Development of a street sweeper fleet management system

Ravi Shenkar Seera

University of Nevada, Las Vegas

Follow this and additional works at: https://digitalscholarship.unlv.edu/rtds

Repository Citation

https://digitalscholarship.unlv.edu/rtds/1909

This Thesis is brought to you for free and open access by Digital Scholarship@UNLV. It has been accepted for inclusion in UNLV Retrospective Theses & Dissertations by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.
DEVELOPMENT OF A STREET SWEEPER FLEET MANAGEMENT SYSTEM

by

Ravi Shenkar Seera

Bachelor of Technology
Jawaharlal Nehru Technological University, Kakinada
2003

A thesis submitted in partial fulfillment
of the requirements for the

Master of Science Degree in Engineering
Department of Civil and Environmental Engineering
Howard R. Hughes College of Engineering

Graduate College
University of Nevada, Las Vegas
December 2005

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
The Thesis prepared by

Ravi Shenkar Seera

Entitled

DEVELOPMENT OF A STREET SWEEPER FLEET MANAGEMENT SYSTEM

is approved in partial fulfillment of the requirements for the degree of

Master of Science in Civil Engineering

Examination Committee Chair

Dean of the Graduate College

Examination Committee Member

Examination Committee Member

Graduate College Faculty Representative
ABSTRACT

Development of a Street Sweeper Fleet Management System

by

Ravi Shenkar Seera

Dr. Shashi Nambisan, Examination Committee Chair
Professor of Civil Engineering
University of Nevada, Las Vegas

Street sweeping is a vital public service that not only facilitates traffic flow and improves the appearance of the neighborhoods. It also helps remove debris and dust from the roadways thus enhancing drainage and reducing air pollution. The need for a street sweeper fleet management system is to help improve the efficiency and effectiveness of street sweeping programs. In this research, an application is developed for the management of street sweepers with the aid of a mapping system interface to query and analyze the data collected using Global Positioning System (GPS) devices installed in the sweeper trucks. Based on the functional needs of the system, an architecture is first developed for the system. Then, existing software and hardware components are used as the basis to design the system. Customized interfaces and processes are developed to integrate various system components. The resulting system is expected to provide accurate records of machine activity, improving fleet cost management and vehicle productivity. The system has the capabilities to generate reports and graphical summaries of various aspects of resource allocation and utilization. Examples of such reporting capabilities include records of machine usage and activity including dates, exact times and locations that vehicles start, drive, idle and stop as well
as the use of various events such as even when and where the brooms and dust suppression systems were engaged. These types of fleet usage data can also greatly reduce manual record keeping and unauthorized vehicle use, minimize billing errors, and improve route efficiencies. Other benefits include accountability of work performed, increased efficiency of resource utilization, better records for legal purposes, optimizing routes and improving service. The street sweeping program managed by the Clark County Department of Public Works, Nevada is used to demonstrate the development and use of the system.
# TABLE OF CONTENTS

ABSTRACT .................................................................................................................................. iii

LIST OF FIGURES ................................................................................................................... vii

LIST OF TABLES .................................................................................................................... viii

ACKNOWLEDGEMENTS ....................................................................................................... ix

CHAPTER 1  INTRODUCTION ............................................................................................ 1
  Background .............................................................................................................................. 2
  Science of GPS Technology ................................................................................................. 3
  Web-based Fleet Management System ................................................................................ 4
  Objective .................................................................................................................................. 5
  Outline of Thesis ..................................................................................................................... 6

CHAPTER 2 LITERATURE REVIEW ................................................................................ 7
  History of Sweepers ............................................................................................................... 7
  Trends in Sweeper Technology ............................................................................................ 8
    Parking Lot Sweepers ........................................................................................................ 10
    Airport Sweepers ............................................................................................................... 10
    Sidewalk Sweepers and Other Specialty Sweeper Types ............................................. 11
  Internet and the WWW ........................................................................................................ 13
  Current State of Spatial Information Provision .............................................................. 14

CHAPTER 3 MANAGEMENT TOOLS AND TECHNOLOGIES ....................................... 17
  Review of Technologies ..................................................................................................... 17
  HTML Forms ....................................................................................................................... 19
    Java ................................................................................................................................... 19
    ActiveX ............................................................................................................................ 20
    Plug-ins ............................................................................................................................ 21
  GIS on the Web .................................................................................................................... 22
  GIS Vendor Products ......................................................................................................... 23
  Assessment Criteria ........................................................................................................... 24
  Autodesk MapGuide ........................................................................................................... 25
    Autodesk MapGuide Components ............................................................................... 25
  ESRI ArcIMS .................................................................................................................. 27
    ESRI ArcIMS Components ......................................................................................... 28
  Discussion of the components of the Major Products .................................................... 30
  Maps, Data and Viewers ................................................................................................. 31

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>A Street Sweeping Vehicle in 2005</td>
<td>9</td>
</tr>
<tr>
<td>Figure 2</td>
<td>A sidewalk sweeper (Litter Gitter) in operation</td>
<td>12</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Sweepers equipped with GPS</td>
<td>13</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Basic Client Server WWW set up</td>
<td>18</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Assessment Components</td>
<td>24</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Autodesk MapGuide Software Architecture (Source: ESRI)</td>
<td>27</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Architecture of ArcIMS Technology (Source ESRI)</td>
<td>29</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Autodesk MapGuide HTML Viewer – plug-in required</td>
<td>33</td>
</tr>
<tr>
<td>Figure 9</td>
<td>ESRI ArcIMS HTML Viewer – No plug-in required (Source: ESRI)</td>
<td>34</td>
</tr>
<tr>
<td>Figure 10</td>
<td>System Architecture</td>
<td>44</td>
</tr>
<tr>
<td>Figure 11</td>
<td>Tables present in the SQL database</td>
<td>48</td>
</tr>
<tr>
<td>Figure 12</td>
<td>Contents in the Tables of the SQL database</td>
<td>49</td>
</tr>
<tr>
<td>Figure 13</td>
<td>Single table containing all the information</td>
<td>50</td>
</tr>
<tr>
<td>Figure 14</td>
<td>First Step in creating a ODBC connection – setup of User DSN</td>
<td>51</td>
</tr>
<tr>
<td>Figure 15</td>
<td>Second Step - Select SQL Server in the driver list</td>
<td>52</td>
</tr>
<tr>
<td>Figure 16</td>
<td>Third Step - Select the SQL server from a list to connect</td>
<td>52</td>
</tr>
<tr>
<td>Figure 17</td>
<td>Fourth Step: Connect to the SQL Server with the administrator login</td>
<td>53</td>
</tr>
<tr>
<td>Figure 18</td>
<td>Fifth Step: Select the database to connect</td>
<td>53</td>
</tr>
<tr>
<td>Figure 19</td>
<td>Sixth Step – Test the Data Source for connection</td>
<td>54</td>
</tr>
<tr>
<td>Figure 20</td>
<td>Typical layout of an Internet Information Server (IIS)</td>
<td>56</td>
</tr>
<tr>
<td>Figure 21</td>
<td>Layout of the services window</td>
<td>57</td>
</tr>
<tr>
<td>Figure 22</td>
<td>Setup local path for the file to be read</td>
<td>58</td>
</tr>
<tr>
<td>Figure 23</td>
<td>Enable default document for viewing</td>
<td>58</td>
</tr>
<tr>
<td>Figure 24</td>
<td>Define users/groups that will have access to map server data sources</td>
<td>59</td>
</tr>
<tr>
<td>Figure 25</td>
<td>Web-based Tool</td>
<td>61</td>
</tr>
<tr>
<td>Figure 26</td>
<td>Lay out of the Map Window</td>
<td>62</td>
</tr>
<tr>
<td>Figure 27</td>
<td>Lay out of pan controls</td>
<td>63</td>
</tr>
<tr>
<td>Figure 28</td>
<td>Lay out of Map Controls</td>
<td>63</td>
</tr>
<tr>
<td>Figure 29</td>
<td>Lay out of the query page</td>
<td>68</td>
</tr>
<tr>
<td>Figure 30</td>
<td>Select vehicle to be displayed</td>
<td>69</td>
</tr>
<tr>
<td>Figure 31</td>
<td>Displayed results on the ASP page</td>
<td>70</td>
</tr>
<tr>
<td>Figure 32</td>
<td>Map Window is reloaded to refresh the data source</td>
<td>71</td>
</tr>
<tr>
<td>Figure 33</td>
<td>Desired results displayed spatially on the map</td>
<td>72</td>
</tr>
<tr>
<td>Figure 34</td>
<td>Route traced by a Sweeper truck</td>
<td>73</td>
</tr>
<tr>
<td>Figure 35</td>
<td>Typical route followed by a sweeper in a neighborhood</td>
<td>74</td>
</tr>
<tr>
<td>Figure 36</td>
<td>Maptips displaying the status as Hopper Dump</td>
<td>75</td>
</tr>
</tbody>
</table>
ACKNOWLEDGEMENTS

I would like to express my sincere acknowledgements to Dr. Shashi Nambisan for his help and guidance in completion of my thesis and throughout my graduate program. I would like to extend my acknowledgements to my committee member, Dr. Srinivas Pulugurtha for his valuable guidance and support both in the academic and the personal front. I would like to thank my committee members Dr. Mohamed Kaseko and Dr. Laxmi Gewali for their support, patience and effort in reviewing my thesis.

I extend my gratitude to the Clark County Department of Public Works and University of Nevada Las Vegas Applied Research Initiative (UNLV-ARI) for their financial support. I thank the Transportation Research Center (TRC) for providing me financial support throughout my coursework and for providing timely infrastructure to finish my thesis.

Finally, I would like to thank my family and friends, without whose support I would not have completed my thesis.
CHAPTER 1

INTRODUCTION

Street sweeping is a vital public service that not only enhances safety but also improves aesthetics in the neighborhoods. Street sweeping involves the use of specialized and costly equipment to remove litter, loose gravel, soil, pet waste, vehicle debris, pollutants, dust, de-icing chemicals, and industrial debris from road surfaces. Street sweeping equipment can consist of a truck or truck-like vehicle equipped with multiple brushes, pick-up deflector, holding bin, water sprayer, vacuum nozzle and filter, or a combination of some or all of these features. When done regularly, street sweeping can remove 50-90% of street pollutants that potentially can enter surface water through storm sewers. Street sweepers remove debris from the roadways and also make road surfaces less slippery in light rains. They improve aesthetics by removing litter, and control pollutants which can be captured by the equipment.

An effective fleet management system can improve the value and productivity of these street sweeping programs. Such a system would enable one to track the position and the functional time of the sweeping trucks. It also helps to reduce overhead cost of fleet ownership and to monitor vehicular events and activities such as speeding, unauthorized usage, asset under-utilization, unnecessary idling, time at the job site and miles driven.
Background

The major function of a fleet management system is to monitor and record system components and functions and when and where they occur. The system reports provide accurate records of machine usage and activity including dates, exact times and locations that vehicles start, drive, idle and stop. The system will collect, store and display information such as vehicle location, stop time, current speed and heading, current location and thematic views of vehicle status (stopped, moving, on-station, etc.) and vehicle paths over a specified period of time for one or more vehicles. Reports on mileage, maximum speed, and specific vehicle input functions as well as pump hours provide necessary information for preventive maintenance scheduling. These types of fleet usage data can also eliminate manual record keeping and unauthorized vehicle use, minimize billing errors, and improve route efficiencies.

The system provides accurate records of machine activity, improving fleet cost management and vehicle productivity. The major benefits include accountability of work performed, efficient use of resources and reduced liability issues like less missed completions and better records for administrative, legal and management purposes. Street sweeping, street blading and snow plowing are similar in applications, and are important topics in the area of public works maintenance of streets. The use of Global Positioning Systems (GPS), has led to a more efficient way to plan, track and monitor operations such as street sweeping and provide advanced customer service at the same time.

With the aid of an effective management systems one can analyze to a great level of detail the routes and actions of the sweeper. Previously, maintenance officials had to search for sweeper’s broom tracks to follow up on resident’s claims that city crews had
ignored their neighborhoods. The officials had to rely on radio contact with their employees to ensure that they were doing their jobs properly. Now, with the aid of system the officials can view a map of the routes serviced by the entire fleet or zoom in to a small area on the map served by a single vehicle.

Science of GPS Technology

The United States Department of Defense first launched a Global Positioning Systems (GPS) satellite in 1978 and achieved a full constellation of 24 satellites in 1994, which the U.S. Government has named Navstar. Today, GPS is used for both civil and military purposes and is controlled by a joint civilian/military executive board of the U.S. Government. The U.S. Air Force on behalf of all users maintains the system.

GPS relies on three components: a constellation of satellites (currently 27) orbiting at about 20,000 km (11,500 miles) above the earth’s surface which transmit ranging signals on two frequencies in the microwave part of the radio spectrum, a control segment which maintains GPS through a system of ground monitor stations and satellite upload facilities, and user receivers (civil and military). In simple terms, the GPS satellites transmit signals to the equipment on the ground.

More specifically, the signals contain a pseudorandom code that identifies which satellite is transmitting the information, ephemeris data that contain information about the status of the satellite and the current date and time, and almanac data that tells the receiver where each satellite should be at any time throughout the day. The receivers use this data to determine how long it takes the signals to travel from the satellite to the receiver. The receiver then uses the speed of light (about 300,000 km per second and
about the same speed at which radio waves travel) to calculate the satellites’ location. By using the exact locations of four or more satellites, the receivers can determine their own latitude, longitude, and elevation. This process of determining a position from measurements of distances is known as trilateration (as opposed to triangulation, which is based on the measurement of angles).

When the GPS was first created, the U.S. Government inserted timing errors into GPS transmissions to limit the accuracy of non-military GPS receivers to about 100 meters. This was known as Selective Availability and was eliminated in the May of 2000. Today, the accuracy of a position determined with GPS depends on the type of receiver, but most hand-held GPS units have about 10 to 20 meter accuracy. If an additional receiver fixed at a nearby location is used, it is possible to obtain much higher accuracy through a method called Differential GPS (DGPS). The accuracy of the GPS is not affected by any weather conditions.

Web-based Fleet Management System

Fleet management encompasses many functions and tasks, from purchasing or leasing the fleet vehicles, to maintenance, routing planning, service and delivery management, dispatch management and much more. One of the fastest growing areas in fleet management is GPS Vehicle Location or GPS Fleet Tracking. These systems enable fleet managers and business owners to better manage their fleet vehicles by knowing where they are, where they have been and plan for future routes and schedules.

Fleet management systems provide significant amounts of data about the vehicles and their travel patterns. GPS fleet tracking is many times referred to as Fleet Management,
although the term Fleet Management encompasses many more functions and tasks. There are two kinds of GPS Vehicle Tracking Systems – Active and Passive. Active systems utilize wireless networks that allow one to receive data from all of the vehicles in one fleet and view the location of any or all of your vehicles in near real-time. Passive systems are the systems in which the data transfer from vehicle to office is done at the end of the day.

**Objective**

The objective of the thesis was to develop an application enabled with Global Positioning Systems (GPS) for the management of street sweepers with the aid of a mapping system interface to query and analyze the data. This GPS application was developed based on the primary need to track the street sweeping process down to an individual street section. The Street Sweeping Application utilizes an on-board computer application called Fleet Manager 2002 (FM2002). A GPS receiver allows the on-board system to time stamp exact vehicle location and provide a breadcrumb trail for mapping programs. At the same time as it captures GPS points, the on-board computer gathers driver and vehicle information and saves the data. All the information is downloaded at the end of the shift via a trip extraction key and loaded into the FM2002 software, which is loaded on the desktop computer, and data are stored in a SQL database.

One of the primary benefits from the new application is the ability to respond to citizen and other customers of the Street Division with accurate sweeping information down to the exact street segment. GPS systems are made up of two parts: the hardware consisting of a black box for each sweeper that receives the satellite signals and the
software that interprets the signals. In passive mode, this GPS box collects and downloads data into the office computer at the end of a sweeping shift using a serial port on the side of the box via a cable. With newer technology equipment, this process may also be performed via wireless modem.

The research topic focuses on developing a street sweeper fleet management system to help integrate work & asset management programs. The need for a street sweeper fleet management system is to help operators, supervisors and managers to analyze the deployment and use of the equipment. For example, queries of interest include where and when the sweepers were on the job, and for how long; the roads that were serviced and even when and where the brooms and dust suppression systems were engaged. Customized reporting and easy to read graphs will enhance the power of such a system. The main aim of the sweeper management system is to improve the efficiency and reduce the overhead cost of fleet ownership by identifying and monitoring vehicular events and activities such as speeding, unauthorized use, asset under-utilization, unnecessary idling, time at the job site, miles driven and more.

Outline of Thesis

A review of the literature on street sweepers and the current state of the practice in sweeper technology and fleet management is presented in chapter 2. Chapter 3 presents the management tools involved in developing the system and the various software programs and their features. Chapter 4 deals with the functional needs of the system. In chapter 5, the system design and development are documented. A summary of the efforts and outcomes of this thesis and recommendations are provided in chapter 6.
The first chapter presents a brief history of street sweepers and some recent trends adopted in sweeper technology. This chapter deals with a review of the literature of the various Internet technologies and GIS technologies which are in use at present for street sweeper management activities.

History of Sweepers

Concerns over cleaning and pollution directly led to other innovations eventually producing the modern street sweeper. Street sweepers evolved out of the need to clean dirt and grime off the newly paved roadways. Up until the advent of the mechanized street sweeper, individuals had been hired to rove streets and collect animal waste and other debris in receptacles on wheels.

However, with increased industrialization, the mechanized street sweeper was developed in the 1910s. One of the first individuals to build such a device was John Murphy of Elgin, Illinois. The town of Elgin provides the name of the enduring sweeping manufacturing company that is still in operation today. Murphy designed a mechanical sweeper that was highly maneuverable and easy to maintain. It was a three-wheeled design with two wheels on the front and one in the back. The waste collection hopper was
in the front of the vehicle and the engine was under and in front of the driver. The front hopper allowed easy loading and unloading of the collected debris. One of the advantages of the three-wheel design was that it allowed the vehicle to move easily around vehicles and horses that might have been present on the streets in the period when the sweeper was developed.

Street sweeping operations have gradually evolved from largely manual and predominantly local operations involving people pushing hand brooms along streets, to encompass large-scale mechanized fleets of street sweepers managed by municipalities. The original goals of street sweeping that focused attention on sanitary conditions and waste removal have given way to broader concerns of storm water pollution reduction, urban cleanliness, and aesthetic considerations. Indeed, sanitary conditions, at least in modern cities, are no longer a significant issue because of relatively efficient sewage systems that quickly remove and treat wastes. However, this is not true for cities in some less wealthy nations, where raw sewage and untreated wastes can still be found running along streets and in drainage ditches in both commercial and residential districts.

Trends in Sweeper Technology

New types of sweepers come on to the market every few years largely building upon the growing need to reduce storm water pollution. This trend will undoubtedly continue. As more is known, therefore, it is expected that more refined sweeping technologies will develop following on from the recent developments in combination sweepers. On the other hand, it would be a service to communities for manufacturers of sweepers to design their equipment so that it can be refined for the specific needs of individual communities.
It would be useful for the street sweeper manufacturers of environmental consultants to develop an evaluative tool to determine the site-specific street sweeping and waste management needs of individual communities. There is an amazing array of other sweeper types available on the market. Some are highly specialized and others are very common, although, like most sweepers, seldom seen. An illustration of a street sweeper used currently in Clark County, Nevada is shown in Figure 1.

Figure 1  A Street Sweeping Vehicle in 2005
Parking Lot Sweepers

Like roads, parking lots are often swept as part of a standard maintenance program. Paved surfaces that provide parking for motorized vehicles present a special challenge for street sweepers, for they must be swept when vehicles are not parked. This is sometimes a challenge for organizations that have twenty-four hour staffing, such as hospitals or airports. In addition, leaking vehicle parts produce particular pollutants that are of concern to waste managers. Parking lots, therefore, tend to concentrate some pollutants and can lead to substantial contamination.

While many parking lots are swept on a regular basis, it is the size of the lot that usually determines the sweeper type used. Many small parking lots that service local businesses are swept by hand or managed by landscape companies. There are a number of different types of mechanical sweepers that may be used to clean parking lots. Often, particularly for smaller lots, the sweeping is subcontracted to a sweeping and parking lot maintenance company. There are three types of these small sweepers:

- Parking lot sweepers that can be driven from place to place on public roads
- Parking lot sweepers that are too small to be driven from place to place on public roads
- Parking lot sweepers that can be temporarily mounted on a truck or other heavy-duty machinery

Airport Sweepers

Airport sweepers generally are effective at removing glycol, a chemical found in deicing fluid from airport runways and gate locations. These machines spray an emulsifier onto the runway that helps to remove the glycol from the concrete surface and
puts it into solution. After this, the material is vacuumed under high suction, and the material stored in a tank until it can be disposed of safely. This sweeper type is very different from other types of sediment removal vehicles. Instead of the waste being concentrated in a load of loose sediment, it is concentrated in liquid form, which requires special waste management.

Sidewalk Sweepers and Other Specialty Sweeper Types

There are a wide variety of sweeper types available for specific uses. A number of different hand-held varieties are used in places such as theme parks, zoos, or factories. These sweepers may be powered by gasoline or electricity. In most settings, these sweepers are for litter management, where other waste handling is not a problem. When used in industrial settings, though, waste management may be problematic. There are also some sweepers that are designed specifically for construction sites. These sweepers pick up debris of a size and weight that other sweepers are unable to remove. Figure 2 shows an example of a sweeper used on sidewalks in Clark County, Nevada.
In addition to functional enhancements, street sweepers have also incorporated information and communications systems. For example, they include on board diagnostic systems that record various aspects of the sweepers operation such as speed, use of brushes, vacuum units, and water. Further, GPS based automatic vehicle location systems have also been integrated with modern street sweepers. An illustration of such a sweeper with a GPS unit is shown in Figure 3.
The rise in popularity of the Internet on the back of the World Wide Web (WWW) has caught more than a few unawares. The pace of technical development in this area has been such that a far greater range of applications and uses are now possible, and are continually appearing. It would be appropriate to note that by the time this thesis is published the WWW industry will have progressed significantly.

In its early years, the WWW was marked by its lack of complexity and ability to facilitate more complicated operations, such as those complementary to online Geographic Information Systems (GIS) applications. However, these early problems were well known and have been documented elsewhere, such as by Li (1996).
Commercial vendors have been quick to realize the potential of the WWW and have made efforts to overcome these problems and make the provision of online spatial information or mapping systems possible.

The World Wide Web provides an attractive and presentable medium for geospatial information. At the present time it is possible to set up a GIS enabled user interface from within a web browser and allow users to perform queries to some remote database. The geospatial data themselves reside somewhere on the remote database, which is often a pre-existing database or GIS that has been extended to allow users on the WWW to query it. The result, in many cases, is a lightweight front end linked to a GIS-like back end that produces an information tool that is powerful yet relatively intuitive to use for a variety of purposes. The Internet at this point in time specializes in providing functional yet lightweight GIS like tools that may or may not look a GIS at all, providing a level of information distribution that requires little in technical capability to extract.

Current State of Spatial Information Provision

GIS is becoming a ‘popular commodity’, but issues such as proper use of data and the quality of data remain (Barr, 1996). Perhaps it is essential to either educate the mass market on how to use such ‘commodities’ properly, or to make it so that the software prevents inappropriate use or places warnings or guidelines on the use of data. The pace of development in this industry is considerable, and web browsers that can handle these concerns; particularly those to do with in-line browser components or security for example, are being continuously developed.
The GIS industry has moved just as quickly to provide web access extensions for their GIS software, as well as by developing the interfaces and linkages to allow users to interact with the GIS through a browser. In these cases, the linkages are becoming transparent, and the user does not know or need to know how the data are served to them. The cadastral industry has also been quick to provide cadastral data for use in such systems, but the pace at which the system has been undergoing reforms to allow it to provide data in the best manner and at a quality that is desired by users, is slower.

The information industry that results from the amalgamation of such disciplines has the potential to provide great benefits to the community through the range of uses that can be derived from it. The ease of use and the transparency of the underlying technologies will provide a potentially global audience with the capability to retrieve information that is of great use to them. The resulting product will not be a GIS, but more of a generic information tool that has more general use, including those of a spatial nature. The GIS will lose the ‘G’ that identifies specific technology oriented systems and become more user-oriented Information Systems. The development of the WWW as an information tool that can seamlessly and easily handle geo-referenced information as well as it handles static information is one that will improve, even more, its usefulness to the world.

GIS vendors are seeking to satisfy the needs of a mass market, in the process producing a range of products that are easier to use by casual users. The value of this idea is demonstrated by the fact that some government organizations have started developing spatial data infrastructures, recognizing that geo-information is being demanded more. The Internet has developed with great speed, and now is capable of delivering GIS
functionality from within the WWW browser. There are some limitations with regard to network data transmission bandwidth, and some issues including security that also need to be taken into account. However, it is generally possible to create information services that seamlessly integrate spatial and non-spatial data that can be used for a variety of purposes, including public information and also to support business.

Today, large amounts of information are suddenly becoming available to large audiences that may otherwise not have had such access. The influence of the user cannot be underestimated either, as they are the ones driving the commercial organizations to develop these technologies, and ultimately they will benefit the most from them. It is unclear what effect that large quantities of readily accessible data will have on society, but it is possible that it could considerably impact the way with which land is perceived and dealt.

The review of the literature identifies technological advancements both from the street sweeper equipment perspective and the information systems equipment perspective. It is clear that incorporating some of the features and capabilities offered by the information technologies could help enhance the effectiveness and efficiency of managing street sweepers. This is subject worthy of further exploration, and thus it is the focus of this research.
CHAPTER 3

MANAGEMENT TOOLS AND TECHNOLOGIES

This chapter provides an in depth study of each of the technologies that comprise an Internet accessible database system. The study identifies for each, the advantages, disadvantages, and an indication of what roles they would be suitable for in a generic context. Also, in each section, the GPS context is briefly considered, given a fleet based model for provision of GPS data over the Internet. In some cases reference is made to a web-based fleet management system, to help highlight issues which are relevant to particular technologies.

Review of Technologies

The advent of plug-in, Java and ActiveX technologies have provided World Wide Web (WWW) administrators a capability that has yet to be fully tested without the limits of network bandwidth. This chapter attempts to explain what technologies are best suited to the provision of web-based fleet management system. Many of the major GIS vendors have released a variety of products designed to extend Internet functionality to their existing GIS products. The Internet has experienced tremendous growth in the last few years, and will continue to do so as more and more services become available.
Conversely, as more individuals and organizations use the Internet, demand will bring more services online. The Internet has proved itself as a data delivery and viewing medium (Wilson, 1997), and as such, it could become a key component to any spatial data viewer and update service. There are a variety of tools and technologies that could implement a data viewing and update service on the WWW. Each has strengths and weaknesses, and some are more suitable than others in particular situations. A typical WWW user/server connection is illustrated in Figure 4. The Static Files, Java Applet / ActiveX, HTML forms, and also plug-ins apply to interface technologies, while Common Gateway interface (CGI) and interoperability of systems are devoted to providing access to GIS and server side databases. These issues of the available security technologies are addressed in the following sections.

![Figure 4 Basic Client Server WWW set up](image)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
HTML Forms

HTML forms provide only for single click operations on images, so that rectangular areas cannot be chosen or selected (Fortunati, 1996). Such capability is important for many GIS operations. The functionality of the CGI program and the underlying GIS in this respect is reliant on the capability of HTML forms, which were never designed for spatial or graphical information requests. The simplistic nature of the HTML forms interface is a ‘bottleneck’ and thus prevents easy to use, powerful GIS functionality (Li, 1996). The advantage is that forms are simpler and faster to implement than the alternatives Java and ActiveX (Bertazzon and Waters, 1996).

The WWW (the HTTP protocol and, as such, browser and server software) provides for only two different image types (GIF and JPEG), both of which are raster based. Mapping outputs, which are commonly vector based and need to be converted to raster format for display in a browser, lose the manipulative power that vector information has. The WWW can only provide for vector display through Java / ActiveX or through browser plug-ins. The lack of capability for rapid, multiple query operations on mainly vector data means that HTML forms would not be ideal for an online GIS style application.

Java

Java was created by Sun Microsystems in 1991 as an attempt to create a platform independent programming language, but was developed further to make it useable through the WWW. Java is a C++ like object-oriented programming language that employs partial compilation and machine translation to achieve platform independence.
Because it is translated, Java is relatively slow when compared to other fully compiled programming languages (Chappell, 1996).

From a security point of view, Java is ideal as security measures were built into it from its inception. Java Applets are Java applications developed specifically for running inside WWW browsers on the user's machine. Java Applets are forbidden access to the user's hard disk and other sensitive areas, making it impossible for a malicious Applet to attack the user's computer. This is a compromise as there are times when it is ideal to write information to the user's computer. Future implementations of Java intend to relax this restriction in certain situations and move the security measures into server and Applet authentication. The leading browsers have introduced these features into their own latest browser implementations in order to allow this fine-grained access control to Applets (Wallach et al., 1997). Java is a completely new language, and as such, its use is still maturing.

More complex functionality can be achieved through the browser using Remote Procedure Interface calls (Java) to a server GIS (Parent, 1997). A detailed appraisal of Java used in integrating GIS to the Internet may be found in Wang (1997). An application providing all data viewing functions could be constructed, one which is able to directly interact with a server GIS (or database software), or, interact with a CGI program that then performs a similar role.

ActiveX

ActiveX is the development of Microsoft's Object Linking and Exchange (OLE) technology for sharing objects between applications, with extensions for use on networks including the Internet. ActiveX Controls are pieces of precompiled binary code that run
on the user's computer, offering performance advantages over Java. This means, however, that ActiveX controls are currently only operable on Microsoft Windows platforms. Security was not one of the original design considerations in OLE, and ActiveX, its successor, suffers as a result. The Microsoft response has been to make use of server authentication using digital certificates to prevent users loading untrusted programs and the encouragement of trust mechanisms that monitor which Internet sites are allowed to do what on a user's computer. On the other hand, the lack of security means that ActiveX programs can provide functionality (for instance, access the user's computers hard drive or memory stacks) in a way that isn’t easily possible in Java.

The big advantage of ActiveX is that it capitalizes on the large base of Windows programmers that exist today. Existing applications can be converted to ActiveX controls without embracing new programming languages, which in turn can potentially lower development costs. ActiveX controls are suited to the PC LAN environment within departments and organizations behind firewalls, where security concerns are not so great, and where Windows based personal computers are more widespread.

Plug-ins

Plug-ins are applications that provide extra functionality to WWW browsers and are an alternative to Java and ActiveX for providing full functional user interfaces. When a browser detects a particular type of content in the page that it is downloading, it redirects that content to the plug-in, which is then capable of dealing with it as required. The plug-in is used extensively in other disciplines providing browsers the capability of presenting multimedia over the WWW. A drawback is that it has to be installed before trying to download pages with special content, and the software resides on the user's machine.
afterwards as well. Another problem is that there are many different plug-ins to cater to the many different types of content that are now available on the WWW. It is a considerable task to have them all installed and is impractical for the average user because of the non-trivial system space they consume. The advantage is that if a type of special content is common, then having the plug-in already loaded prevents the download time associated with Applets and ActiveX components.

Plug-ins, however, are platform dependent and browser dependant, which increases development costs and limits deployment across different platforms. Because the browser starts the plug-in when a particular type of content is loaded, that content will typically reside in the publicly accessible space on the WWW server. This could present security problems for data from commercial databases. However, if the plug-in software was smart enough to retrieve data from more protected areas, this problem could be overcome in most cases.

GIS on the Web

Distributing GIS data over a web-based network is a powerful method for effective communication. One of the strengths of web-based GIS is that users can view GIS data using an inexpensive, standard Internet browser. With a desktop GIS an user typically must purchase, install, and learn how to use a general GIS software tool to load, manipulate, query, and analyze the data. Conversely, a common difficulty associated with web-based GIS is the variable and sometimes limited bandwidth for data flow between the data server and user. As a result, web-based GIS must be highly scalable, successful applications take advantage of networks with high bandwidth while working efficiently

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
to avoid problems with slow networks and low bandwidth. Therefore, when choosing a web-based GIS, scalability is essential. The demands placed on GIS solutions today show a great need for high performance and scalable solutions designed and built specifically for Internet applications.

GIS Vendor Products

The appearance of the WWW as a medium for data presentation, distribution and communication, with its immediate popularity, was a somewhat surprising development when it appeared in the early 1990s. Many commercial vendors provided add on software patches to enable their products to produce WWW useable files leading to more integrated software in subsequent generations.

The GIS industry is another example of this move, with a flurry of WWW products designed to present GIS data on the WWW. For the most part, they included software that converted output from native GIS into WWW content destined for WWW pages, allowing the use of existing operational GIS. Subsequent generations saw refinement of these packages into stand-alone products, providing more mature services than earlier implementations.

This review will focus on the components that comprise GIS for the WWW systems. The major parts of these systems will be assessed on their suitability for satisfying an online fleet management system, and for the types of components that they have. A result of this is a perspective on the major technological components that have been incorporated by commercial vendors, possibly leading to a justification for the use of particular technologies / techniques in the online model.
Assessment Criteria

By examining certain components of each product or suite of products, it should be possible to see how typical systems functions. The components examined are what could be construed to be the major components of any online database system. The key components are listed next and their relationships are illustrated in Figure 5:

- User Interface, including supported platforms, interface methodology, etc;
- WWW Server, including any extensions to normal Hypertext Transfer Protocol (HTTP) WWW servers, Common Gateway Interface (CGI) etc.;
- Data Server, platforms, data formats, restrictions etc.;
- Overall Comment, including miscellaneous comments.

Figure 5  Assessment Components
Autodesk MapGuide

Autodesk MapGuide® software helps to develop, manage, and distribute GIS and design applications on the Internet or an intranet, broadening one’s access to mission-critical geospatial and digital design data. Autodesk MapGuide is one of the leading solutions for web-based GIS and distributing maps online because of its ease of use with map authoring, highly scalable server built for network environments, and customizable viewer application programming interface (API).

Autodesk was the first major software company to bring dynamic, vector-based, and interactive online maps to the consumer market and has been developing web-based GIS software since the mid-1990s. From the beginning, the Autodesk MapGuide architecture was designed expressly for Internet and intranet applications. Autodesk MapGuide is a full-featured Internet GIS authoring and viewing environment, complete with an API for application-specific customization. In addition, because of Autodesk’s focus on data integration in their GIS product line, Autodesk MapGuide can handle data from a variety of existing formats (including SHP, MIF/MID, DGN, DWG, and ESRI coverages), which can be brought into Autodesk MapGuide from different servers and then published to the Web.

Autodesk MapGuide LiteView enables the Autodesk MapGuide Server to display maps in a browser without a plug-in. LiteView is a Java program that runs as a servlet and converts Autodesk MapGuide MWF files into a PNG file.

Autodesk MapGuide Components

Autodesk MapGuide consists of three major components that were developed for distributed network environments. These three components work together with a web
server to serve dynamic maps to a web browser. A few other components are also available for additional functionality and customization.

1. Autodesk MapGuide Server: Handles requests from a viewer and delivers the appropriate data.

2. Autodesk MapGuide Author: Creates the map (saved as an MWF file), which is then embedded into a web page. All of the map’s properties (color, line style, layers accessible at different map scales, and more) and viewer functionality are defined and created in the Autodesk MapGuide Author. (Alternatively, users can author maps with Autodesk® Envision, a separate companion desktop product for Autodesk® Land Desktop, Autodesk Map™ and Autodesk MapGuide.)

3. Autodesk MapGuide Viewers: The following four viewers are available for Autodesk MapGuide:
   - Plug-in for Netscape®
   - ActiveX® Control for Microsoft Internet Explorer
   - Java™ Viewer for Sun® and Macintosh® operating systems
   - Autodesk MapGuide LiteView—no plug-in required.

Other components include SDF Loader, SDF COM Toolkit, Dynamic Authoring Toolkit, Data Provider for SHP (allows for direct Autodesk MapGuide reading of SHP files), Data Provider for Oracle® Spatial (allows Autodesk MapGuide to read directly from an Oracle Spatial or Oracle Locator database), Raster Workshop, and Symbol Manager. The product architecture of Autodesk MapGuide is shown in Figure 6.
ArcIMS is an ESRI product, offers a solution for delivering dynamic maps and GIS data and services via the web. It provides a highly scalable framework for GIS Web publishing that meets the needs of corporate Intranets and demands of worldwide Internet access. ArcIMS services can be used by a wide range of clients including custom Web applications, the ArcGIS Desktop, and mobile and wireless devices. Using ArcIMS, city and local governments, businesses, and other organizations worldwide publish, discover, and share geospatial information.

ArcIMS is a server-based product that provides a scalable framework for distributing GIS services and data over the Web. ArcIMS provides Web publishing of GIS maps, data, and metadata for access by many users both inside the organization and outside on
the World Wide Web. ArcIMS enables Web sites to serve GIS data, interactive maps, metadata catalogs, and focused GIS applications.

ArcIMS users access these services through their Web browsers using HTML or Java applications that are included with ArcIMS. In addition, ArcIMS services can be accessed using many different clients including ArcGIS Desktop, custom applications created using ArcGIS Engine, ArcReader®, ArcPad®, ArcGIS Server, MapObjects—Java Edition, and a wide variety of mobile and wireless devices.

ESRI ArcIMS Components

ArcIMS consists of five major components that interact with each other to enable users to view and query GIS data with an Internet browser. Additional components provide different features for customization:

1. ArcIMS Spatial Server: Processes requests for maps and attribute information.
2. ArcIMS Application Server: Written in Java, this component tracks client requests for information and distributes them to the appropriate ArcIMS Spatial Server.
3. ArcIMS Application Server Connectors: Connects the web server to the ArcIMS application server. Any of the four connectors can be used to translate client requests into ArcXML. (ArcIMS Servlet Connector uses ArcXML, ColdFusion Connector translates Macromedia® ColdFusion® into ArcXML, ActiveX Connector translates ASP or VB into ArcXML and ArcIMS Java Connector uses JSP).
4. ArcIMS Manager: A web wrapper that combines three separate ArcIMS applications (ArcIMS Author, ArcIMS Designer, ArcIMS Administrator) into one
user interface. Because of its web framework, the ArcIMS Manager can be used remotely. The three individual applications can also be used separately on a local machine.

5. ArcIMS Viewers: There are three viewers available for ArcIMS. They are:

- HTML Viewer,
- Java Standard Viewer,
- Java Custom Viewer.

Other components include ArcSDE, ArcMap, and Route Server. The system architecture of ArcIMS technology is shown in Figure 7.

![Figure 7: Architecture of ArcIMS Technology (Source ESRI)](image)
Discussion of the Components of the Major Products

Autodesk was the first major software company to bring dynamic, vector-based, and interactive online maps to the consumer market and has been developing web-based GIS software since the mid-1990s. From the beginning, the Autodesk MapGuide architecture was designed expressly for Internet and intranet applications. Autodesk MapGuide is a full featured Internet GIS authoring and viewing environment, complete with an API for application-specific customization. In addition, because of Autodesk’s focus on data integration in their GIS product line, Autodesk MapGuide can handle data from a variety of existing formats (including SHP, MIF/MID, DGN, DWG, and ESRI coverages), which can be brought into Autodesk MapGuide from different servers and then published to the Web.

In comparison, ESRI ArcIMS technology allows desktop-based shape files to be viewed with a standard web browser using a proprietary programming language (ArcXML). Although enabling users to view GIS data over the Web was an important step, ESRI’s IMS software was first engineered using desktop-based technology, not web-based. ArcView IMS was released in the mid to late 1990s. It used the desktop software ArcView as its engine and was prone to crash. The Internet component was essentially an extension to the desktop software, and so ESRI’s first venture into Internet mapping was really just an attempt to web-enable their desktop GIS software, a system never designed with the network in mind.

ESRI also released MapObjects IMS, a more effective tool but one with a difficult development environment. Version 2 of MapObjects IMS was released in 1998, and ESRI invested its next effort into a different code stream. Despite its version number,
ArcIMS 4 is a second release of that code stream. ESRI numbered the premier release of ArcIMS as version 3 since it followed release 2.1 of MapObjects IMS, but these code streams are completely separate. As a result, ArcIMS is a relatively new technology compared to Autodesk MapGuide and has the performance and reliability problems one might expect from a new software line. Although the latest release of ArcIMS, version 4, addresses some of these issues, the problematic legacy remains.

Although both are powerful tools, the differences between Autodesk MapGuide and ArcIMS parallel the differences in their origins—between technology developed for the desktop and technology developed for the Web. The mature legacy of Autodesk MapGuide and the historic focus on ease of customization provide a more effective solution.

Maps, Data and Viewers

The Autodesk MapGuide Author pulls vector data and raster images from the Autodesk MapGuide Server as layers to create an MWF file. The MWF file is the map file that gets embedded in a web page (or otherwise published to the Web). A big advantage of Autodesk MapGuide is that once the MWF file is authored there is no special publishing or formatting necessary, whereas with ArcIMS, application developers must first create an Image Service or a Feature Service. An Autodesk MapGuide LiteView application and an application developed for the Autodesk MapGuide ActiveX Control can both point to the same MWF. In this way, Autodesk MapGuide makes it easier to deliver maps on the web to different client viewers since all viewers—raster or vector—can point to a single source file.
ArcIMS provides two different formats with which to deliver maps on the Web after they have been created and authored in the ArcIMS Author: an Image Service or a Feature Service.

- The Image Service is a map customized for the ArcIMS HTML Viewer (no download required) or either of the two ArcIMS Java Viewers (download required) and is used for basic viewing and querying. With the Image Service, a user may search on types of restaurants in a city, choose a restaurant, zoom in to the area of interest, and pan around that area to find other points of interest. The Image Service combined with the ArcIMS HTML Viewer technologies is similar to Autodesk MapGuide LiteView technology. Both technologies take a snapshot image of the requested extents of data on the server and pass back an image file (PNG or JPEG, for example) to the client.

- The Feature Service is a map customized or processed to deliver data to the ArcIMS Java Standard or Custom Viewer only and is used for more advanced, customized user interaction. The HTML Viewer does support limited customization, but the Java Viewer is where the most flexible custom application can be developed within the ArcIMS system.

The Feature Service combined with the ArcIMS Java Viewer is similar to the Autodesk MapGuide Viewer plug-in, ActiveX Control, and Java Edition. Examples of displays at the user’s end using AutoDesk’s MapGuide viewer and ESRI’s ArcIMS are shown in figures 8 and 9 respectively.
Figure 8  Autodesk MapGuide HTML Viewer – plug-in required

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Advantages of SDF over SHP for Web Delivery

Autodesk MapGuide software’s native file format, the SDF file, has only five attributes. As a result, SDFs are much smaller (and faster for web delivery) than shape files, which use a DBF file to store attribute information. By comparison, SDF files use DBMS data sources to access attribute information, making them more suitable for distributed network environments. Though ESRI shape files work well for desktop GIS, they hold extra information that can slow web delivery. Shape files consist of three files.
with the same name but different extensions: \texttt{.shp}, \texttt{.shx}, \texttt{.dbf}. The \texttt{.shp} file stores the geometry. The \texttt{.shx} file indexes the geometry. The \texttt{.dbf} file stores the attribute information of the \texttt{.shp} file. When information is accessed from a shape file, the entire file must be opened. Other files (unique to the shape file) can also be associated with the shape file to create definitions for other indexes and references. Another benefit of SDF over shape files is the ability to quickly compile a large number of similar shape files into a single SDF.

Technical Comparison

Both Autodesk MapGuide and ArcIMS provide client technologies with or without a required plug-in download. As with most client technologies, whether an application requires a plug-in depends on the end-user functionality needed. There are differences in functionality, potential functionality, and performance between applications that are based on plug-ins and those that are not. For example, if the user needs only to view data and query attribute data related to spatial data, then perhaps a developer will build a simple raster image–based application that does not serve live vector data and thus does not require a plug-in.

Autodesk MapGuide LiteView enables the Autodesk MapGuide Server to display maps in a browser without a plug-in. The ArcIMS HTML Viewer is the only non-Java based viewer option available for ArcIMS. The ArcIMS HTML Viewer sends an ArcXML request another proprietary language from Servitor the ArcIMS Server and receives an ArcXML response. The ArcIMS HTML Viewer has more built-in functionality than Autodesk MapGuide LiteView, such as buffering, spatial
selections (select by rectangle or circle), and measuring distances, but much of the same functionality can be added to Autodesk MapGuide LiteView through customization. Table 1 shows the comparison of the functionality of web mapping systems that require no plug-in download and installation.

**TABLE 1** Comparison of the functionality of web mapping systems that require no plug-in download and installation.

<table>
<thead>
<tr>
<th></th>
<th>AutodeskMapGuide LiteView</th>
<th>Autodesk MapGuide Customized</th>
<th>ArcIMS HTML Viewer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Download required</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Output Image</td>
<td>PNG</td>
<td>PNG</td>
<td>JPEG, GIF, PNG</td>
</tr>
<tr>
<td>Zoom In, Zoom Out, Zoom Full, Pan</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Create Buffer</td>
<td></td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Select within Buffer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identify - select geographic object and view data</td>
<td>√</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Measure Distance</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The ArcIMS Java Viewer using Feature Services has two functions called MapNotes and EditNotes (EditNotes is available only for the Java Standard Viewer). The MapNotes function enables the user to add text or graphics to the map (on the MapNotes layer). These edits are then sent to the MapNotes folder on the ArcIMS Server. The EditNotes function enables the user to edit features on the displayed map. Similar to MapNotes, once the user is finished editing, the changes are submitted to the EditNotes folder on the ArcIMS Server for the server administrator to review. For true data creation, EditNotes must be converted to SHP files or XML. With the SDF COM Toolkit, developers can extend the Autodesk MapGuide Viewer to enable users to create, edit, or delete actual SDF files—no conversion necessary.
TABLE 2 Comparison of functionality of viewer technology that require software
download and installation

<table>
<thead>
<tr>
<th></th>
<th>Autodesk MapGuide Viewers</th>
<th>ArcIMS Java Viewer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Download required</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Support for Netscape</td>
<td>✓</td>
<td>*</td>
</tr>
<tr>
<td>Select objects by rectangle</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Select objects by radius</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Select multiple objects on different layers</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

* - Plug-in required

Autodesk MapGuide provides the ability to create complex buffers from disparate feature types (points, lines, and polygons) that are close together or far apart and creates a single buffer that can be either joined together (when close) or separate, but act as a single entity (when far apart). ArcIMS does not offer this powerful feature. Another powerful Autodesk MapGuide feature is the ability to select or deselect map features or objects using SHIFT-pick. This capability is especially powerful after a buffer has been created because the user can add or remove parcels, roads, or other objects from the selection set using the standard Windows method of SHIFT-pick. ArcIMS does not provide this capability. In addition, the selection of objects in ArcIMS viewers is restricted to only the active layer. This means users can select either Parcels or Buildings but not both.

ArcIMS uses the standard web printing feature of Microsoft Internet Explorer. Another feature in the ArcIMS Java Viewers enables users to add shape files and ArcSDE layers from local machines and map services from other ArcIMS websites. It is interesting to note that the ArcIMS Author does not allow users to access data from other
TABLE 2 Comparison of functionality of viewer technology that require software download and installation

<table>
<thead>
<tr>
<th>Feature</th>
<th>Autodesk MapGuide Viewers</th>
<th>ArcIMS Java Viewer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Download required</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Support for Netscape</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Select objects by rectangle</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Select objects by radius</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Select multiple objects on different layers</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Autodesk MapGuide provides the ability to create complex buffers from disparate feature types (points, lines, and polygons) that are close together or far apart and creates a single buffer that can be either joined together (when close) or separate, but act as a single entity (when far apart). ArcIMS does not offer this powerful feature. Another powerful Autodesk MapGuide feature is the ability to select or deselect map features or objects using SHIFT-pick. This capability is especially powerful after a buffer has been created because the user can add or remove parcels, roads, or other objects from the selection set using the standard Windows method of SHIFT-pick. ArcIMS does not provide this capability. In addition, the selection of objects in ArcIMS viewers is restricted to only the active layer. This means users can select either Parcels or Buildings but not both.

ArcIMS uses the standard web printing feature of Microsoft Internet Explorer. Another feature in the ArcIMS Java Viewers enables users to add shape files and ArcSDE layers from local machines and map services from other ArcIMS websites. It is interesting to note that the ArcIMS Author does not allow users to access data from other
ArcIMS websites. Autodesk MapGuide puts this functionality in the Author rather than the viewer.

TABLE 3 Technical Comparisons of MapGuide Viewer and ArcIMS Java Viewer

<table>
<thead>
<tr>
<th></th>
<th>Autodesk MapGuide Viewers</th>
<th>ArcIMS Java Viewer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select objects with SHIFT-Pick</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td><strong>Buffering</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create buffer</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Buffer creates new layer</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Select within buffer</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Complex buffer creation</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td><strong>Querying</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identify - select geographic object and view data</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Set map units</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Set selection mode centroid or intersection</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Set mouse position display units (Lat/Lon or Mapping Coordinate System)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td><strong>Presentation</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Map Tips / Map Tooltips</td>
<td>✓</td>
<td>Feature Service only</td>
</tr>
<tr>
<td>Customized printing</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Online help files</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

Authoring Maps and Publishing on the Web

The Autodesk MapGuide Author enables users to add spatial and attribute data to an MWF file as layers and edit properties of these layers. The ArcIMS Author also enables the user to bring in data as layers and edit layer properties, but saves the map as an AXL file—ESRI’s version of the MWF/MWX file. Connections to remote databases (IBM DB2, IBM Informix, Microsoft SQL Server, and Oracle only) must be made through ESRI’s SDE software.

Autodesk MapGuide provides more tools for map authoring and a more helpful approach to web-based GIS development. Generally, ArcIMS provides many tools for authoring maps, but most of the more powerful functions reside on the client side. Autodesk MapGuide provides a superior map-authoring program that contains all the
functionality of its viewer and much more. The Autodesk MapGuide authoring and viewing products are consistent, while ArcIMS allows the ArcIMS Author to perform only a subset of the functionality of its Java viewers.

Autodesk MapGuide users can right-click to access all the tools available in the menu, whereas ArcIMS users cannot. Both technologies have set scale features for bringing in data layers, and different view properties can be scale dependent. ArcIMS Author cannot edit the properties of multiple layers at the same time, nor can it create layer groups (layers within layers), which allow for better data organization. Nor can users change the data source of a layer defined in the ArcIMS Author. Autodesk MapGuide treats layer properties (such as line width and color) and layer data sources similarly in that they are defined at the property level. In fact, a layer’s data source is one of its properties. This illustrates the ease of use and intuitive design of Autodesk MapGuide.

ArcIMS Author lacks printing functionality and uses the Windows® default print feature. In contrast, Autodesk MapGuide offers its own printing function (with Windows printing) and enables the user to set many properties, including scale bar, north arrow, title, time and date, and URL. It also provides a print preview. Further, all printing properties can be customized and are available to the Autodesk MapGuide Viewer through the API. The map explorer and print preview are two examples of Autodesk MapGuide Author’s ease of use. The authoring tool in Autodesk MapGuide has the same look and feel as the Autodesk MapGuide viewer, making development easier and enabling the developer to be more creative and productive.
ArcIMS Designer provides an easy-to-use wizard to set up a simple web application in about eight steps. The ArcIMS Designer guides the user through a process of choosing many map properties that in Autodesk MapGuide are defined through the Autodesk MapGuide Author. This wizard approach to web development creates a local directory for the web files, and links that directory to the map service (Image or Feature) created in the ArcIMS Server Administrator. Although the ArcIMS Designer may appeal to the needs of some potential users, functionality is inconsistent throughout the product.

With Autodesk MapGuide the developer must be familiar with HTML and web design. As with most wizard approaches to application development, there are pros and cons to the ArcIMS Designer: it may save time in the short run, but automated code generation and access to only a predefined set of functionality mean that extending and customizing such applications may take longer. Both the Autodesk MapGuide Author and the ArcIMS Author enable the user to create thematic maps by setting theme values for different layers. However, with ArcIMS the theme field must be added from a shape file. With Autodesk MapGuide the theme field can come from the SDF file or an OLE DB data source, resulting in more sophisticated and easier GIS analysis over the Web.
<table>
<thead>
<tr>
<th></th>
<th>Autodesk MapGuide 6</th>
<th>ESRI ArcIMS 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>.mwf, .mwx</td>
<td>.axl</td>
</tr>
<tr>
<td>Save as</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Save individual layer</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Copy map as..</td>
<td>.emf, URL</td>
<td>.jpeg</td>
</tr>
<tr>
<td>Open file from HTTP location</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Navigation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zoom width</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Zoom scale</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Zoom selected object</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Zoom goto address - address matching</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Selection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Select objects by rectangle</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Select objects by radius</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Select objects by map feature</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Select objects by polygon</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Select multiple objects on different layers</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Select objects with SHIFT - Pick</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Buffering</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create buffer</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Buffer creates new layer</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Select within buffer</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Create complex buffer</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Authoring</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Link map features to URL</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Measure distance</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>MapTips</td>
<td>√ one layer only</td>
<td></td>
</tr>
<tr>
<td>Add scale bar</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Labeling</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Map preview</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Change coordinate system</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Create queries / stored queries</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Thematic mapping based on OLE DB data source</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Graduated symbols</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Map password protected setting</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Track map usage</td>
<td>√</td>
<td></td>
</tr>
</tbody>
</table>
MapGuide Vs ArcIMS

Most GIS users would like to simply view and query data over the web, and both Autodesk and ESRI provide technologies to help them do that. However, as demand for more sophisticated analysis increases, Autodesk MapGuide is in a better position to meet those needs. ArcIMS is still in an early phase of development, demonstrated by inconsistencies in features and functions and by processes that do not work as expected. Its central strength is that it works with ESRI products (showing images of SHP files on the web). Because Autodesk MapGuide also works with ESRI products, its powerful capabilities and scalability make it a better choice for implementing a web-based system.

Chapter Summary

This chapter reviewed the components of some web GIS. A number of common traits were identified in both commercial / vendor products and real world online applications. The commercial products demonstrated the latest available technology, and what the GIS companies had identified as necessary in web GIS products. The real world applications demonstrated practical solutions and the concepts used in those solutions, with less focus on the technologies themselves. The final section of the chapter outlined the preferred software for a web-based fleet management application.
CHAPTER 4

SYSTEM DESIGN

The chapter deals with the design process to develop a system to collect, analyze and manage data for street sweeper management. This system is based on GPS data obtained from individual sweepers. The steps in system design are discussed including the phases of GPS data collection, data processing, database management and the application development process.

System Architecture and Design Process

The key components of System Architecture are:

1. Data logger in the vehicle, data transfer, database design, data analysis and management, display and communication of results.

2. User interfaces in the sweeper, for data transfer, end user

3. Web enabled applications.

The overall system architecture is illustrated in Figure 10.
The system design process includes the phases of GPS data collection with the aid of GPS receivers installed on the street sweeper trucks. The collected GPS data are transferred to a computer in the base station using a DECT module. The data are stored in database. The data can be managed using the SQL Enterprise manager software. The SQL database is used for all the data post processing. The information present in several tables of the database is queried and the result set is kept in a single table. This single table has all the information which can be displayed on the maptips of the MapGuide website.
The SQL database is connected to MapGuide Author using an ODBC connection and creating a user DSN. The MapGuide Author is connected to the single table in the SQL database where all the necessary information to be displayed is present. A web-based tool is developed with the help of AutoDesk MapGuide software. This web-based tool will display the streets swept by the sweeper truck. The search page is developed using Active Server Pages (ASP) to help the user to query the database by Vehicle ID and date. When this query is submitted the web tool developed will display the queried information in a spatial display. The web-tool will display the routes swept by the Sweeper truck on that particular day. The maptips will display the Vehicle ID, Status, Speed and the Date and Time of the Sweeper truck.

GPS Data Collection Process

GPS technology has matured into a resource that goes far beyond its original design goals. These days scientists, sportsmen, farmers, soldiers, pilots, surveyors, hikers, delivery drivers, sailors, dispatchers, lumberjacks, fire-fighters, and people from many other walks of life are using GPS in ways that make their work more productive, safer, and sometimes even easier. The GPS systems used for the sweeper trucks are made up of two parts: the hardware consisting of a black box for each sweeper that receives the satellite signals and the software that interprets the signals.

A GPS receiver receives the signal broadcast by satellites and uses information contained in the signal to calculate the position of the receiver. The positional accuracy that can be achieved with GPS ranges from 100 meters to millimeters, depending on the
type of receiver used and if the data collected are differentially corrected either in real
time or in a post process fashion.

The hardware unit selected for this study is the Fleet Manager 200, made by VDO
and approximately costs $600 per unit. VDO deploys GPS and communications
technologies to help create a comprehensive range of powerful fleet management system.
The Fleet Manager 200 has a Digital Enhanced Cordless Telecommunications (DECT)
module which has been connected and configured and a base station has been installed at
the “home base/ depot” for the street sweeper fleet. The DECT can be used over a short
distance of approximately 300 meters. This negates the need for the physical access to the
vehicle required by the code-plug downloading method and is intended for vehicles that
return regularly to home base. These downloads are scheduled and initiated by the DECT
Downloader software.

The code plug mechanism is the original means of extracting data from a vehicle. A
plug is inserted into the vehicle’s code-plug socket, and then inserted into a download
module connected to a computer. The plug can store more than 900 trips, including event
and violation data, which caters for up to 30 days of driving at the rate of 30 trips per day.

The GPS data recorded by the FM 200 units is downloaded into the FM 2002
Professional software by any of the standard downloading techniques described above.
The data are stored in the FM 2002 Professional database and can be analyzed using the
GPS Log Viewer. The following data elements are stored with every GPS point in the
database:

- Vehicle ID
- Driver ID
The Fleet Manager 200 is modular in design and can interface with a wide range of accessory products including fuel meters, tachographs, hand held computers, and other products in the VDO Fleet Manager range. The FM 200 unit supports the use of the black passenger code plug for identifying passengers.

Data Processing

The second part of the GPS system is the client-server software application that interprets the signals from the satellite. The VDO Fleet Manager 2002 Professional is a client-server software application is designed to front end FM 200 on board computer. The software enables users to manage vehicle, driver and passenger information, define and manage events and unit configuration, and analyze data. The data stored in the VDO Professional database are managed using the SQL Enterprise manager software. The data
are stored in several SQL tables and the information is queried and processed to obtain the required data. Figure 11 show the layout of tables in the SQL Enterprise manager.

Figure 11 Tables present in the SQL database
The VDO Fleet Manager Software allows the users to run the software on a single PC via the free SQL server engine installed with the software or can be integrated into the existing SQL server database server. In this case, it was integrated into the existing SQL server database so that data can be shared over a wide network. The tables contain information for every single event triggered by the GPS receiver. The GPS data is the most important aspect in our case along with the event data. Figure 12 shows the layout of the contents present in the table of the SQL database and Figure 13 shows a single table containing all the information.

<table>
<thead>
<tr>
<th>GPSID</th>
<th>VehicleID</th>
<th>DriverID</th>
<th>OriginVehicleID</th>
<th>BlockSeq</th>
<th>DateTime</th>
<th>Latitude</th>
<th>Longitude</th>
<th>Altitude</th>
<th>Heading</th>
<th>uSatellites</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12260</td>
<td>0</td>
<td>0</td>
<td>-3147463848</td>
<td>6/11/2004 4:37:00 36.11422</td>
<td>-115.049</td>
<td>520</td>
<td>116</td>
<td>12</td>
<td>12</td>
</tr>
</tbody>
</table>

Figure 12 Contents in the Tables of the SQL database

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>IFV Value</th>
<th>Status</th>
<th>Sweeping Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Water Level</td>
<td>Low</td>
<td>Low</td>
<td>Sweeping 0.89944</td>
</tr>
<tr>
<td>Broom</td>
<td>Left 1744</td>
<td>Sweeping 3.5</td>
<td></td>
</tr>
<tr>
<td>Broom</td>
<td>Right 553</td>
<td>Sweeping 1.5</td>
<td></td>
</tr>
<tr>
<td>Broom</td>
<td>Right 556</td>
<td>Sweeping 1.5</td>
<td></td>
</tr>
</tbody>
</table>

Figure 13  Single table containing all the information

MapGuide Association with SQL Server

Connections to databases and/or database servers are independent of the publishing of MapGuide maps. However, maps can employ database information as data sources for
maps. Therefore, the MapGuide Server must have access to the database(s) either locally (on the computer) or via a network connection. MapGuide Server employs ODBC driver connections to attach to data sources (databases). This is accomplished via one or more System DSN, File DSN, or OLEDB connections on MapGuide Server computer. Figure 14 to 19 show the step-by-step process setting up ODBC driver connections to attach to data sources.

![ODBC Data Source Administrator](image)

Figure 14 First Step in creating a ODBC connection – setup of User DSN

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Create New Data Source

Select a driver for which you want to set up a data source.

<table>
<thead>
<tr>
<th>Name</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft FoxPro VFP Driver (*.dbf)</td>
<td>1</td>
</tr>
<tr>
<td>Microsoft ODBC for Oracle</td>
<td>2</td>
</tr>
<tr>
<td>Microsoft Paradox Driver (*.db)</td>
<td>4</td>
</tr>
<tr>
<td>Microsoft Paradox-Treiber (*.db)</td>
<td>4</td>
</tr>
<tr>
<td>Microsoft Text Driver (*.txt; *.csv)</td>
<td>4</td>
</tr>
<tr>
<td>Microsoft Text-Treiber (*.txt; *.csv)</td>
<td>4</td>
</tr>
<tr>
<td>Microsoft Visual FoxPro Driver</td>
<td>1</td>
</tr>
<tr>
<td>Microsoft Visual FoxPro-Treiber</td>
<td>1</td>
</tr>
</tbody>
</table>

SQL Server

Figure 15 Second Step - Select SQL Server in the driver list

Create a New Data Source to SQL Server

This wizard will help you create an ODBC data source that you can use to connect to SQL Server.

What name do you want to use to refer to the data source?

Name: Mapguide

How do you want to describe the data source?

Description: Mapguide

Which SQL Server do you want to connect to?

Server: PWSOL1

Finish Next > Cancel

Figure 16 Third Step – Select the SQL server from a list to connect
Create a New Data Source to SQL Server

How should SQL Server verify the authenticity of the login ID?

- With Windows NT authentication using the network login ID.
- With SQL Server authentication using a login ID and password entered by the user.

To change the network library used to communicate with SQL Server, click Client Configuration.

Login ID: JTrcintern
Password: |

Figure 17  Fourth Step: Connect to the SQL Server with the administrator login

Create a New Data Source to SQL Server

- Change the default database to:
  - tempdb
  - msdb
  - Northwind
  - pubs
  - SweeperGPS
  - tempdb

- and drop the stored procedures:
  - Only when you disconnect.
  - Only when you disconnect and as appropriate while you are connected.

- Use ANSI quoted identifiers.
- Use ANSI nulls, paddings and warnings.
- Use the failover SQL Server if the primary SQL Server is not available.

Figure 18  Fifth Step: Select the database to connect
A new ODBC data source will be created with the following configuration;

- **Microsoft SQL Server ODBC Driver Version 03.85.1117**
- **Data Source Name**: Mapguide
- **Data Source Description**: Mapguide
- **Server**: PWSQL1
- **Database**: SweeperGPS
- **Language**: (Default)
- **Translate Character Data**: Yes
- **Log Long Running Queries**: No
- **Log Driver Statistics**: No
- **Use Integrated Security**: No
- **Use Regional Settings**: No
- **Prepared Statements Option**: Drop temporary procedures on disconnect
- **Use Failover Server**: No
- **Use ANSI Quoted Identifiers**: Yes
- **Use ANSI Null, Paddings and Warnings**: Yes
- **Data Encryption**: No

![ODBC Microsoft SQL Server Setup](image)

**Figure 19 Sixth Step – Test the Data Source for connection**

AutoDesk MapGuide products do not include any specific database product, but can employ and connect to any database provider which has ODBC or OLEDB drivers. Typical database providers are Microsoft Access, Microsoft SQL Server, Oracle and other database engines.
Application Development Process

MapGuide employs a patented streaming http delivery system to deliver real vector object information and raster files from a server component to a client-based viewer component. The MapGuide Author is sold as a separate product and exists solely to create and author maps. MapGuide Author builds the set of parameters that are included in the configuration file of each of the individual maps you publish via the server to the viewer. The MapGuide Viewer reads this configuration file for the map and this tells the Viewer where to get the information to display and how to display the information. A single map can include multiple layers of information delivered from multiple MapGuide Servers and multiple database sources.

The map is commonly delivered via HTML pages inside a web browser interface, a running web server (HTML) will be used to serve the HTML pages. For most users, the web server will be the Microsoft Internet Information Server (IIS) that comes inbuilt with Microsoft NT and Windows 2000 Servers. For security and performance reasons the latest release of IIS (at least 4.0) is required. It is possible to run the MapGuide Server with almost any web server. The MapGuide Server need not be on the same machine as the web server, but this is not usually necessary. Figure 20 shows a typical layout of an Internet Information Server (IIS):
Figure 20 Typical layout of an Internet Information Server (IIS)

The end user’s computer must be able to locate (resolve http) and connect to the MapGuide Server directly via the http protocol to actually publish maps. In other words, the MapGuide Viewer makes simultaneous and completely separate connections to the web server computer and the MapGuide server computer. Behind the authoring and viewing of maps is the MapGuide Server. It handles requests from the Author and Viewer programs, determines which data to provide, and then returns the data to the clients. The data can come from several sources such as native MapGuide SDF files and a database server like Microsoft SQL.

The interface between the web server and MapGuide server is called a MapAgent. There are three different types of MapAgents; ISAPI, NSAPI, and CGI. ISAPI is the best
choice if Microsoft Internet Information Server (IIS) is used. NSAPI is used for Netscape Enterprise servers. The CGI version works on most other web servers like Apache. Figure 21 shows the typical layout of a services window:

![Services Window](image)

Figure 21 Layout of the services window

After installing the Map Agent and MapGuide Server, a set up is done to get the system ready for MapGuide Author users. The users/groups are defined who will have access to the map server data sources. Figures 22 to 24 show sweeper-gps property window and selection of the user groups respectively.
Figure 22  Setup local path for the file to be read

Figure 23  Enable default document for viewing
Authentication Methods

- **Anonymous access**
  No user name/password required to access this resource.
  Account used for anonymous access: [Edit]

- **Authenticated access**
  For the following authentication methods, user name and password are required when:
  - anonymous access is disabled, or
  - access is restricted using NTFS access control lists

- Basic authentication (password is sent in clear text)
  Select a default domain: [Edit]

- Digest authentication for Windows domain servers
- Integrated Windows authentication

---

Figure 24   Define users/groups that will have access to map server data sources
CHAPTER 5

IMPLEMENTATION AND RESULTS

This chapter summarizes the development of the web-based tool and the various sections which comprise the tool. The various functions included in the tool are discussed in individual sections of this chapter. The results obtained with the tool are discussed and the various outputs are shown. This tool can be applied to any fleet system.

Web-based Tool

The purpose of the web-based tool is to provide an outlet of information dissemination in a dynamic environment that the user can browse and interact with. The web-based tool is created using Autodesk’s MapGuide software package. The MapGuide Server software parses out the data from the map document. The map document itself was created using MapGuide Author. The map document is displayed in the map window and shows all the data associated with the web-tool. The user would need the latest MapGuide Viewer, provided free by Autodesk, to read and process the data. The web-tool functions and layout are incorporated using JavaScript and HTML, respectively. The web-tool is divided into four main sections: Map Window, Pan Controls, Map Tools, and Layer Controls. Figure 25 shows the screen shot of the web-based tool homepage.
Figure 25  Web-based Tool

Map Window

The Map Window is where all the spatial data are displayed. The map window is designed to maximize the amount of space used for displaying the map while still providing easy access to the map tools. Map tools, pan controls, and the layer controls interact with and control what shows up on the map window. The map window is the display where all the tool functions and navigation takes place. This window also shows any selected features, layers currently turned "on", and the dynamic pop-up label tool.
tips. The layers can be activated by checking the box against the layer name. Lay out of the map window can be seen in the figure below, Figure 26.

Figure 26  Lay out of the Map Window

Pan Controls

The Pan Controls provide easy navigation around the map at the current scale. Functionality is provided for all eight directions of movement. The four primary directions are labeled, north, east, west, and south. The northwest, northeast, southwest, and southeast directions are depicted by an arrow in their respective directions. The eight pan controls are located on the sides of the map window, and clicking on any of them will
move the map in the direction that the arrow indicates. These are handy for navigating quickly without having to change the current map tool. Figure 27 shows the layout of pan controls.

![Figure 27 Lay out of pan controls](image)

**Map Controls**

The Map Tools panel contains tools for exploring the map and gives the user the ability on selecting various map elements. Essentially, this toolbar provides access to the various mapping functions, such as user specified zooming controls, select and unselect, and the measure tool. The map tool provides the best interface for interacting with the map, and the function of each of the map tools will be discussed in detail in the following section regarding the Web-based Tool Manual. Map control layouts can be seen in the figure 28.

![Figure 28 Lay out of Map Controls](image)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Map Tool Descriptions

The Map Tools, as previously discussed, provides the best interface for interacting with the map. The function of each of the tools on this toolbar is described next.

Select:

The Select tool allows one to select objects on the map. If the pointer changes to the hyperlink indicator (pointing hand cursor icon), then double-clicking on the feature, and detailed information or images will be displayed.

Pan:

The Pan tool allows one to interactively move the extents of the map. The user clicks and holds on the Map Window, and drags it in the desired direction for the move.

Zoom In:
The Zoom In tool function can be activated by clicking or drawing a zoom box on the Map Window to allow for zooms on the location of interest.

Zoom Out:

Clicking on the Map Window when the zoom out tool is active zooms the map out a fixed distance centered on the location you have clicked.

Zoom Previous:

Zoom previous acts similar to the web browser's "Back" button, taking one to the previous zoom levels and extent of the map.

Zoom to Full Extent:

The Zoom to Full Extent tool takes you to the initial view of the Clark County wide map (the home page for the tool).
Measure:

The Measure Distance tool allows one to measure and estimate distances on the map. Click once on the start point and then continue to click along the line one wants to measure. As one moves, the distance should appear in a pop-up box. Double-clicking ends the line and allows one to start measuring another. Single-clicks will continue the measurement distance until one double-clicks to end the measure tool.

Print:

The print tool allows one to take the print of the image which is present in the map window.

Layer Restrictions

The layer controls, as previously discussed, have certain scale and draw priority restrictions. The scale restriction means that the map window needs to be at a certain scale in order for the layer to become visible. This is most commonly done to prevent a bottleneck in requesting data from the server. For example, the aerial photography is divided according to the Clark County book and section number and set to be visible at 0-
20,000 feet because each individual image would need to be processed before it is displayed in the map window.

At the overall Clark County extent, and no scale restrictions are set, it would take the server too much time to process each individual image and display it on the map window. For this reason, it is been opted to restrict the display of certain layers based on scale. The draw priority restriction means that a layer is in front of another layer, thus preventing the lower priority layer to be shown. For example, when the user is at the appropriate scale to view the aerial photography image, the City Boundaries layer would no longer be seen. The draw priority is set for the best display for the users. MapGuide assigns the draw order of the layers based on the draw priority so that higher priority layers will always be drawn on top of a lower priority layer.

Query by Vehicle and Date

The query page is developed with Active Server Pages (ASP) software. The query page is connected to the SQL database with the help of an OLEDB connection. The page can be used to query up to four vehicles with desired dates. The main function of the query page is to query the database and display the desired dates. The figure 29 shows the lay out of the query page:
Figure 29 Lay out of the query page

The desired vehicle is selected and the desired dates are entered and query is submitted to display the results in a tabular manner for the first vehicle on the ASP page. The vehicle is selected from a drop down menu. The user types in the desired dates. The results are displayed to confirm the user that the user passed a valid query. If no results are obtained then it is an invalid query or sweeper truck was not in operation on the desired dates. The selection of an individual vehicle is illustrated in Figure 30.
Figure 30  Select vehicle to be displayed

Results

The results obtained after submitting the query are displayed in the ASP page immediately after the user clicks submit. The results are displayed for the first vehicle only. This result page is generated just to confirm the user that the query was valid and it was passed to the database for processing. Figure 31 shows a typical layout of the result on an ASP page.
Results displayed for only one vehicle

<table>
<thead>
<tr>
<th>ID</th>
<th>fLat</th>
<th>fLon</th>
<th>Head</th>
<th>SweeperTest</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>36.11469</td>
<td>-115.0472</td>
<td>0</td>
<td>VehicleN: 12260uSpeed(mph): 5</td>
</tr>
</tbody>
</table>

Figure 31 Displayed results on the ASP page

The map window is right clicked and the map window is reloaded for the data source of the map to refresh. Figure 32 shows the map window when reloaded.
The map now displays the desired spatial data after the reload and the user can view the routes of the sweeper trucks. The routes are discriminated by color and the layers are super imposed over each other. The user can view the route traced by a specific truck by checking the layer controls. Figure 33 show the spatial display of the results on the map. Figure 34 shows the display of the route traced by a selected sweeper truck.
Figure 33  Desired results displayed spatially on the map
The Layer Controls provide the user with a dynamic legend showing the layers on the map. The layer list allows the user to dynamically turn on/off any layer depending on their needs and desires at any given time. The layers portion of the web-tool allows the user to select which map features are displayed on the map window. While all layers are available on the map, some layers have scale restriction or draw priority restrictions depending on the layer that covers it. The maptips are displayed when the mouse is placed over any desired point. The maptips display pertinent information such as Vehicle ID, Speed at that time, Status and the Date and time when the GPS reading was noted. Maptips are beneficial for users to know the description of the sweeper at a particular point.

Figure 34 Route traced by a Sweeper truck
location or time. Figures 35 and 36 show the typical route followed by a sweeper in a neighborhood and display of status with maptips respectively.

Figure 35  Typical route followed by a sweeper in a neighborhood
Figure 36 Maptips displaying the status as Hopper Dump
CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

This chapter contains a brief summary of work documented in this thesis. Conclusions and recommendations for further study are also presented.

Summary

Fleet management systems provide significant amounts of data about the vehicles and their traveling patterns. These systems enable fleet managers and business owners to better manage their fleet vehicles by knowing where they are, where they have been. The objective of this thesis was to develop a street sweeper fleet management system to help improve the efficiency and effectiveness of street sweeping programs. The developed application is a reliable Fleet management system with a versatile web-based mapping tool.

An effective street sweeper management system would help reduce the overhead cost of fleet ownership and to identify and monitor vehicular events and activities such as speeding, unauthorized use, asset under-utilization, unnecessary idling time at the job site and miles driven. In this research an application was developed for the management of street sweepers with the aid of a mapping system interface to query and analyze the data.
collected using GPS devices installed in the sweeper trucks. The use of the system was demonstrated using street sweepers used by the Clark County, Nevada.

Conclusions

The major benefit of the system developed in this research is that it provides the ability to respond to citizen and other customers of the Street Division with accurate sweeping information down to the exact street segment. This system will serve as an integrated approach to work and asset management programs. This system will serve as a valuable tool for the end users who will be interested to see exactly where and when their sweepers were on the job and for how long, the roads they serviced and even when and where the brooms and dust suppression systems were engaged.

The developed system will help in reducing manual record keeping and unauthorized vehicle use, minimizes billing errors, and improves route efficiencies. The system provides accurate records of machine activity and is therefore assumed to improve fleet cost management and vehicle productivity. The assumed benefits from the system include accountability of work performed, efficient use of resources and reduced liability issues like less missed completions and better records for court in case of accidents.

Recommendations for Future Work

Future research could involve developing web-based mapping interfaces to display data from active GPS devices. Further, a tool that incorporates reporting feature can be developed. Future work may also involve the optimization of routes of the sweeper trucks.

77

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
BIBLIOGRAPHY


Edward, Mac Gillary. (2000). Cartographic aspects of WebGIS software
Department of Cartography, Utrecht University, The Netherlands, April 2000


VITA

Graduate College
University of Nevada, Las Vegas

Ravi Shenkar Seera

Local Address:
969 E. Flamingo Road, # 122
Las Vegas, Nevada 89119

Home Address:
1710 West Hillcrest Drive, # 116
Newbury Park, California 91320

Degrees:
Bachelor of Technology, Civil Engineering, 2003
Jawaharlal Nehru Technological University, Kakinada

Thesis Title: Development of a Street Sweeper Fleet Management System

Thesis Examination Committee:
Chairperson, Dr. Shashi Nambisan, Ph.D.
Committee Member, Dr. Srinivas Pulugurtha, Ph.D.
Committee Member, Dr. Mohamed Kaseko, Ph.D.
Graduate Faculty Representative, Dr. Laxmi Gewali, Ph.D.