ArcGIS API Flex Resource Center	39
Figure 5-11:	Source Code from the ArcGIS API Flex Resource Center.....	40
Figure 5-12:	Query results table	41
Figure 5-13:	Export search results from th einterface data tabl to a CSV file.....	42
Figure 5-14:	Search by Selection tool.....	42
Figure 5-15:	Search by Selection buffer tool.....	43
Figure 5-16:	Search by Attribute tool	43
Figure 5-17:	Find an Address tool	44
Figure 5-18:	Table of Contents	45
Figure 5-19:	Bookmarks	45
Figure 5-20:	Adobe Flex Builder Design Interface	46
Figure 5-21:	Adobe Flex Builder Export Release Build.....	47
Figure 5-22:	Application login request.....	47
Figure 6-1:	Web application user interface.....	49

Figure 6-2:	Predetermined bookmarks	50
Figure 6-3:	Identifying features – members near surfing spots	52
Figure 6-4:	Identifying features – members near universities	53
Figure 6-5:	Search by Polygon tool	54
Figure 6-6:	Record identification from the table	55
Figure 6-7:	Record attributes	56
Figure 6-8:	Select by Buffer	57
Figure 6-9:	Select by Attribute – Member ID.....	58
Figure 6-10:	Select by Attribute – Zip Code	59
Figure 6-11:	Find an Address	60
Figure 6-12:	Finding areas of high donation	62

List of Tables

Table 1.	Summary of requirements.....	14
Table 2.	Effectiveness of the file geodatabase.....	26
Table 3.	Zip codes yielding highest donations.....	61
Table 4.	Regional chapters yielding highest donations.....	63

List of Acronyms and Definitions

API	Application Programming Interface
ESRI	Environmental Systems Research Institute
CSV	Comma-Separated Value
DBF	dBase
FTP	File Transfer Protocol
GIS	Geographic Information Systems
GPS	Global Positioning System
LQ	Location Quotient
REST	Representational State Transfer
SQL	Structured Query Language
SWF	Shockwave File
TIGER	Topologically Integrated Geographic Encoding and Referencing
UML	Unified Modeling Language
URL	Uniform Resource Locator
WGS-84	World Geodetic System 1984

Chapter 1 – Introduction

With its advanced analytic toolsets, geographic information systems (GIS) software is seen as a valuable tool designed to find, display, and analyze geospatially distributed data. GIS is replacing old “push pin” methods of finding locations on a paper map and attempting to identify patterns and relationships (Fox, 1997). Because of this, many businesses, churches, and environmental organizations have implemented GIS applications to manage membership records and analyze the spatial distribution of members in order to determine growth potential.

The Surfrider Foundation, a non-profit environmental organization dedicated to the health and preservation of the world’s oceans, shorelines, and surfing areas, has also investigated the possibility of implementing a GIS application to manage past and present membership growth and explore possible expansion opportunities. With an extensive membership distributed throughout the United States, the Surfrider Foundation particularly hopes to use GIS technology to display membership and donation activity within a geographic context in which other data, such as chapter areas, popular surf spots, and universities, are present. In doing so, the existing membership patterns and anomalies will be easier to identify and the relationships between donation and potential geographic indicators can be further explored. This information can then be used by the organization to develop and implement new methodologies to promote membership growth, especially in areas identified to have limited activity. Once areas of sluggish growth are identified, successful initiatives in areas experiencing rapid increase may also be applied to encourage Surfrider Foundation donations and support, as well as stimulate environmental and ocean awareness through educational programs. This project was developed with these needs in mind and GIS technology was used as an integral part of this analysis.

The remainder of this chapter introduces the client, problem, and proposed solution for this project. In addition to this, the goals, objectives, and scope of this project are presented and the methods used to construct this project are discussed. The chapter then concludes with a description of the intended audience for this document and an overview of the remaining chapters.

1.1 Client

The Surfrider Foundation is a non-profit organization with a grassroots approach to conservation and environmental activism. It has developed many community-based programs which educate and encourage the protection of coastal waters and beaches. The organization strives to maintain the biodiversity and ecological integrity of the world’s oceans and coastlines through research, instruction, and activism. The Surfrider Foundation also fights to preserve free and open access to waves and beaches for all people. Founded in 1984 by a group of surfers living and surfing near Malibu, California, the Surfrider Foundation has grown to include members throughout the United States, Canada, Australia, Mexico, France, Germany, Spain, Portugal, Italy,

Japan, and Brazil. Members have proven to be powerful spokespeople in the preservation of the ocean environment and have fought on numerous issues including water quality and pollution, wetlands protection, and shoreline and surfing area preservation. In order to better organize its membership database, the Surfrider Foundation has specifically asked for assistance in creating a GIS application to explore the spatial characteristics of its membership and look for new methodologies to promote future growth and activity.

The point of contact for this project is Mr. Mark Rauscher, Assistant Environmental Director, and Mr. Steve Blank, Director of Development at the Surfrider Foundation National Office in San Clemente, California. Mr. Rauscher provides scientific and technical assistance to the Surfrider Foundation's network of members and partners, and manages all educational programs and activities. As Director of Development, Mr. Blank manages all fundraising efforts and oversees the grant writing and membership departments. All data for this project was extracted from the Surfrider Foundation's SQL server databases and delivered in an appropriate format by Mr. Olaf Lohr, Membership Coordinator at the Surfrider Foundation.

1.2 Problem Statement

Because the Surfrider Foundation has approximately 155,000 records of current and historic membership associated with 75 active chapters across the United States, it has become difficult to visualize the extent of its membership network and how spatial trends may affect the continued success of its programs. With no effective tools to analyze the spatial distribution of membership, the organization has difficulty pinpointing areas of potential growth.

The Surfrider Foundation, which is supported by donations from its members, also needed to understand which areas had a greater potential for fund-raising activities. Finding areas throughout the United States where the generosity of existing membership was more prevalent was likewise a problem which the Surfrider Foundation wished to address. By analyzing the spatial distribution of member donations, the Surfrider Foundation wished to geographically focus future campaign strategies and mailers in order to petition additional contributions.

1.3 Proposed Solution

To address the issues faced by the client, a GIS solution was proposed. GIS is a powerful tool which allows the application of spatial, temporal, and statistical analyses to membership distribution in order to discover patterns, trends, and anomalies. By studying individual records in the membership spreadsheet, it is easy to conclude where an individual member is located based on address information. However, by looking at the distribution of membership on a map rather than individual members, patterns become apparent and a better understanding of the spatial distribution of Surfrider Foundation supporters can be discovered (Mitchell, 1999). By identifying these spatial patterns and relationships, GIS tools and methodologies can help the Surfrider Foundation understand the past and future growth of its membership base and its fund-

raising potential. This project was designed using GIS tools and methodology in order to visualize the Surfrider Foundation membership records in order to gain a better understanding of their spatial distribution.

To discover the geospatial pattern and distribution of its membership and their donation activity, the Surfrider Foundation required a means of visualizing this data in order to perform further analysis and gain a better understanding of these datasets. Although the Surfrider Foundation has received ArcGIS software through a grant from the Environmental Systems Research Institute (ESRI), no one within the organization has adequate training or experience to manage, analyze, or display the GIS data utilized in this project. For this purpose, an online GIS application was created to display all the datasets included in this project. This application allows the Surfrider Foundation to visualize, explore, and analyze membership distribution for themselves with no prior knowledge or skill with GIS software and theory. Simple GIS tools were also developed to allow the Surfrider Foundation to analyze its membership data.

1.3.1 Goals and Objectives

The ultimate goal of this project was to help the Surfrider Foundation develop new methodologies to boost membership numbers and donations in the future. More specifically, this project aimed to provide the Surfrider Foundation with effective yet simple tools that allowed the decision makers within the organization to visualize and analyze the spatial distribution of membership and donation records.

Based on specific requests expressed by the Surfrider Foundation, three major objectives were established that could be addressed using an online GIS application or through other means approached in this project. These objectives included the examination of membership concentration, distribution, and donation activity. These objectives were then further separated into two separate categories dealing with membership and donation activity. Further questions were then identified, many of which can be addressed using the online mapping application designed and built for the project:

- Identify areas of high member concentrations: Is membership clustered around active chapter locations? Does Surfrider Foundation membership cluster around popular beaches and surf spots? Do membership numbers appear higher near universities, especially those having strong environmental programs?
- Analyze past donation activity: Which areas have produced the largest donations? Can records within these areas be identified to focus additional fundraising campaigns and mailers?

1.3.2 Scope

The Surfrider Foundation has a global mission which includes chapters located in the United States, Japan, Australia, Mexico, Brazil, Canada, and Europe. The scope of this

project included data for the approximately 155,000 membership records located within the United States. All foreign-based records were excluded from the study area. The main project deliverables included a geodatabase containing the domestic records and a GIS web application for data visualization, exploration, and analysis.

Because of privacy concerns, only member addresses, identification numbers, the date and amount of donations given, and the date of enlistment or expiration of membership were included in the data received from the Surfrider Foundation for this project. No personal membership information was handled during the process of this project. Membership records from the last twenty-five years were provided by the Surfrider Foundation and once membership data was acquired from Mr. Lohr on March 16th, 2009, no additional records were included in this study.

As part of this project, the Surfrider Foundation received documentation of all major developmental phases, analytic findings, and necessary login information for the online mapping application allowing the organization to continue this project in the future as seen fit. It is the responsibility of the Surfrider Foundation to supplement this application as needed in the future and circulate findings or impressions within the organization via the World Wide Web or other distributable mediums.

All data management, GIS analysis, and web application assembly was conducted on a Dell Precision M4400 laptop computer. No additional hardware or software requirements were required to perform this analysis and complete this project. ArcGIS 9.3, developed and distributed by ESRI, was the primary GIS software used to perform all spatial, temporal, and spatial statistical analysis and database modeling necessary for this project. Additional software used to organize and manage data for this project included Microsoft Access and Microsoft Excel. To build the web interface for the online mapping application, Adobe Flex Builder 3.0 was used to compile the code necessary to run the application and its integrated tools.

1.3.3 Methods

The membership and donation datasets necessary to complete this project were exported from SQL Server 2000 and received from the Surfrider Foundation in comma-separated value (CSV) file format. Since this file format is used predominately for the digital storage of tabular data, some data cleaning and manipulation was required. Microsoft Access was used to organize, manage and export the data into dBase (DBF) formatted tables which could then be easily ingested into ArcGIS. Using an ESRI-provided U.S. street network, all membership records were then located in a GIS using address information. Because of ArcGIS geocoding constraints, all membership records containing post office boxes were discarded and those records containing positive address matches were included in the membership dataset.

All relevant datasets were then imported into a file geodatabase developed using ArcGIS desktop software. Many of the datasets, including donation records, chapter locations, and chapter victories, were all contained within separate data tables. Tabular joins were performed using ArcGIS to append these datasets to the membership dataset.

This data-processing method allowed additional geo-spatial statistics and location quotient-based analysis to be performed.

Once the geodatabase was populated, selected datasets, including present and historic membership and chapter boundaries, were published to a web server maintained by the University of Redlands MS GIS program. A web GIS application was then designed using Adobe Flex Builder 3.0 and, once compiled, was also published to the server. This web application allows the Surfrider Foundation to visually explore its membership data in a geographic environment and perform common GIS functionality including data queries, buffering, and geocoding.

1.4 Audience

This project may be of interest not only to the Surfrider Foundation, but also to other environmental, government, civic, business, or religious groups interested in gaining a better understanding of GIS applications and how they can be used to manage and display membership data. It is assumed that the majority of these people have no prior GIS knowledge or experience and this document is written to reflect this.

1.5 Overview of the Rest of this Report

The following chapters will provide the reader with an overview of the development, design, implementation, and conclusive results of this project. Chapter 2 provides an explanation of how GIS technology is being utilized within grassroots organizations. Chapter 2 will also provide background on prior membership-themed studies done using GIS software and what geo-spatial, temporal, or socio-demographic methods were used to approach the visualization and analysis of point-based data within a geospatial context. Web-based GIS methodology and uses are also discussed in this chapter. Chapter 3 will present the requirements and development of the system design used as a backbone for this project. Also discussed in Chapter 3 are the project plan, the major tasks, and resources that were originally required, and how they changed during the course of this project. Chapter 4 will provide an explanation of the database that was designed for this project and an in-depth explanation of the conceptual and logical data models, sources, and data collection will be provided. Chapter 5 will discuss how the database was employed as the foundation for an online GIS mapping application and how these web-based mapping tools were developed. Chapter 6 provides an explanation of the results of the implementation and analysis and how the web-based mapping application allows the client to visual, explore, and analyze membership data. Finally, Chapter 7 will present a conclusion of this work, how it satisfied the original requirements of the project, and future recommendations and work that could be pursued to further this research.

Chapter 2 – Background and Literature Review

In order to gain a broader understating of how GIS technology has been implemented in other organizations and how to successfully apply past research and methods to this study, a literature review was conducted prior to the commencement of this project.

This review investigates the use of GIS in other grassroots organizations and how it has been used effectively to analyze membership growth and fundraising in a spatial or temporal environment using GIS technology. Additional information on how the socio-demographic status of geographic locations may impact membership distribution and growth within private organizations is discussed. Diverse concepts and examples demonstrating the effectiveness of web GIS applications is also presented. The final section of this review is a summary of how past research can be linked to the Surfrider Foundation membership project.

2.1 Analyzing Membership and Fundraising in Space and Time

The major goals of many nonprofit organizations, especially within the environmental realm, is to rally around a cause, sway public opinion, and make a difference in their local communities and throughout the world. To reach this goal, it is important to maintain a healthy membership base and to constantly engage in fundraising, which requires a thorough understanding of membership patterns. The changes in membership or fundraising vary for different places and different periods of time. Therefore, both spatial and temporal dimensions of the membership patterns are critical and have been examined in the literature.

Recently, GIS technology has been widely used in many grassroots organizations to study membership patterns across space and time. Sieber (2000) explains why these organizations are looking to GIS technology to achieve their various objectives: “The greater promise of [GIS] technology, to nonprofits such as grassroots organizations and nongovernmental organizations, is that it may assist in influencing public policy, through the sophistication of analysis and presentation of powerful images” (p. 15). For this purpose, these organizations are looking to GIS tools and methodology as an indispensable advantage as they move into the future.

Traditionally, grassroots organizations rely on spreadsheets to store and manipulate their membership and donation records. But there comes a time, as membership numbers begin to swell and donation monies accumulate, when a small organization begins to see the need to identify patterns and trends in their data and find effective ways to analyze and display their findings. In such situations, tables and charts are not enough for analysis because they do not explicitly present the spatial component associated with geographic data and are therefore insufficient for displaying the spatial patterns and trends (Granados, 2003). Granados further explains that by using maps to display this information, spatial patterns are easily identified and the geographic element associated with the data becomes easier to analyze and understand. GIS tools and methodology,

therefore, provide a useful approach to the capture, analysis, and storage of spatial information and offer the organization an additional process to understand their data.

Many organizations, however, fail to employ a successful GIS program by themselves because of the expense and complexity of developing a full GIS implementation. “For GIS work”, says Sieber (2000), “even high levels of commitment are insufficient; quality and continuity of work are essential. Because a GIS is only as accurate as the data entered...good data-entry personnel are crucial. Even if digital spatial data is procured from other sources, technical knowledge is required to use the data properly in a spatial analysis” (p. 18). Because of this, outsourcing to external contractors, institutions, and universities for its GIS needs, including data, systems, trained personnel, and expertise, has become a desirable option for many organizations.

As an organization begins to implement GIS technology, more analytic options become available. The ability to visualize and explore large datasets becomes available. GIS tools can be implemented to further analyze socio-demographic, spatial, and temporal characteristics in membership growth and distribution.

2.1.1 Spatial and Temporal Aspects of Membership Distribution

Due to its usefulness in spatial data exploration, GIS technology has been applied by some grassroots organizations to manage membership and donation datasets and examine the spatial correlation and patterns of its activities and supporters. For example, by representing membership distribution on a map, officials from the Notre Dame church were able to conduct a successful fund-raising campaign by locating and visiting 2,700 families from their congregation (Fox, 1997). During the campaign, membership records were displayed in a GIS and broken down into geographic areas. This new technique replaced the use of push-pins on a wall map and allowed church officials to establish collection areas and routes making fund-raising more manageable and convenient. Using GIS to isolate membership clusters and focus collection efforts, time and resources were better managed during the campaign.

Another study conducted by Wikle (1998) investigated the spatial patterns of the World Wildlife Fund, The Nature Conservancy, and the National Resources Defense Council to determine if similar membership patterns exist for different types of environmental organizations across the United States. By using a statistical method called the location quotient (LQ), Wikle compared membership numbers at the county level to determine where membership ratios are higher or lower than the national average for all three organizations. The LQ method, explains Wikle, is used to “determine the concentration of an activity (membership in environmental organizations) in an areal unit in comparison to the presence of some other activity (population)” (p. 43). The LQ method, along with GIS tools used for spatial analysis, allow an organization to identify the number of members living in a specific area and how those numbers compare with other areas sharing similar characteristics.

Examining spatial trends and patterns introduces correlations between members and regional environmental issues. This correlation leads to a better understanding of

member interests and the extent of their willingness to support environmental activism. Spatial patterns may be found in or between spatial datasets. Gregory (2008) expands on the important role of GIS in recognizing and analyzing spatial patterns by identifying spatial patterns in data. With different spatial datasets, it is possible to “confirm relationships that were believed to have existed, thus providing evidence for an existing explanation.” These “patterns may suggest new relationships that require new explanations” and “may challenge an orthodox explanation by showing that the expected pattern is not actually present” (p. 124).

Being in close proximity to a specific location or event can lead to the increase of other related incidents or situations. In describing the spatial relationships between hazardous waste sites and cardiovascular disease and heart attacks, many suggest “that proximity is a surrogate for exposure” (Golden & Schell, 2008). When working with the membership distribution of an organization, identifying which geographic, social, environmental, or economic variables have led to past membership growth and activity is beneficial. This leads to a greater understanding of the variables which directly affect membership growth or decline.

By mapping the change or movement of individuals through time, additional insights into how the overall population behaves becomes apparent (Mitchell, 1999). An in-depth study of spatial growth and trends allows an organization to anticipate future membership distribution, evaluate the results of current enlistment programs, and decide on further courses of action. According to Mitchell (1999), there are three time patterns that can be represented on a map: trends, cycles, and before and after. Trends identify the increase or decrease of a population or event, while cycles show the recurring patterns that may reveal behaviors in population growth or decline. By mapping conditions before and after an event, the impact of that event can be observed.

2.1.2 Socio-demographic Characteristics of Members

Socio-demographic variables such as age, race, education, income and size of city of residence can be directly associated with environmental membership rates (Wikle, 1998). Wikle suggests that most members of mainstream environmental organizations can be categorized as affluent, educated, and most likely to live in urban or suburban areas.

Monroe (2002) uses socio-demographic variables to examine the recent decline of Freemason membership in Oklahoma. Along with socioeconomic influences and patterns in his data, Monroe identifies valid predictor variables in membership levels which explain which population groups are more likely to join and remain members. Although difficult to isolate when working with a country-wide membership dataset, these socio-demographic variables, once identified, will help the Surfrider Foundation target specific geographic areas for contributions, recruitment, and retention.

In his study of spatial patterns in Kiwanis membership, Wikle (1997) was able to construct a socioeconomic profile of its affiliates by examining membership distribution. From his findings, Wikle concluded that membership numbers were highest in rural counties which had experienced recent losses in population and retail sales. Urban

counties which had experienced a population increase and economic growth, however, were found to have low membership numbers. By linking Kiwanis membership and local economic circumstance, Wikle was able to ascertain that Kiwanis membership was strongest in most rural communities. This conclusion offsets the previous assumption that membership numbers would be higher in urbanized counties and helped Wikle identify a socioeconomic profile of Kiwanis membership. This study, along with Wikle's (1998) evaluation of membership patterns for mainstream national environmental agencies, help smaller organizations identify membership profiles and how they may be associated with specific geographic regions

2.2 Web GIS

GIS concepts and technology, which were at one time solely utilized by a select few, are now becoming available to the world via the Internet. As Harder (1998) suggests, "The internet doesn't change the fundamental nature of GIS, it just gets it online" (p. 1). Millions of people are accessing geographic data on a daily basis via the Internet to locate addresses, check traffic conditions, plan driving routes, or query real estate databases.

Many of these online GIS services deliver interactive content that the user can manipulate. For the majority of this online audience, the technology behind GIS is foreign and complex, being traditionally conducted by those with extensive software, data, and technical training. Thus, the use of GIS technology is often reserved for those possessing these skills and the necessary hardware and software programs. Nonetheless, online users are able to access geographic data, perform GIS analysis, and present results with no prior knowledge of GIS (Peng, 2002). Online GIS applications can be used to present geographic information and analysis to anyone accessing the Internet. The user no longer needs to purchase expensive GIS software or receive additional training in order to view and analyze geographic information using spatial analysis tools. Timeliness is also an advantage since users can now "log onto the Web, query the information and make the map, almost instantly" (Peng, 2002).

When developing a web-based GIS application, knowing the type of information and tools to include is an important consideration to insure that the user's needs are met. In Gwinnett County, Georgia, eight separate County departments were using GIS desktop software at one level or another. Realizing that many of these users mainly viewed and queried data, the County decided to create a straightforward web-based GIS application for these "query-only" users. Search functions were incorporated into the application allowing users to query data from four spatial and tabular databases. By holding workshops with each of the eight participating departments, functional, informational, and operational requirements were determined which allowed developers to build a single application that could be used by the entire County (Stevenson, 2005).

Many online GIS applications are being published to the Internet using ArcGIS Server. There are many advantages to using ArcGIS Server technology to distribute data and maps to the world. According to the Environmental Systems Research Institute (2009), ArcGIS Server can be used to publish fast, intuitive web maps which can be

customized to a specific audience allowing easy access to real-time data. The ability to centrally manage large datasets while ensuring data security and integrity is also an advantage. Application developers also use ArcGIS Server application programming interfaces (APIs) to build and deploy interactive websites. This is allowing many businesses and government entities to distribute a variety of geographic information over the web.

2.3 Summary

This project greatly benefited from insights of past membership-related studies using GIS. Other grassroots organizations have successfully implemented GIS tools and technology to achieve their various objectives. The Surfrider Foundation would also benefit from a GIS application which can facilitate their exploration and examination of their membership data. The spatial, temporal, and socio-demographic concepts and methodologies presented in this literature review provide insights into the functions that would be useful for spatial data exploration, which were eventually implemented in this project. As other applications have demonstrated, a web-based GIS application could be useful to the Surfrider Foundation as an additional data visualization and analysis tool to gain further insights into membership and donation distribution. In summary, the approach taken to address the Surfrider Foundation's needs was to provide the organization with an intelligible tool to perform custom GIS analysis in an on-line environment.

Chapter 3 – Systems Analysis and Design

In order to demonstrate an understanding of the client's problem and meet the goals and objectives of the project, a systems analysis and design was an integral step in the early stages of planning. Through client interaction and feedback, a set of system requirements were established that helped guide the further development of a system design and project plan.

This chapter outlines the approach taken to identify the requirements necessary to develop a GIS system architecture that would meet the needs of the client. An understanding of the major elements involved in the system architecture is also discussed to illustrate how raw data collected for this project was used to conduct analysis, produce maps, and provide data visualization through an online GIS application. Project workflow is also presented in this chapter to describe the approach taken to design and implement this project.

3.1 Problem Statement

The Surfrider Foundation has a membership base comprised of approximately 155,000 current and historic individual records associated with 75 active chapters across the United States. Due to the large number of records and their extensive distribution, the Surfrider Foundation finds it difficult to visualize the extent of its members, the financial support those members provide, and how spatial trends may affect the continued success of its recruitment efforts and environmental programs. The Surfrider Foundation sees GIS technology as an effective tool to visualize, manage, and analyze the spatial patterns of its membership and member donation distribution. In doing so, new patterns and anomalies which could lead to additional growth and environmental awareness may be discovered and therefore allow the Surfrider Foundation to geographically focus future campaign strategies and mailers to petition additional contributions. However, the Surfrider Foundation currently has neither knowledge nor experience using GIS, though they have the ArcGIS desktop software. To address this issue, this project developed an online GIS application for the client to visualize, explore, and analyze membership data.

3.2 Requirements Analysis

After multiple discussions with the Surfrider Foundation, a requirements analysis was conducted based on the proposed objectives of the project. The requirements analysis, being comprised of details related to the requirements, design, and implementation of the project, was separated into non-functional and functional requirements. The non-functional requirements are the “what” of the project which identify the major objectives which are proposed. The functional requirements are the “how” of the project and the approaches being taken to satisfy the non-functional requirements. The goal of the requirements analysis was to ensure that the Surfrider Foundation's needs were met and that all data acquisition, database organization, analysis, and applications were completed within the allotted timeframe. Table 1 provides a summary of the functional and non-

functional requirements which were determined from the proposed objectives of this project.

Table 1. Summary of requirements

Non-functional requirements	Functional requirements
Organize and store data	Develop a file geodatabase
Identify the distribution, relationships, and patterns of growth/decline in membership and donation data.	Provide effective analytical tools for spatial data exploration and analysis and conducting required analysis
Ability to easily visualize, query, and geocode data.	Develop an Internet mapping application

As part of the non-functional requirements, the Surfrider Foundation required a basic yet adequate system to organize and store the shapefiles which were being assembled for this project. The solution for this was the creation of a file geodatabase using ArcGIS desktop software to house and organize all relevant shapefiles used in the project. The major objectives of the project were to identify the distribution, relationships, and patterns of growth and decline of Surfrider Foundation membership records over the last twenty-five years. These objectives were met by developing a set of analytical tools to explore and visualize the multiple datasets within a geographic environment and performing necessary analysis.

The final non-functional requirement included an uncomplicated method of visualizing this data and performing various geospatial functions, because the users are those having very little GIS expertise. Instead of requiring the client to interact with professional GIS software, a web-based mapping application was developed to allow the graphic display of Surfrider Foundation datasets. As part of this online GIS application, various geospatial tools were developed to allow the Surfrider Foundation to perform basic GIS tasks to continue the research included in this project. The following functionality was projected for the web mapping application in order to satisfy the requirements associated with this application:

- The application would use a mapping service to communicate Surfrider Foundation data on the MS GIS Program web server to users for a year.
- The application would allow the user to graphically display Surfrider Foundation membership records, chapter locations and areas, and popular surf spots and universities where membership accumulation is possible.
- The application would allow the user to interactively navigate the map display through panning, zooming, and preset geographic extents.

- The application would allow the user to query membership records using a draw tool to select areas of interest or by creating buffers around a user-specified area of interest.
- The application would allow the user to search for individual membership records on the map by querying a specific identification number associated with the member.
- The application would allow the user to view queried results with a separate table and identify the location of specific records on the map by selecting those records in the table.
- The application would allow the user to find an individual address or a list of addresses on the map.

Through the functional and non-functional requirements established during the early planning stages of the project, an application could be designed to allow the client to effectively visualize and explore the data being stored within the geodatabase.

3.3 System Design

The system design revolves around the file geodatabase, the data storage framework assembled for the project. The data is the most important element for most GIS projects and without good data, the various GIS outputs, including analysis, maps, and applications, are poor. The design for this project started with the data received from the client. The workflow of the system, including the various outputs and infrastructure requirements, is presented in Figure 3-1.

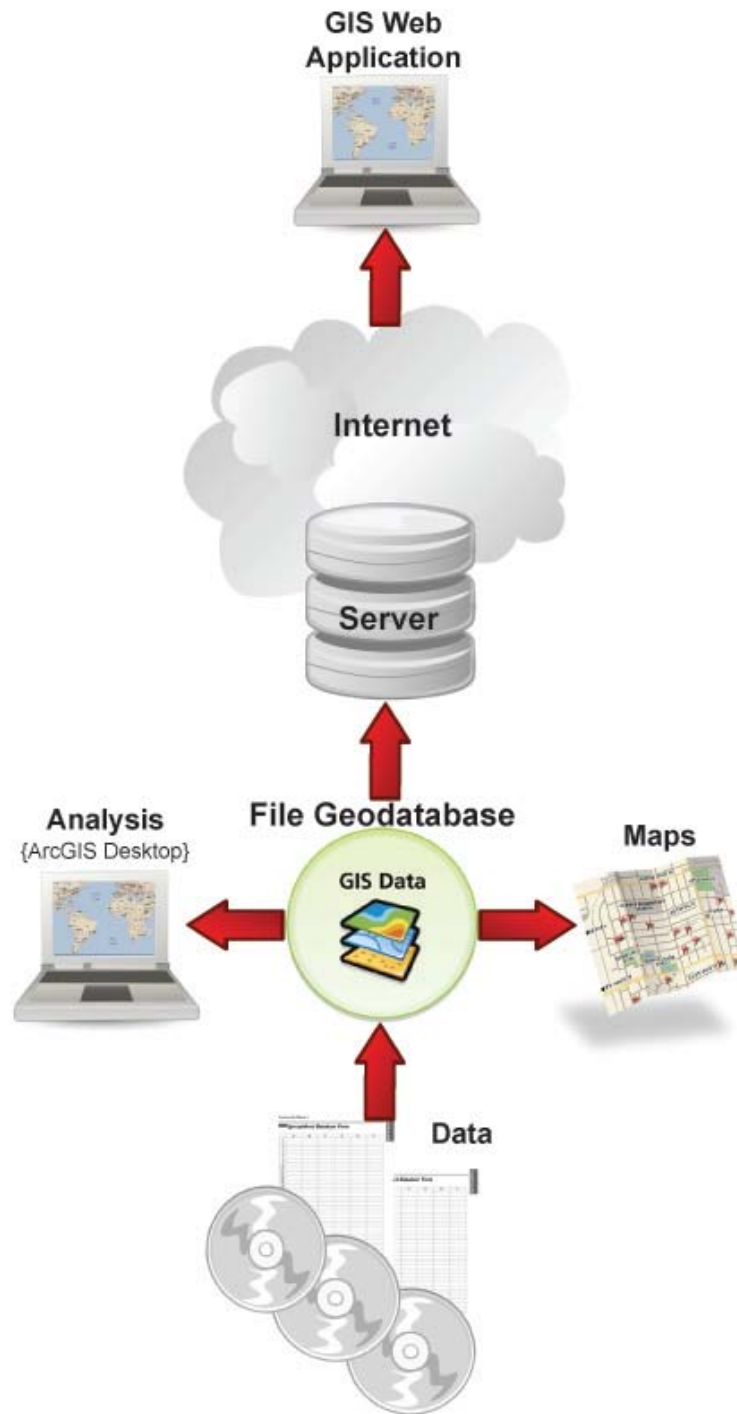


Figure 3-1: System design diagram

The development environment for this system was ESRI's ArcGIS 9.3 desktop software. Using ArcGIS, the data was ingested and the file geodatabase was constructed. ArcGIS desktop was used to complete the analytic requirements and map creation for this project. To build the online GIS application, the data and maps were then uploaded to the

web server and made available to the client via the Internet. The web interface and application tools were designed and compiled using Adobe Flex Builder 3.0 and utilized the data that was previously uploaded to the server. The web application is accessible via the World Wide Web using a standard Internet browser such as Internet Explorer or Mozilla Firefox. Users can request and receive data and information through the network, allowing them to gain access to membership data and perform common GIS-related tasks.

3.4 Project Plan

Eight major tasks were proposed as a workflow strategy during the initial planning of this project. This initial project plan was followed during the course of the project without any major setbacks. This project workflow is presented in Figure 3-2 and further explained in the following subsections.

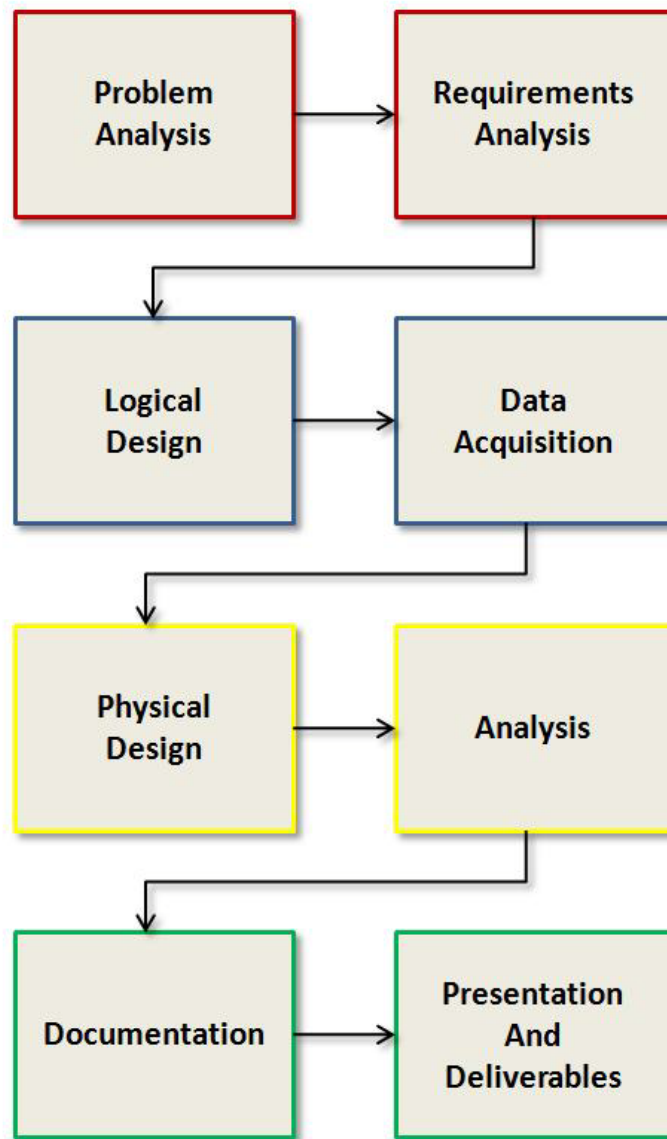


Figure 3-2: Project workflow

3.4.1 Problem Analysis

At the beginning of the project, a problem analysis was performed to identify a client and develop a suitable project. Once a client was identified and contacted, a project was designed to satisfy a specific requirement established by the client. During the problem analysis phase, any foreseen problems or risks associated with the project were addressed before moving on the requirements analysis phase of the project.

3.4.2 Requirements Analysis

As discussed earlier in this chapter, a series of requirements were established to aid in the design of the project. Meetings with the client were conducted to identify relevant requirements for the project. Once these requirements were identified, major objectives and the overall scope of the project was established. The requirements analysis included the functional and non-functional requirements, as well as the system and application architecture that would be designed and implemented to satisfy the objectives of the project.

3.4.3 Logical Design

After the requirements analysis was completed and the design requirements for the system architecture and web application were defined, a logical design was then created. This design included a data organization plan explaining all hardware, software, and data necessary to complete the project. Data requirements were discussed with the Surfrider Foundation and arrangements were made to assemble all essential datasets.

3.4.4 Data Acquisition

Once the logical design had been completed, all necessary datasets were acquired from the client. The Surfrider Foundation maintains a SQL 2000 database where all membership and donation history is recorded and stored. Membership and donation information was extracted from this server and provided in comma-separated value (CSV) file format for this project. Since this file format is predominately used for the digital storage of tabular data, some data cleaning and manipulation was required for use in GIS software. Microsoft Access was used to organize, manage, and export the data into dBase (DBF) formatted tables which were then easily ingested into ArcGIS. Other datasets received from the client were in Microsoft Excel format and were easily ingested and manipulated into ArcGIS software.

3.4.5 Physical Design

During the physical design phase of the project, all requisite data were processed and organized before analysis could begin. A geodatabase was designed and built using ESRI's ArcGIS 9.3 desktop software, which was used as the primary storage environment for the data. Any additional hardware and software elements essential for the project were also coordinated.

The major part of the physical design phase of the project was the design and assembly of an online GIS application. This application was created to allow the Surfrider Foundation to interactively explore, query, and analyze its historic membership dataset. This application was built and compiled using Adobe Flex Builder 3.0 and published to a web server where it was made available to the client via secure login. The File Geodatabase was also pushed to the server and can be accessed through the web application's user interface.

3.4.6 Analysis

Once the database was built, the analysis portion of the project was conducted. The analysis phase ran concurrent with the design and assembly of the online GIS application. This analysis consisted of geocoding the data and conducting select spatial and temporal analytic processing in ArcGIS in order to meet the major goals and objectives previously established as part of the requirements analysis. This analysis was seen as an additional element of this project and, along with the web application, was presented to the client for their use and dissemination.

3.4.7 Documentation

Project documentation, including the system and database design, project implementation, and final analytic results, was constructed concurrently with the system design and analysis phases. The documentation provides a written account of the major steps taken to complete the project.

3.4.8 Presentation and Deliverables

Once the project had been completed, the analytic results and GIS web application were presented during a formal meeting with Surfrider Foundation membership coordinators, environmental analysts, and managers. The web mapping application was explained and demonstrated at this meeting and all analytic results of the project discussed.

3.5 Summary

The success of the project was a direct result of the project plan that was established during early planning stages and the system design that was created to illustrate the project architecture and workflow. The project plan did require a fair degree of flexibility due to time constraints and unforeseen events that competed with this project. The plan, however, was seen as a guide to the necessary steps for completion and was closely followed during the course of this project. The system design was also a necessary outline to visualize the integral elements of the project. It identified the essential elements and how those elements would interact and rely on each other to form a successful architecture suitable for the completion of all major objectives.

Chapter 4 – Database Design

Every successful GIS requires complete and adequate data, as well as a systematic manner of storing and managing that data. Once the data have been assembled, a well-structured geodatabase is an essential element model geographic features, attributes, and relationships necessary for analysis.

This chapter begins by addressing the conceptual data model designed to represent the inherit relationships between the data elements involved in this project. The logical data model, consisting of the database design, will then be discussed and further explanation given to the separate data elements and their identifiable attributes included in this project. All relevant layers, data sources, and manipulation necessary to prepare the data for analysis will then be explained.

4.1 Conceptual Data Model

The conceptual model for this project was developed using Unified Modeling Language (UML) to visualize the various data elements associated with the database and how those elements interact with each other. Two major data elements were described in this project using a UML diagram: Surfrider Foundation membership and membership donations (Figure 4-1). It is understood that these two components are directly correlated with the potential growth of any non-profit organization and without them the organizations would cease to exist.

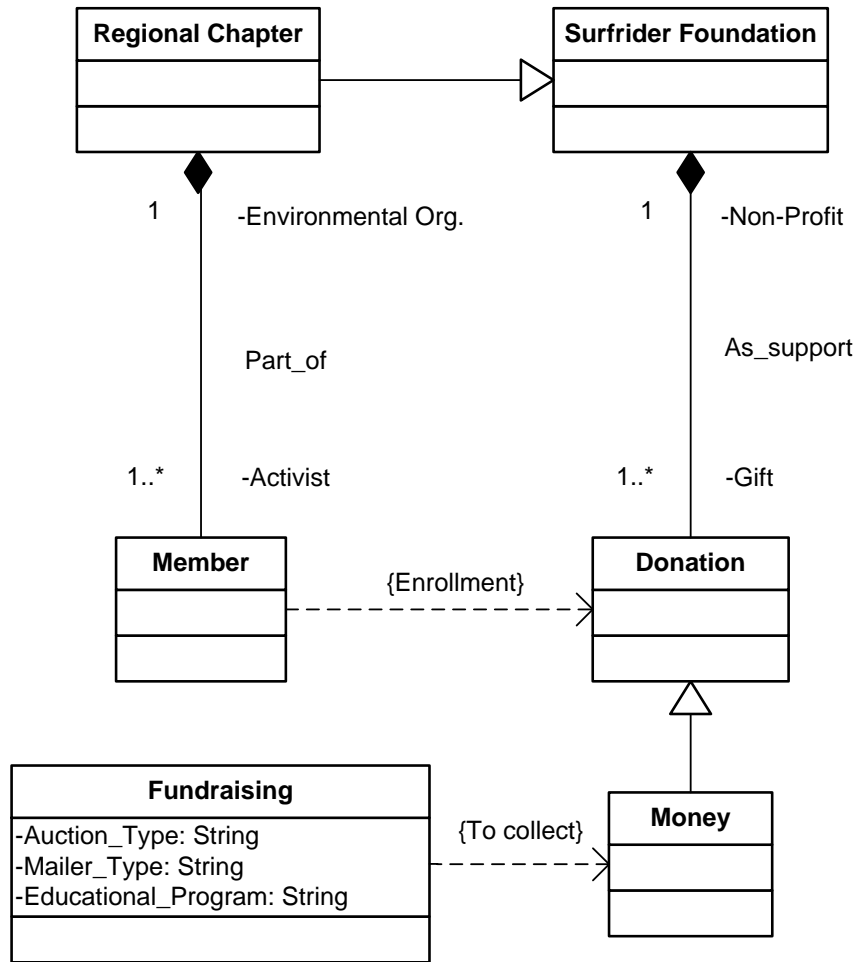


Figure 4-1: Surfrider foundation UML diagram

There is a direct relationship between membership numbers and donations which determine the success of the Surfrider Foundation programs. In order to become a member of the Surfrider Foundation, a small donation is required with enrollment. Additional donations are invited thereafter. The donations are gifts from the members to support the Surfrider Foundation's non-profit activities. Donations can be monetary or volunteer work associated with Surfrider Foundation activities and initiatives. In order to increase monetary donations, the Surfrider Foundation often holds fundraising events such as auctions or concerts, sends out mailers to its membership, or invests in environmental awareness and other educational programs in the community. As membership numbers continue to grow, so too will the donations required for the organization's success.

There are numerous variables that can be identified to determine the catalyst for membership growth in the Surfrider Foundation. Three of these variables described in this project are location, demographics, and historic environmental events, including Surfrider victories (Figure 4-2).

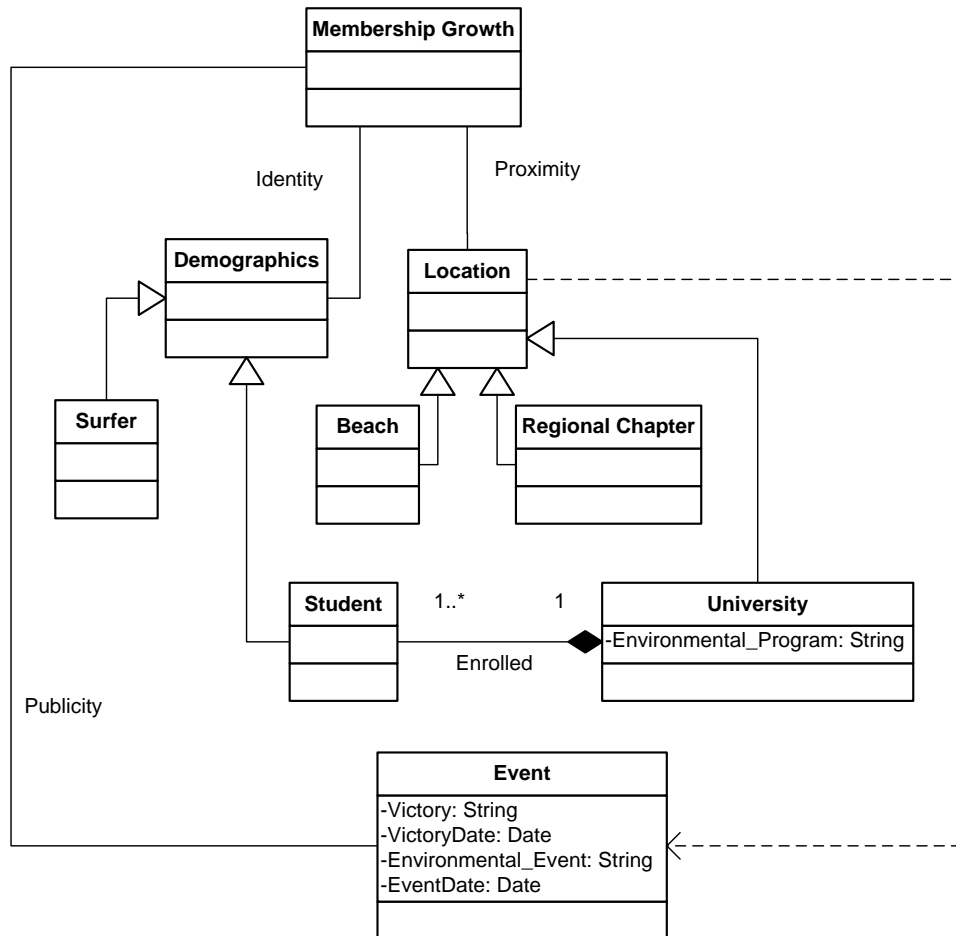


Figure 4-2: Membership growth UML diagram

Surfers and beach-goers are a major demographic because they are identified as individuals having a great appreciation for the ocean and a desire to keep it healthy and clean. College-aged individuals may also be a major demographic for Surfrider Foundation membership due to their willingness to be included in what they see as a worthy cause.

Historic events may determine if membership numbers have spiked in certain locations during or after an event in that area. These events may include environmental incidents such as major pollution or the threatening of a popular beach or surf spot. These events may also include Surfrider Foundation victories won in the courtroom, through rallies, or through the press. These events are often well publicized and introduce the public to the Surfrider Foundation's focus of preserving the environment and possibly encourage others to offer support.

4.2 Logical Data Model

The logical model was derived from the data and relationships presented in the conceptual model. This model describes the feature classes and tables that were implemented in the study of Surfrider Foundation membership and donation records. Figure 4-3 is a representation of the data layers and the data type and associated attribute fields of each layer.

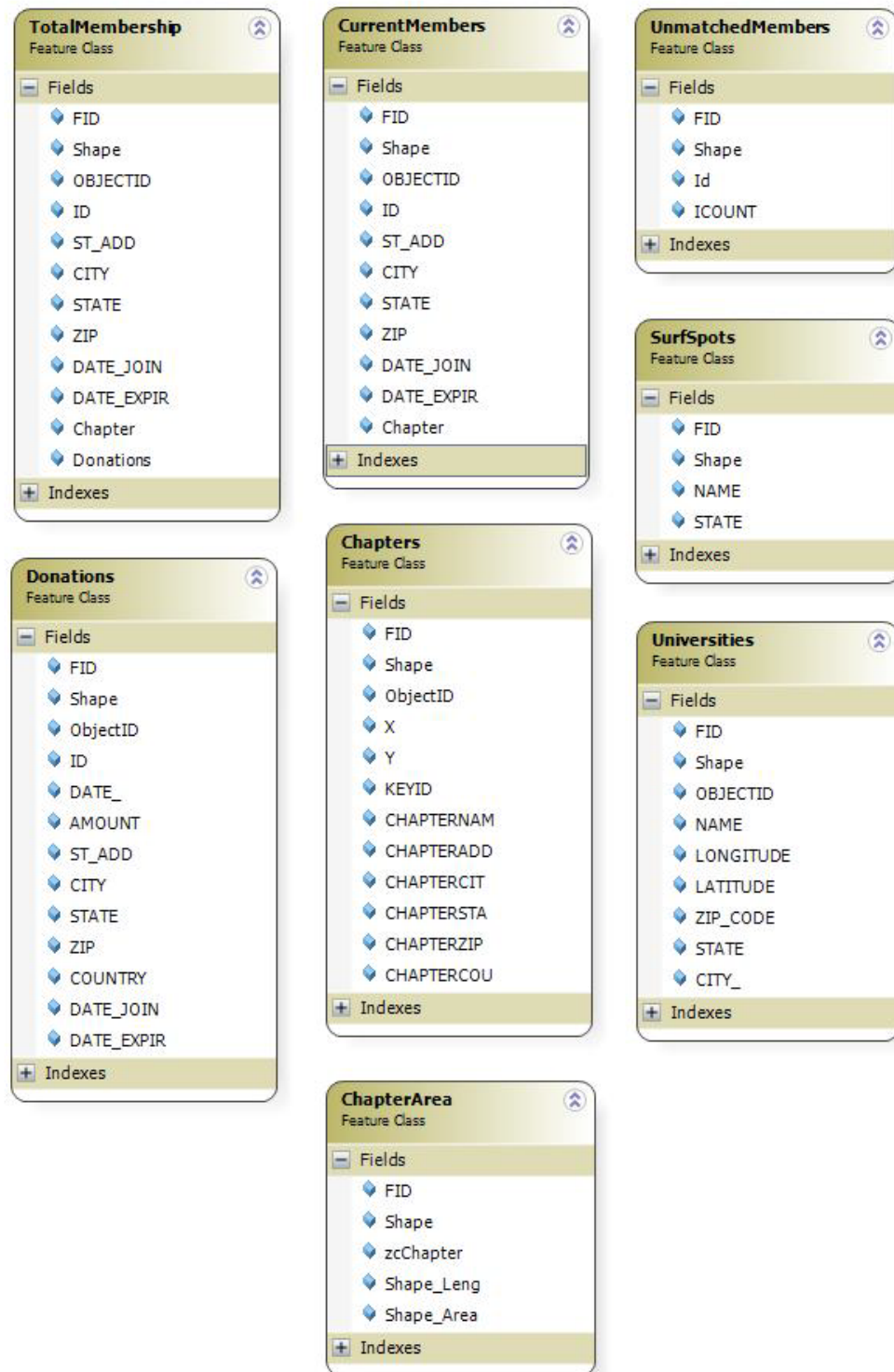


Figure 4-3: Project database layers and tables

An ArcGIS 9.3 file geodatabase was created using ArcCatalog and the data was imported and organized within the database environment. A geodatabase is an effective storage device designed to store, manipulate, and manage geospatial data. Childs (2009, p. 12) presents nine reasons why a file geodatabase should be used when working with a collection of shapefiles in ArcGIS. He separates these nine reasons into three categories, structural, performance, and data management, to distinguish their use and effectiveness (Table 2).

Table 2. Effectiveness of the file geodatabase

Structural	<ol style="list-style-type: none"> 1. Improved versatility and usability 2. Optimized performance 3. Few size limitations
Performance	<ol style="list-style-type: none"> 4. Easy data migration 5. Improved editing model 6. Storing rasters in the geodatabase
Data Management	<ol style="list-style-type: none"> 7. Customizable storage configuration 8. Allows updates to spatial indexes 9. Allows the use of data compression

The file geodatabase is the ideal data structure for the performance and storage of geospatial data and outperforms the operational characteristics, especially those involving attributes and scaling dataset size limits, of individual shapefiles. Many of the datasets received from the client were very large and difficult to manage and scalability was a major factor in choosing to work with this type of geodatabase. These large datasets were easily managed within the database and an increase in performance and usability was prevalent. The file geodatabase data structure is designed to store these large datasets and use only one-third of the feature geometry storage required by shapefiles (Childs, 2009, p. 14).

The data represented in Figure 4-3 were the primary data used in all analysis and the GIS web application created for this project. The *TotalMembership* feature class contains all membership records from the Surfrider Foundation's inception in 1984 until March, 2009. Many of these memberships, however, have since expired so an additional feature class (*CurrentMembers*) was created of current membership only. The *Donations* feature class was created from a tabular join between the *TotalMembership* feature class and the historic donation records received from the client in tabular form. The membership feature class was joined to the donations table using a common member ID field and all donation records were retained since many members had donated more than once. The tabular join preserved all donation records, including data and amount, and added the corresponding address and enrollment date for each member. Once the join was complete, the table was geocoded by member address to create a new donations feature class. The donation amount was also summarized and included as a new field in the *TotalMembership* feature class to identify the total amount of money each member has donated. The *UnmatchedMembers* feature class contains all membership records that were not found using the geocode method. These records could not be located because

they contained a post office box or other type of address that was unrecognizable by the geocoding tool. These records were geocoded by zip code and then summarized using the Collect Events tool in ArcGIS in order to symbolize the layer based on the number of records located within each zip code. The *Chapters* feature class is point data including all the Surfrider Foundation chapter addresses. The *ChapterAreas* feature class was created using an additional table provided by the client containing all the zip codes associated with each local chapter. The chapter zip codes were then joined with a U.S. Census zip code layer and the zip code boundaries were dissolved to form comprehensive chapter regions. The *ChapterVictories* data table contains information related to individual cases around the United States in which the Surfrider Foundation was involved. These cases were associated with chapter locations and membership enrollment to conduct further analysis. The *Universities* and *SurfSpots* feature classes were not datasets received from the client but were collected by other means, which will be briefly discussed in Section 4.3.

Additional datasets created during the analysis phase of this project were also added to the geodatabase. Also included in the database are secondary datasets, including United States and zip code boundaries, road networks, and U.S. city data. All primary datasets were made available to the Surfrider Foundation for visualization and analysis as part of the online web-mapping application created for this project. Before this could happen, however, extensive data editing and testing were conducted at the database level. This was necessary in order to uncover any inconsistencies within the database before the development of the web service which would push the data from the server to the web application.

4.3 Data Sources

The membership, donation, chapter, and chapter victories datasets were acquired from the Surfrider Foundation. The Surfrider Foundation maintains a SQL 2000 database where all membership and donation history is stored. This data was provided in comma-separated value (CSV) file format. Microsoft Access was used to organize, manage, and export the data into dBase (DBF) formatted tables which were then easily ingested into ArcGIS. The chapter and chapter victories datasets were received in Microsoft Excel spreadsheets and were easily ingested and geocoded in ArcGIS. The universities dataset is a data layer which was compiled by the United States Department of Education in 2002. This layer was downloaded from the U.S. Department of Education website in ESRI shapefile format and no further manipulation was required for ArcGIS compatibility. No pre-assembled surf spot data could be located for this project so this layer was compiled using global positioning system (GPS) data of popular U.S. surfing beaches assembled by WannaSurf, an online forum comprised of surfers from around the world. This forum allows surfers to interact with each other and share information and history related to the surfing beaches in their area. This data was downloaded from the WannaSurf website (www.wannasurf.com) as a Microsoft Excel spreadsheet and easily ingested and geocoded in ArcGIS.

Additional datasets utilized in this study include a U.S. street dataset compiled by StreetMap USA using TIGER 2000 and ESRI Tele Atlas data and numerous Address Locator files created in ArcCatalog for geo-coding purposes.

4.4 Summary

There are evident relationships between the datasets compiled for this project. These relationships have governed the attainment and management of this data. The database for this project was designed with these relationships in mind. All necessary elements required to attain the analytic objectives of this project were included in the database. Some data manipulation was needed to achieve compatibility with ArcGIS software; however, no major interruptions were experienced during data preparation.

Chapter 5 – Implementation

Once the geodatabase was designed, the project moved to the next stage – implementation. It was during this phase that a web-based GIS application was designed and developed.

In this chapter, the steps taken to build the online mapping application are explained, including data processing, geodatabase creation, map document assembly, the design and construction of the web application, and the introduction of the application to the World Wide Web.

5.1 Data Manipulation and Processing

Before any steps to build the online mapping application could begin, extensive data manipulation and processing were necessary. All major datasets received from the client were delivered in a format that was not easily managed by ESRI software. The Surfrider Foundation currently utilizes a SQL Server 2000 database to store and manage its membership records. Once the membership dataset was targeted for use in this project, it was exported to a comma-separated value (CSV) file format and uploaded to the Surfrider Foundation's online file transfer protocol (FTP) website where it could be retrieved. The CSV file format is used predominately for the digital storage of tabular data and is not easily ingested by ArcGIS. For this reason, Microsoft Access 2007 was used to import, manipulate, and organize the CSV tables. The data within the tables was separated and field headers were created. Once the tables were organized, they were exported to dBase (DBF) formatted tables which are the native tabular format used by ESRI products. Other datasets received from the client or collected by other means were organized in Microsoft Excel spreadsheets and could be easily ingested in ArcGIS and converted to shapefile format.

5.2 Geodatabase Loading

All relevant datasets, either received from the Surfrider Foundation or collected by other means, were loaded into the designed ArcGIS file geodatabase (discussed in Chapter 4). The file geodatabase was built using ArcCatalog, part of the suite of tools available with ESRI's ArcGIS desktop software. Once created, all relevant shapefiles (feature classes) and tables used in the project were imported to the geodatabase using the Import to Geodatabase Python script (Figure 5-1). With the feature classes successfully imported to the file geodatabase, work could begin on preparing the data for the online mapping application and all analysis necessary for the project.

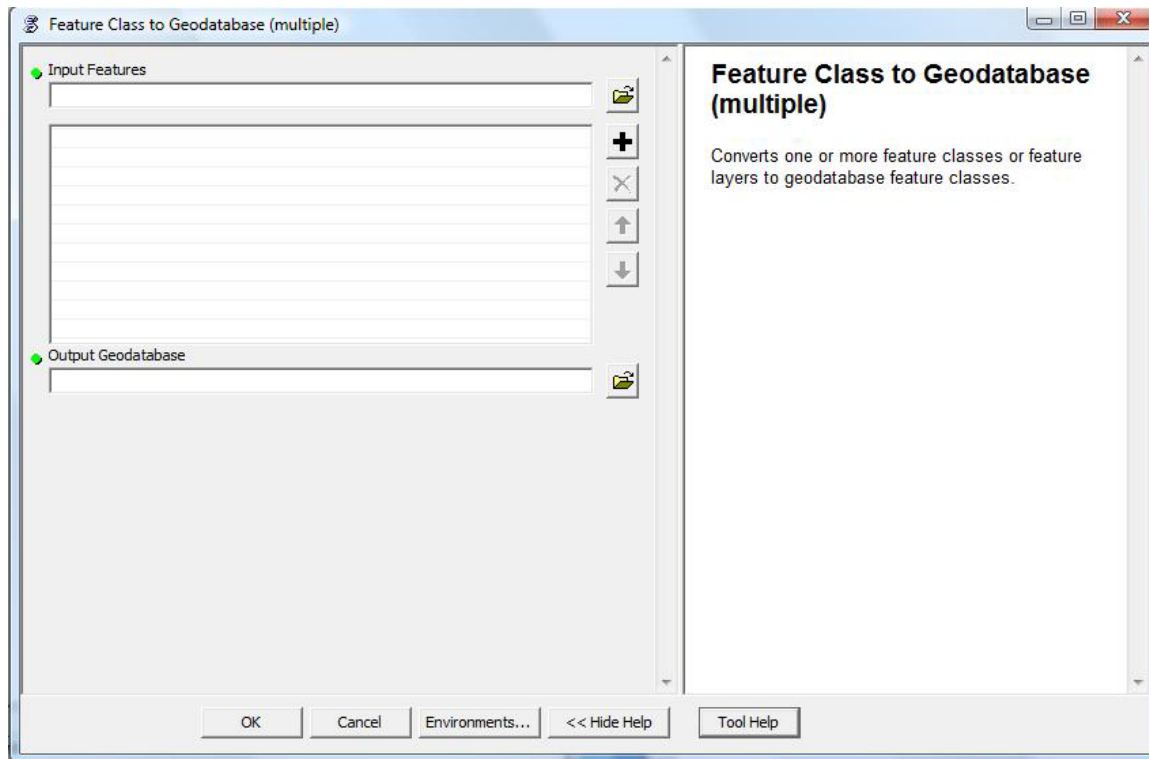


Figure 5-1: Geodatabase file import interface

In order to locate all the membership records and their related attributes on the map, an Address Locator file was created in ArcCatalog and stored in the geodatabase. An Address Locator file is used in the geocoding process to identify address information from a table and locate it on a map using a pre-defined street network. Since each membership record contains street, city, state, and zip code address information, the Address Locator file was created using a US Streets with City, State, and ZIP road network provided by ESRI (Figure 5-2).

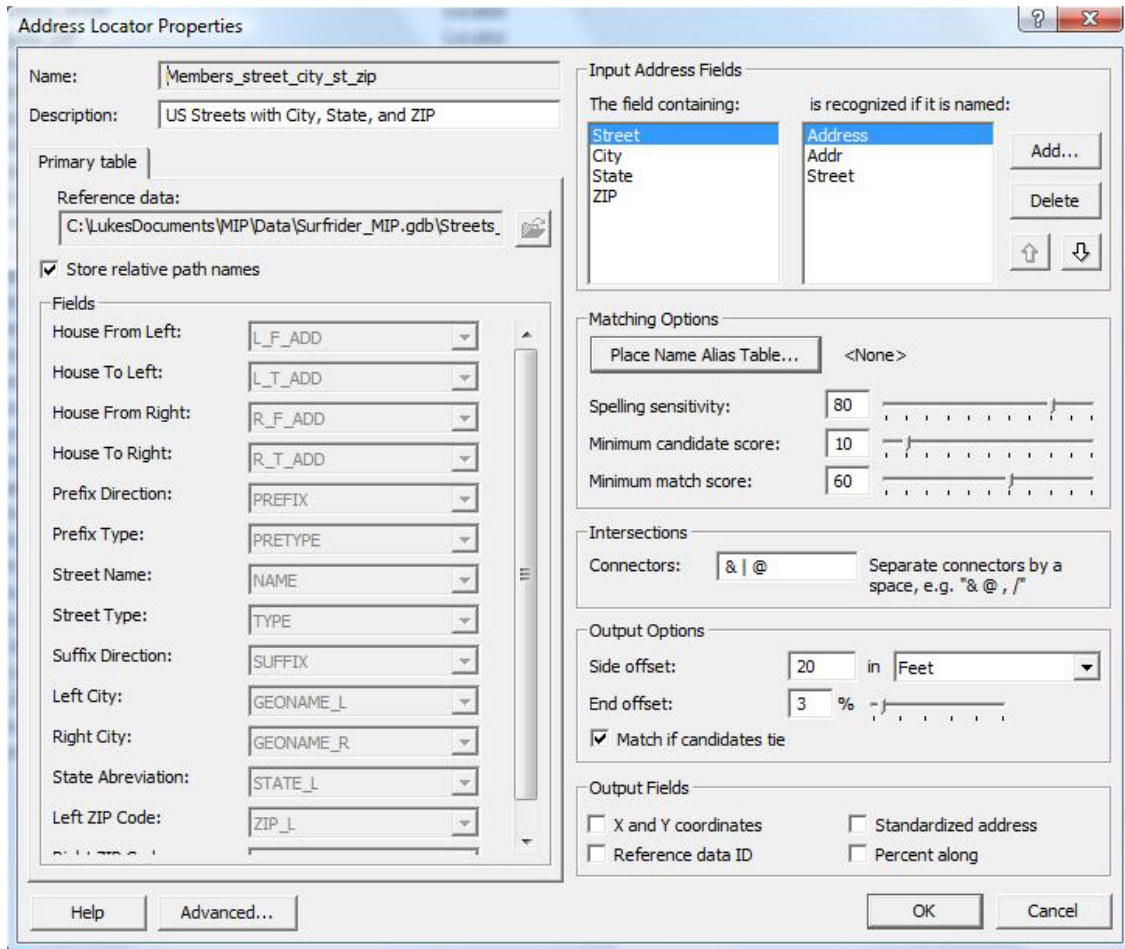


Figure 5-2: Address Locator properties

The geocoding process was able to locate 85% of the membership records from the last 25 years. The Address Locator was unable to process the remaining 15% of the records because they had been recorded as post office boxes in the Surfrider Foundation's database or were incorrect or misspelled addresses that could not be referenced to the US Streets with City, State, and ZIP road network. As an alternative to using a street address to geocode these remaining records, an additional Address Locator file was created using a US Zip Codes point file from the United States Census Department. This zip code file was based on approximate post office locations and zip codes represented as points instead of areas such as post office box zip codes and single site, building, or organization zip codes. This Address Locator was able to locate the unmatchable records by their corresponding zip code. The result of this method created a stacking effect at any zip code point where multiple records shared the same zip code. In order to represent the number of members at each zip code point, the Collect Events tool was used in ArcGIS to summarize the number of members at each location. The data was divided into five groups and each point was then symbolized by size on the map based on the number of members at each zip code location (Figure 5-3). This symbolized layer was included in the web application's table of contents to allow the user to identify the

number of members which were located based on their zip code and not street address. This was done to insure that all membership records were accounted for in the application, including records with a less accurate location based on zip code. The records located by zip code were also included with the membership records previously geocoded by street address. This total membership dataset provided the client with an accurate number of members when a search was conducted on the entire membership dataset within a specific geographic area in the web application.

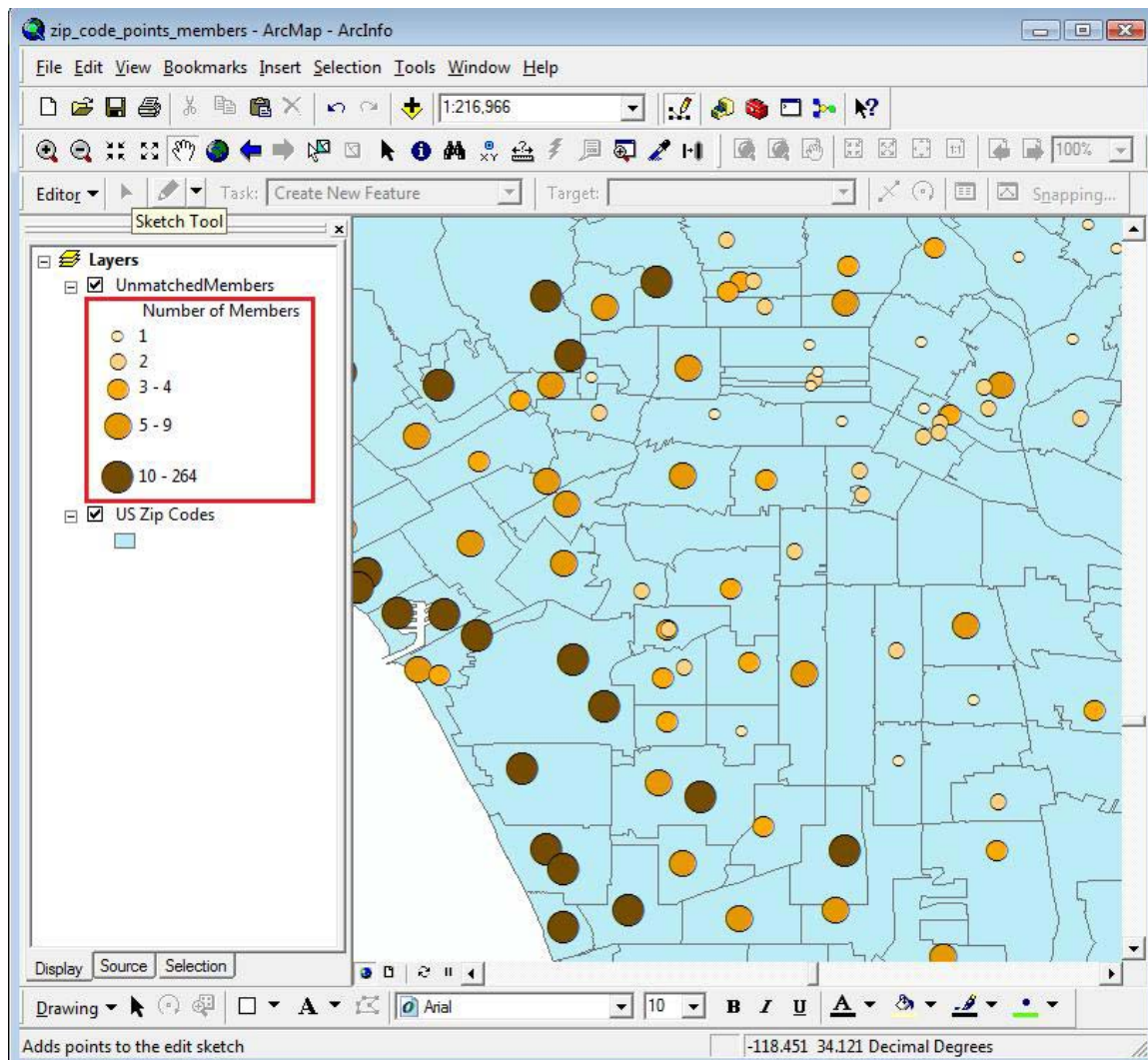


Figure 5-3: Membership records geocoded by zip code and symbolized by number of members at each point

The membership dataset included all member records from the last 25 years, including many which had expired. This comprehensive dataset was important because it displayed the overall growth of the organization and how members have been distributed across the United States over time. However, because some of these memberships had expired, it was also important to have a feature class containing only current member records. This was done by producing a query which selected membership records still

active as of January 1, 2009, and exporting them to a new feature class. These two datasets could then be compared to identify areas of membership decline. Analysis could also be conducted using current membership records to promote donation to the organization.

The membership feature class was joined to the donations table using a common member ID field. Because of the many-to-one relationship between donations and members, the Keep All Records join option was selected (Figure 5-4).

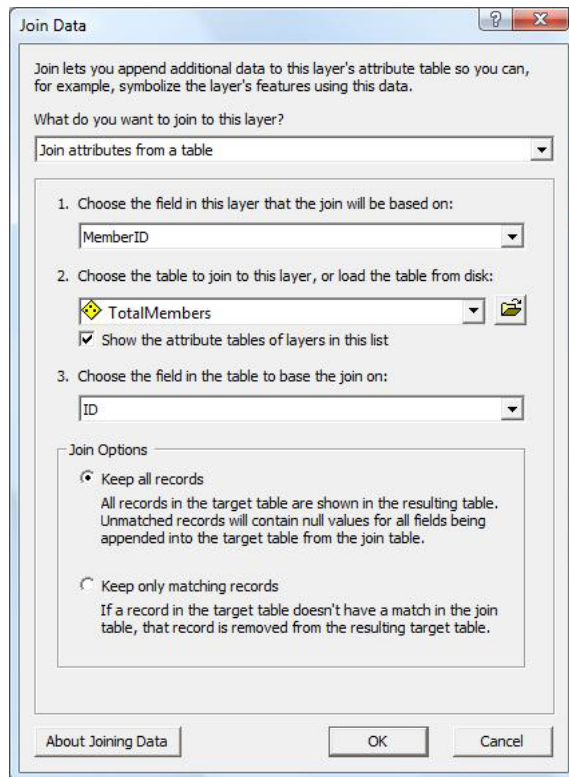


Figure 5-4: Join Data tool

This was done in order to retain all donation records within the table since many members had donated more than once. This preserved all the donation records, including date and amount, and joined the corresponding member information to each donation record (Figure 5-5).

MEMBER ID	DATE	AMOUNT	STREET	CITY	STATE	ZIP	EXPIRATION DATE	JOIN DATE
890	2001-05-15	100	1451 PUEBLO CT	CLAREMONT	CA	91711	2002-05-15 00:00:00	1988-01-01 00:00:00.000
890	2001-12-26	100	1451 PUEBLO CT	CLAREMONT	CA	91711	2002-05-15 00:00:00	1988-01-01 00:00:00.000
893	1994-03-02	15	25711 CUMBERLAND WAY	BLACK DIAMOND	WA	98010	2000-11-22 00:00:00	1988-01-01 00:00:00.000
893	1999-11-22	25	25711 CUMBERLAND WAY	BLACK DIAMOND	WA	98010	2000-11-22 00:00:00	1988-01-01 00:00:00.000
893	2000-07-21	10	25711 CUMBERLAND WAY	BLACK DIAMOND	WA	98010	2000-11-22 00:00:00	1988-01-01 00:00:00.000
898	1994-04-18	40	354 CLAREMONT AVE	LONG BEACH	CA	90803	2009-04-01 00:00:00	1988-01-01 00:00:00.000
898	1995-06-01	40	354 CLAREMONT AVE	LONG BEACH	CA	90803	2009-04-01 00:00:00	1988-01-01 00:00:00.000
898	1996-08-01	25	354 CLAREMONT AVE	LONG BEACH	CA	90803	2009-04-01 00:00:00	1988-01-01 00:00:00.000
898	1997-12-29	25	354 CLAREMONT AVE	LONG BEACH	CA	90803	2009-04-01 00:00:00	1988-01-01 00:00:00.000
898	1999-04-22	40	354 CLAREMONT AVE	LONG BEACH	CA	90803	2009-04-01 00:00:00	1988-01-01 00:00:00.000
898	2008-04-01	25	354 CLAREMONT AVE	LONG BEACH	CA	90803	2009-04-01 00:00:00	1988-01-01 00:00:00.000

Figure 5-5: Donation and membership join

Once the join was complete, the table was geocoded to create a new donations feature class. This layer contained the amount of all individual donations received by the Surfrider Foundation from the last 25 years and the identification number of the associated member. Through the tabular join, the donation information was paired with the membership information in that membership feature class, including a dollar amount field identifying the monetary value of each member's donations, the member's address, and the dates of membership commencement and expiration. The dollar amount field in the donation table was then summarized in order to identify the overall donation amount for each member. The new summary table was joined to the total membership layer using the common member ID field to allow additional analysis to be done once the layer was available in the online mapping application.

5.3 Map Document

Having prepared all relevant data layers for use in ArcGIS, the next step included the creation of a map document to be uploaded to the MS GIS server. Once uploaded to the server, the layers represented in the map document, along with their associated attributes and symbology, could then be carried to the GIS web application for analysis and display.

ArcGIS 9.3 desktop software was chosen to prepare the map document because this is GIS software that the client has. Layer symbology was determined and visual hierarchy was established using the software, all of them being carried to the online application (Figure 5-6). Each layer had an associated attributes table containing various fields describing the data. This table could be edited in the software and the attributes accessed by the web service and displayed using query or identification tools built into the online application.

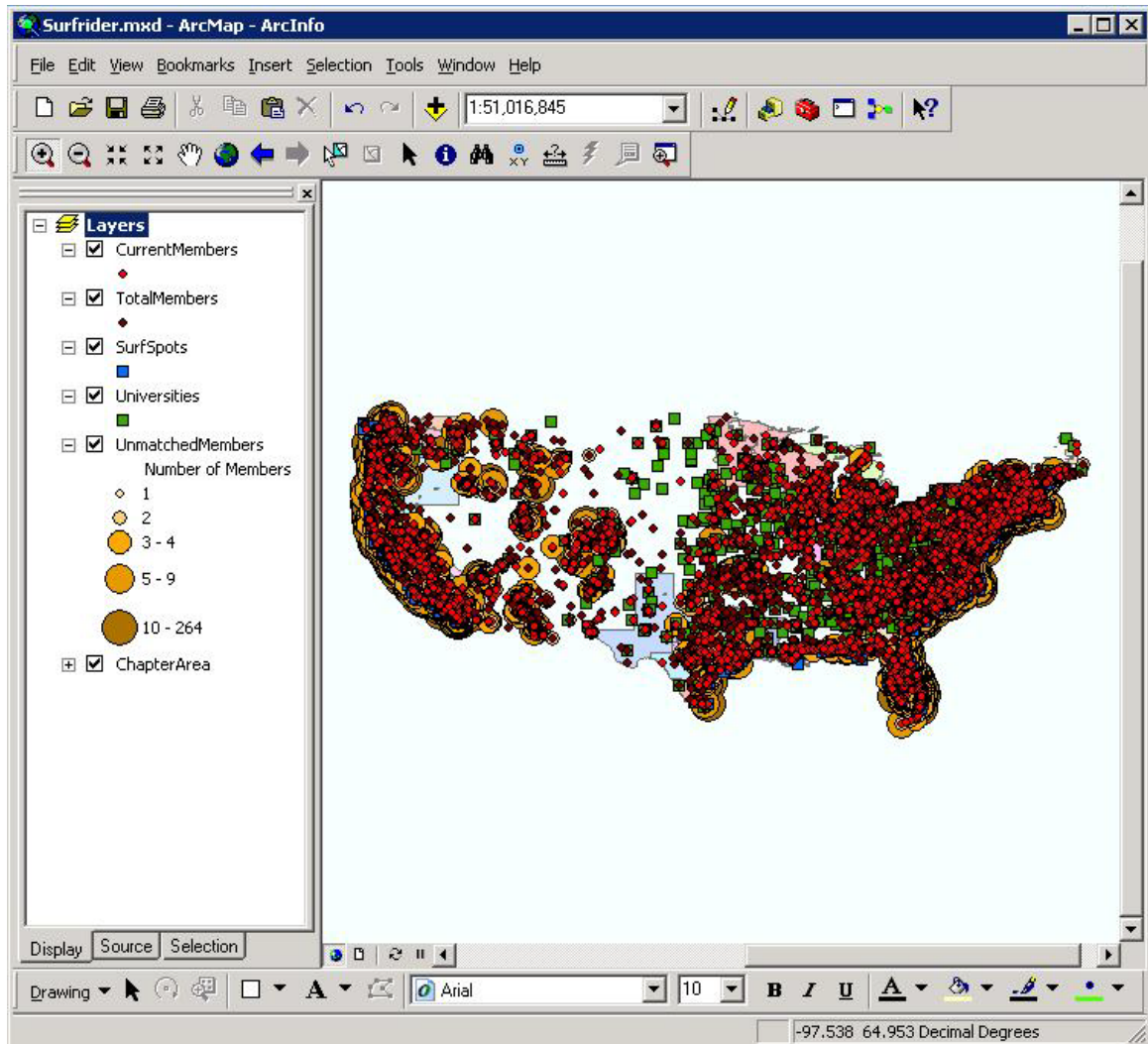


Figure 5-6: Map document layers

The data layers incorporated into the map document included the current and all membership datasets, Surfrider Foundation chapter locations and areas, universities, and surf spots. These were the primary layers referenced and displayed in the online mapping application. No base map information was included in the map document. The base map layers were displayed within the web application that has direct online access to the ESRI StreetMap World 2D MapServer. The ESRI StreetMap World 2D MapServer includes highway and street-level data, railways, water features, administrative boundaries, cities, parks, landmarks, and a shaded relief imagery layer for added content (Figure 5-7). When accessing the base map from the ESRI MapServer in the web application, a slider bar is included to allow the user to choose the extent of the map. Additional navigation features such as zoom in, zoom out, and pan are also provided. The datum used for the ESRI World 2D MapServer is World Geodetic System 1984 (WGS-84) and cannot be adjusted within the application.

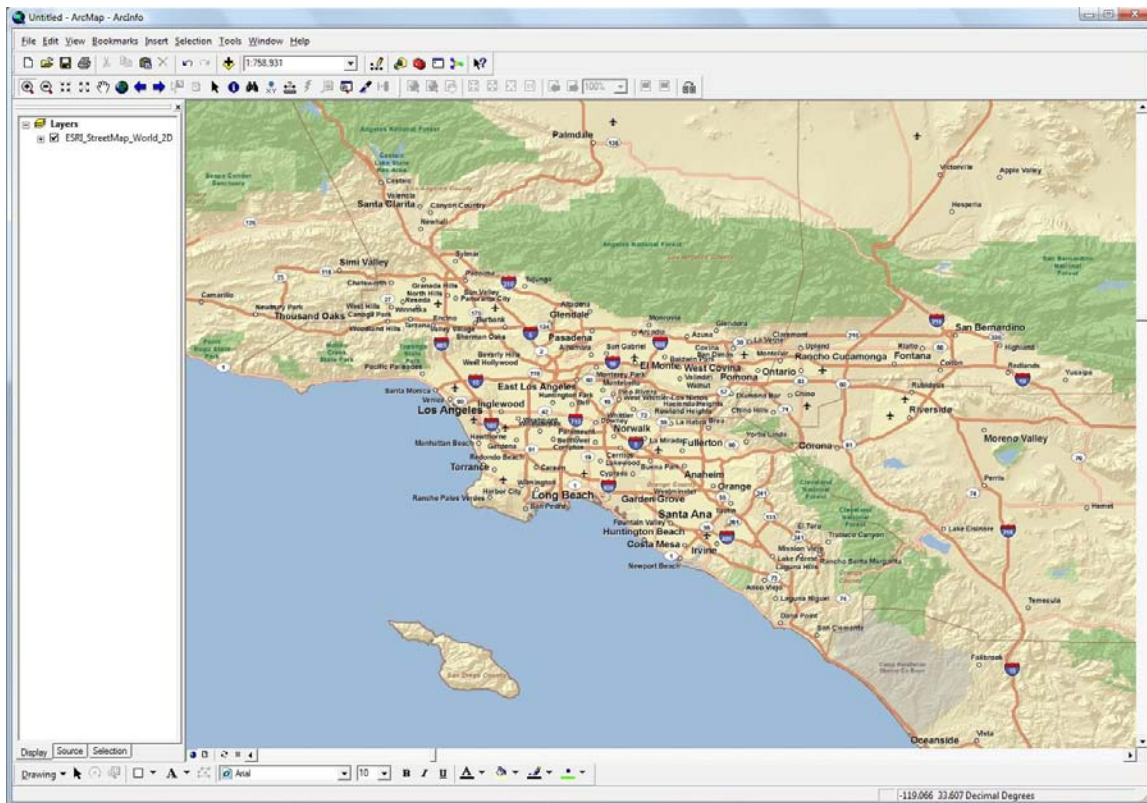


Figure 5-7: ESRI StreetMap World 2D Basemap

Once the map document was completed, the document and all its associated layers were copied to a virtual server created for the project. It was on this virtual server that the web application, the map document, and all associated data layers were stored and served to the web. ArcCatalog was used to publish the map document to ArcGIS Server, which also created an online Services Directory for the project on the virtual machine. The Services Directory can be used to browse and obtain information about the layers published to the ArcGIS Server (Figure 5-8). The user can view service level metadata and preview the overall appearance of the project in ArcMap, in a web browser, or in GoogleEarth. The Services Directory also provides information to help the user develop web applications.

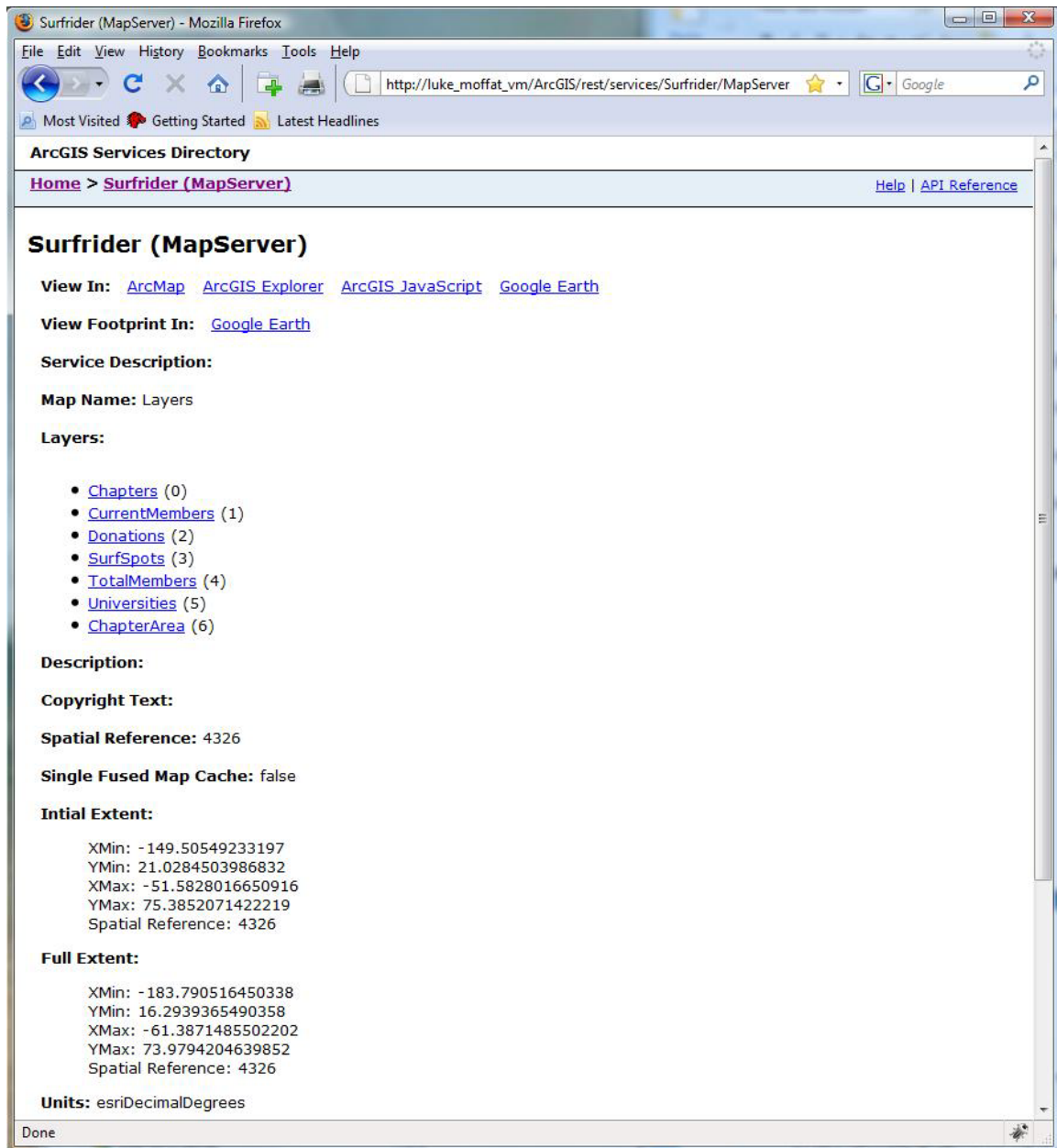


Figure 5-8: ArcGIS Services Directory for the Surfrider MapServer

When a web application is developed using a JavaScript or Adobe Flex programming interface, a Uniform Resource Locator (URL) where all services, layers, and functionality are stored is required to access the content within the application. The Services Directory provides an interactive way to construct these URLs. The Services Directory works using REST (Representational State Transfer), an architectural style used by ArcGIS Server to acquire information about itself via URLs. Using the Services Directory, the user can preview available data and discover information about the server which is retrieved through REST using a URL. Up to this point, the Services Directory was established for this project, and the web mapping application was ready to be built on the web.

5.4 Building the Web GIS Application

The web-based GIS application was designed and built using Adobe Flex 3.0 software. Flex is a software development toolset used to develop and deploy rich Internet applications based on the Adobe Flash platform. These Internet applications can be built on top of the ArcGIS Server framework, which indicates that Flex can be used to develop a web application which can display an interactive map with predefined layers, execute GIS-related tools, search for features and attributes in the data, locate addresses, and display results. These components can also be customized for various web applications. Due to these advantages, Flex was chosen as a suitable tool to assemble the necessary programming code to build and deploy the web-based mapping application for the project (Figure 5-9).

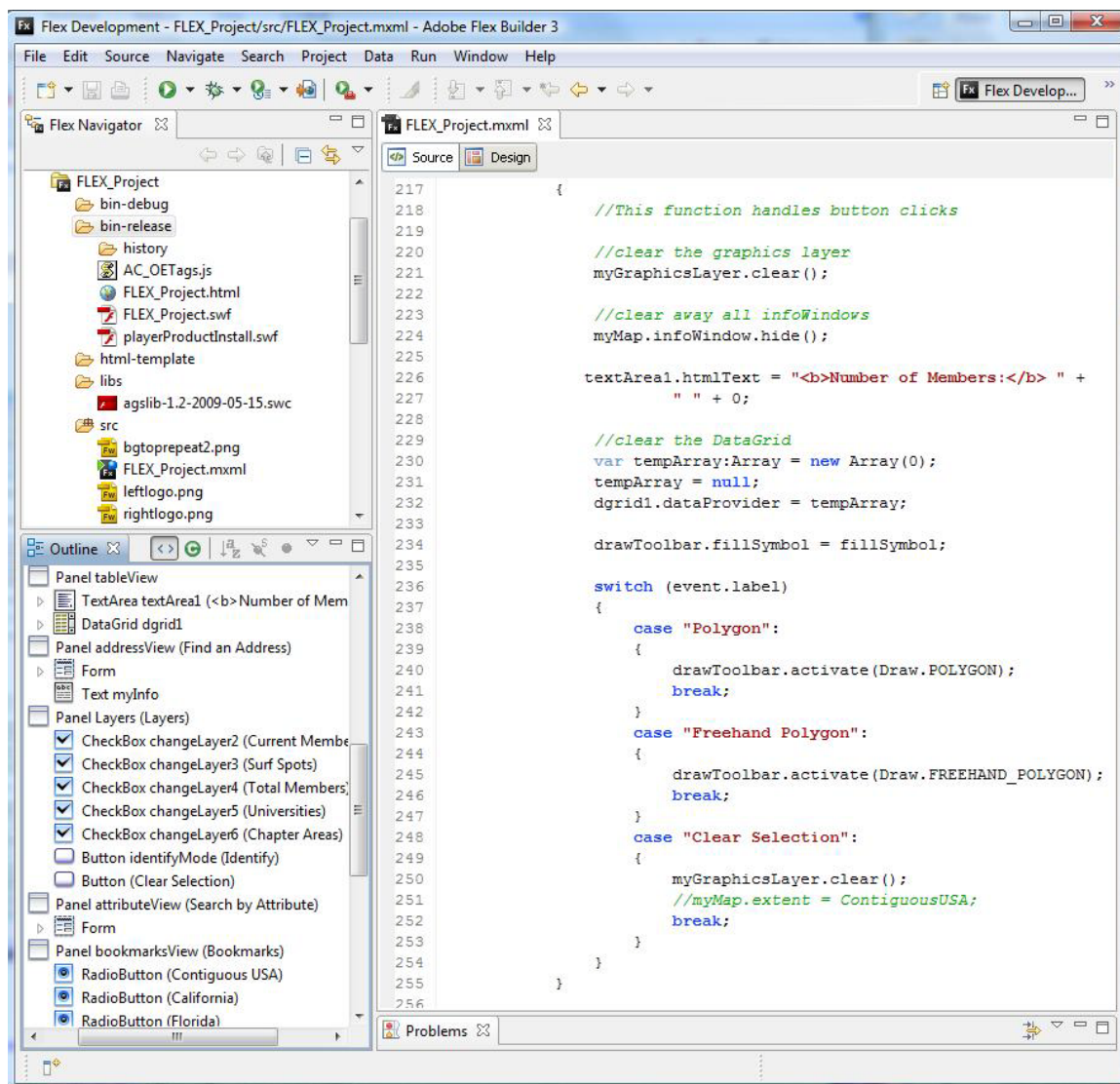


Figure 5-9: Adobe Flex Builder interface

In order to provide the Surfrider Foundation with a web mapping application that allows the user to visualize, search, and analyze membership data, a set of custom GIS-related tasks were designed and implemented using Flex for the application. The code for many of these tasks had been built by ESRI and was available for reference on the ArcGIS API for Flex Resource Center webpage. ESRI's Flex Resource site has 76 different GIS tasks that have been built and deployed using Flex. The site allows the developer to view a sample of the task functionality (Figure 5-10). If the developer wants to include this task in another project, the source code is also made available and can be further customized if needed (Figure 5-11).

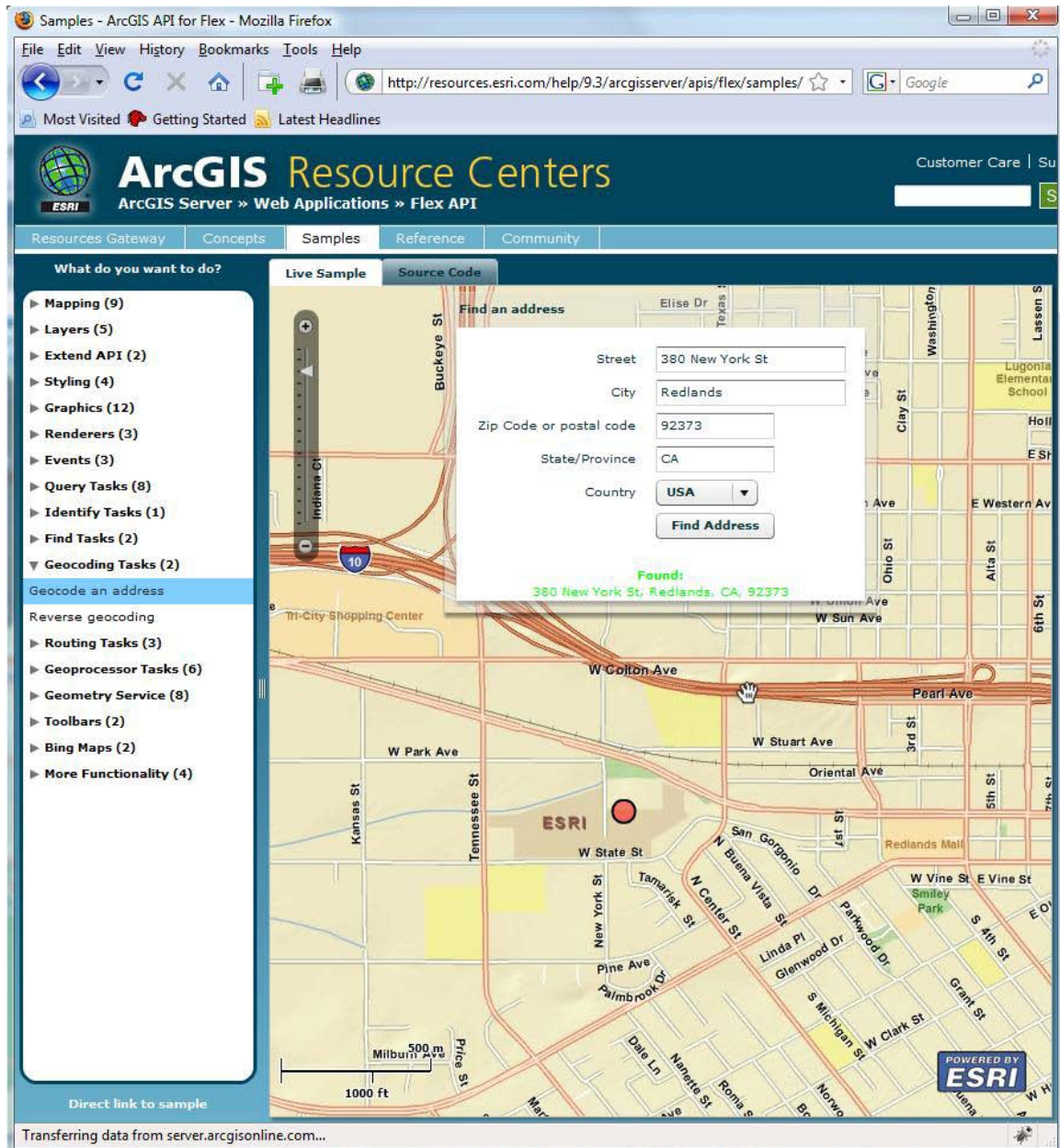


Figure 5-10: GIS Task from the ArcGIS API Flex Resource Center

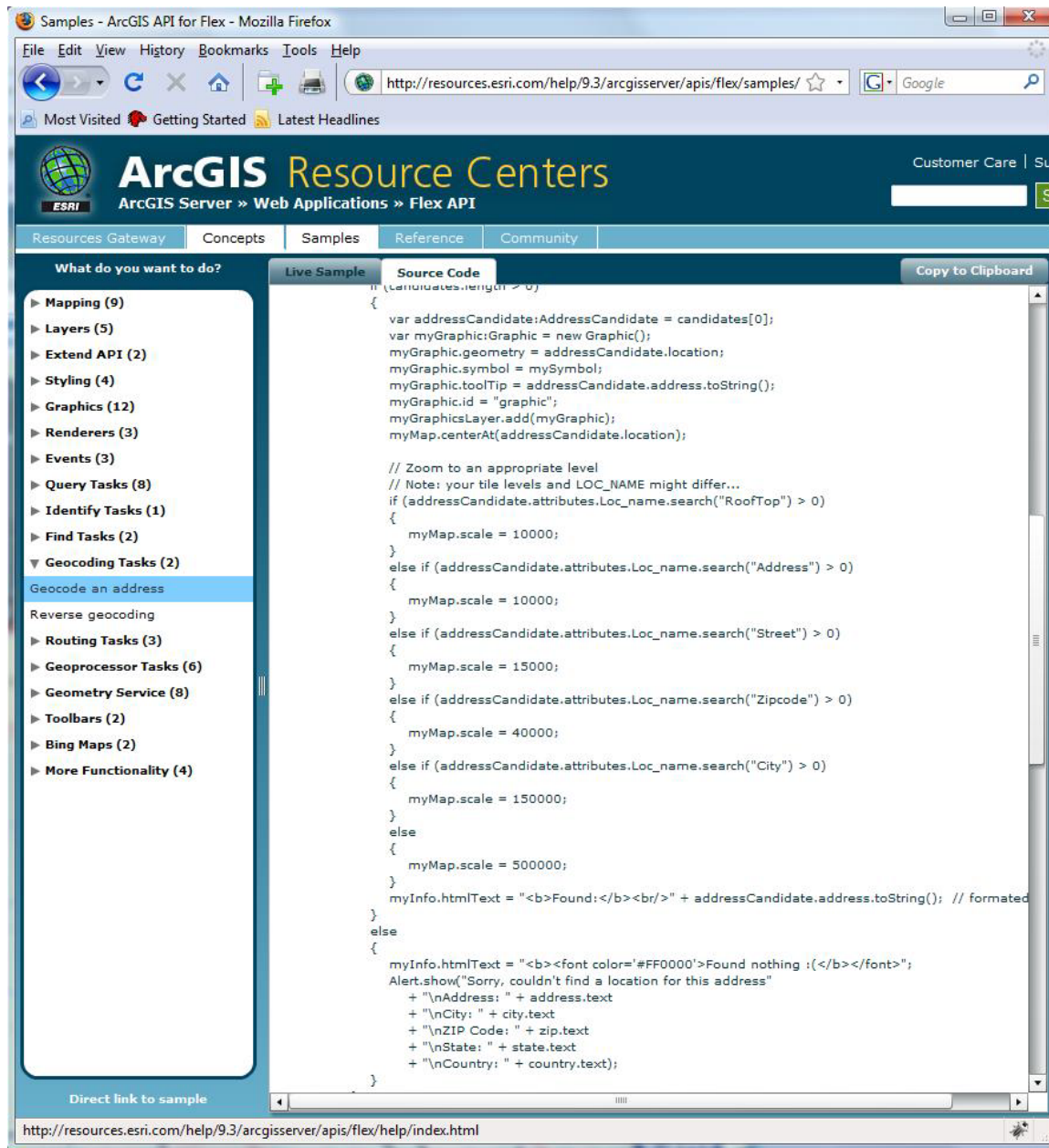
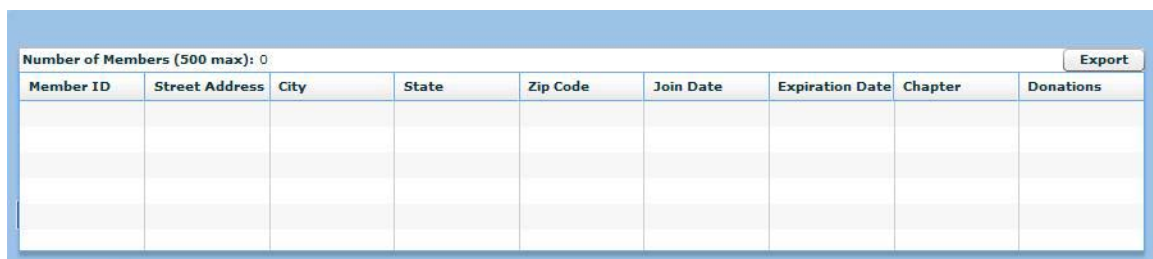


Figure 5-11: Source Code from the ArcGIS API Flex Resource Center

Once a list of GIS tasks was determined for the application, the source code for many of those tasks was referenced from the ArcGIS Resource Center. The tasks to be included in the Surfrider Foundation's web-mapping application, however, differed in appearance and functionality from those found in the ESRI Flex sample website. Thus, the source codes were further customized in Flex Builder to reflect those differences. Four GIS tasks were incorporated into the application to allow the user to visualize and explore the Surfrider Foundation membership data: Search by Selection, Search by Attributes, Find an Address, and a table of contents which includes an identify tool. The programming code used to build these tools is included in Appendix A.

The search tools query the total membership dataset and provide results as point symbols on the map and the corresponding attributes are also provided in a data table incorporated within the application. The data table displays the results of the various search tools when the results are called from the server (Figure 5-12). This table also allows the user to scroll and highlight individual records for display on the map. There is also a records counter associated with the table that provides the user with the number of membership records found as a result of the search. A 500 record limit was set for the number of results the search tools can identify. When a search task is performed, the first 500 results are returned and displayed on the map and in the table. If there are additional records located in the search area, they are not returned as part of the query result. This limitation was set to prevent the server from being overloaded by a search task and maintains a quick and efficient return on results.



Number of Members (500 max): 0								Export
Member ID	Street Address	City	State	Zip Code	Join Date	Expiration Date	Chapter	Donations

Figure 5-12: Query results table

Once the query results are displayed in the data table, the user can then export the table to a CSV file which can be opened in Microsoft Excel. This allows further analysis to be performed in Excel with specific member records identified using the online mapping application. The records can also be printed once they are exported to a CSV file. When the user selects the export function in the table, the mapping application references an ASP.NET application that was assembled in Microsoft Visual Studio and uploaded to the server. ASP.NET is a web application framework developed by Microsoft that allows programmers to build dynamic web applications and services. This ASP.NET application converts the records in the data table to a CSV file and allows the user to either view or save the file in Microsoft Excel (Figure 5-13). The programming code used to build the table export function is available in Appendix B.

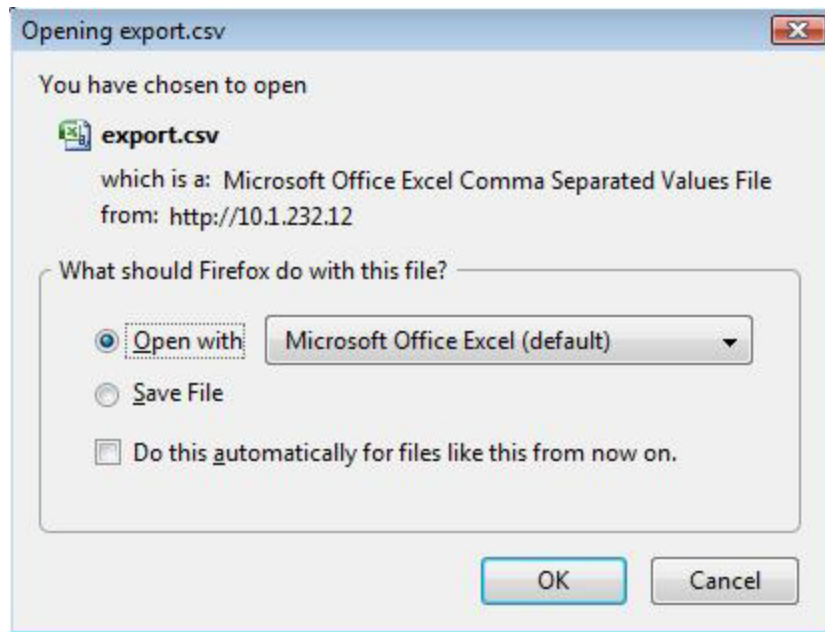


Figure 5-13: Export search results from the interface data table to a CSV file

The Search by Selection task allows the user to either search the membership dataset by drawing a polygon or by using a pre-defined buffer (Figure 5-14).

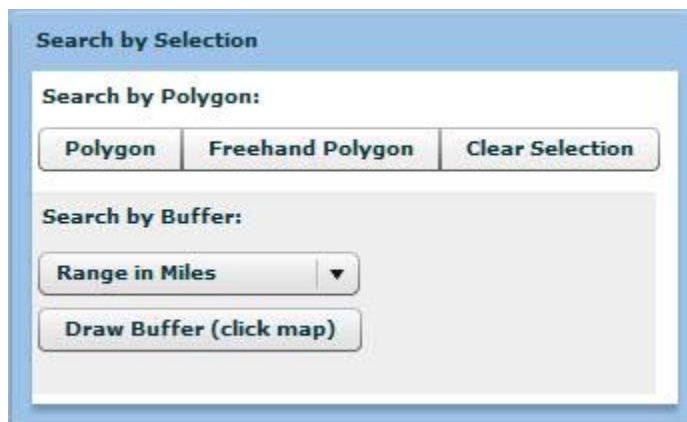


Figure 5-14: Search by Selection tool

Searching the dataset using the polygon feature allows the user to draw a polygon around an area of interest on the map; all the members which reside within that area are then returned as points on the map and as records in the data table which is included as part of the user interface. The freehand polygon feature allows the user to draw a polygon in any desired shape and not be regulated to drawing a straight-line polygon.

By using the search by buffer tool, the user selects a pre-defined buffer range (1, 10, 20, 50, or 100 miles) from a drop down list, clicks the draw buffer button, and selects the

center point location on the map (Figure 5-15). All the members which reside within the drawn buffer are returned to the map as points and as records in the data table.

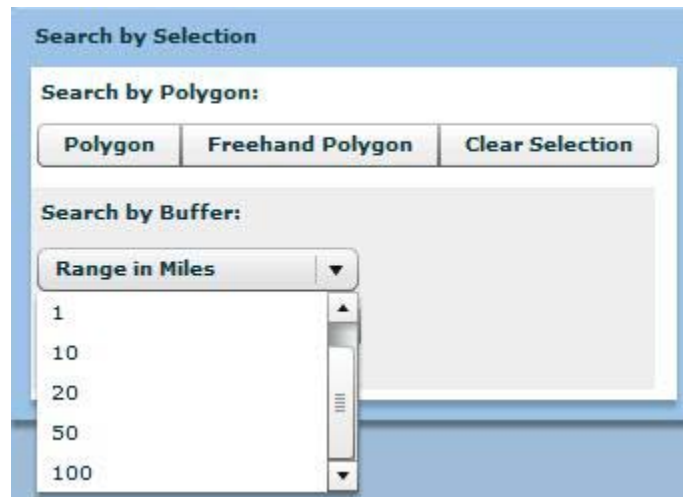


Figure 5-15: Search by Selection buffer tool

The Search by Attributes task allows the user to query the membership dataset using a selected attribute in the membership table located on the server (Figure 5-16). If results are found for the query, the member locations are displayed on the map and the records are displayed in the data table. By using the Search by Attributes task, the user can locate an individual member by querying a specific member identification number or by querying a specific zip code.



Figure 5-16: Search by Attribute tool

An address locator has been included in the toolbar to allow the user to locate a specific member or an event based on a given address (Figure 5-17). The tool interface includes pre-assigned street address, city, state, and zip code values as examples which can be replaced by the user to search an address of interest. Once an address is located, an address match will appear in the tool panel to inform the user that the address of

interest has been located. If the tool was unable to locate the specified address, the results window will indicate that the nearest address match was selected or no address was found at all. This tool utilizes a geocoding service located on the ESRI Online Server to locate addresses within the United States.

Find an Address

Street: 1111 E. Central Ave

City: Redlands

State: CA

Zip Code: 92374

Find Address

Clear Address

Found:
1111 E Central Ave, Redlands, CA, 92374

Figure 5-17: Find an Address tool

A table of contents is also available for the user to turn layers on and off within the application (Figure 5-18). Integrated into the table of contents is an identify tool which allows the user to select a layer feature and receive attribute information about that feature. This information is contained in the layer table and accessed from the server when the feature is selected in the application.

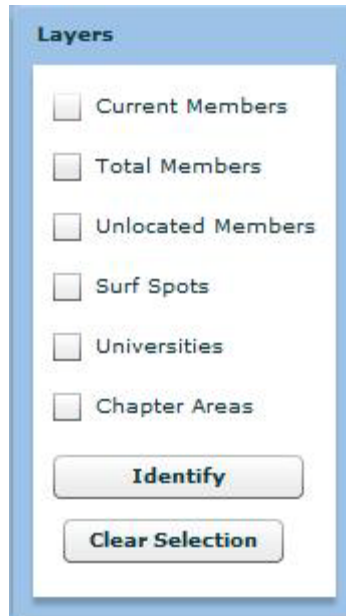


Figure 5-18: Table of Contents

There are six preset bookmarks to aid in map navigation (Figure 5-19). Bookmarks have been created for the contiguous United States, as well as Alaska, Hawaii, California, Florida, and the Northeast United States. Additional instructions to help the user navigate the map within the application are also included.



Figure 5-19: Bookmarks

After the code for the GIS tasks was assembled, a layout for the application's graphical user interface was built using the Flex Builder Design feature (Figure 5-20). The layout design included the color, position of toolbar, and additional content necessary to create a visually appealing, yet highly functional, web application. The

importance of each map element was considered and sized accordingly in the map. The position of each element in relation to other elements was also considered and an overall balance to the layout was sought. In order to allow more space for the map itself, the various GIS tasks were collapsed and hidden in an expandable toolbar at the top of the map layout.

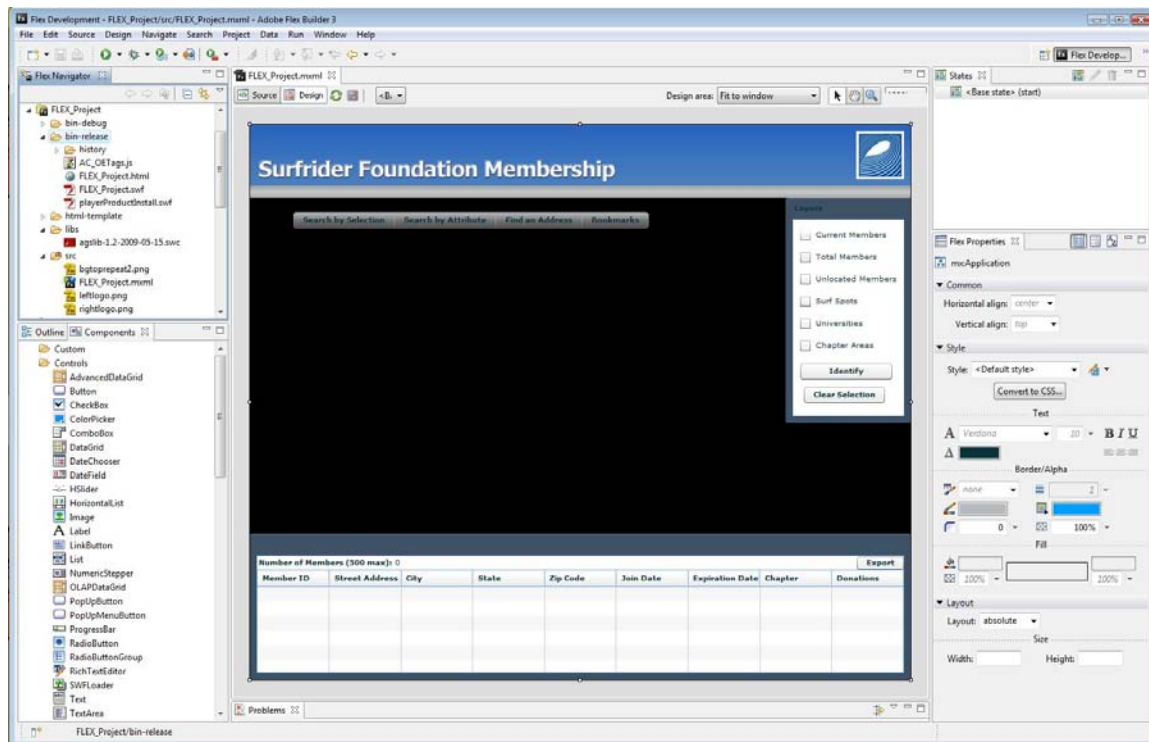


Figure 5-20: Adobe Flex Builder Design interface

5.5 Introduction to the World Wide Web

With the completion of the project in Flex, the necessary steps were taken to export the code and move it to the virtual server where it could be made available to the client on the World Wide Web. The Export Release Build function was used in Flex Builder to compile the code and export it from debug mode to a deployable Shockwave File (SWF). The Export Release Build function is shown in Figure 5-21. The SWF and its associated files were then loaded on the virtual server set up for the project. From this server, the application could be accessed via the World Wide Web using a standard Internet browser such as Internet Explorer or Mozilla Firefox.

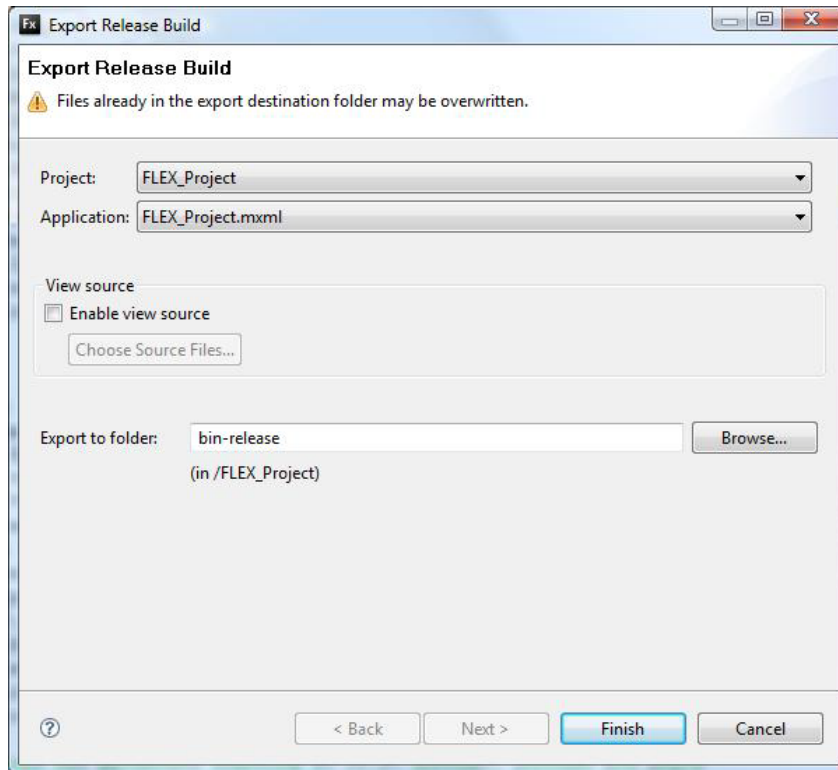


Figure 5-21: Adobe Flex Builder Export Release Build

Working with membership address information and making that information available on the Internet raised privacy concerns for the application. It was decided to install a username and password login feature to gain access to the web application interface. This was done using the Internet Information Services Manager on the virtual server. A username was chosen and a password selected, without which the user is unable to gain access to the application (Figure 5-22).

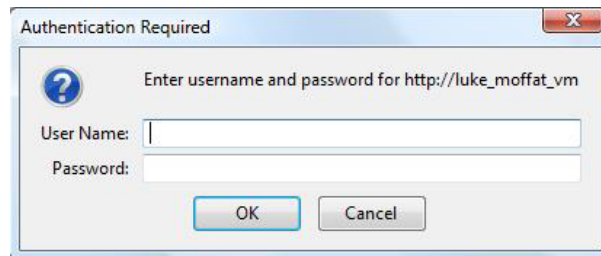


Figure 5-22: Application login request

The application is housed on the MS GIS server and is available to the client for a twelve month period. During this twelve month period, the Surfrider Foundation will have free access to the application and its associated base map and layers. It will be during this time that the usefulness of the application will be determined by the client and its future evaluated. After the courtesy period expires, it will be up to the Surfrider

Foundation to provide the necessary hardware and software required to host the application on its own server.

Chapter 6 – Results and Analysis

The online mapping application created for the project provides the Surfrider Foundation with an effective tool to visualize and explore current and historic membership data. This chapter presents the benefits of using a web-based GIS application to understand and analyze membership growth and distribution. Examples of GIS tasks that may be performed by the Surfrider Foundation are included in this chapter to illustrate the usefulness of the application.

6.1 Web GIS Interface

The Surfrider Foundation has 25 years worth of historic member data in an SQL Server database, a relational model database server which stores data in a collection of tables with typed columns. The data could be queried and exported from the server but no additional tools were available to visualize the data in a geospatial environment and gain additional insights into member distribution and patterns. The web GIS application developed for this project addresses this issue and allows the Surfrider Foundation to visualize and explore membership dispersal and perform valuable analysis. Through a user-friendly interface, the entire membership database can be displayed and analyzed using simple GIS tasks which lead to a greater understanding of the data (Figure 6-1).



Figure 6-1: Web application user interface

The main window of the interface includes a map display of the United States made available from the ArcGIS Online Server and accessed directly from the application. If the user wishes to analyze membership data in California, a fixed bookmark is available under the Bookmarks tab on the toolbar (Figure 6-2). Additional bookmarks are made available for states with areas containing high membership numbers. Remote areas, including Alaska and Hawaii, are also included to help the user navigate to those regions quickly. A slider bar is provided in the upper left corner of the interface to allow the user to zoom in and out on the map using preset map scales included in the ESRI StreetMap World 2D MapServer base map.

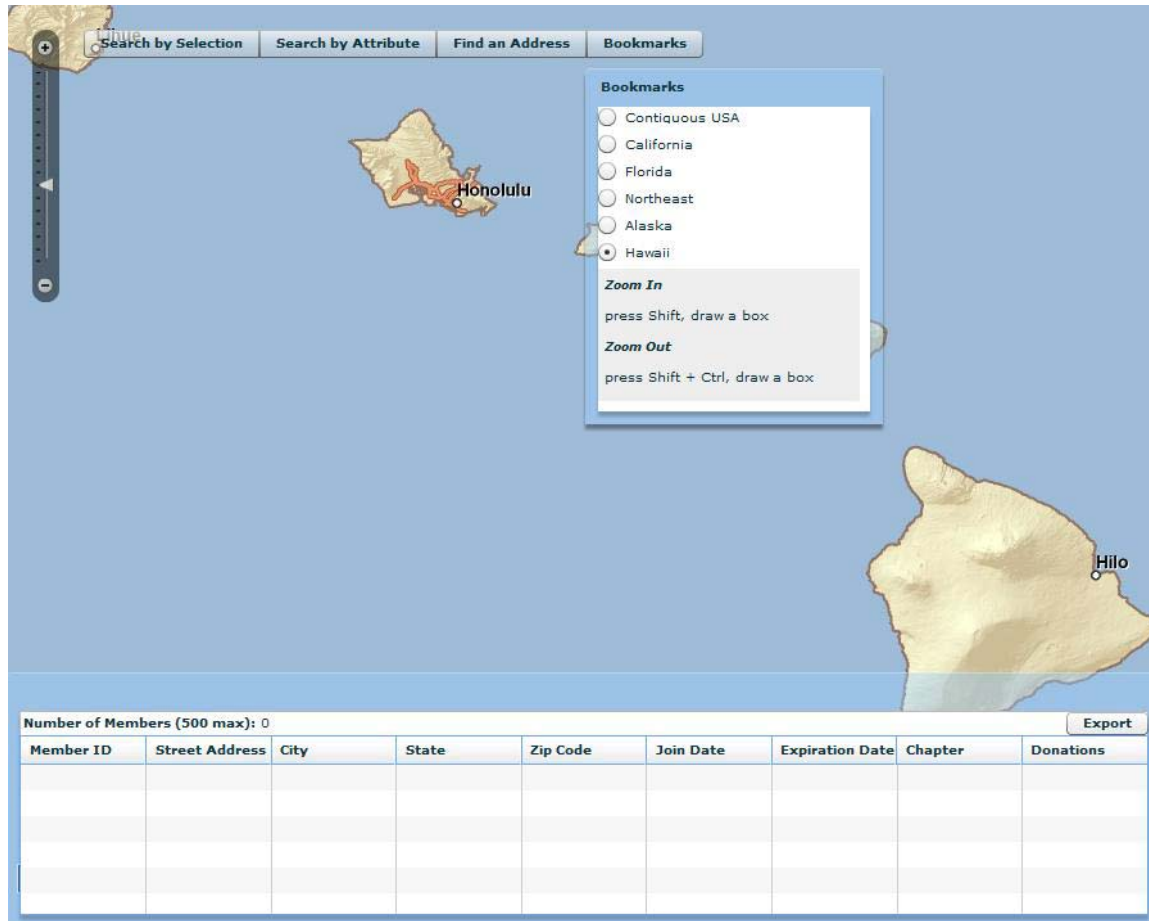


Figure 6-2: Predetermined bookmarks

Once the user navigates to an area of interest using a fixed bookmark, the tool is collapsed. Another tool can then be opened. This design feature was implemented to avoid cluttering the map space, allowing only one tool panel to be visible at any given time.

6.2 The Table of Contents

Unlike the tasks integrated in the toolbar, the layers panel and data table are anchored in opposite corners of the interface and cannot be collapsed. Using the table of contents, the user can activate multiple data layers in the map, overlaying them to reveal potential relationships in the data. The table of contents contains two membership layers. The first is a layer of individuals whose Surfrider Foundation memberships are still current. The other is a layer of historic membership from the last 25 years and contains records which are both current and expired. By overlaying these two layers on the map, the user can identify which individuals have let their membership expire. This is a quick and easy way to identify these individuals without having to sort and compare the membership dates in the database tables. Once identified, efforts can be made to contact these individuals and reactivate their memberships.

When a data layer has been activated, an identify tool is available to display the attributes associated with a given feature. Once the Identify button is selected, the user can then select a layer feature on the map. The feature will then be highlighted and an identify box will be displayed. This identify box has two fields, one for the key, or field name, and one for the value, or attribute. This information is contained in the layer's data table which is accessed from the server when the feature is selected in the application. The identify feature allows greater understanding of layer content and enhances data exploration.

Though Surfrider Foundation membership is dispersed throughout the country, membership clusters can be identified by visualizing the data on the map. Once those areas are identified, further data exploration can be conducted by activating additional layers and determining if the proximity of one feature to another is a catalyst for this clustering. It has been found that Surfrider Foundation members, the majority of whom are surfers or beach-goers, tend to gather near the coast. Figure 6-3 illustrates how dense clusters tend to gather in areas where there are multiple surfing spots or popular beaches. Using the Identify tool, the user can also find what each of these surfing spots is. The opposite may hold true for areas of the coast where membership levels may be low due to a scarcity of quality waves or beaches.

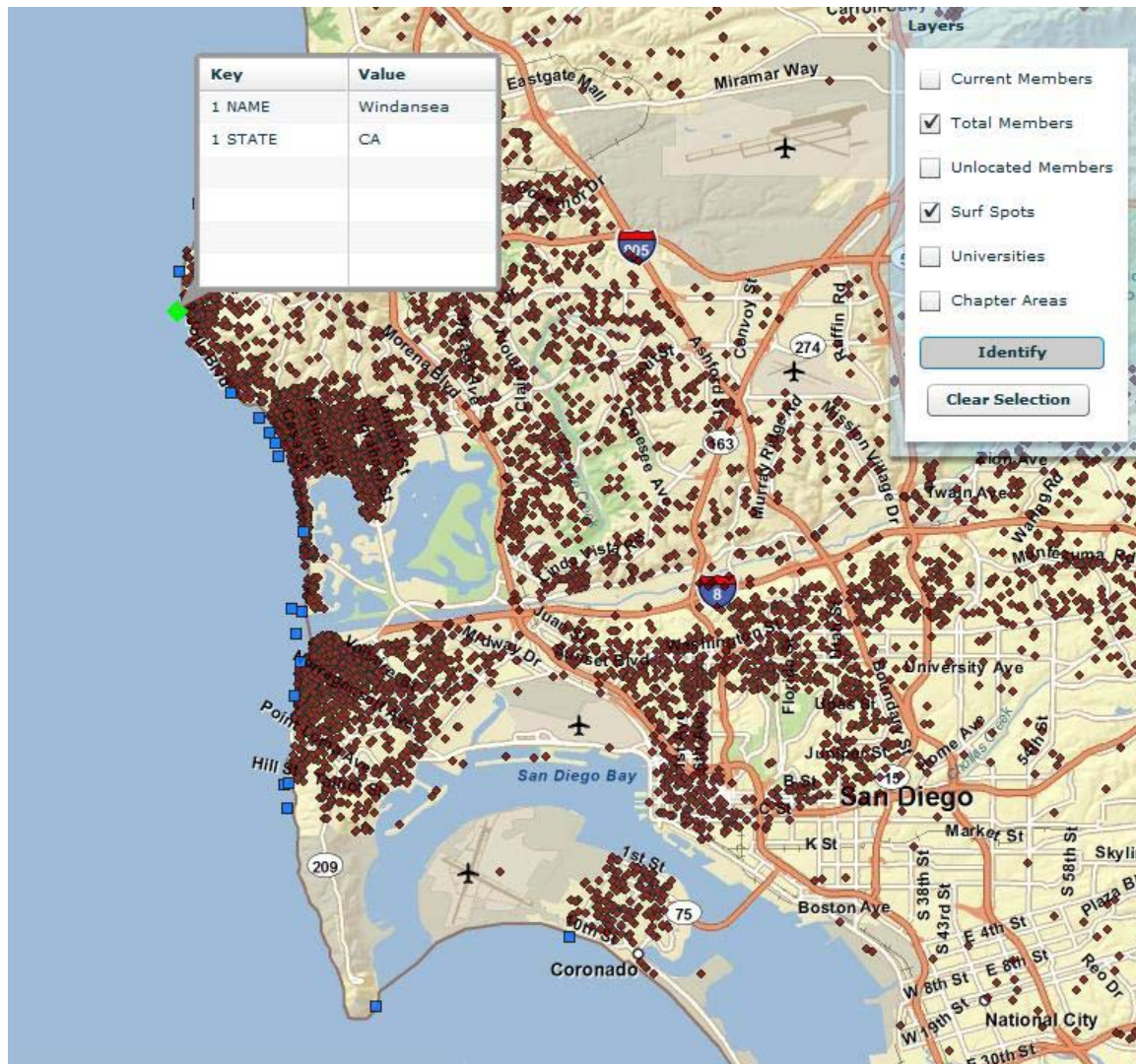


Figure 6-3: Identifying features – members near surfing spots

Members may also congregate near universities, especially those with active environmental or marine conservation programs. For example, we can find a local cluster of members around Oregon State University which has an active marine conservation program (Figure 6-4). Students and faculty involved in conservation programs are often aware of environmental issues. Comparing membership records and their proximity to these universities allows the Surfrider Foundation to discover if these students have knowledge of and support the organization. This may lead to additional student research and volunteer work. In addition to this, the Surfrider Foundation can identify universities with environmental and marine conservation programs having only a few members in close proximity in order to focus environmental activism and recruitment efforts in the future.

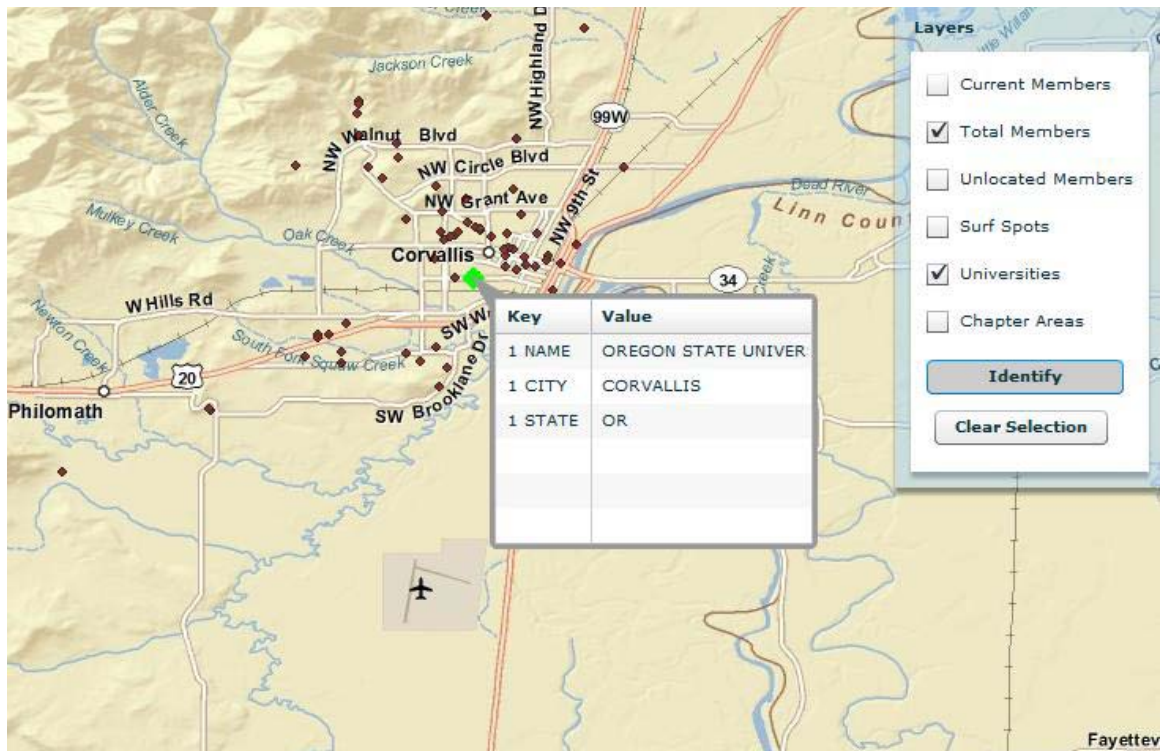


Figure 6-4: Identifying features – members near universities

6.3 Search Tools

In order to explore the membership dataset and generate a list of specific members which can be further analyzed, three different search tools were developed for the project. These include the Search by Polygon, Search by Buffer, and Search by Attribute tools. The Search by Polygon and Search by Buffer tools are both located under the Search by Selection tab on the toolbar. The Search by Attribute tool resides in its own panel, which can be accessed by selecting the Search by Attribute tab on the toolbar.

The Search by Polygon tool allows the user to draw a polygon around an area of interest and retrieve member locations within that area. There are two drawing tools and a clear selection tool. The polygon draw tool allows the user to draw a straight-line polygon by selecting more than two points on the map. The user then double clicks the map and the final line is drawn, closing the polygon. The freehand polygon draw tool, however, allows the user to draw a polygon in any desired shape and not be regulated to drawing a straight-line polygon. After the polygon is drawn, the map interface then zooms to the extent of the shape, making it easier to identify the number and distribution of the selected member records. The clear selection feature can then be used to clear any polygon and search results that were added to the map interface.

Once a polygon has been drawn, a request is sent to the MapServer and a search of the current and total members dataset is performed based on the geographic extent of the

polygon. All members living within the boundaries of the drawn shape are returned to the map in a highlighted color different from the Total Members layer in the table of contents (Figure 6-5). This gives some contrast to the selected points and distinguishes them from those in the map layer. If the map layer is not already drawn when the search feature is deployed, then only the highlighted member records will appear within the polygon boundary.

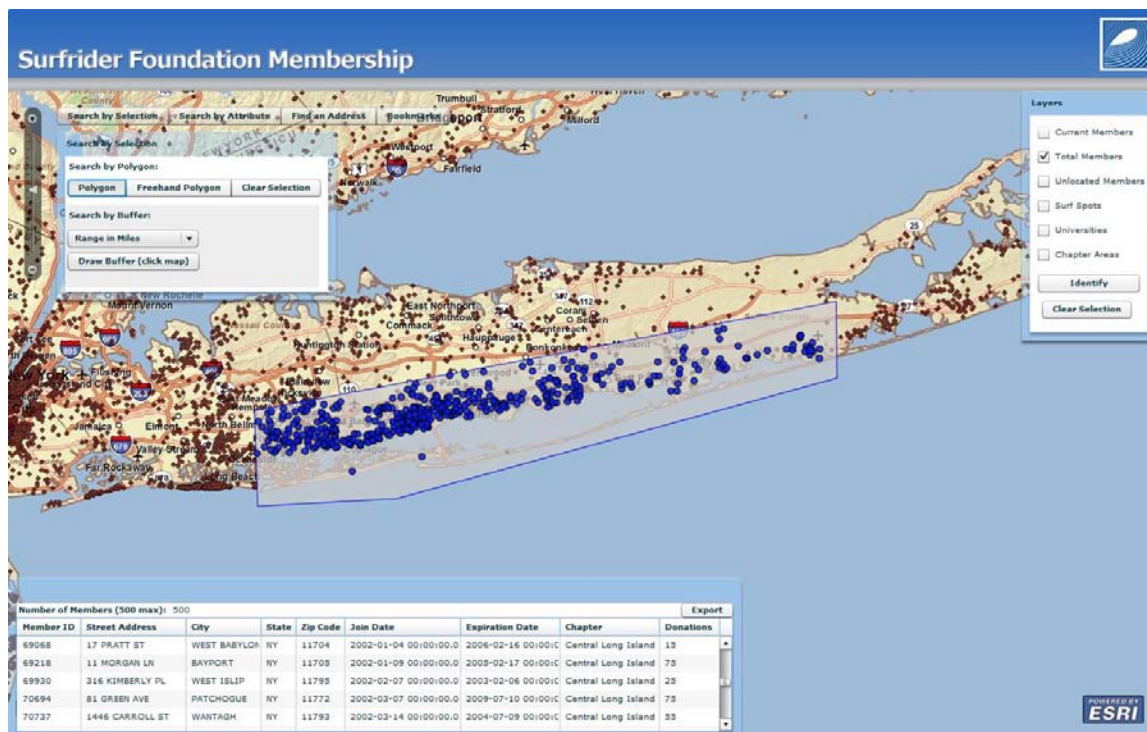


Figure 6-5: Search by Polygon tool

Once a selection has been made using the Search by Polygon tool, the tool panel is collapsed and all selected member records appear in a table locked to the lower left corner of the map interface. This table is designed to mimic the attribute table associated with the current and historic members dataset located on the MapServer and includes various fields which display attributes, including member ID, address information, membership enrollment and expiration dates, the regional chapter, and how much money each member has donated. These fields can be sorted by numerical value or alphabetical listing to help the user quickly find specific records in the table. A membership counter at the top of the table also records the number of members found within the search area.

In order to identify the exact location of a record in the table on the map interface, the user can select the record and see an identification box with the member's identification number displayed at the location of that record (Figure 6-6).

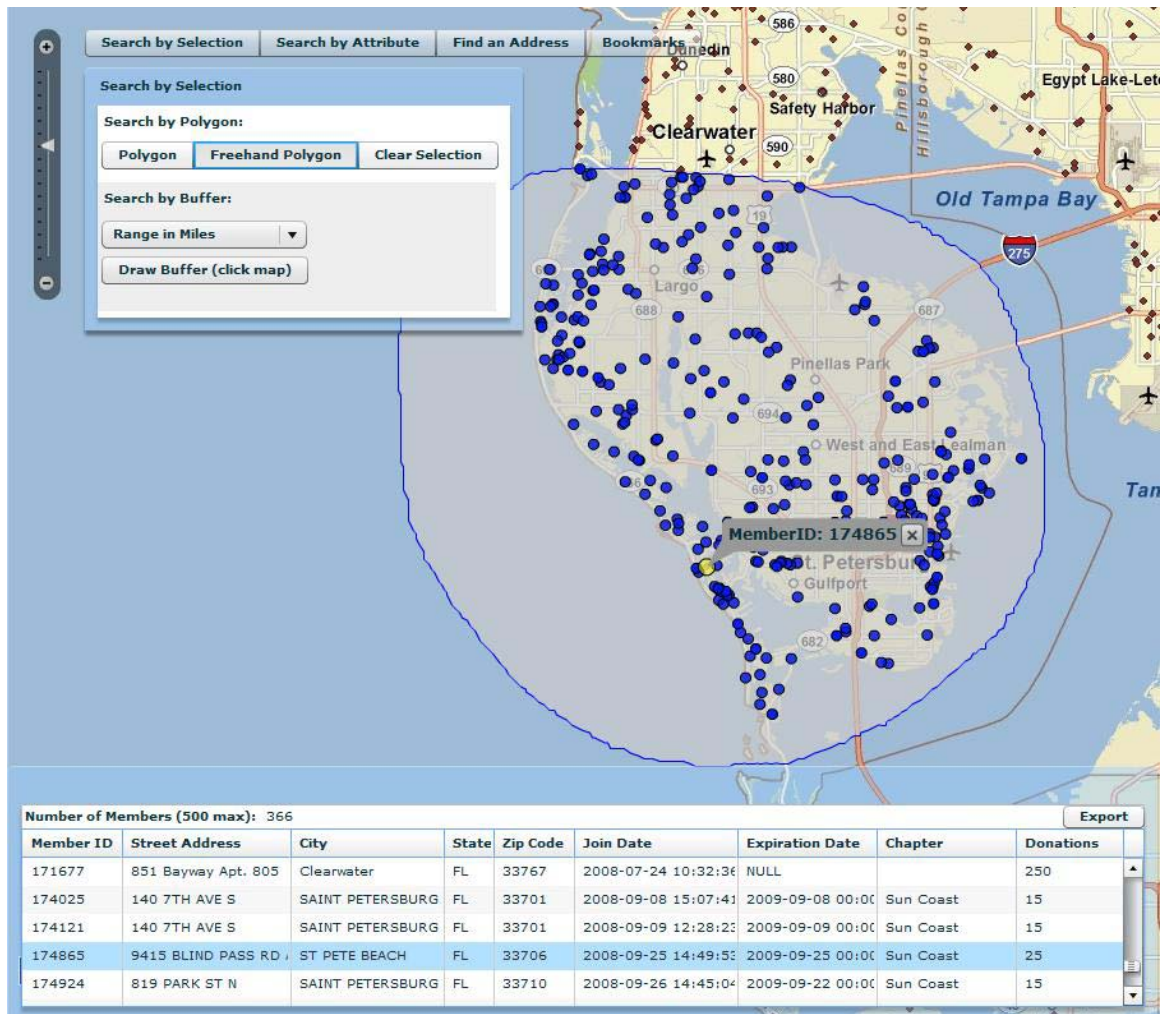


Figure 6-6: Record identification from the table

Records can also be found in the table by hovering the cursor on a member of interest on the map and seeing the corresponding record highlighted in the table. When the record is highlighted, a box appears which includes all the attribute data contained in the table (Figure 6-7). These features allow the user to quickly identify the location of an individual member and retrieve corresponding attribute data from the table.

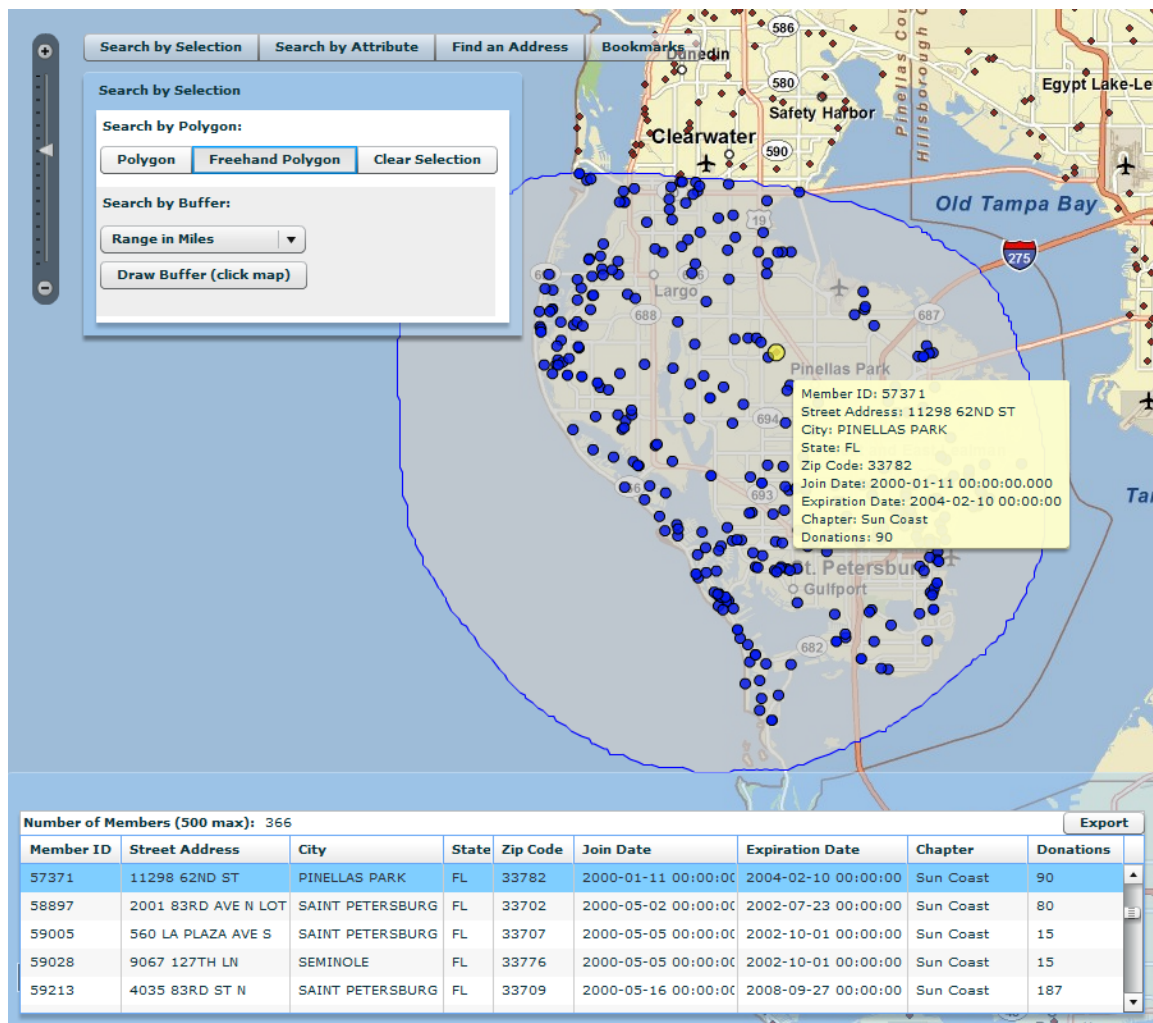


Figure 6-7: Record attributes

The Search by Buffer allows the user to draw a buffer around a specified center point. All member records located with the geographic extent of that buffer are then returned to the application from the MapServer. To draw the buffer, the user selects the buffer range from a drop down box. These preset distances are set at 1, 10, 20, 50, or 100 mile increments. The user, however, can select only one buffer range at a time. Once the range is set, the user then selects the draw buffer button, clicks a location on the map, and the map interface zooms to the extent of the buffer drawn around that point (Figure 6-8). The ESRI StreetMap World 2D MapServer base map data, using the WGS-84 datum, is unprojected which may cause slight stretching of the buffer area as seen in Figure 6-8. This may have a slight effect on the accuracy of the buffer range especially in areas at higher latitudes, but does not pose a significant problem for the application.

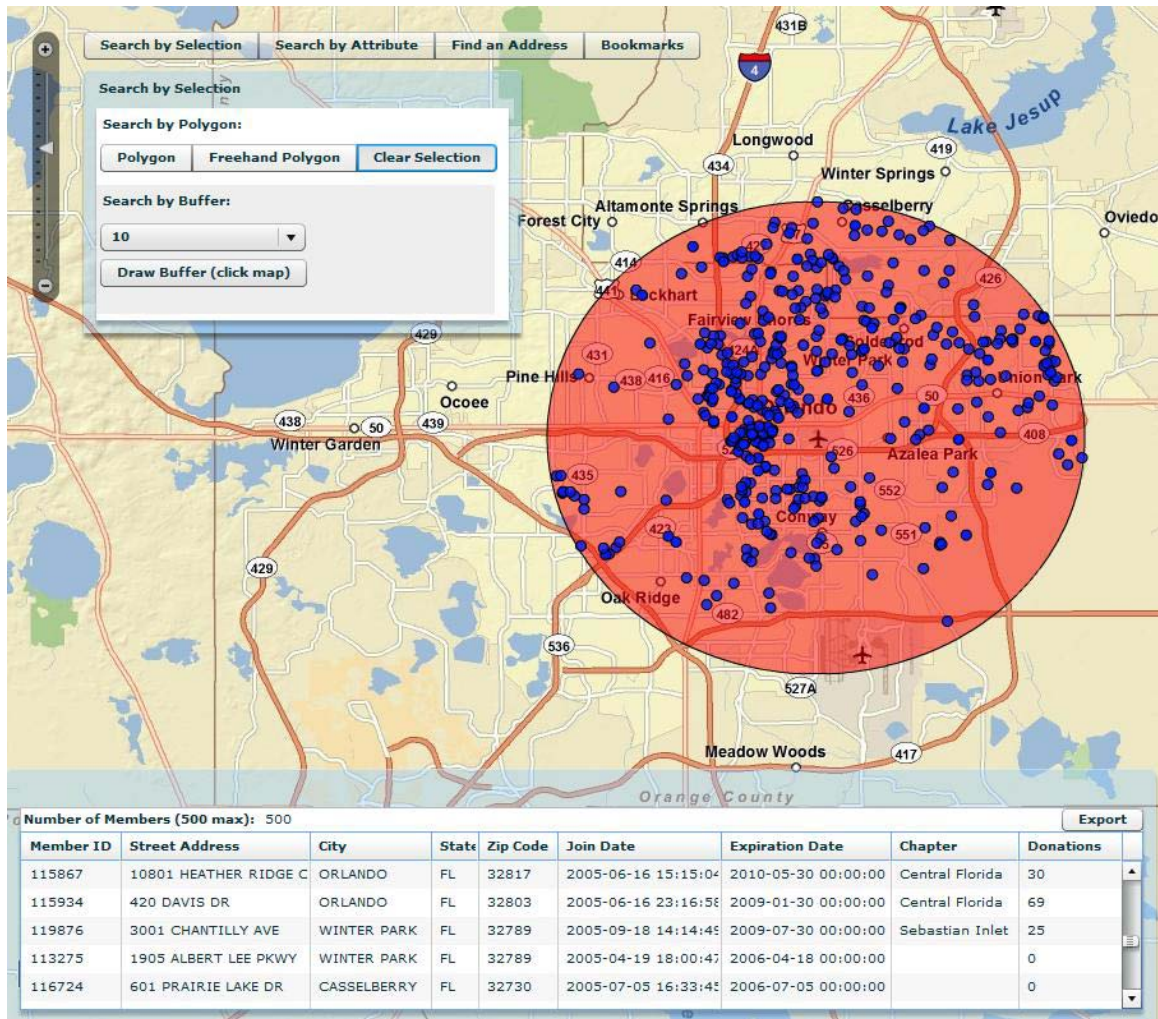


Figure 6-8: Select by Buffer

As with the Search by Polygon tool, the member records located within the buffer extent are counted and displayed in the table at the bottom left corner of the map interface. These records can also be identified on the map by moving the cursor over the record. The attribute data is also displayed in a pop-up window when the cursor hovers over the member location on the map. The clear selection tool can also be used to clear the buffer and all selected membership records from the map and table. The Select by Buffer tool can be used to find the number of members living within a specific range of a given point. This proves useful to the Surfrider Foundation in finding the number of members living near a popular beach, surfing spot, or chapter headquarters.

In order to perform advanced tools available in Microsoft Excel, an export function is available in the data table (Figure 6-9). This function exports the membership records found using the search tools to a CSV which can be opened in Excel. Having the ability to access Excel's wide range of calculation and graphing tools allows the user to perform advanced analysis using select records. The user also has the ability to print the data table after it has been export to Excel.

The Select by Attributes tool was designed to allow the user to query the current and historic members dataset located on the MapServer and display the results in the map interface. An individual member can be located using the Member ID field in the attribute table (Figure 6-9).

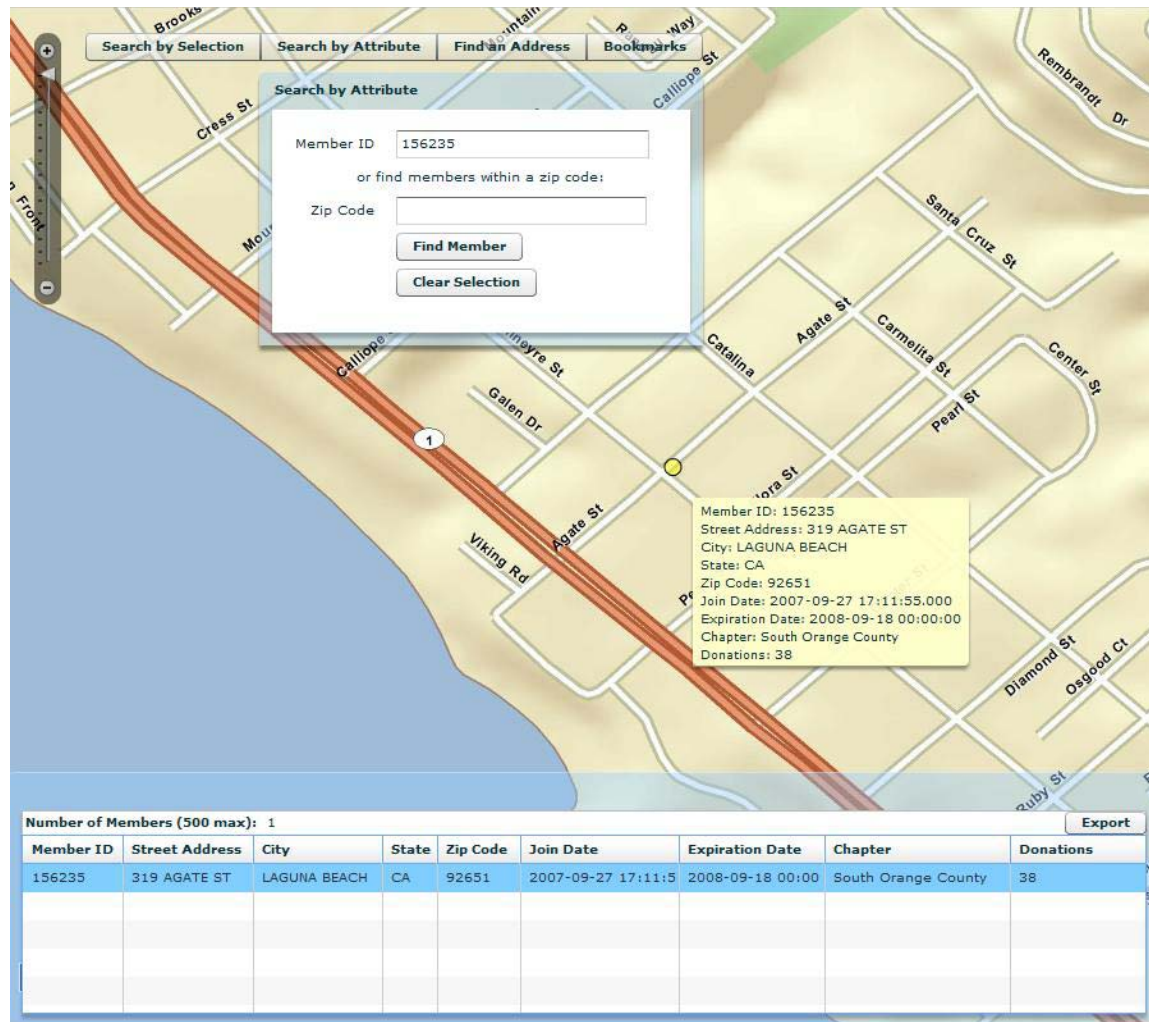


Figure 6-9: Select by Attribute – Member ID

All the members living within a specific zip code can also be located using the Zip Code field in the attribute table (Figure 6-10). The user, however, cannot select both the member ID and zip code search options simultaneously. Once the user enters a value in one of the two search boxes, the query results are retrieved from the MapServer and the member locations are displayed in the map interface. The member records are also counted and displayed in the table at the bottom left corner of the map interface. Similar to the search tools, these records can also be identified on the map by moving the cursor over the record. A pop-up window containing all member attributes is displayed when the cursor hovers over the member location on the map. The clear selection tool is used to clear selected members and records from the map and table.

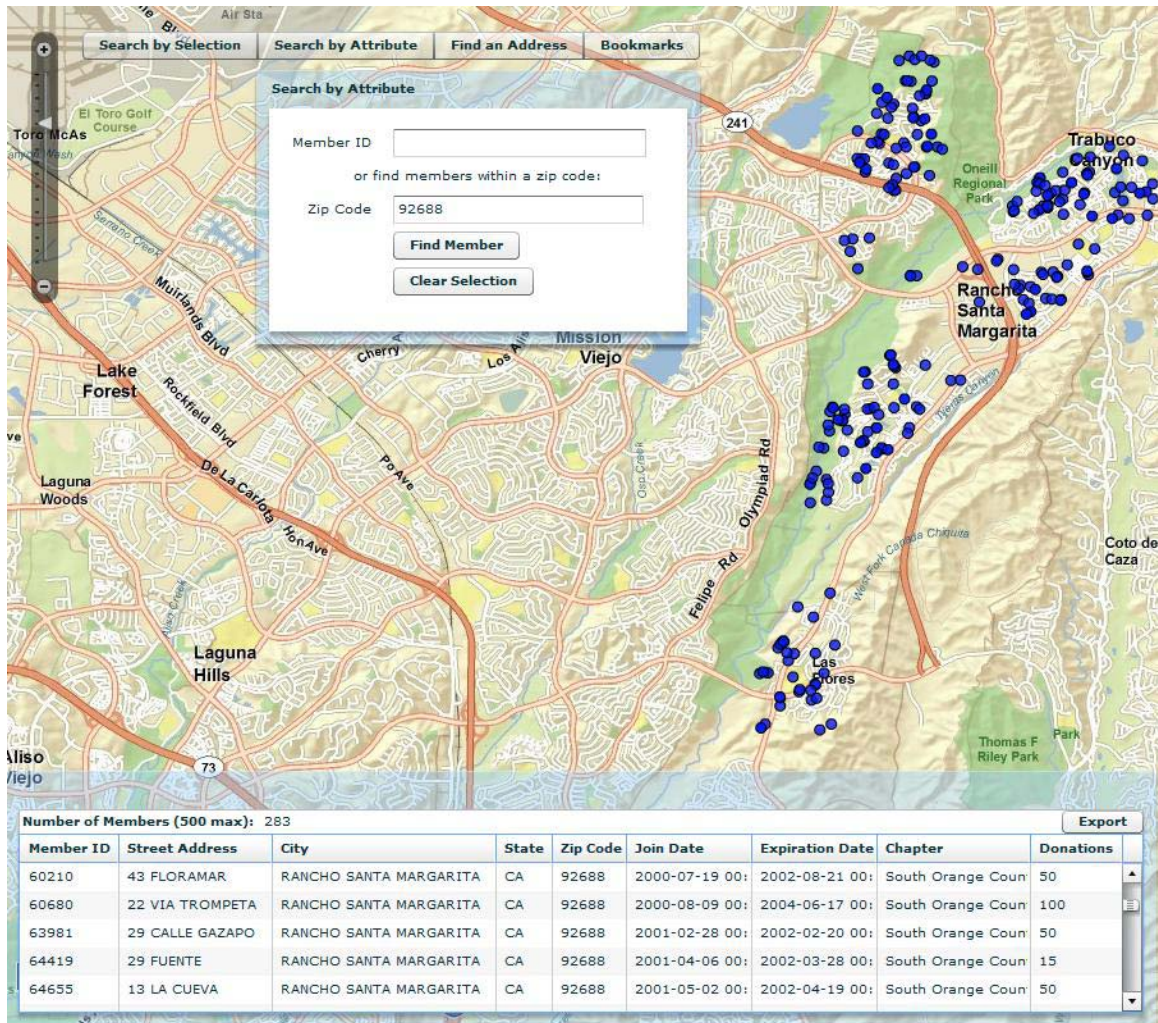


Figure 6-10: Select by Attribute – Zip Code

6.4 Find an Address Tool

The Find an Address tool allows the user to locate a specific address on the map. This tool runs off an ESRI geocoding service which is accessed directly from ESRI ArcGIS Online by the application. Upon opening the tool, the necessary fields are populated with a default address which serves as an example of the address format read by the geocoding service. The required address fields include a street address, city, state, and zip code. Once an address is located, an address match will appear in the tool panel to inform the user that the address of interest has been located (Figure 6-11). If the tool was unable to locate the specified address, the results window will indicate that the nearest address match was selected or no address was found at all. Once the address match is confirmed, the map interface zooms to the local street map extent and places a highlighted dot on at the location. A pop-up box is also displayed when the cursor hovers over the point.

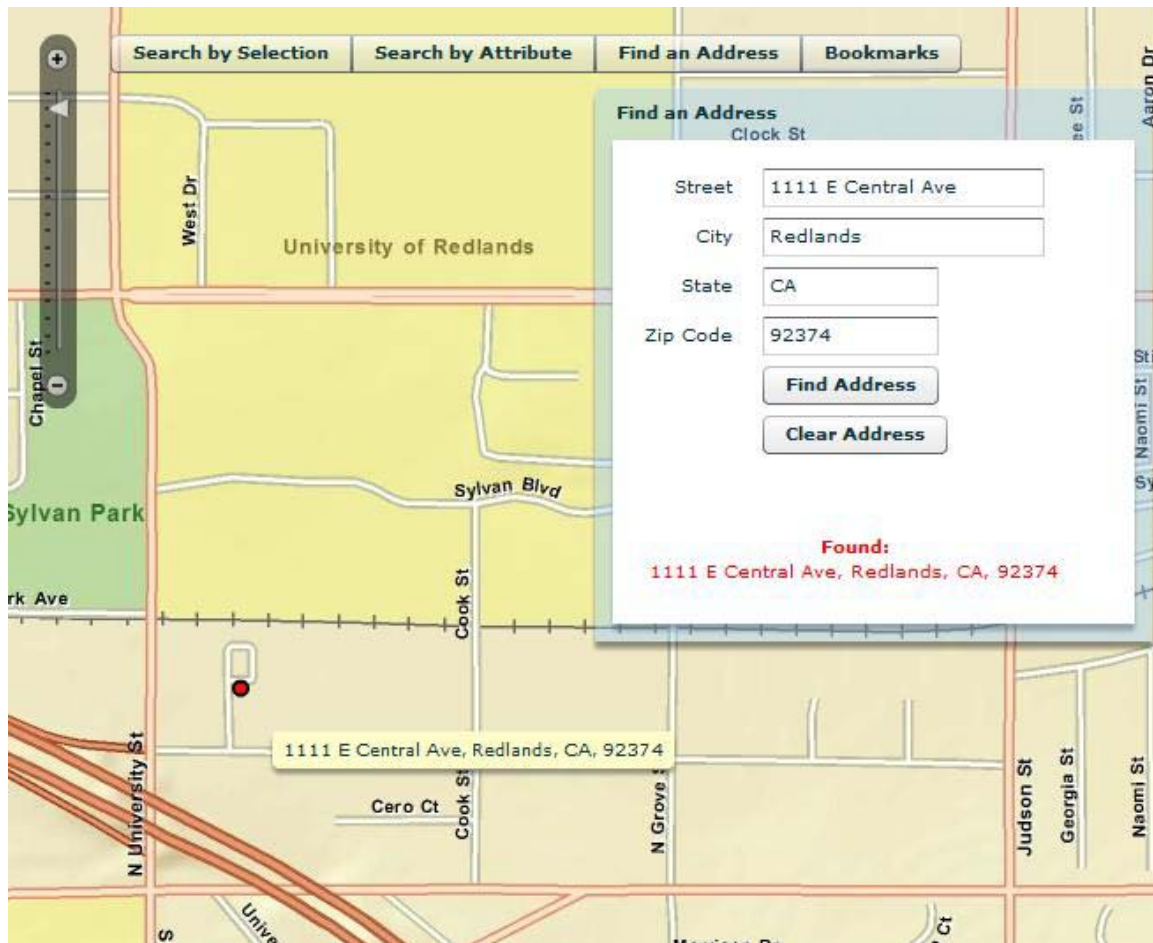


Figure 6-11: Find an Address

The Find an Address tool allows the Surfrider Foundation to find the location of new members as their enrollment information is received. This is especially important as the organization gauges the effectiveness of its community education programs and recruitment efforts. If an address is provided for a beach clean-up project, environmental awareness event, or rally the location can be identified on the map and current members in the area can be notified to participate. The office address for one of the 75 regional chapters can also be located and the Search by Buffer tool can be used to discover the number of members living within a certain range of that location. By analyzing the enrollment dates of the records associated with those members, the organization is able to identify chapters which are experiencing significant growth and which chapters have not seen growth for some time. This information can help the Surfrider Foundation better manage chapter programs and recruitment activities.

6.5 Analyzing Member Donations

To continue its environmental research and education programs, the Surfrider Foundation relies on monetary donations from its membership. Since 1993, the organization has received close to \$22 million in member donations. Historically, people in certain areas

of the United States have donated more than others. Table 3 lists the top twenty zip codes which have produced the most donations in the past eleven years.

Table 3. Zip codes yielding highest donations

Number	Zip Code	Post Office Name	State	Amount
1	94022	Los Altos	CA	\$507,934
2	77006	Houston	TX	\$361,083
3	95814	Sacramento	CA	\$308,291
4	90025	Los Angeles	CA	\$306,407
5	33483	Delray Beach	FL	\$186,092
6	92656	Aliso Viejo	CA	\$182,938
7	92651	Laguna Beach	CA	\$180,818
8	90808	Long Beach	CA	\$148,518
9	92625	Corona Del Mar	CA	\$129,229
10	90024	Los Angeles	CA	\$105,525
11	94123	San Francisco	CA	\$103,809
12	60611	Chicago	IL	\$100,459
13	94109	San Francisco	CA	\$75,645
14	92008	Carlsbad	CA	\$65,298
15	90405	Santa Monica	CA	\$63,334
16	10011	New York	NY	\$52,759
17	92660	Newport Beach	CA	\$51,107
18	85018	Phoenix	AZ	\$51,050
19	97209	Portland	OR	\$50,680
20	02109	Boston	MA	\$50,150

The members living within these zip codes can be identified using Select by Attribute tool in the online mapping application. Once selected, these members will be displayed on both the map and the data table located in the lower left corner of the interface. Using the data table, the user can identify which members have donated the most money (Figure 6-12). This tool helps the user find areas where donations are the highest and allows the organization to focus campaign efforts, including petitions, for additional contributions.

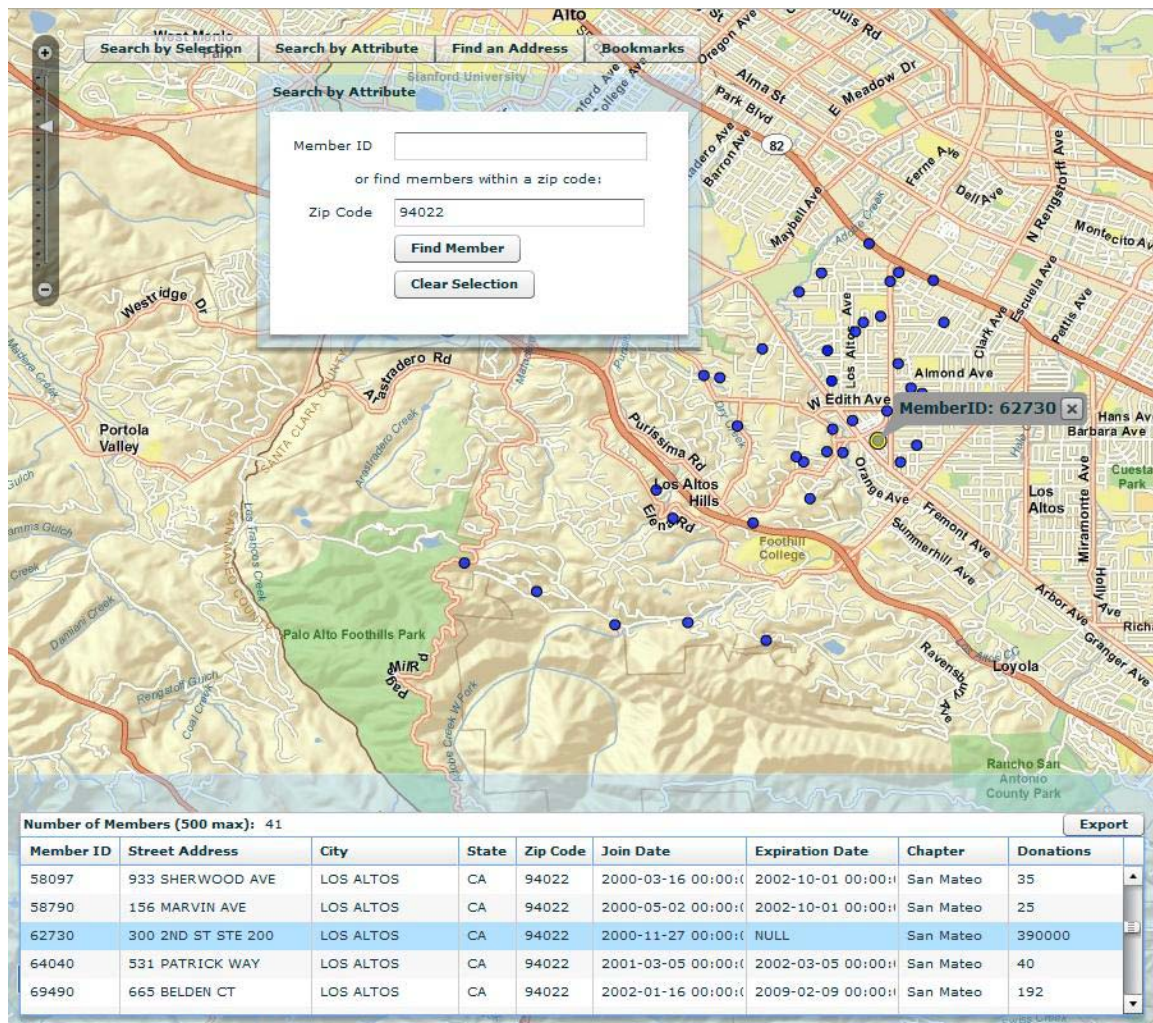


Figure 6-12: Finding areas of high donation

Certain regional chapters may also produce more donations than other. Many factors may contribute to the amount of donations received at the chapter level. These factors include the size of the chapter, number of wealthy areas in the chapter, and the number of completed fundraising activities and their success. Table 4 identifies the ten regional chapters which have received the most donations over the past eleven years. These chapters can be further analyzed by the Surfrider Foundation to identify fundraising methods or specific events which may have proven profitable in the past. These methods or events can then be applied to other chapters where members have not donated as much. The online mapping application can also be used to identify the geographic extent of the chapter. Using the map interface and application tools to identify major cities, popular beaches and surfing spots, universities, and areas of high member concentrations, the Surfrider Foundation can plan fundraising events and focus future campaign strategies and mailers in the right locations.

Table 4. Regional chapters yielding highest donations

Number	Chapter	Amount
1	San Mateo	\$918,417
2	Malibu/West Los Angeles	\$782,189
3	San Diego	\$539,640
4	South Orange County	\$521,578
5	Newport Beach	\$427,692
6	New York City	\$418,722
7	Texas Upper Coast	\$414,382
8	San Francisco	\$315,180
9	Long Beach	\$242,860
10	Palm Beach	\$210,493

6.6 Benefits of a Web GIS Application

The tools discussed in this chapter will add a much needed geospatial perspective to the Surfrider Foundation membership dataset. Before the online mapping application was made available to the organization, membership coordinators were unable to visualize the thousands of membership records housed within their SQL Server database. The application now allows them to not only visualize the geographic distribution of Surfrider Foundation membership on a map, but search, explore, and analyze the dataset as well. These new techniques add considerable context to the methods used by the organization to manage membership growth and evaluate the effectiveness of educational programs and recruitment activities.

Chapter 7 – Conclusions and Future Work

7.1 Conclusion

As a non-profit environmental organization dedicated to the protection of the world's oceans, beaches, and waves, the Surfrider Foundation is at the forefront of grassroots conservation and activism. As the world's population moves into a heightened sense of conservation and sustainability, Surfrider Foundation membership support will continue to grow for years to come. For this purpose, the Surfrider Foundation must obtain a greater understanding of membership distribution, membership growth, and necessary steps to increase this support in the future.

In order to analyze past growth patterns and trends, innovative geospatial tools are a necessity. Now available to the Surfrider Foundation through an online GIS application, these tools allow the organization to visualize, explore, and analyze current and historic membership distribution in a geographic environment. Membership and donation information can also be linked and analyzed using the application interface. Without the benefit of GIS, it is difficult to perceive the extent of Surfrider Foundation reach and influence. Having geospatial knowledge of membership distribution, however, allows the organization to visualize its geographic scope and focus the deployment of its educational and recruitment programs.

The major objective of this project was to provide the Surfrider Foundation with an online mapping application to visualize, explore, and analyze its membership distribution and discover new approaches to membership growth and retention. As a result of this, the Surfrider Foundation now has the ability to perform basic GIS tasks and obtain additional knowledge of its membership. As an essential component of the project, this GIS empowerment allows the client to complete vital analysis and take necessary steps for future growth.

The application developed for the project satisfies the functional and non-functional requirements originally proposed by the Surfrider Foundation. The geodatabase created for the project organizes and stores the data that was provided. Tools included in the online mapping application allow the user to easily visualize, query, and geocode data. These tools also allow the organization to identify the distribution, relationships, and patterns of growth of membership and donation data.

7.2 Future Work

The web GIS application designed for the project is currently retained on the University of Redlands MS GIS server. Once deployed to the Internet, the application will be available to the client for a period of twelve months. When this grace period expires the Surfrider Foundation will be required to obtain the hardware and software needed to administer the online application, as well as the personnel necessary to manage the data and maintain the server. During the next twelve months, the Surfrider Foundation is encouraged to evaluate the application and its potential use in membership and donation

management and analysis. If seen as a valuable resource to the organization, necessary steps can then be taken to transfer complete ownership of the application to the Surfrider Foundation.

Because the application and its corresponding datasets reside on the MS GIS server, administrative access to the data is prohibited by outside users. For this reason, the online application does not allow users to edit the data layers in any way. Providing users the ability to add or delete membership records within the application interface could damage the integrity of the datasets included in the application and cause potential harm to the server. Having the ability to manage and edit feature datasets within the application interface cannot be incorporated until the application has been permanently moved to the Surfrider Foundation server.

The web GIS application is currently designed to visualize, explore, and analyze historic membership growth and distribution across the United States. Future additions to the application could include data input tools having direct access to the membership dataset residing on the server. These input tools could allow the membership coordinator to add new member information to the dataset through the application interface. This would create a living dataset containing current membership and donation information which could be displayed on the map interface and analyzed using available GIS tools. Additional datasets could also be included to allow the user to visualize and manage worldwide Surfrider Foundation membership within the same application.

Works Cited

- Childs, C. (2009, 03). The Top Nine Reasons to Use a File Geodatabase. *ArcUser, The Magazine for ESRI Users*, 12(2), pp. 12-15.
- Environmental Systems Research Institution. (2009). *ArcGIS Server: A Complete and Integrated Server GIS*. [Brochure] Redlands, CA: ESRI.
- Fox, G., Zamudio, G., Redmond, Q., & Redmond, J. (1997). GIS for GOD: Geographical Organization of Data for Notre Dame Church Membership. *Proceedings of the Seventeenth Annual ESRI User Conference*. San Diego: ESRI.
- Golden, R., & Schell, J. D. (2008). Using ZIP Code and GIS Studies to Assess Disease Risk. *Environmental Health Perspectives, Vol. 116, No. 1*, A18.
- Granados, M. (2003). Mapping Data on Enrolled Students. In D. Teodorescu, *Using Geographic Information Systems in Institutional Research* (pp. 23-35). San Francisco: Jossey-Bass.
- Gregory, I. N. (2008). "A Map is Just a Bad Graph": Why Spatial Statistics are Important in Historical GIS. In A.K. Knowles, *Placing History: How Maps, Spatial Data, and GIS are Changing Historical Scholarship* (pp. 123-149). Redlands: ESRI Press.
- Harder, C. (1998). *Serving Maps on the Internet*. Redlands: ESRI Press.
- Kouznestova, M., Huang, X, Ma, J., & Carpenter, D.O. (2007). Increased Rate of Hospitalization for Diabetes and Residential Proximity of Hazardous Waste Sites. *Environmental Health Perspectives, Vol. 115, No. 1*, 75-79.
- Krivoruchko, K., & Gotway, C. (2002). Expanding the "S" in GIS: Incorporating Spatial Statistics in GIS. *CSISS Specialist Meeting on Spatial Data Analysis Software Tools*. Santa Barbara: ESRI; Centers for Disease Control and Prevention.
- Mitchell, A. (1999). *The ESRI Guide to GIS Analysis, Volume 1: Geographic Patterns and Relationships*. Redlands: ESRI Press.
- Mitchell, A. (2005). *The ESRI Guide to GIS Analysis, Volume 2: Spatial Measurements and Statistics*. Redlands: ESRI Press.
- Monroe, A. D., & Comer, J. C. (2002). *Spatial and Socioeconomic Patterns of Freemason Membership in Oklahoma*. Stillwater, OK: Oklahoma Academy of Science.

- Murray, A. T., & Estivill-Castro, V. (1998). Cluster Discovery Techniques for the Exploratory Spatial Data Analysis. *International Journal of Geographic Information Science*, Vol. 12, No. 5 , 431-443.
- Peng, Z., & Yu, D. (2002, October). *GIS on the Internet: A state of the Art Review*. Paper presented at the URISA 2002 Annual Conference, Chocago, IL.
- Sieber, R. (2000). GIS Implementation in the Grassroots. *URISA Journal*, Vol. 12, No. 1.
- Stevenson, S., Hasham, F., & Albee, J. (2005, July). *Web-GIS Integration in Gwinnette County: A Multi-Faceted Approach*. Paper presented at 25th Annula ESRI User Conference, San Diego, CA.
- Wikle, T. (1995). Geographic Patterns in Membership in U.S. Environmental Organizations. *The Professional Geographer*, Vol. 47, No. 1 , 41-48.
- Wikle, T. (1997). Those Benevolent Boosters: Spatial Patterns of Kiwanis Membership in the United States. *Journal of Cultural Geography*, Vol. 17, No. 1 , 1-19.

Appendix A. Web GIS Application Code

The programming code used for the Surfrider Foundation Membership application was developed in Adobe Flex 3.0. All GIS tools and graphic user interface design elements are made available in the following MXML script.

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application
    xmlns:mx="http://www.adobe.com/2006/mxml"
    xmlns:esri="http://www.esri.com/2008/ags"
    xmlns:samples="com.esri.ags.samples.*"
    creationComplete="init()"
    cornerRadius="0" alpha="1.0" layout="absolute"
    horizontalScrollPolicy="off" verticalScrollPolicy="off">
    <!-- Style definition for the entire application. -->
    <mx:Style>
        Application {
            paddingLeft: 10px;
            paddingRight: 10px;
            paddingTop: 10px;
            paddingBottom: 10px;
            horizontalAlign: "center";
            backgroundImage: "";
            backgroundColor: 000000;
        }
    </mx:Style>

    <mx:Script>
    <![CDATA[
    /*
        Developer: Luke Moffat, Cohort 15, MS GIS Program, University of
        Redlands

        Build Date: 11-01-09

        Created Using: FlexBuilder 3

        Tested on: FireFox 3.0.15, Internet Explorer 8.0.6001.18828

        Purpose: This standalone application incorporates Surfrider
        Foundation membership information from 1-1984 to 3-2009.
        Membership data can be searched by drawing a polygon or
        buffer or by attribute. Search results are displayed
        as point symbols on the map and within a data table
        included in the interface. Additional functionality
        includes an address geocode tool. These tools were
        referenced from samples available on the ESRI Resource
        webpage
        (http://resources.esri.com/help/9.3/arcgisserver/apis/flex/
        samples/index.html)
    */
    import flash.utils.getQualifiedClassName;
```

```

import mx.messaging.AbstractConsumer;
import mx.controls.CheckBox;
import flash.trace.Trace;
import com.esri.ags.geometry.Extent;
import flash.events.MouseEvent;
import mx.controls.Alert;
import com.esri.ags.tasks.AddressCandidate;
import mx.controls.dataGridClasses.DataGridItemRenderer;
import mx.collections.ArrayCollection;
import mx.rpc.events.FaultEvent;
import mx.rpc.AsyncResponder;
import mx.logging.LogEventLevel;
import mx.events.ItemClickEvent;
import mx.events.DataGridEvent;
import mx.events.ListEvent;
import com.esri.ags.events.QueryEvent;
import com.esri.ags.events.DrawEvent;
import com.esri.ags.tasks.FeatureSet;
import com.esri.ags.tasks.QueryTask;
import com.esri.ags.tasks.Query;
import com.esri.ags.Graphic;
import com.esri.ags.layers.GraphicsLayer;
import com.esri.ags.geometry.MapPoint;
import com.esri.ags.geometry.Geometry;
import com.esri.ags.geometry.Polygon;
import com.esri.ags.toolbars.Draw;
import com.esri.ags.symbol.SimpleFillSymbol;
import com.esri.ags.symbol.SimpleLineSymbol;
import com.esri.ags.symbol.SimpleMarkerSymbol;
import com.esri.ags.SpatialReference;
import com.esri.ags.events.GeometryServiceEvent;
import com.esri.ags.tasks.BufferParameters;
import com.esri.ags.events.MapMouseEvent;
import com.esri.ags.geometry.MapPoint;
import com.esri.ags.tasks.ExecuteResult;
import com.esri.ags.tasks.ParameterValue;
import com.esri.ags.symbol.InfoSymbol;
import com.esri.ags.tasks.IdentifyParameters;
import com.esri.ags.tasks.IdentifyResult;

public var mouseOverGraphic : Graphic = new Graphic();
public var lineSymbol : SimpleLineSymbol = new
    SimpleLineSymbol("solid",0, 1, 1);
public var ContiguousUSA : Extent = new Extent(-140.076640625,
    16.01974609375, -55.70164062500001, 60.53634765625);
public var Alaska : Extent = new Extent(-171.06408166503905,
    50.791956665039066, -128.87658166503905, 73.77535510253907);
public var Hawaii : Extent = new Extent(-161.944267578125,
    18.398268125, -153.4964404296875, 23.0455029296875);
public var California : Extent = new Extent(-131.99669885253905,
    31.445032836914066, -110.90294885253905, 42.936732055664066);
public var Florida : Extent = new Extent(-95.14855432128905,
    23.304163696289066, -74.05480432128905, 34.795862915039066);
public var Northeast : Extent = new Extent(-88.34801721191405,
    33.730189086914066, -67.25426721191405, 45.221888305664066);
public var tempSymbol : SimpleMarkerSymbol = new SimpleMarkerSymbol();
private var usa:Rectangle = new Rectangle(-140, 16, -56, 60);

```

```

private var alaska:Rectangle = new Rectangle(-179, 45, 112, 82);
private var hawaii:Rectangle = new Rectangle(-161, 18, 153, 23);
private var mapPoint:MapPoint = new MapPoint();
public var mouseClickedGraphic: Graphic = new Graphic();

//SimpleMarkerSymbol
public var pointSymbol : SimpleMarkerSymbol =
    new SimpleMarkerSymbol( SimpleMarkerSymbol.STYLE_CIRCLE , 5 ,
        0x0000FF, 0.5, 0, 0, 0, lineSymbol);
public var mouseOverPointSymbol : SimpleMarkerSymbol =
    new SimpleMarkerSymbol( SimpleMarkerSymbol.STYLE_CIRCLE , 13 ,
        0xFFFF00 , 0.5, 0, 0, 0, lineSymbol);
public var drawToolbar:Draw = new Draw();

private function init():void
{
    //things to do when the page creation is complete
    drawToolbar.fillSymbol = fillSymbol;
    drawToolbar.lineSymbol = lineSymbol;
    drawToolbar.map = myMap;
    drawToolbar.addEventListener(DrawEvent.DRAW_END,onDrawEnd);
}

private function rb_change(evt:Event):void {
    var rb:RadioButton = evt.currentTarget as RadioButton;

    bookmarksView.visible = false; //hide tool once bookmark is
    selected

    switch (rb.label)
    {
        case "Contiguous USA":
        {
            myMap.extent = ContiguousUSA;
            break;
        }
        case "Alaska":
        {
            myMap.extent = Alaska;
            break;
        }
        case "Hawaii":
        {
            myMap.extent = Hawaii;
            break;
        }
        case "California":
        {
            myMap.extent = California
            break;
        }
        case "Florida":
        {
            myMap.extent = Florida
            break;
        }
        case "Northeast":

```



```

        {
            myMap.extent = Northeast
            break;
        }
    }
}

public function onRollOverEventHandler(event:Event):void
{
    var e:ListEvent = ListEvent(event);

    //use the DataGridItemRenderer to retrieves information from the
    DataGrid row mouseover
    var dgrid:DataGridItemRenderer =
    DataGridItemRenderer(e.itemRenderer);
    var dgridBlock:Object = dgrid.data.MemberID;

    //display the infoWindow on a point
    var newPoint:MapPoint = new
    MapPoint(dgrid.data.graphic.geometry.x,dgrid.data.graphic.
    geometry.y);
    myMap.infoWindow.label = "MemberID: " + dgridBlock.toString();
    myMap.infoWindow.show(newPoint);

    //display mouseOverGraphic on a point
    mouseOverGraphic = dgrid.data.graphic; //retrieve the graphic
    from the hidden field in the table
    tempSymbol = SimpleMarkerSymbol(mouseOverGraphic.symbol);
    mouseOverGraphic.symbol = mouseOverPointSymbol;
}

public function onRollOutEventHandler(event:Event):void
{
    var e:ListEvent = ListEvent(event);

    //use the DataGridItemRenderer to retrieve information from the
    DataGrid table row mouseover
    var dgrid:DataGridItemRenderer =
    DataGridItemRenderer(e.itemRenderer);
    var dgridGraphic:Graphic = dgrid.data.graphic;

    myMap.infoWindow.hide();
    mouseOverGraphic = dgridGraphic;
    mouseOverGraphic.symbol = tempSymbol;
}

protected function graphicMouseOutEventHandler(event:MouseEvent):void
{
    //reset Graphic symbol one mouseout
    Graphic( event.target ).symbol = tempSymbol;

    //return DataGrid to default behavior
    dgrid1.selectedIndex = -1;
}

protected function graphicMouseOverEventHandler(event:MouseEvent):void
{

```

```

//change the symbol when mousing over a Graphic
var mgraphic:Graphic = Graphic(event.target);
tempSymbol = SimpleMarkerSymbol(mgraphic.symbol);
mgraphic.symbol = mouseOverPointSymbol;

//scroll the DataGrid so that it shows the row that relates to
Graphic
var x:int = findInList(mgraphic);

//highlight the DataGrid row designated by index 'x'
dgrid1.selectedIndex = x;

//validates and updates the properties and layout of the control
and redraws if necessary.
dgrid1.validateNow();

//auto scroll the DataGrid to the row with index 'x'
dgrid1.scrollToIndex(x);
}

public function findInList( graphic : Graphic ) : int
{
    for each( var attributes : Object in dgrid1.dataProvider )
    {
        //compare DataGrid datafield called 'graphic' with geometry
        of Graphic that is being moused over
        if (attributes.graphic.geometry == graphic.geometry)
        {
            return (dgrid1.dataProvider as
                ArrayCollection).getItemIndex(attributes)
        }
    }
    return -1;
}

public function itemClickHandler(event:ItemClickEvent):void
{
    //This function handles button clicks

    //clear the graphics layer
    myGraphicsLayer.clear();

    //clear away all infoWindows
    myMap.infoWindow.hide();

    textArea1.htmlText = "<b>Number of Members:</b> " +
        " " + 0;

    //clear the DataGrid
    var tempArray:Array = new Array(0);
    tempArray = null;
    dgrid1.dataProvider = tempArray;

    drawToolbar.fillSymbol = fillSymbol;

    switch (event.label)
    {
        case "Polygon":

```

```

        {
            drawToolbar.activate(Draw.POLYGON);
            break;
        }
        case "Freehand Polygon":
        {
            drawToolbar.activate(Draw.FREEHAND_POLYGON);
            break;
        }
        case "Clear Selection":
        {
            myGraphicsLayer.clear();
            //myMap.extent = ContiguousUSA;
            break;
        }
    }
}

public function onDrawEnd(event:DrawEvent):void
{
    drawToolbar.deactivate();

    if (event.graphic.geometry is Polygon)
    {
        var drawPolygon : Graphic = new Graphic();
        drawPolygon.geometry = event.graphic.geometry;
        drawPolygon.symbol = fillSymbol;
        drawPolygon.autoMoveToTop = false; //keep graphics from
        moving to top
        myGraphicsLayer.add(drawPolygon);
        myMap.extent =
        Polygon(event.graphic.geometry).extent.expand(2);

        runQueryTask(event.graphic.geometry);

        selectionView.visible = false; //hide tool once area is
        drawn
    }
    else if (event.graphic.geometry is MapPoint)
    {
        bufferMapPoint(event.graphic.geometry);
    }
}

//Geocode tool (single address) from ESRI
private function doClear():void
{
    myGraphicsLayer.clear();
    address.text = "";
    city.text = "";
    state.text = "";
    zip.text = "";
    //myMap.extent = ContiguousUSA;

```

```

}
private function doFind():void
{
var myAddress:Object = {
Address: address.text,
City: city.text,
State: state.text,
Zip: zip.text

};
// Use outFields to get back extra information.
// The exact fields available depends on the specific Locator used.
var myOutFields:Array = ["Loc_name"];
locateTask.addressToLocations(myAddress, myOutFields, new
AsyncResponder(onResult, onFault));

function onResult(candidates:Array, token:Object = null):void
{
    if (candidates.length > 0)
    {
        var addressCandidate:AddressCandidate = candidates[0];
        var myGraphic:Graphic = new Graphic();
        myGraphic.geometry = addressCandidate.location;
        myGraphic.symbol = mySymbol;
        myGraphic.toolTip = addressCandidate.address.toString();
        myGraphic.id = "graphic";
        myGraphicsLayer.add(myGraphic);
        myMap.centerAt(addressCandidate.location);

        // Zoom to an appropriate level
        if (addressCandidate.attributes.Loc_name.search("RoofTop")
        > 0)
        {
            myMap.scale = 10000;
        }
        else if
        (addressCandidate.attributes.Loc_name.search("Address
        ") > 0)
        {
            myMap.scale = 10000;
        }
        else if
        (addressCandidate.attributes.Loc_name.search("Street"
        ) > 0)
        {
            myMap.scale = 15000;
        }
        else if
        (addressCandidate.attributes.Loc_name.search("Zipcode
        ") > 0)
        {
            myMap.scale = 40000;
        }
        else if
        (addressCandidate.attributes.Loc_name.search("City")
        > 0)
        {

```

```

        myMap.scale = 150000;
    }
    else
    {
        myMap.scale = 500000;
    }
    myInfo.htmlText = "<b>Found:</b><br/>" +
        addressCandidate.address.toString(); // formatted
        address
    }
    else
    {

    }

    }
    function onFault(info:Object, token:Object = null):void
    {
        myInfo.htmlText = "<b>Failure</b>" + info.toString();
        Alert.show("Failure: \n" + info.toString());
    }

//Buffer tool script (ESRI)
private function onMapClick(event:MapMouseEvent):void
{

    mapPoint = event.mapPoint;
    mapPoint.spatialReference = myMap.spatialReference;

    mouseClickGraphic.geometry = event.mapPoint;

}

private function bufferMapPoint(point:Geometry):void
{
    var myMapClickPoint:Graphic = new Graphic();
    myMapClickPoint.geometry = point;

    var bufferParameters:BufferParameters = new BufferParameters();
    bufferParameters.features = [myMapClickPoint];
    bufferParameters.distances = [comboMile.text];
    bufferParameters.unit = BufferParameters.UNIT_STATUTE_MILE;
    bufferParameters.bufferSpatialReference = new
        SpatialReference(102113);

    myGeometryService.addEventListener(GeometryServiceEvent.BUFFER_CO
        Mplete, bufferCompleteHandler);
    myGeometryService.buffer(bufferParameters);

    function bufferCompleteHandler(event:GeometryServiceEvent):void
    {
        myGeometryService.removeEventListener(GeometryServiceEvent.
            BUFFER_COMPLETE, bufferCompleteHandler);
        for each (var graphic:Graphic in event.graphics)

```

```

        {
            selectionView.visible = false; //hide tool once
            buffer is drawn on the map

            graphic.symbol = sfs;
            graphic.autoMoveToTop = false;
            myGraphicsLayer.add(graphic);
            runQueryTask(graphic.geometry);
            myMap.extent = graphic.geometry.extent.expand(2);
        }
    }
}

public function doClearExtent():void
{
    memberid.text = "";
    memberzip.text = "";
    myGraphicsLayer.clear();
}

public function doClearWhere():void
{
    memberid.text = "";
    memberzip.text = "";
    //myMap.extent = ContiguousUSA;
    myGraphicsLayer.clear();
}

public function doSearch():void
{
    var queryTask : QueryTask = new QueryTask();
    queryTask.url =
        "//luke/ArcGIS/rest/services/Surfrider/MapServer/4";

    var query : Query = new Query();

    query.returnGeometry = true;
    query.outSpatialReference = myMap.spatialReference;
    query.spatialRelationship = "esriSpatialRelContains";
    query.outFields = ['*'];

    var id:Number = parseInt(memberid.text);
    var zip:String = memberzip.text;

    attributeView.visible = false; //hide tool once the find member
    button is selected

    if (!isNaN(id))
        query.where = "ID = " + id.toString();
    else
        query.where = "ZIP = '" + zip + "'";

    queryTask.execute( query, new AsyncResponder( onResult,
    onFault));
}

```

```

function onResult( featureSet : FeatureSet, token : Object =
null):void
{

    //set up the Array that will be used to push featureSet data to
    DataGrid

    var tempArray:Array = new Array();
    var memberID:int = 0;
    var address:Object;
    var city:Object;
    var state:Object;
    var zip:int = 0;
    var datejoin:Object;
    var dateexpire:Object;
    var chapter:Object;
    var donations:Object;
    var membertotal:int = 0;

    var minX:Number;
    var minY:Number;
    var maxX:Number;
    var maxY:Number;

    //iterate through the featureSet
    for (var i:uint = 0; i < featureSet.features.length; i++)
    {

        memberID = featureSet.features[i].attributes['ID'];
        address = featureSet.features[i].attributes['ST_ADD'];
        city = featureSet.features[i].attributes['CITY'];
        state = featureSet.features[i].attributes['STATE'];
        zip = featureSet.features[i].attributes['ZIP'];
        datejoin = featureSet.features[i].attributes['DATE2'];
        dateexpire = featureSet.features[i].attributes['DATE1'];
        chapter = featureSet.features[i].attributes['Chapter'];
        donations = featureSet.features[i].attributes['Donations'];

        if (i == 0)
        {
            minX = featureSet.features[i].geometry.x;
            maxX = featureSet.features[i].geometry.x;
            minY = featureSet.features[i].geometry.y;
            maxY = featureSet.features[i].geometry.y;
        }
        else
        {
            if (featureSet.features[i].geometry.x < minX)
            minX = featureSet.features[i].geometry.x;

            if (featureSet.features[i].geometry.x > maxX)
            maxX = featureSet.features[i].geometry.x;

            if (featureSet.features[i].geometry.y < minY)
            minY = featureSet.features[i].geometry.y;

```

```

        if (featureSet.features[i].geometry.y > maxY)
            maxY = featureSet.features[i].geometry.y;
    }

    var point : MapPoint = new
    MapPoint(featureSet.features[i].geometry.x,featureSet.features[i]
    .geometry.y);
    var myGraphic : Graphic = new Graphic();

    myGraphic.symbol = memberPoint;
    myGraphic.geometry = point; //geometry of each feature in the
    featureSet

    //listeners for controlling events relating to individual
    graphics

    myGraphic.addEventListener(MouseEvent.CLICK,graphicClickOver
    EventListener);

    myGraphic.addEventListener(MouseEvent.CLICK,graphicClickOutEv
    entHandler);

    myGraphic.toolTip = "Member ID: " + memberID + "\n" +
        "Street Address: " + address + "\n" +
        "City: " + city + "\n" +
        "State: " + state + "\n" +
        "Zip Code: " + zip + "\n" +
        "Join Date: " + datejoin + "\n" +
        "Expiration Date: " + dateexpire + "\n" +
        "Chapter: " + chapter + "\n" +
        "Donations: " + donations;
    myGraphicsLayer.add( myGraphic);

    //set up array so that this data can be sent to the datagrid
    tempArray.push(
    {
        MemberID:memberID,
        StreetAddress:address,
        City:city,
        State:state,
        ZipCode:zip,
        DateJoined:datejoin,
        ExpirationDate:dateexpire,
        Chapter:chapter,
        Donations:donations,
        graphic:myGraphic
    }
    );

    //calculate totals
    membertotal = membertotal + 1;

}

var newExtent:Extent = new Extent(minX, minY, maxX, maxY, new
SpatialReference(4326));

```



```

myMap.extent = newExtent.expand(2);

//send totals to the text area
textArea1.htmlText = "<b>Number of Members (500 max):</b> " +
    " " + membertotal;

//populate the datagrid
dgrid1.dataProvider = tempArray;
}

function onFault( info:Object, token : Object = null) :void
{
    log.text = info.toString();
}

}

public function runQueryTask(geometry:Geometry):void
{

    var queryTask : QueryTask = new QueryTask();
    queryTask.url =
        "/luke/ArcGIS/rest/services/Surfrider/MapServer/4";

    var query : Query = new Query();
    query.geometry = geometry;

    query.returnGeometry = true;
    query.outSpatialReference = myMap.spatialReference;
    query.spatialRelationship = "esriSpatialRelContains";
    query.outFields = ['*'];

    //run the query task
    queryTask.execute( query, new AsyncResponder( onResult,
        onFault));

    function onResult( featureSet : FeatureSet, token : Object =
        null):void
    {
        //set up the Array that will be used to push featureSet data to
        DataGrid

        var tempArray:Array = new Array();
        var memberID:int = 0;
        var address:Object;
        var city:Object;
        var state:Object;
        var zip:int = 0;
        var datejoin:Object;
        var dateexpire:Object;
        var chapter:Object;
        var donations:Object;
        var membertotal:int = 0;

        //iterate through the featureSet
        for (var i:uint = 0; i < featureSet.features.length; i++)
        {

```

```

memberID = featureSet.features[i].attributes['ID'];
address =
featureSet.features[i].attributes['ST_ADD'];
city = featureSet.features[i].attributes['CITY'];
state = featureSet.features[i].attributes['STATE'];
zip = featureSet.features[i].attributes['ZIP'];
datejoin =
featureSet.features[i].attributes['DATE2'];
dateexpire =
featureSet.features[i].attributes['DATE1'];
chapter =
featureSet.features[i].attributes['Chapter'];
donations =
featureSet.features[i].attributes['Donations'];

var point : MapPoint = new
MapPoint(featureSet.features[i].geometry.x,featureSet
.features[i].geometry.y);
var myGraphic : Graphic = new Graphic();

myGraphic.symbol = memberPoint;
myGraphic.geometry = point; //geometry of each
feature in the featureSet

//listeners for controlling events relating to
individual graphics

myGraphic.addEventListener(MouseEvent.MOUSE_OVER,grap
hicMouseOverEventHandler);

myGraphic.addEventListener(MouseEvent.MOUSE_OUT,graph
icMouseOutEventHandler);

myGraphic.toolTip = "Member ID: " + memberID + "\n" +
"Street Address: " + address + "\n" +
"City: " + city + "\n" +
"State: " + state + "\n" +
"Zip Code: " + zip + "\n" +
"Join Date: " + datejoin + "\n" +
"Expiration Date: " + dateexpire + "\n" +
"Chapter: " + chapter + "\n" +
"Donations: " + donations;
myGraphicsLayer.add( myGraphic);
myGraphicsLayer.moveToTop(myGraphic);

//set up array so that this data can be sent to the
datagrid
tempArray.push(
{
    MemberID:memberID,
    StreetAddress:address,
    City:city,
    State:state,
    ZipCode:zip,
    DateJoined:datejoin,
    ExpirationDate:dateexpire,
    Chapter:chapter,

```

```

        Donations:donations,
        graphic:myGraphic
    }
);

//calculate totals
membertotal = membertotal + 1;

}

//send totals to the text area
textArea1.htmlText = "<b>Number of Members (500
max):</b> " +
    " " + membertotal;

//populate the datagrid
dgrid1.dataProvider = tempArray;
}

function onFault( info:Object, token : Object = null) :void
{
    log.text = info.toString();
}

}

public function changeLayer (source:CheckBox, id:Number):void
{
    if(source.selected)
    {
        //add layer
        if (!defaultLayers.contains(id))
            defaultLayers.addItem(id);
    }
    else
    {
        //remove layer
        if (defaultLayers.contains(id))
        {
            defaultLayers.removeItemAt(defaultLayers.getItemIndex
            (id));
        }
    }

    myDynamicService.visibleLayers = defaultLayers;
}

//Identify tool
private function mapClickHandler(event:MouseEvent):void
{
    if(!identifyMode.selected)
        return;

    myGraphicsLayer.clear();

```

```

var clickGraphic:Graphic = new Graphic(event.mapPoint);
myGraphicsLayer.add(clickGraphic);

var identifyParams:IdentifyParameters = new
IdentifyParameters();
identifyParams.returnGeometry = true;
identifyParams.tolerance = 3;
identifyParams.width = myMap.width;
identifyParams.height = myMap.height;
identifyParams.geometry = event.mapPoint;
identifyParams.mapExtent = myMap.extent;
identifyParams.spatialReference = myMap.spatialReference;
identifyParams.layerOption = "all";
identifyParams.layerIds = defaultLayers.toArray();
identifyTask.execute(identifyParams, new
AsyncResponder(myResultFunction, myFaultFunction,
clickGraphic));
}
private function myResultFunction(results:Array, clickGraphic:Graphic =
null):void
{
    var myAttributes:Object = new Object();
    for (var i:Number = 0; i < results.length; i++)
    {
        var myResult:IdentifyResult = results[i];
        var myGraphic:Graphic = myResult.feature;
        switch (myGraphic.geometry.type)
        {
            case Geometry.MAPPOINT:
            {
                myGraphic.symbol = smsIdentify;
                break;
            }
            case Geometry.POLYGON:
            {
                myGraphic.symbol = sfsIdentify;
                break;
            }
        }

        myGraphicsLayer.add(myGraphic);
        myGraphic.autoMoveToTop = false;

        for(var key:String in myGraphic.attributes)
        {
            myAttributes[(i + 1) + " " + key] =
myGraphic.attributes[key];
        }

        clickGraphic.symbol = new InfoSymbol();
        clickGraphic.attributes = myAttributes;
    }
}

private function myFaultFunction(error:Object, clickGraphic:Graphic =
null):void

```

```

{
    Alert.show(String(error), "Identify Error");
}

private function controlClicked(event:ItemClickEvent):void
{
    //first hide all panels
    selectionView.visible = false;
    bookmarksView.visible = false;
    attributeView.visible = false;
    addressView.visible = false;

    //show the appropriate panel
    switch(event.index)
    {
        case 0:
            selectionView.visible = true;
            break;
        case 1:
            attributeView.visible = true;
            break;
        case 2:
            addressView.visible = true;
            break;
        case 3:
            bookmarksView.visible = true;
    }
}

private function exportTableToCSV():void
{
    if (dgrid1.dataProvider == null)
        return;

    var csvTable:String = "";
    var rawData:ArrayCollection = dgrid1.dataProvider as
        ArrayCollection;

    for (var i:int = 0; i < rawData.length; i++)
    {
        var thisRow:String = "";
        if (csvTable == "")
        {
            for (var key:String in rawData[i])
            {
                if (key != "mx_internal_uid" && key !=
                    "graphic")
                {
                    if (thisRow == "")
                        thisRow += "\"" + key + "\"";
                    else
                        thisRow += ",\"" + key +
                            "\"";
                }
            }
            csvTable += thisRow + "\n";
        }
    }
}

```

```

        thisRow = "";
    }
    for (var key:String in rawData[i])
    {
        if (key != "mx_internal_uid" && key
            != "graphic")
        {
            if (thisRow == "")
                thisRow += "\"" +
                    rawData[i][key].toString() + "\"";
            else
                thisRow += ",\"" +
                    rawData[i][key].toString() + "\"";
        }
    }
    csvTable += thisRow + "\n";
}

var variables:URLVariables = new URLVariables();
variables.table = csvTable;

var request:URLRequest = new URLRequest
("http://ri-arcgis93-dev/PostToCSV/");
request.data = variables;
request.method = URLRequestMethod.POST;
navigateToURL(request, "_blank");
}

]]>

</mx:Script>

<esri:SimpleFillSymbol id="sfs" color="0xFF0000">
<esri:SimpleLineSymbol color="0x000000"/>
</esri:SimpleFillSymbol>
<esri:GeometryService id="myGeometryService"

url="http://sampleserver2.arcgisonline.com/ArcGIS/rest/services/G
eometry/GeometryServer"/>

<!--ESRI Geocode tool (single)-->
<esri:Locator
    id="locateTask"
    url="http://tasks.arcgisonline.com/ArcGIS/rest/services/Loc
ators/TA_Address_NA/GeocodeServer"/>

<esri:SimpleMarkerSymbol id="mySymbol" size="8" style="circle"
color="0xFF0000" alpha="0.5">
<esri:SimpleLineSymbol width="2"/>
</esri:SimpleMarkerSymbol>

<!--Identify Tool -->
<!-- Symbol for where the user clicked -->
<esri:SimpleMarkerSymbol id="clickPtSym" style="x"
    color="0xFF0000" size="12"/>
<!-- Symbol for Identify Result as Point -->

```

```

        <esri:SimpleMarkerSymbol id="smsIdentify" style="diamond"
        color="0x00FF00" size="15"/>
<!-- Symbol for Identify Result as Polygon -->
        <esri:SimpleFillSymbol id="sfsIdentify"/>
<!-- Identify Task -->
        <esri:IdentifyTask id="identifyTask"
        concurrency="last"
        url="/luke/ArcGIS/rest/services/Surfrider/MapServer"/>

<mx:Image source="@Embed('bgtoprepeat2.png')" top="0" left="0" />
<mx:Image source="@Embed('leftlogo.png')" top="47" left="10" />
<mx:Image source="@Embed('rightlogo.png')" top="10" right="10" />

<!-- Set up the Map and it's various components -->
<esri:Map id="myMap" mapClick="mapClickHandler(event)"
        openHandCursorVisible="false"
        left="0" right="0" bottom="0" top="100">
<esri:extent>
        <esri:Extent xmin="-140.076640625" ymin="16.01974609375" xmax="-
        55.70164062500001" ymax="60.53634765625">
        <esri:SpatialReference wkid="4326"/>
        </esri:Extent>
</esri:extent>
        <esri:ArcGISTiledMapServiceLayer
        url="http://server.arcgisonline.com/ArcGIS/rest/services/ESRI_Str
        eetMap_World_2D/MapServer" />
<esri:ArcGISDynamicMapServiceLayer
        id="myDynamicService"
        alpha="0.8"
        url="/luke/ArcGIS/rest/services/Surfrider/MapServer"
        visibleLayers="{defaultLayers}" />
<esri:GraphicsLayer id="myGraphicsLayer" />

<!-- IDENTIFY TOOL -->
</esri:Map>
<mx:ArrayCollection id="defaultLayers">
</mx:ArrayCollection>

        <mx:Panel id="tableView" left="0" bottom="0" layout="absolute"
        height="199" borderColor="#99CCFF" width="900">
                <mx:TextArea id="textArea1" height="20" width="880">
                        <mx:htmlText>
                                <![CDATA[<b>Number of Members (500 max):</b> 0]]>
                        </mx:htmlText>
                </mx:TextArea>

                <mx:Button y="1" label="Export" right="0" height="18"
                click="exportTableToCSV();" />
                <mx:DataGrid id="dgrid1" top="19" left="0" right="0" bottom="0"

                allowMultipleSelection="true"
                itemRollOver="onRollOverEventHandler(event)"
                itemRollOut="onRollOutEventHandler(event)"
                borderColor="#6AA9F6">
                        <mx:columns>
                                <mx:DataGridColumn dataField="MemberID"
                                headerText="Member ID"/>

```

```

        <mx:DataGridColumn dataField="StreetAddress"
        headerText="Street Address"/>
        <mx:DataGridColumn dataField="City"
        headerText="City"/>
        <mx:DataGridColumn dataField="State"
        headerText="State"/>
        <mx:DataGridColumn dataField="ZipCode"
        headerText="Zip Code"/>
        <mx:DataGridColumn dataField="DateJoined"
        headerText="Join Date"/>
        <mx:DataGridColumn dataField="ExpirationDate"
        headerText="Expiration Date"/>
        <mx:DataGridColumn dataField="Chapter"
        headerText="Chapter"/>
        <mx:DataGridColumn dataField="Donations"
        headerText="Donations"/>
        <mx:DataGridColumn dataField="graphic"
        visible="false"/>
    </mx:columns>
</mx:DataGrid>
</mx:Panel>

<mx:Panel id="addressView" title="Find an Address"
borderColor="#99CCFF" height="314" x="335" y="150" visible="false"
width="316">
    <mx:Form width="292" height="216">
        <mx:FormItem label="Street">
            <mx:TextInput width="100%" id="address" text="942 Calle
            Negocio"/>
        </mx:FormItem>
        <mx:FormItem label="City">
            <mx:TextInput width="100%" id="city" text="San Clemente"/>
        </mx:FormItem>
        <mx:FormItem label="State">
            <mx:TextInput width="100" id="state" text="CA"/>
        </mx:FormItem>
        <mx:FormItem label="Zip Code">
            <mx:TextInput width="100" id="zip" text="92673"/>
        </mx:FormItem>
        <mx:FormItem width="255" height="52">
            <mx:Button label="Find Address" click="doFind()"
            textAlign="center"/>
            <mx:Button label="Clear Address" click="doClear()"/>
        </mx:FormItem>
    </mx:Form>
    <mx:Text id="myInfo" width="275" color="#F60707" textAlign="center"
    height="46"/>
</mx:Panel>

<mx:Panel height="272" layout="absolute" id="Layers" title="Layers"
borderColor="#99CCFF" right="0" top="100" width="171">
    <!--mx:CheckBox id="changeLayer1" x="10" y="130" label="Chapters"
    change="changeLayer(changeLayer1, 0)" selected="false"-->
    <mx:CheckBox id="changeLayer2" x="10" y="10" label="Current
    Members" change="changeLayer(changeLayer2, 1)" selected="false"
    borderColor="0xEEEEEE"/>

```



```

<mx:CheckBox id="changeLayer3" x="10" y="70" label="Surf Spots"
change="changeLayer(changeLayer3, 3)" selected="false"/>
<mx:CheckBox id="changeLayer4" x="10" y="40" label="Total
Members" change="changeLayer(changeLayer4, 4)" selected="false"
width="121" fillAlphas="[1.0, 0.4, 1.0, 1.0]"/>
<mx:CheckBox id="changeLayer5" x="10" y="100"
label="Universities" change="changeLayer(changeLayer5, 5)"
selected="false"/>
<mx:CheckBox id="changeLayer6" x="10" y="130" label="Chapter
Areas" change="changeLayer(changeLayer6, 6)" selected="false"/>
<mx:Button id="identifyMode" toggle="true" label="Identify"
selected="false" x="10" y="165" width="125"/>
<mx:Button label="Clear Selection" click="doClearExtent()"
textAlign="center" x="15" y="198"/>
</mx:Panel>

<mx:Panel id="attributeView" title="Search by Attribute"
borderColor="#99CCFF" height="214" width="347" x="196" y="150"
visible="false">
<mx:Form width="326" height="172">
<mx:FormItem label="Member ID">
<mx:TextInput width="198" id="memberid"/>
</mx:FormItem>
<mx:Label text="or find members within a zip code:" width="100%"
textAlign="center"/>
<mx:FormItem label="Zip Code">
<mx:TextInput width="196" id="memberzip"/>
</mx:FormItem>
<mx:FormItem width="277" height="52">
<mx:Button label="Find Member" click="doSearch()"
textAlign="center"/>
<mx:Button label="Clear Selection" click="doClearWhere()"
textAlign="center"/>
</mx:FormItem>
</mx:Form>
</mx:Panel>

<mx:Panel id="bookmarksView" title="Bookmarks" paddingBottom="0"
paddingTop="0" height="277" borderColor="#99CCFF" width="232" x="451"
y="150" visible="false">
<mx:RadioButton label="Contiguous USA" groupName="continent"
change="{rb_change(event)}" height="15"/>
<mx:RadioButton label="California" groupName="continent"
change="{rb_change(event)}" height="15"/>
<mx:RadioButton label="Florida" groupName="continent"
change="{rb_change(event)}" height="15"/>
<mx:RadioButton label="Northeast" groupName="continent"
change="{rb_change(event)}" height="15"/>
<mx:RadioButton label="Alaska" groupName="continent"
change="{rb_change(event)}" height="15"/>
<mx:RadioButton label="Hawaii" groupName="continent"
change="{rb_change(event)}" height="15"/>
<mx:Form backgroundColor="0xEEEEEE" paddingBottom="4"
paddingTop="4" paddingLeft="3" paddingRight="3" height="103"
width="204">
<mx:Label text="Zoom In" width="59" fontWeight="bold"
fontStyle="italic"/>

```

```

        <mx:Label text="press Shift, draw a box" width="151"/>
        <mx:Label text="Zoom Out" width="117" fontWeight="bold"
fontStyle="italic"/>
        <mx:Label text="press Shift + Ctrl, draw a box" width="187"/>
    </mx:Form>
</mx:Panel>

<!-- This is the DataGrid that will be populated with features from the
featureSet -->

<mx:TextArea height="0" width="0" id="log"/>
<mx:TraceTarget level="{LogEventLevel.DEBUG}" includeCategory="true"
includeLevel="true"/>
<!-- The output from TraceTarget is available when running Debug within
Flex Builder -->

<esri:SimpleFillSymbol id="fillSymbol" color="0xCCCCCC" style="solid"
alpha="0.7">
    <esri:SimpleLineSymbol id="lineSymbol1" color="0x0000FF"
style="solid" />
</esri:SimpleFillSymbol>

<esri:SimpleFillSymbol id="clearFillSymbol" color="0xCCCCCC"
style="solid" alpha="0">
    <esri:SimpleLineSymbol id="lineSymbol2" color="0x0000FF"
style="solid" />
</esri:SimpleFillSymbol>

<!--the properties of the query results symbol-->
<esri:SimpleMarkerSymbol id="memberPoint" style="circle" size="8"
color="#051CF6" alpha="0.8">
    <esri:SimpleLineSymbol color="0x000000" width="0" alpha="1"
style="solid"/>
</esri:SimpleMarkerSymbol>

<mx:Panel id="selectionView" title="Search by Selection"
paddingBottom="0" paddingTop="0" height="206" borderColor="#99CCFF"
x="60" y="150" visible="false" width="343">
    <mx:Form width="100%" paddingBottom="4" paddingTop="4" paddingLeft="3"
paddingRight="3">
        <mx:FormItem label="Search by Polygon:" fontWeight="bold" width="289">
        </mx:FormItem>
        <mx:ToggleButtonBar id="tbb"
creationComplete="tbb.selectedIndex = -1"
itemClick="itemClickHandler(event)"
valueCommit="if (tbb.selectedIndex == -1) { drawToolbar.deactivate()
}">
        <mx:dataProvider>
            <mx:Array>
                <mx:String>Polygon</mx:String>
                <mx:String>Freehand Polygon</mx:String>
                <mx:String>Clear Selection</mx:String>
            </mx:Array>
        </mx:dataProvider>
    </mx:ToggleButtonBar>
    </mx:Form>

```

```

<!-- buffer tool-->
<mx:Form backgroundColor="0xEEEEEE" paddingBottom="4" paddingTop="4"
paddingLeft="3" paddingRight="3" height="101" width="312">
  <mx:FormItem label="Search by Buffer:" fontWeight="bold"
width="284" height="20">
  </mx:FormItem>
  <mx:ComboBox id="comboMile" width="161">
  <mx:ArrayCollection>
    <mx:String>Range in Miles</mx:String>
    <mx:String>1</mx:String>
    <mx:String>10</mx:String>
    <mx:String>20</mx:String>
    <mx:String>50</mx:String>
    <mx:String>100</mx:String>
  </mx:ArrayCollection>
  </mx:ComboBox>
  <mx:Button label="Draw Buffer (click map)"
click="drawToolbar.activate(Draw.MAPPOINT)" />
</mx:Form>

</mx:Panel>

<mx:ButtonBar id="controls" left="60" top="120"
itemClick="controlClicked(event);">
  <mx:dataProvider>
    <mx:Array>
      <mx:String>Search by Selection</mx:String>
      <mx:String>Search by Attribute</mx:String>
      <mx:String>Find an Address</mx:String>
      <mx:String>Bookmarks</mx:String>
    </mx:Array>
  </mx:dataProvider>
</mx:ButtonBar>

</mx:Application>

```

Appendix B. Data Table Export Code

The web GIS application allows the user to search the Surfrider Foundation membership dataset and view query results in a data table. This data table can then be exported to a CSV file for further analysis and printing in Microsoft Excel. The ASP.NET programming code used to build the CSV file outside of the application was compiled using Microsoft Visual Studio 2008.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        //set the response type appropriately
        Response.AddHeader("Content-Disposition", "attachment;
filename=export.csv");

        Response.Write(Request.Params["table"]);
        Response.ContentType = "text/csv";
    }
}
```