

# Multimedia Data Integration and Retrieval in Planning Support Systems

by

Carleton Tsui

Bachelor of Environmental Studies  
University of Waterloo, 1994

Submitted to the Department of Urban Studies and Planning in partial  
fulfillment of the requirements for the degree of

Master in City Planning

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1998

© 1998 Carleton Tsui. All Rights Reserved.

The author hereby grants to MIT permission to reproduce  
and distribute publicly paper and electronic  
copies of this thesis document in whole or in part.

Author .....  
Department of Urban Studies and Planning  
May 21, 1998

Certified by .....  
Michael J. Shiffer  
Lecturer / Principal Research Scientist  
Thesis Advisor

Accepted by .....  
Lawrence Bacow  
Professor of Law and Environmental Policy  
Chairman, M.C.P. Committee

JUN 22 1998

LIBRARIES



# **Multimedia Data Integration and Retrieval in Planning Support Systems**

by

Carleton Tsui

Submitted to the Department of Urban Studies and Planning on  
May 21, 1998 in Partial Fulfillment of the Requirements for the  
Degree of Master in City Planning

## **Abstract**

The use of multimedia data such as images, movies clips, animations, audio, or hypertext can facilitate the understanding of the physical environment for planning. In order to further this goal, computational techniques were developed to use a GIS compatible system architecture to organize and retrieve multimedia data in planning support systems. These techniques can help better integrate visual data into traditional database and GIS analyses performed by planners.

A prototype distributed multimedia planning support system was developed to provide a platform for experimenting with integration and retrieval strategies for digital images. Trials with the prototype showed that intelligent image retrieval techniques can enhance the usability of multimedia planning support systems and helps users to locate image data.

Thesis Supervisor: Michael J. Shiffer

Title: Lecturer / Principal Research Scientist



## **Biographical Note**

In addition to the MCP Degree, Carleton Tsui received a Bachelor of Environmental Studies (Honours Co-operative Urban and Regional Planning) from the University of Waterloo's School of Urban and Regional Planning in 1994. During his undergraduate years, Carleton focused on urban design and computer based visual simulation. His multimedia oriented Senior Honours Essay won the 1994 Ontario Professional Planners Institute Communications Award in Audio-Visual Presentation.

A Canadian citizen, Carleton Tsui worked as a private planning consultant in for nearly three years in Ontario prior to his admission to MIT. His professional planning experience has primarily been in the area of recreational planning, urban design, arts and culture planning.

While at MIT's Department of Urban Studies and Planning, Carleton studied as part of the Planning Support Systems Group with research interests that included: multimedia information interpretation / display / delivery; GIS; and Internet software development.



## Acknowledgements

Credit goes to Michael Shiffer for his thoughts on developing Internet accessible online annotation and many other good ideas over the years. It has been a great pleasure to work with one of the long time leaders in the development of multimedia applications in planning. Mike and my reader, Professor Joseph Ferreira, Jr., were important influences throughout the entire thesis process. They asked the hard questions that challenged me and left me always searching for a better answer. I am grateful for this vital critical input that helped transform imagination to reality.

The research and development of the software component of this thesis would not have been possible without the facilities of the DUSP Computer Resources Lab. My thanks to Mike Shiffer, Phil Thompson, and Tom Grayson for providing an excellent (yet often underrated) computing environment where I could learn and play free of resource constraints. No mention of the CRL would be complete without acknowledging the Backyard crew of Qing Shen, Anne Thompson, Jennifer Johnson, Raj Singh, Sue Delaney, Anne Beamish, John Evans, and Ming Zhang who all helped to make life in the CRL a bit more fun over the past two years. Special thanks to Laura Wilcox for making available much needed sugar at all times; “GUI Girl” Jennifer Johnson for laughs, lunches, and spiffy RC cars; and Raj Singh for all the late night Java talk!

Finally, I would like to dedicate this thesis to all the family and friends across Canada that have helped me get to where I am today and continue to support me wherever I may be in the world. Whatever success I can claim from my two year venture here at MIT deserves to be shared with these people who have all along held an unbelievable amount of confidence in me.





# Table of Contents

<b>1</b>	<b>Multimedia Applications in Planning</b> .....	<b>15</b>
1.1	Problem Statement .....	15
1.2	Approach.....	16
1.3	Structure of Thesis .....	17
<b>2</b>	<b>Previous Work in Multimedia Planning Support Systems</b> .....	<b>19</b>
2.1	The Role of Multimedia Applications in Planning Today .....	19
2.1.1	Interactive Multimedia for Information Presentation .....	20
2.1.2	Multimedia as a Tool for Collaboration.....	21
2.1.3	Embedding Multimedia into Planning Support Systems .....	22
2.2	New Directions in Multimedia Planning Support Systems .....	23
2.2.1	The Design of Multimedia Planning Support Systems.....	24
2.2.2	Further Research into Multimedia Planning Support Systems .....	25
<b>3</b>	<b>The Integration of Multimedia Objects in Planning Support Systems</b> .....	<b>27</b>
3.1	Scenarios Describing Multimedia Use.....	27
3.2	The Challenge of Integrating Multimedia Data .....	29
3.2.1	Limitations in the Imagemap Based Interface .....	29
3.2.2	Limitations in Point-and-Click Retrieval of Multimedia.....	31
3.2.3	Metadata for Integration .....	32
<b>4</b>	<b>Techniques for Retrieving Images Based on Spatial Metadata</b> .....	<b>37</b>
4.1	Providing Computable Attributes for Multimedia Objects.....	37
4.2	How Spatial Metadata is Used.....	37
4.3	Image Retrieval Techniques Based on Queries on Spatial Metadata .....	39
4.3.1	Keyword Search.....	40
4.3.2	Simple Point Query.....	40
4.3.3	Area Query .....	41
4.3.4	Location-to-Location Query .....	42
4.3.5	Including / Excluding Extreme Areas.....	43
4.3.6	Deducing Subject Proimnence by Distance from Origin.....	44
4.3.7	Juxtaposing GIS Coverages or Databases With Spatial Metadata.....	45
4.4	Extending Spatial Metadata .....	46
<b>5</b>	<b>Review of the MapVision MMPSS Prototype</b> .....	<b>49</b>
5.1	MapVision Client.....	49
5.1.1	Basic Navigation Functions .....	49
5.1.2	Keyword Searches .....	50
5.1.3	Area Searches.....	50
5.1.4	View Point Searches .....	52
5.1.5	Submitting Notes .....	53
5.1.6	Submitting Images .....	54
5.1.7	Saving and Deleting Data .....	55

5.1.8	Displaying Retrieved Multimedia Objects.....	55
5.2	MapVision Server .....	56
5.2.1	The MapWindow .....	57
5.2.2	Layer Control Panel .....	58
5.2.3	Multimedia and Annotation Layer Creation .....	58
5.2.4	Storage of Notes From Client .....	59
5.2.5	Query by Other Data Sources .....	60
5.3	Observations About The System .....	60
<b>6</b>	<b>MapVision System Architecture.....</b>	<b>63</b>
6.1	Server Design.....	63
6.2	Client Design .....	64
6.3	Overview of Client / Server Functionality .....	65
6.3.1	System Design Strengths and Weaknesses .....	66
6.4	Detailed Descriptions of Client / Server Interactions .....	66
<b>7</b>	<b>Advancing Multimedia Planning Support Systems .....</b>	<b>73</b>
7.1	Evaluation of Prototype .....	73
7.2	Distributed Data Submission and Collaboration.....	74
7.3	Methods for Generating View Geometry .....	76
7.3.1	Manually Sketching View Geometry.....	76
7.3.2	Calculating View Geometries Through Visibility Analysis .....	77
7.3.3	Calculating View Geometries Using Three Dimensional Space .....	77
7.4	Applications .....	78
7.4.1	Visualization .....	79
7.4.2	Collaborative Repository of Multimedia Data.....	80
7.4.3	Added Functionality to Existing Planning Support Systems.....	80
7.5	Multimedia Information Planning Systems .....	81
7.6	Conclusion .....	81
	<b>Bibliography .....</b>	<b>83</b>

# List of Tables

Table 2.1: Characteristics of Multimedia Applications in Planning Over Time .....	24
Table 3.1: Structure of Metadata Database in MapVision.....	35
Table 5.1: Annotation Layer Database Fields.....	59
Table 6.1: Client/Server Transactions for Initializing theMapVision System .....	67
Table 6.2: Client/Server Transactions for Navigation Functions .....	68
Table 6.3: Client/Server Transactions for Image Queries .....	69
Table 6.4: Client/Server Transactions for Adding a Note .....	70
Table 6.5: Client/Server Transactions for Adding an Image .....	71
Table 7.1: Recommended Access Privileges for Public and Private Users.....	75

# List of Figures

Figure 3.1: Example of an Imagemap.....	30
Figure 3.2: Problems With Integrating Multimedia Data Using Imagemaps .....	31
Figure 3.3: Example of the Complex Semantics Contained in an Image.....	32
Figure 3.4: The Role of View Geometry in Determining What is Visible .....	34
Figure 3.5: The Process of Attaching Various Metadata Types to an Image. ....	36
Figure 4.1: The Use of Spatial Metadata Within MapVision .....	38
Figure 4.2: Simple Point Query .....	41
Figure 4.3: Region Query .....	42
Figure 4.4: Location-to-Location Query.....	43
Figure 4.5: Example of Image With Extremely Large View Geometry.....	44
Figure 4.6: Deducing Subject Prominence by Distance from Origin .....	45
Figure 5.1: MapVision Java Client .....	50
Figure 5.2: MapVision Implementation of Area Searches .....	51
Figure 5.3: MapVision Implementation of View Point Searches .....	52
Figure 5.4: Submitting Notes in MapVision.....	53
Figure 5.5: Submitting Images in MapVision .....	54
Figure 5.6: Displaying Retrieved Multimedia Objects in MapVision.....	56
Figure 5.7: MapVision Server Map Window .....	57
Figure 5.8: Layer Control Panel.....	58
Figure 5.9: Multimedia Queries Utilizing Other Data Sources .....	60
Figure 6.1: MapVision System Architecture .....	65
Figure 7.1: Variations in the Recording of View Geometry .....	76





# **Chapter 1**

## **Multimedia Applications in Planning**

Multimedia data types, such as digital images, audio, video, animations, or documents, can be an important aspect of planning because they have the capability to provide added insight to an analysis of a place or problem. Photographs, illustrations, or maps allow people to visit (or revisit) a site or project. Video or animation can illustrate the effects of changes over time. Audio recordings document ambient sound, what people say in meetings, or help in studying noise. Finally, many planning activities inevitably express their results as text based documents for future reference.

In this paper, the term planning support systems loosely refers to those computer systems and software used to assist planners in the act of planning. Examples range from the simple spreadsheet model, to transportation network analysis systems, to Geographic Information Systems (GIS). The inclusion of multimedia data types into planning support systems has been the subject of research, development, and even some commercial products, for well over a decade. The fundamental belief, of course, is that a planning support system that can work with multimedia data types provides added value and insight for its users. Some of the most commonly held goals of constructing multimedia planning support systems (MMPSS) are to:

1. help the general public and planners alike to better understand their working environment and solutions;
2. help visualize abstract quantitative concepts;
3. distribute information about planning practices and projects; and,
4. help promote public dialogue and debate through the provision of alternative communication channels.

### **1.1 Problem Statement**

There have been many approaches to incorporating computer-based multimedia objects into planning support systems. These approaches range from stand-alone systems built to convey information about a specific project, to dedicated multimedia relational databases, to multimedia objects embedded in Geographic Information Systems (GIS), to World Wide Web (WWW) pages.

Despite the variety of design approaches, many multimedia applications are limited by the fact that their multimedia data is often tightly linked to their delivery mechanism (e.g. HTML imagemaps, a particular multimedia authoring software, CD-ROM). This makes it difficult to universally juxtapose multimedia objects with other planning data sources (e.g. census data, other digital maps, multimedia objects from other studies) and often limits the role of multimedia systems to presentation of compiled results. This is one reason why multimedia objects, despite their demonstrated usefulness in augmenting the understanding of many planning problems, have not played a more prominent role in planning support systems.

Furthermore, most multimedia planning support systems have typically taken simple approaches to retrieving multimedia objects. Web pages, desktop or kiosk multimedia applications, and recent experiments with multimedia GIS seem to ignore the development of a more powerful multimedia object retrieval strategy capable of moving beyond the standard point-and-click metaphor.

This thesis presents one possible solution to these problems in the form of: i) a platform neutral distributed system architecture; and, ii) specifications for metadata describing multimedia objects (and images in particular) that are optimized for multimedia object retrieval by spatial queries. The open system architecture tries to address the problem of multimedia objects being too closely integrated with one particular platform or software. Meanwhile, the metadata descriptors offer powerful yet intuitive ways for users to find multimedia objects that relate to a certain place.

## **1.2 Approach**

This thesis builds upon the generally accepted notion that the ability to integrate and retrieve qualitative multimedia data alongside traditional sources of quantitative data provides beneficial insights into planning problems. In order to further this objective, this thesis documents the design, construction, and testing of a working prototype of a distributed multimedia planning support system.

Specifically, the research conducted for this thesis explores the following issues:

1. Based on some of the common uses of multimedia in planning, and by studying past multimedia planning support system efforts, what approach can be developed to better integrate multimedia data with planning support systems?
2. Since there are many different types of multimedia data, each with distinct integration and retrieval issues, this thesis focuses mainly on the development of a methodology to handle digital images about the built environment.



3. What techniques can be developed that would assist in the retrieval of digital images from a large database of images given that the stated goal of exploring the urban environment through images?
4. How might either the system design or image retrieval strategies explored be used to support planning functions?

The development of the system is not framed with a particular planning organization or operation in mind as the goal of this work is to specifically *not* provide a targeted implementation. Most prototype multimedia planning support systems have followed the path of constructing a specialized application for a particular planning case or organizational structure. While this provides a powerful specific implementation of technology to the demands of one particular organization, it often limits the system design's transferability to a broader planning domain. The work in this thesis strives to present strategies for constructing more generalized systems for multimedia planning support. Likewise, the techniques for image retrieval are based on the premise of reusability in other specific implementations of MMPSS beyond those in the developed prototype.

### **1.3 Structure of Thesis**

This chapter has provided a brief introduction to the field of multimedia in planning and identified issues and approaches. The potential uses of multimedia in urban and environmental planning situations has been the subject of numerous academic studies, demonstration applications, and professional applications since the mid-1980s. A discussion of some these studies, their premises, and findings can be found in Chapter 2.

Chapter 3 examines some common scenarios for using multimedia in planning and identifies some deficiencies in the current methods of integrating multimedia data into planning support systems. This chapter argues that the careful consideration of how multimedia data is stored in the system is vital to providing better multimedia data retrieval strategies.

Chapter 4 documents the variety of multimedia data retrieval techniques that can be constructed by using the indexing method described in this thesis. These are techniques that can provide planning support systems with the ability to retrieve and analyze multimedia objects intelligently for urban visualization purposes.

Chapter 5 is a visual tour of the major implemented functions in the prototype MMPSS (named MapVision). This chapter is provided as an example of how an MMPSS based on this thesis' design concepts might work and how it may look like. This useful for solidifying the reader's understanding of the interactions that are desired before moving

into detailed explanations of how the interactions work. The chapter concludes with some observations realized through programming and operating the prototype.

Chapter 6 provides the design schematics for developing a distributed client/server multimedia planning support system to demonstrate the viability of the concepts explored throughout this document. This chapter explains how the various components of the MMPSS interact and provides the rationale for the chosen design.

Chapter 7 concludes the thesis by providing an evaluation of the developed prototype and identifies some areas that need to be strengthened through further research. Some applications for multimedia planning support systems are discussed along with speculation for important future directions for multimedia in planning.

## Chapter 2

# **Previous Work in Multimedia Planning Support Systems**

The use of multimedia computer applications for planning has been the subject of a fair amount of study in both academic and professional settings. This chapter provides a survey of past work in multimedia applications and their planning contexts, reviews the design approaches taken by the various authors in constructing their multimedia applications, and then explains why still more work in MMPSS is necessary.

### **2.1 The Role of Multimedia Applications in Planning Today**

Computer based multimedia can be thought of as the art and science of combining relevant digital data into a coherent format capable of offering a number of different perspectives on a given subject. The goal is to leverage qualitative data sources such as images, video, animation, and audio towards a better understanding of a particular subject. The broad number of multimedia data types and possible ways of combining them into a computer application makes it difficult to definitively state what a multimedia application should contain, what it looks like, or how it is designed. This is in contrast to other analytical computer planning tools such as CAD, GIS, databases, or spreadsheets. In a spreadsheet, planners provide numbers for calculations in a matrix. Databases are among the most structured applications for storing and retrieving data that planners can use. Computer aided design typically takes place in an application with a well-equipped graphical drafting and modeling environment. GIS, being a fusion of databases and CAD, generally has system attributes common to both as well as a dedicated core of spatial analysis capabilities such as thematic map creation or address matching.

Multimedia applications, on the other hand, are typically specialized in form and function with diverse purposes and audiences. As will be discussed later, this diversity is both an advantage and disadvantage for using multimedia in planning. Within the domain of planning, the leading models of multimedia applications can be broadly categorized as follows:

1. Multimedia applications as a prepared interactive tool to assist the public and professionals alike in understanding a planning subject.
2. Multimedia applications as a tool for collaboration in either a synchronous or asynchronous fashion.

3. The combination of multimedia data with other planning support systems such as relational databases or geographic information systems.

The three categories above are not exclusive, but rather provide a convenient model for discussing the primary focuses of the multimedia applications in this literature review. The subsections to follow discuss each of the three categories in greater detail.

### **2.1.1 Interactive Multimedia for Information Presentation**

The earliest, and still most prevalent, use of multimedia in planning is for the construction of interactive applications that allow a user to explore a particular topic in greater detail. These interactive experiences range from simple point and click presentation slides to explorations of the physical environment complete with a library of digital video.

The most common examples of this type of multimedia application today are the HTML (HyperText Markup Language) pages found on the World Wide Web covering almost every topic imaginable. The power of the WWW distribution channel to reach global audiences along with a relatively easy to understand mechanism for construction has made the HTML page an emerging standard in interactive multimedia both on and off the WWW.

More sophisticated examples of this type of multimedia application have also been developed outside of the Internet and WWW environments. An early multimedia application in the form of a 'city advisor' that provides information to planners and tourists in the City of Halmstad in Sweden is documented by Christiansson (Christiansson 1989). The City Advisor development environment is also notable in its foresight to provide multiple retrieval techniques for multimedia data including support for keyword searching, intelligent agents, and image browsing, in addition to hyperlinked attributes.

A large scale multimedia system is The St. Louis Riverfront Planning and Design System constructed to interactively synthesize and portray the issues surrounding the Mississippi shoreline (Kindelberger, 1992). This system included an extensive digital video documentary on laserdisc as well as some basic tools for manually modifying images in the system and automatically generating reports based on text and graphics in the system.

In 1993, the author completed a multimedia application to illustrate a design proposal bikeway system for downtown Windsor, Ontario, Canada (Tsui, 1993). This stand-alone system included excerpts from the city's Bikeways Master Plan, the author's design proposal and sketches, a photographic study of the surrounding environment, and 3-D animated sequences to assist in the visualization of both the design problems and solutions.

Quantitative measures can sometimes be difficult to fully understand if one has not been directly involved in its creation or works in a profession that constantly deals with such matters. In these cases, interactive multimedia representational aids such as sequenced images can be used to assist in understanding. This type of multimedia application was developed to help visualize real world traffic conditions based on parameters specified in the Transportation Research Board's Highway Capacity Manual (Shiffer, 1995). Time lapsed digital video was also used help explain traffic. Another good example of conveying information using multimedia representations is the Rantoul, Illinois, Collaborative Planning System (Shiffer, 1994). Here, digital maps, photos, and calibrated audio clips were used to demonstrate the probable impacts of re-opening a former airfield for commercial flight operations. Both the traffic and airport noise multimedia applications effectively demonstrate the value of using multimedia data in assessing planning issues.

Interactive multimedia applications for planning education is another area that has been an area for experimentation. Some examples include a multimedia system for better understanding GIS (Wiggins and Shiffer, 1990) and a system for teaching design students about spatial qualities in an urban environment using navigable digital video (George, 1997).

In this section, a number of examples of multimedia applications oriented towards conveying information to augment the understanding of a planning issue has been reviewed. With the continuing growth of the WWW, there will be no shortage of planning organizations pursuing the design of interactive web sites to disseminate information about important planning issues.

### **2.1.2 Multimedia as a Tool for Collaboration**

Moving beyond the interactive presentation of information, a more sophisticated class of multimedia applications can be found in the form of Collaborative Planning Systems (Shiffer, 1992). The multimedia applications described in the previous section or the average web page is generally designed with an audience of one in mind. The user is to move at his or her own pace through the material until the information they need has been found. By comparison, the Collaborative Planning System (CPS) is designed to function as a multimedia based information system to assist users engaged in planning activities as a small group. As envisioned by Shiffer, the CPS is designed with an emphasis on the ability to display, and retrieve multimedia data in real time. This is because users in a meeting environment tend to be intolerant of any delays incurred as a result of interacting with an information system (Shiffer, 1995).

It should be noted that the growing importance of networked computer systems as well as the emergence of a standard communication protocol in the form of the Internet's TCP/IP has lead to many different types of collaborative software. As of this writing, both

Netscape Communicator from Netscape Inc. and Internet Explorer from Microsoft Corp. have software packages that allow two or more remote parties to collaborate in real-time in a virtual whiteboard setting. These are, of course, only two such examples in a large market of groupware products; the point being that collaboration is one area where many good generic tools are being developed by the software industry that could have an impact on planning in the future. The challenge for those working with developing planning support systems will be in determining what sets of collaborative tools work best in the planning environment and less on building unique systems.

Collaborative planning systems need not always be operating in real time. Again, the popularity of the WWW has made the use of distributed asynchronous communication among interested parties a practical reality. Since users can access the WWW from both public and private locations, one model for a CPS is to design a web site that incorporates web-based bulletin board systems as a medium for disparate parties to converse and debate planning matters. This has been documented in a prototype for an environmental review process (Gouveia, 1996). The primary advantage of a web based system is its distributed nature which maximizes its availability to the intended audience.

### **2.1.3 Embedding Multimedia into Planning Support Systems**

Proprietary multimedia authoring tools dominated the development of multimedia applications in the early years of experimentation. While possessing powerful multimedia presentation capabilities, these tools were not designed with analytical functions, or large scale data storage and retrieval in mind. Lately, research has begun to focus less on developing multimedia applications with specialized tools and more towards incorporating multimedia objects into existing systems such as relational databases or geographic information systems.

As a planning support system, GIS provides a good foundation for multimedia applications because it provides a consistent standard for measuring, manipulating, and storing information about geographic locations. The embedding of multimedia objects into GIS has given rise to the term Multimedia GIS. In practice, the term Multimedia GIS has been applied to a broad range of applications. These range from web sites that feature 'clickable' maps as a primary metaphor for data retrieval, to the inclusion of multimedia data formats such as VRML and MPEG directly into the GIS software by leading GIS vendors (Hughes, 1996).

The issues governing the design, value, and uses of multimedia data in a GIS context are carefully outlined in an article by Fonseca, Gouveia, Camara, and Ferreira (1995). For the purpose of this discussion it is suffice to say that the benefits ascribed to multimedia usage in the previous sections apply equally in the multimedia GIS context. But what exactly is a multimedia GIS?

Examples in the research and trade literature seem to afford the title of multimedia GIS to any application that has spatial context whether they are actually constructed on top of a GIS infrastructure or not (Lang 1992, Hughes, 1996). However, it is very important to make the conceptual distinction between multimedia applications constructed using dedicated multimedia authoring software and those built upon a true GIS. The former usually has greater restrictions on its broad applicability given that the content and delivery scheme is usually embedded in the design of the system (Hartman and Sanford, 1994). By contrast, a multimedia application built atop a GIS should have the ability to integrate multimedia objects into the GIS' display and/or analysis capabilities. In this way, the multimedia application can have direct ties to the GIS or database as opposed to being an undesirable 'snapshot' of a dataset at a given time (Shiffer, 1995). This can be simply implemented as multimedia objects that are linked to particular GIS layers (e.g. pictures of houses tied to particular parcels). Another possibility is to simply store the reference to various multimedia objects in a standard relational database and construct a map based user interfaces to query and retrieve multimedia content (Rossel, 1994). This approach is made much easier by the widespread acceptance of the HTTP standard to reference web based resources. A more comprehensive approach to multimedia GIS is the design of a customized front-end interface that allows access to both multimedia objects and GIS functionalities. This can be done by either customizing the GIS itself through its accompanying scripting languages, by constructing additional functionalities in separate software modules and linking it to a GIS, or some combination of the two (Moreno-Sanchez et al, 1997).

Work with embedding multimedia content into various planning support systems is a relatively new area of interest in planning support systems. As a result, there are no definitive multimedia planning support system examples. Furthermore, the diversity of multimedia content and its possible applications in planning may mean that a standard definition never emerges.

## **2.2 New Directions in Multimedia Planning Support Systems**

The previous sections have explored some of the leading examples in the development of multimedia applications for planning. From this review, it can be said that multimedia applications in planning have evolved along three general paths: communication and presentations, collaborative planning, and augmentation of traditional planning support systems. These are not mutually exclusive categories (e.g. it may be possible to use a multimedia GIS in a collaborative setting or as a presentation tool), but are useful for examining the approaches to using and designing multimedia systems. Having reviewed the various uses of multimedia in planning, this section now turns to some general design issues and why it is important to continue pursuing this research.

### 2.2.1 The Design of Multimedia Planning Support Systems

The early usage of multimedia in planning was characterized by multimedia applications that enhanced communication about planning issues. While this application of multimedia may continue to be dominant in planning (especially since the emergence of the WWW), current research initiatives and technological developments are pushing the field of multimedia PSS into new directions. Consider the table below comparing several major characteristics of multimedia PSS.

**Table 2.1: Characteristics of Multimedia Applications in Planning Over Time**

	Past	Present and Future
Platform	Stand-alone systems; platform specific	Networked systems; platform independent
Design Standards	Proprietary systems designs	Open standards system design
Data Sources	Embedded “databases” of limited size tightly coupled with system design	Diverse interconnected data sources
Data Access	More static; localized data access	Dynamic and distributed data access
Goals	Emphasis on display and presentation of information	Towards using multimedia alongside analysis, evaluation, collaboration, decision-making

Research in multimedia planning support systems (including this thesis) are moving towards the construction of applications that are less likely to be independent systems and much more likely to be integrated with other data sources. In this way, the role of multimedia in planning support systems becomes much more accessible during the whole of the planning analyses, evaluation, collaboration, and decision-making process.

In many ways, the evolution of MMPSS closely follows the trends in the general computer industry. The Internet, and the WWW in particular, has provided a way for all sorts of organizations to distribute data. Software continues to be developed to take advantage of the networking paradigm and help developers create networked applications. Multimedia planning support systems developed in the future must seriously consider incorporating some or all of the emerging standards shown at the far right of the table. This is because many of the deficiencies identified in previous MMPSS research can be addressed in this fashion.



Consider the problem of updating data in a multimedia based planning support system. In the past, multimedia applications relied on having all its data (including multimedia objects) more or less “hardwired” into the tool itself. Updating the data most likely required some reprogramming of the application’s logic. This problem was exacerbated if the application was distributed to multiple parties in that either a redistribution would need to occur or the changes would need to be made individually to each separate application. Furthermore, who would do the work if the personnel is no longer familiar with the authoring system used to develop the application? What if the organization decided to switch from one type of computer system to another? Is it really worth the investing in the development of a multimedia application if only a relatively few people could have access to it?

The above issues were ones that only a few years ago were deemed overwhelmingly technical or expensive to address for a planning organization. While not all of the issues have been solved, the solutions are more practical to implement today than ever before. For example, the emergence of the WWW as a standard data network has made huge advances in distributing multimedia applications to the general public. Many government agencies provide web-based mapping or GIS services using technology that is a possible base for advanced multimedia applications. Industry movements for open standards surrounding Internet data transmission protocols and GIS data formats are making it easier for organizations to share digital data.

Although one specific form of MMPSS may not materialize it is very likely that the systems will take on the characteristics described in this section. The multimedia PSS prototype developed for this thesis has tried to consider as many of these new design objectives as possible.

### **2.2.2 On Further Research into Multimedia Planning Support Systems**

Having reviewed multimedia applications both recent and past, and having touched briefly on changing system design paradigms, it is worth pausing to consider the question of: why conduct further research into MMPSS?

Multimedia augmenting planning activities range from simple broadcast of information, to demystifying complex quantitative concepts, to real-time collaborative planning assistants, to enhanced understanding of the environment. These are concepts that have already been demonstrated in the research to date (Shiffer 1994,1995; Fonseca et al, 1994). On the purely technical level, further research into any of these or other areas is always needed given that the rapid advances in computer technology often allow us to solve or re-examine problems that were not previously addressable.

Research into planning support systems continues to be vital in order to provide quality information acquisition, processing, and management techniques. Such systems

are critical to the informed planning of communities, targeting social programs, and providing communication channels. The future of planning support systems is the provision of an information infrastructure that facilitates interaction between all parties involved in planning. Such a system should contain structured and accessible information (not just raw data) about the real world features of concern to the public (Klosterman 1997). Multimedia enhancements to current planning support systems features quite prominently on Klosterman's proposed research agenda including the augmentation of GIS to display graphics, sound, video, and other media. The ideal PSS should also have the ability to view analyses with charts, maps, and interactive video/sound (Klosterman 1997).

Klosterman and others (Moreno-Sanchez et al, 1997) advocate the integration of different computing application modules in order to construct planning support systems. Multimedia capabilities is one of the most important. The next chapter will describe one possible approach to integrating multimedia capabilities into planning support systems.

## Chapter 3

# **The Integration of Multimedia Objects in Planning Support Systems**

As mentioned in the Approach section of Chapter 1, this thesis chose to design and implement a prototype MMPSS with specific focus on a methodology for integrating and retrieving digital images about urban environments. This approach was taken for several reasons:

1. Attempting to intelligently incorporate all the different multimedia data types into an MMPSS, except in a very rudimentary way, would be an enormous amount of work given the thesis time constraints.
2. Digital images are an easily understood data type, easy to generate, and have many practical applications in planning.
3. The basic theory developed to index images can be extended to help characterize other multimedia data types in any case.

This chapter begins by describing some common planning scenarios where multimedia may be useful. In the course of examining these scenarios a number of general requirements are brought forth that the MMPSS described in this document tries to meet. This chapter then goes on to describe a technique for indexing images by use of 'spatial metadata'.

### **3.1 Scenarios Describing Multimedia Use and its Relevance on MMPSS System Design**

Multimedia objects have the potential to greatly enhance the communication of ideas about planning projects. The ability to see the configuration of physical space in the land use planning, urban design, architecture, and civil engineering fields is particularly vital in terms of understanding existing urban conditions. For example, land use planners frequently need to make site visits or refer to photos or maps of sites where some type of development or change is under consideration. Urban designers and architects rely extensively on being able to understand the built environment and its relationship to the proposed forms in their own design projects. Civil engineers might also take advantage of multimedia objects to investigate things such as existing road conditions or traffic congestion at intersections.

In each case, the integration of multimedia objects into analyses can be beneficial in clarifying problem parameters. For example, the engineer could certainly rely on textual descriptions or traffic counts to analyze the existing conditions, but images or video of the road surface would describe the specifics of pavement cracking better than a written paragraph. Likewise, the sight of a congested intersection or on-ramp could enhance understanding of the problem better than a quantitative value by pinpointing exactly where in the system bottlenecks occur. An open distributed multimedia system could not only show the engineer what is happening based on information collected by his department, but also allow other agencies or even members of the public to identify congestion areas or road defects.

The land use planner who needs to prepare a rezoning application could also do a better job if he were able to study the land parcel(s) the application would impact. A multimedia system that could systematically archive images, video, maps, or past studies (recall that multimedia objects also include other documents) about land throughout a city would be a major benefit. This would be especially true if the land use planner were a consultant working for a client in a city many miles away (where frequent site visits are more difficult). The ability to retrieve multimedia objects from a variety of governments, agencies, private developers, or the public, about a particular place could be of enormous benefit to policy makers trying to understand the dynamics of a place in relation to the current planning project. Today's multimedia application structure does not easily or efficiently facilitate this kind of interaction even if they are on the WWW.

Finally, in the case of the urban designers and architects, it is visual data about the environment that is often the foundation of design projects. The ability to see the architectural form or style of the surrounding areas and observe their existing functions is an often a critical ingredient to design. Furthermore, designers often need to collect data about places at a detail higher than the abstracted features a drafted map typically provides. This is typically done through sketches, photography, or video. Why not store and reference these multimedia objects in a way that would facilitate easy retrieval during the life span of the design project, for future use in similar projects, or even by other professionals and the public?

To summarize, the use of multimedia objects can have direct benefits to planners in many situations. This section has presented some ideas for the direction multimedia planning support system should take, including:

1. Open distributed architecture to increase accessibility of the MPSS and reduce storage capacity problems associated with centralized storage of multimedia objects.
2. A scalable and standardized multimedia object storage technique that supports distributed access and would allow for data exchange between organizations collecting multimedia objects for their own needs.

3. Tools to assist in the retrieval of multimedia objects beyond the simple point-and-click interface in use today.

Notice that there is a close relationship between the identified future directions of MMPSS in Chapter 2 and some of the more common working requirements of the users of MMPSS.

## **3.2 The Challenge of Integrating Multimedia Data**

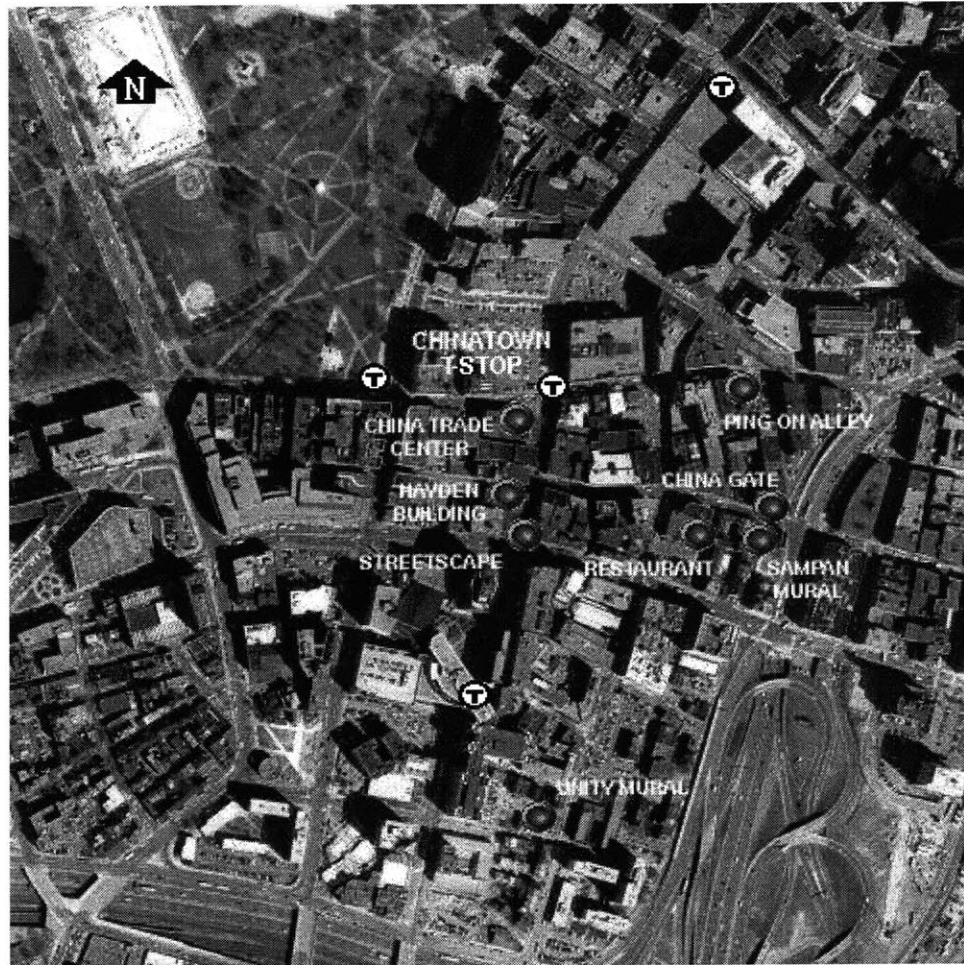
As described already, the primary objective of the MMPSS prototype (named MapVision) described in this thesis is to try to demonstrate an alternative way of integrating and retrieving multimedia data (specifically images). Naturally, before anything can be retrieved, data must be entered into the system and categorized. This section discusses how digital images can be systematically stored in an MMPSS.

The traditional approach to integrating multimedia data in planning support systems has been by way of an associative hyperlink. A hyperlink can be a piece of text, a button, a point on a map, or any one of numerous other variations on the theme. When a hyperlink is clicked it sends a signal to the controlling software to fetch and display an associated image, video, sound, or document. This is the primary metaphor that multimedia GIS applications use to retrieve multimedia objects stored in the system. But wait! Wasn't it the purpose of this section to talk about how multimedia data is integrated? As it turns out, the two concepts are intrinsically linked and the explanation for how to better integrate multimedia data is very much related to how it is to be retrieved. Therefore a slight diversion is in order.

### **3.2.1 Limitations in the Imagemap Based Interface**

The hyperlink metaphor has remained the dominant method of interacting with multimedia data because it is easy to implement and is the most obvious design strategy. Most MMPSS use a variant of the hyperlink (point-and-click) strategy of relating and retrieving multimedia data known generally as the imagemap.

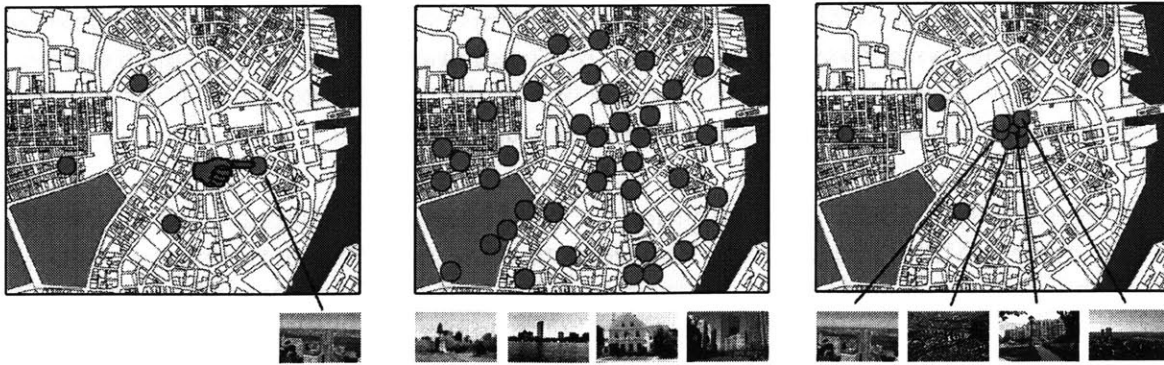
The imagemap - as its name implies - is an image. Within that image, a number of regions are designated as hyperlinks. Clicking within any one of these regions starts the process of fetching multimedia data. In many planning related multimedia applications the imagemap is actually a map or aerial photo of a defined study area (see Figure 3.1; next page).



**Figure 3.1:** Example of an Imagemap

*This imagemap was originally part of a WWW based multimedia application to provide information about Boston's Chinatown district. Users could click on the various 'buttons' to learn more about those particular points of interest. This is the predominant method of integrating multimedia objects into an application.*

The imagemap works well in systems where there are a limited number of hyperlinks, but begins to break down when there are many hyperlinks on the map and/or when the hyperlinks are very close together. The composition of a good imagemap is also subject to the design skills of the individual constructing the map. There is also the constant trade off between placing hyperlinks on the map and the readability of the imagemap itself. Figure 3.2 shows this problem with a series of hypothetical imagemaps of downtown Boston. Note that as the number of multimedia objects associated with the map increases, the actual legibility of the map itself begins to degrade (middle). Areas that are content rich in terms of multimedia objects can cause the hyperlink interface itself to break down (i.e. it becomes difficult to isolate one hyperlink from another and be able to click on it).



**Figure 3.2:** Problems With Integrating Multimedia Data Using Imagemaps

*The imagemap is a standard way of integrating multimedia objects and works well for small collections of objects (left). However, large collections based on the point-and-click metaphor of retrieval clutters the referencing map (middle). Many objects in very close proximity begin to degrade the usability of the interface (right).*

Part of the problem is that a standard imagemap is typically of a fixed resolution and dimension without the capability for display at varying levels of magnification. The ability to "zoom in or out" without loss of image resolution is the first step to building a more scalable interface. This has been accomplished by either dynamically generating new imagemaps as the user changes magnification or by placing the hyperlinks as entities in a GIS framework. The former solution (dynamically generating new imagemaps) is commonly seen in web sites that distribute street maps (e.g. MapQuest Interactive Atlas web site) or in emerging WWW-based GIS applications. The latter approach (placing hyperlinks within a GIS) also offers a solution to the interface problem since GIS packages have always had innate abilities to zoom and pan on a map. While both approaches can help reduce clutter on the visible imagemap by enlarging the scale of the visible area, there may still be problems with multiple hyperlinks clustered tightly together (e.g. imagine a series of images taken from roughly the same location).

### 3.2.2 Limitations in Point-and-Click Retrieval of Multimedia

While some of the deficiencies associated with the imagemap/hyperlink have been addressed through more sophisticated system design, it has not changed the fundamental nature of the interaction with multimedia objects. The basic metaphor remains the same - the user clicks on a location on a map to retrieve an object. The problem with designing the MMPSS strictly around the point-and-click interaction metaphor is that it considerably limits the way a user can interact with the multimedia data.

Suppose that Beth, an urban planner, is studying streetscape design for downtown revitalization and decides that Commonwealth Avenue in Boston, Massachusetts, might be a source of inspiration. This is a picturesque tree-lined boulevard contained by fine Victorian era housing and terminating at the famous Boston Commons park downtown.

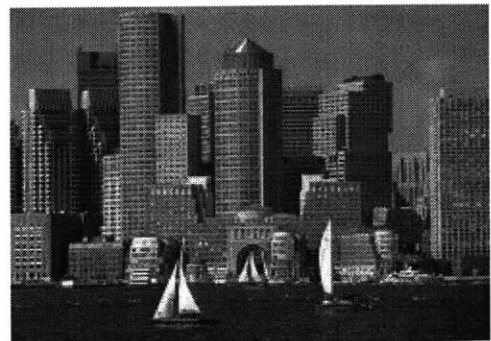
Beth has access to a MMPSS that she knows contains images of Boston so she decides to go through the system looking for pictures of Commonwealth Avenue. For the moment, do not worry about who has built the system or how the data was generated for the system. These issues will be addressed during later parts of the thesis.

Fortunately for Beth, there seems to be a lot of images in and around Commonwealth Ave. as the MMPSS imagemap display seems to show many ‘hotspots’ that she can click on. In fact there seems to be almost too many photos! Since Commonwealth Avenue is a rather interesting place in Boston, the contributors to the MMPSS’ database seem to have referenced a large amount of multimedia objects to this location over time. Beth begins to click on various hyperlinks that she thinks might be interesting, but soon realizes that the time it takes to transfer even relatively small images over the Internet is a time consuming task during mid-afternoon. Furthermore, although the locations of where images were taken are generally demarcated by icons on the imagemap she finds that many are not that useful. Some are merely images taken *from* Commonwealth Avenue, but actually look at other places (e.g. down intersecting streets). Other times she finds an interesting image or video, but cannot ascertain exactly what direction she is looking at (especially when looking at images of individual building facades). After some consideration, Beth decides that what she really is interested in are views along Commonwealth Avenue looking towards the Boston Common area from a variety of locations along the street. Unfortunately, the hyperlink icons littered along the map of Commonwealth Avenue do not offer any indication of what images in the system might be vista shots (much less shots looking towards Boston Common). Unable to tell the system either what she wants to look at, or where she wants to look from, Beth is consigned to continue manually browsing through the collection of images

As this scenario illustrates, there is a limit to the usefulness of relying exclusively on the point-and-click metaphor to retrieve multimedia data from a non-trivial repository of images. It is now finally time to turn developing a strategy for integrating images into an MMPSS to enable the functions that Beth would have ideally had in the scenario above.

### 3.2.3 Metadata for Integration

In order to provide for better ways of accessing images, it is first necessary to provide a better way of storing images - or more precisely, storing information *about* the images. Information about information (images in this case) is commonly referred to as *metadata*. It is important to provide metadata about images because a picture can be a representation of the world with highly complex semantics.



**Figure 3.3:** Example of the Complex Semantics Contained in an Image

*This simple image can mean many things to many people. How is a computer to understand what it means to the user?*



Take the picture shown in Figure 3.3 (previous page) as an example. What is this picture about? To residents of the Boston area it may be a picture of Boston's waterfront. To others from other places it may simply be a nice picture of a large city. Some may describe the picture as an example of a downtown or a central business district. However, it could also be a picture about sailing. Perhaps the picture represents even more abstract concepts like a sunny day, water based recreation, or high density. To a computer system, however, the picture is nothing more than a collection of bits. For the user to be able to tell the computer that it should give back images of Boston's waterfront the computer first needs to understand what is meant by the term "waterfront" or "Boston". This presents a fundamental unsolved problem that has been the subject of ongoing research in the field of artificial intelligence. To quote from a research article about querying images by examining its contents:

"Perceptual organization - the process of grouping image features into meaningful objects and attaching semantic descriptions to scenes through model matching - is an unsolved problem in image understanding. Humans are much better than computers at extracting semantic descriptions from pictures. Computers, however, are better than humans at measuring properties and retaining these in long-term memory." (Flickner et al, 1996)

Metadata plays an important part in helping computers understand human intentions because it helps to bridge the gap between requiring complete understanding by the computer and having a human manually do all the work of identifying images.

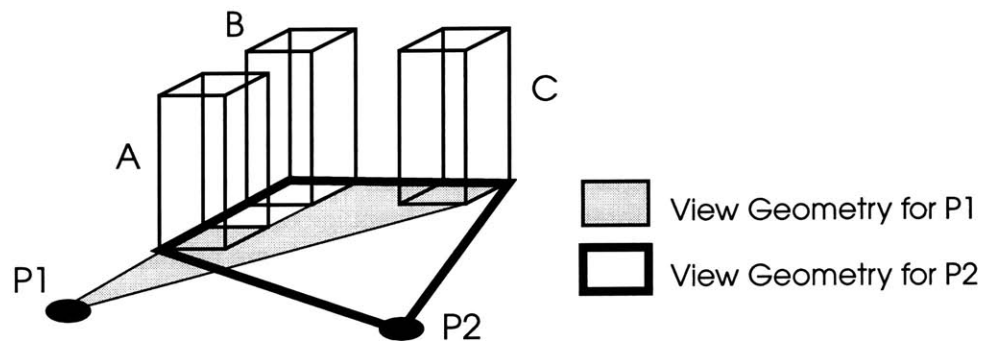
The most common form of metadata for all multimedia objects is textual description or keywords. For example, the Federal Geographic Data Committee's (FGDC) National Spatial Data Infrastructure (NSDI) project relies heavily on a very comprehensive and structured text metadata standard to catalog GIS data. In this case, the FGDC strived to describe a core set of metadata for spatial objects by constructing a standard with 334 different elements (FGDC, 1998). Among other things, this metadata is then used to help with the FGDC's task of providing a geospatial data clearinghouse (FGDC, 1998). Clearinghouse nodes are indexed searchable metadata catalogues that conform to Z39.50 library catalogue search protocols. Other geospatial data repositories such as the Alexandria Digital Library also take advantage of extensive metadata standards (UCSB, 1998). However, these metadata standards are primarily aimed at planimetric images and maps covering large areas (e.g. aerial or satellite photographs) as opposed to the oblique angled ground level images captured by individuals.

Text descriptions and keywords provide for a powerful encapsulation of semantics attributed to multimedia objects by humans. However, fully describing all images is time consuming and can be subject to a writer's vocabulary, his or her ability to draw broad meaning from a given image. Nevertheless, free text keyword descriptions are an important part of the metadata system employed in MapVision.

Keyword metadata schemes are useful because they can categorize images for nearly any application domain (i.e. keywords can contain concepts that are not limited to planning). Indeed, much of the work in multimedia databases by engineers and computer scientists have strived to build multimedia integration and retrieval systems that fit as broad as an application domain as possible. However, by realizing that the MMPSS does indeed operate in the specific domain of planning we can develop metadata that is optimized for this function. This is the fundamental concept that the MapVision integration strategy is built upon.

Knowing where a particular image was taken is also an important piece of metadata because it helps link the image to a particular space in the world. MapVision stores the real world geographic coordinates of each image in a coordinate system of that matches the rest of the MMPSS map contents (e.g. UTM, NAD, State Plane, etc.). Geography is important because so many planning tasks are about a particular place on earth. In the traditional design of MMPSS this is often the only piece of metadata that exists to help retrieve the image. Image location is important, but by itself is not nearly enough information (as previously discussed).

The most important piece of metadata in the MapVision system by far is the view geometry of each image. This is simply the field of view stored as a series of points that make up a polygon. More specifically, this view geometry is stored as a polygon in a map layer in a GIS compatible format. Theoretically, what falls within this polygon is what can be seen in the image. Again, planning tasks typically ask for images about places on the earth. One of the problems experienced by Beth the planner in the example scenario is that although she could find images taken *at* a certain location it was just as important that she find images *about* a certain location. In its most rudimentary application, the view geometry metadata helps MapVision determine if a certain requested point on earth is likely to be seen in the image referenced by the view geometry.



**Figure 3.4:** The Role of View Geometry in Determining What is Visible in an Image

*P1 and P2 represents two points where a digital photograph has been taken. A, B, and C represents buildings along a street. The visibility of each building in an image can be determined by whether the building intersects the view geometry for P1 and P2. In the example above, buildings A and C are visible for P1; buildings A, B, and C are visible for P2.*

This concept of view geometry represents a type of *spatial metadata* - defined as spatially referenced geometry that helps index and provide limited intelligence about data (e.g. images). As will be discussed in Chapter 4, *Techniques for Retrieving Images Based on Spatial Metadata*, spatial metadata can be used to make powerful inferences about what the image might contain and what are better choice images to return to the user given their specified interests and location on a map.

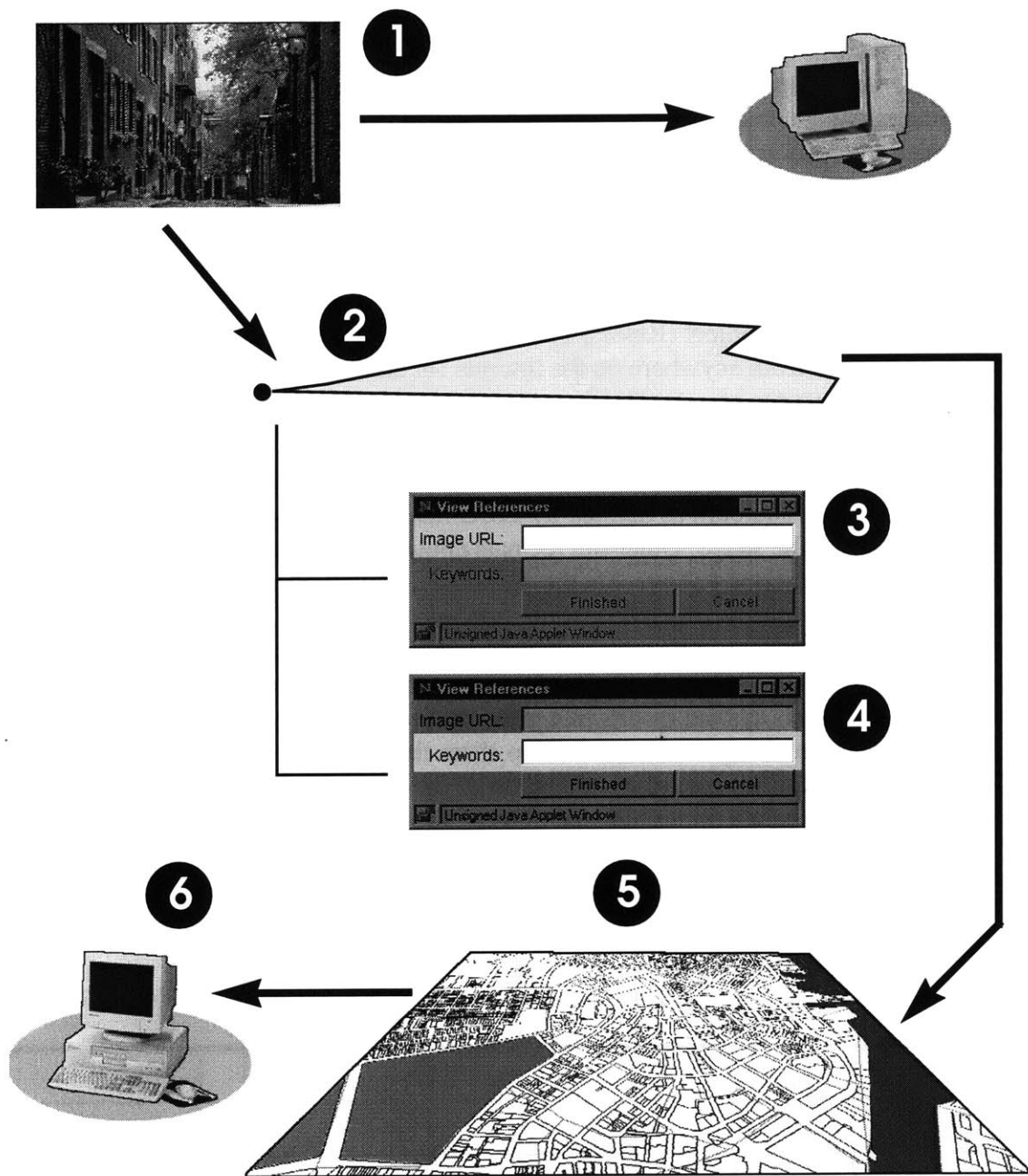
MapVision also needs to be able to determine where to find an image (or any other multimedia object) once it has been requested. Given that a distributed system based on open standards is the objective, the simplest way to do this is to store the URL (Uniform Resource Locator) of the object (Rossel, 1994). This allows the multimedia object to be located and retrieved from anywhere on the Internet (or an intranet) and helps alleviate the need for massive amounts of centralized storage to maintain the MMPSS.

In the MapVision system, all of the metadata types mentioned above is conveniently stored in the shapefile standard as defined by ESRI Inc. The view geometry defines the actual appearance of the shape. Other file formats could have been used based on the metadata standards described above, but the shapefile standard was adopted due to the availability of tools to easily write the information into this format. The multimedia component of the MMPSS can be relatively easy to adapt to a variety of planning support systems with a GIS base and is not confined to the methodology used to develop MapVision. Finally, this system results in a great deal of portability for multimedia content. Should different organizations wish to share their multimedia data, this methodology allows for the convenient transport of the content in the form of multimedia “layers” of spatial metadata rather than the objects themselves.

**Table 3.1: Structure of Metadata Database in MapVision**

SHAPE	ID	ORIG X	ORIG Y	AREA	URL	KEYWORDS

Table 3.1 shows how the various metadata components that describe any image would appear in the shapefile’s associated DBF database table. Not previously mentioned are the ID and AREA fields. The ID field simply provides a unique identifier to each image which could be used for joins with other tables in a RDBMS or GIS attribute tables. The AREA field contains the computed area of the view geometry (represented by the SHAPE field) in the corresponding map units. Figure 3.5 (next page) provides a general overview of the entire process of indexing and storing an image in the MapVision MMPSS. A more specific description can be found in the prototype description chapter.



**Figure 3.5:** The Process of Attaching Various Metadata Types to an Image

*The general steps needed to encode an image with metadata in the MapVision system. 1) The image is shot using a digital camera or created from photographs using a scanner and stored on a network accessible computer; 2) The view geometry describing where the picture was taken, its field of vision, and bearing is created and associated with the image; 3) A URL indicating the network location of the image is recorded; 4) Keywords are used to better describe the image's content; 5) The view geometry, its URL and keywords are saved as a separate "multimedia" layer in a GIS; and 6) Image can now be queried spatially as part of an MMPSS either locally or via the Internet.*

## **Chapter 4**

# **Techniques for Retrieving Images Based on Spatial Metadata**

Referencing images or other multimedia objects in a GIS is not new. A simple point on a map layer that a user can click on is the simplest form of spatial metadata. However, the MapVision prototype MMPSS allows the user to retrieve images about places in ways that would be very difficult, if not impossible, using the standard method of geo-referencing multimedia (e.g. points or simple polygons).

In Chapter 3, the concept of spatial metadata was introduced as a method for overcoming some of the problems in image retrieval in the planning domain. This chapter describes the many different uses that spatial metadata has in helping to intelligently retrieve images. MapVision makes use of many of the basic concepts described in this chapter, but does not implement all of the retrieval strategies presented here due to time constraints imposed on the software development stage. In fact, some of the techniques were developed only after experimentation with the prototype indicated potential problem areas not addressed by the original design. Inclusion of the non-implemented techniques was deemed necessary to demonstrate the full range and flexibility of the spatial metadata concept.

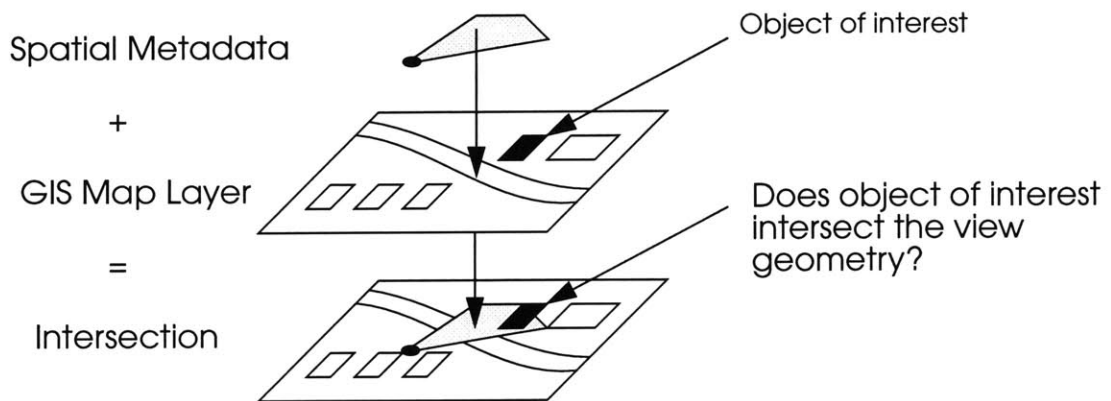
### **4.1 Providing Computable Attributes for Multimedia Objects**

The fundamental concept behind the use of spatial metadata is to extend the computability of multimedia objects. Recall that multimedia objects such as images are typically rich in complex (even abstract) meaning. Furthermore, computers perform very poorly in tasks requiring the understanding of semantic descriptions. Since it is not feasible at this point to endow a computer system with human-like reasoning abilities then the other option is to utilize the computer in areas that it does perform well in - areas such as rapid computation or searching. This is one reason why keyword indices are popular first steps in organizing multimedia objects. A computer knows how to search through a database of text. Spatial metadata is simply another way of indexing multimedia objects. It adds computable attributes to an otherwise difficult to compute multimedia object.

### **4.2 How Spatial Metadata is Used**

The advantage that spatial metadata provides is that it has the ability to not only geo-reference the location of the image, but provides information needed to determine if a certain object in the real world appears in the referenced image. This is possible because

properly recorded view geometry for each image will intersect certain regions on the earth (the region that the image shows). The computer merely needs to determine if the user's requested area intersects with the view geometries of any images in a database. Where an intersection occurs, the requested object on the map must be in one or more images. The process of handling these intersections and other spatial relationships between view geometry and objects in the real world is known as spatial querying.



**Figure 4.1:** The Use of Spatial Metadata Within MapVision

*Basic use of spatial metadata relies on the intersecting the view geometry of the image and map layers that contain objects of interest. When an object is wholly or partially within the view geometry then there is a high probability that the image contains the object of interest.*

Knowing this, it is now easier to understand why a GIS back end was chosen for MapVision. Geographic information systems have long possessed advanced spatial querying algorithms for use with digital mapping and analysis. MapVision simply reuses these same algorithms but applies them to the problem of retrieving images.

A summary of the types of spatial queries possible within the MapObjects framework is given below (the list is from selected functions found in ESRI, 1996). To clarify some terminology, a shape can be thought of as any entity on a GIS map layer. A feature can be thought of as another entity used for comparison with the shape (e.g. a user drawn rectangle, a polygon, a point, etc.)

- Searching features by distance
- Shape and feature boundaries overlap
- Shape and feature share a common point
- Shape and feature cross edges
- Shape and feature share a common line
- Shape and feature share common point or cross edges

- Shape and feature intersect
- Shape and feature intersect on interior
- Shape and feature intersect without touching edges
- Shape contains feature
- Feature contains shape
- Shape completely contains a feature
- Feature completely contains shape
- Feature contains first point of shape
- Shape contains feature centroid
- Feature is identical to shape

This list is not a comprehensive treatment of GIS capabilities even within the MapObjects development environment, but serves to highlight some of the more common querying functions available. In theory, since MapVision stores spatial metadata as a shapefile, most GIS querying algorithms are available for processing an image's view geometry. However, only a small number of GIS spatial queries were necessary to prepare the techniques described in the next section.

### **4.3 Image Retrieval Techniques Based on Queries on Spatial Metadata**

One important point needs to be addressed before performing an in-depth review of possible image retrieval techniques. If spatial metadata is simply a polygon layer denoting the field of view of an image, then why not just display these polygons as another layer on the imagemap and not develop retrieval strategies at all? Users can then simply return to the well known point-and-click interaction metaphor to retrieve the image. The reason why this is not feasible is that the view geometries will overlap in any system with a substantial number of images. Imagine photographs of a proposed development site. Even in a case with as few as two photographs, the view geometries of each image will overlap. Overlap increases as: i) more images about a particular subject are entered into the system; ii) more images are entered into the system over time (e.g. greater chance of different people taking pictures of similar area), and iii) images are stored which have vast fields of view (e.g. picture taken from atop a tall building overlooking the city or an airplane). As explained in Chapter 3, a cluster of overlapping hyperlinks has the effect of rendering a traditional point-and-click interface ineffective. If this is the case, then hasn't using spatial metadata merely made the problem worse while adding another layer of indexing?

In a very simple multimedia system, the answer to the question posed above is a resounding "Yes". However, this thesis is not interested in how to integrate and retrieve a limited amount of multimedia data (where simpler approaches could be used). This thesis is interested in finding solutions for integrating large amounts of multimedia data, allowing many organizations to participate, distributing access to the data over the Internet, and addressing some of the problems associated with the traditional point-and-

click metaphor that will only be exacerbated in a large system (as described in Sections 3.2.2 and 3.2.1). In such a system, spatial metadata becomes extremely useful by adding considerable intelligence to a simple mouse click as well as providing useful retrieval methods not possible any other way. The following review of spatial querying on spatial metadata will provide clarifications.

#### 4.3.1 Keyword Search

Out of all the techniques described in this chapter, keyword searching is one of the few retrieval strategies that does not make use of spatial metadata. Keyword searching is still very useful because it allows users to retrieve images by expressing complex or abstract concepts. For example, imagine a query that states, “Find images of 4th of July celebrations in Boston Common.” While it is easy to tell the computer where Boston Common is through a geographically based searching system, how is the computer supposed to deal with the concept of celebration? No known computational technique today could extract such a concept from an image. However, if the image were associated with keywords such as “festival”, “party”, “holiday”, “gathering” then it may be possible for the user to find such a picture by querying with a number of synonyms for “celebration”.

One of the problems of an data retrieval system based entirely on keywords is the potentially overwhelming variety of descriptions that any one picture, video, sound, or document can produce. Under the MapVision system, the problem is alleviated slightly in that references and relationships to place are handled by spatial metadata. This means locational keywords can now be eliminated and focus can be placed on describing less tangible concepts. It simply is not necessary to provide descriptions such as, “[picture of] Main Street looking North standing beside Joe’s General Store at Fifth Street”. None of these words are necessary because the user can directly transfer these concepts to MapVision by performing queries on the interface. Instead, keywords should focus on capturing more subtle qualities, “[picture of] commercial activity during Christmas holidays with crowds of pedestrians moving through snow”. Under the MapVision system, the proper coding and usage of keywords can provide a powerful supplement to spatial queries.

#### 4.3.2 Simple Point Query

Although much has been made of the problems associated with retrieving multimedia data exclusively with a point-and-click interface, a place remains for this standard interaction within MapVision. The user can always simply point to any location displayed on a map layer and click to retrieve any available data. Unlike other multimedia systems where a click is sent to an object (e.g. button) that is waiting for the action, a click in MapVision transmits a real world locational coordinate to the MapVision Server. This

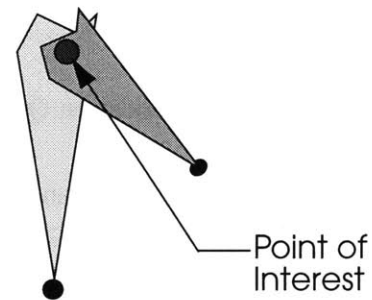


coordinate can then be used in a search against a database of spatial metadata that is linked to images.

From the user's perspective the simple point query seems to behave very similarly to the standard imagemap (only without the icons marking hyperlinks). However, there is a great difference in how MapVision decides what to send present to the user. Since MapVision uses spatial metadata to determine what objects on earth are most likely to be visible in an image it can do a much better job of gathering multimedia data that a user requests. Suppose that a user is interested in finding images about a house. Under the standard imagemap implementation, the user will generally try to look for hyperlinks on or near the house to click on. This works fine if the images of the house are relatively close by and/or the designers of the imagemap has decided to place links about a particular object ON the object. However, what if there is an equally good image of the house that is from across the street, or down the street, or as seen from a hill many blocks away? This presents some problems.

One, if the MMPSS were to be designed to properly geo-reference *where* the image was taken then any images that are taken from points further away from the object of interest may be missed - even if the object shows up in those images. Two, if the MMPSS were to be designed so that it assigns hyperlinks to all subjects that appear in an image then this can lead to an unbearable proliferation of redundant links (e.g. consider how many links one high angle photograph of a neighborhood could generate).

The Simple point query based on spatial metadata eliminates the above problems because it performs the query by seeing if a coordinate supplied by the user falls within the view geometry. Assuming that an image's view geometry covers everything that is shown in the image, then MapVision is able to retrieve said image(s) regardless of the photo's location. Figure 4.2 (right) shows how both the short range shot of an object and a long range shot of an object are returned whereas in the traditional implementation there is a much greater chance of missing more distant views. These distant views may be very important; such as in the case of urban design when it is desirable to not only see an area of interest, but the surrounding area as well.

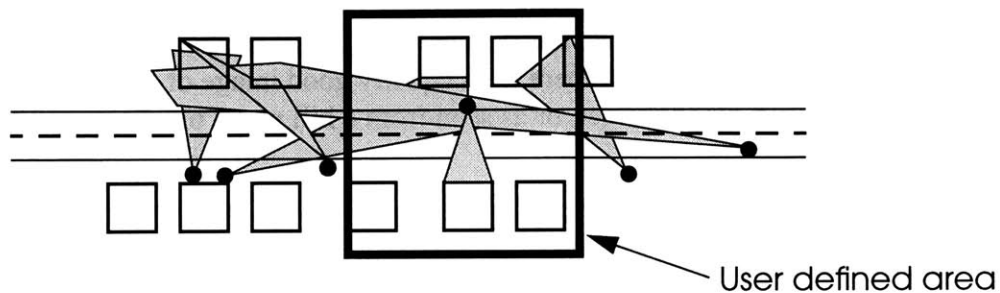


**Figure 4.2:** Simple Point Query  
*Spatial metadata helps ensure that all images containing the point of interest are retrieved.*

### 4.3.3 Area Query

The area query is merely an extension of the simple point query in that it allows users to search for images over a larger area. In MapVision, tools are provided for the user to interactively define several useful types of regions using rectangles (e.g. for a city block),

polygons (e.g. along a waterfront), or circles (e.g. distances from a point). MapVision uses all of these regions in similar ways. The system intersects the user's region of interest with the image database and returns all those images that have view geometries falling within the user's region. Figure 4.3 (below) illustrates this concept. Area queries are useful for helping to return many multimedia objects in one interaction. Considering that each interaction in a distributed system generates network traffic (at least under the current state-of-the-art in web client/server models), area queries can help the user gather information quickly by eliminating the need to make repeated calls to the server through repeated simple point queries.



**Figure 4.3:** Area Query

*In an area query, the user defines an area of interest and the system will retrieve all images with view geometry that intersects that region. In the example above, four images will be retrieved to show part of a streetscape.*

#### 4.3.4 Location-to-Location Query

Both the simple point and region queries are merely minor enhancements to the standard imagemap interface. Queries on spatial metadata has a lot more to offer. Many times in physical planning exercises it is important to be able to visualize what a certain building, street, or neighborhood looks like from certain vantage points. MapVision has the ability to support such exercises by using a location-to-location query.

As its name implies, the location-to-location query allows the user to interactively specify two points on the earth - a view FROM point and a view TO point. MapVision then takes the given points as a defining a line of sight and finds images in its database that approximates that sight line. Figure 4.4 (next page) provides a good starting point for explaining how this technique works. MapVision uses the FROM point to try and situate the origin of the viewer to the origin of an image and the TO point to locate an object of interest. The latter is easily performed using the simple point query, but the astute reader will note that it is nearly impossible (even for a very large database) to match the viewer's requested position on earth exactly with those in its image database (e.g. what are the odds that the database will contain an image taken from exactly [235645, 889403] in State Plane coordinates?).

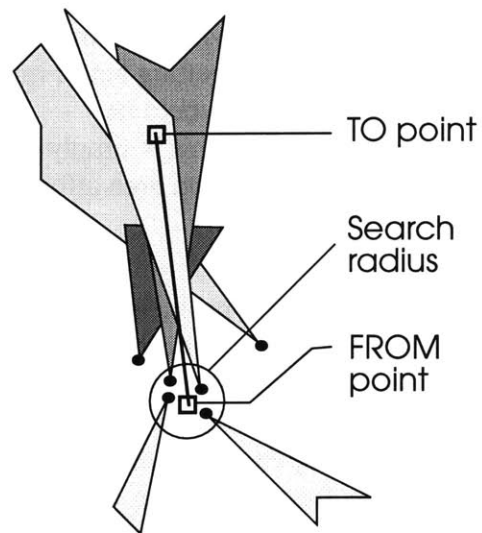
Given that the only way of achieving an exact view of any location in the real world is to just go there, a computer system could do worse than return the closest possible matches to the describe sight line. This is achieved by applying varying levels of tolerance on the specification of the origin of the image/viewer. Recall that spatial metadata stores the origin of the picture in real world coordinates. MapVision takes the user specified vantage point and begins to search from that point out to a determined distance (user or system specified) looking for the origin points of images in its database. The images that are returned are those with an origin within “x” distance of the viewer’s requested vantage point *and* has view geometry that contains the TO point.

Although not implemented in the Map-Vision prototype, a more advanced location-to-location query would be able to measure each returned image’s origin from the requested origin so that the system could note what images have origins that are closer to the requested point than others. Furthermore, some computation might be performed on the TO point in order to note whether the point is barely contained by a certain image’s view geometry or whether it is closer to being a dominant object in an image. A theory for how this might be assessed is covered in Section 4.3.6.

#### 4.3.5 Including/Excluding Extreme Areas

The area covered by an image’s view geometry is directly related to the breadth of subjects that can be seen in the image. A close-up of an entrance to a building would be an instance where the view geometry would be very small. The door may only be a matter of meters away. On the otherhand, an image of a city skyline taken from a few kilometers away will have a view geometry that covers very vast distances. The latter is the problematic case.

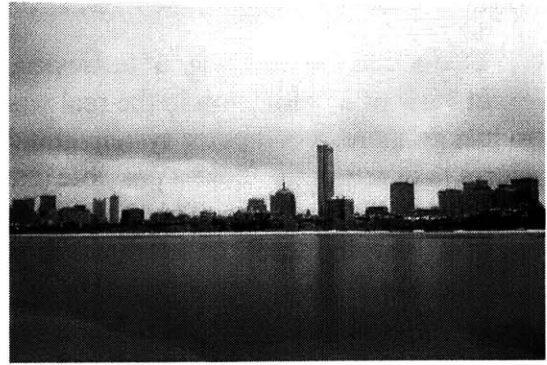
Long distance shots of the urban environment creates large view geometries which will then be frequently returned in queries even if they have no bearing on the subject matter. Figure 4.5 (next page) shows an image of the Boston skyline taken from the City of Cambridge (across the Charles River). Note the prominence of the tall buildings, but also the hundreds of the small buildings in the foreground that are at least partially visible.



**Figure 4.4:** Location-to-Location Query  
*The system searches within a certain radius of the vantage point (from point) to locate image origins and compares it with what the user wants to see (to point). In the above example, two images would be retrieved.*

Although these small buildings do technically show up in the image, the value of continually receiving this image in queries focused on getting better images of the small buildings would be counterproductive. This is simply a function of the large intersection area afforded to long distance shots.

To counter this problem, an extreme area filter can be used to automatically suppress any images that have areas above a certain value. This value could be determined by the user, computed as number of standard deviations from a mean, or an “xth” percentile cut-off could be devised. Of course, this technique could also be applied in reverse to force the system to only consider very large sweeping landscapes by filtering away view geometries below a certain threshold. Indeed, the area measurement provides one way of making inferences about the type of shot contained in an image.



**Figure 4.5:** Example of Image With Extremely Large View Geometry

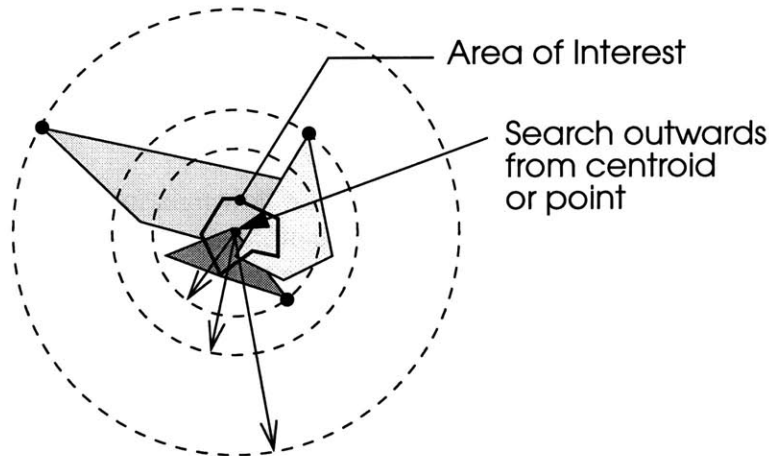
*This image is a may be a good reference for the tall buildings or the Charles River, but not so great for the hundreds of small and illegible buildings along the shore.*

#### 4.3.6 Deducing Subject Prominence by Distance from Origin

Sometimes it is useful to know if an object or area of interest figures prominently in an image or not. Again, a good example is if the user is interested in seeing images about a very specific building or feature. Imagine that a user wants to find visual references to the Massachusetts State House. This is a building with some visual prominence near downtown Boston, but it can also be seen from further locations due to its unique gold dome and position atop higher ground. This means that parts of the State House could potentially be found in many images - some of which are more useful than others. The extreme area filtering process described above can help in this situation, but there may still be instances where the State House may show up in an image that is not one with an “extreme area” (e.g. the State House appears in the background of some other focus of attention).

There is no way of definitively stating whether a given image will feature an object of interest prominently (e.g. one that shows the object in a way that makes it the primary focus of attention) using spatial metadata techniques. However, it is possible to find images with a greater probability of being good images using spatial metadata. One simple way of approaching the problem is to make the assumption that: of those images that show the object of interest, those images with origins closest to the object will feature the object most prominently.

This is illustrated in Figure 4.6. First, a point is specified to be the point of interest (if a region is used then take the centroid). Next, a set of images that intersect the point is formed. The system then sorts the set by distance of each image's origin to the point of interest in ascending order. Under this technique, the result set highlights those images that were taken closest to the point of interest hoping that close proximity will equal subject prominence. Similarly, the technique can be used in reverse by simply sorting in descending order. This will give the most emphasis to shots that are taken furthest from the object of interest.



**Figure 4.6:** Deducing Subject Prominence by Distance from Origin

*Where the subject of interest may be in an image can be estimated by looking at the distance of the image's origin from the subject of interest. Those images with closer origins may feature the subject more prominently.*

Of course, this technique may only be partially effective in helping to isolate images featuring an object of interest. Being very close to an object such as a building is not always helpful in forming an image of the building as a whole (e.g. a close-up shot of an entrance). Also, this technique has no way of knowing whether intervening objects obscure the object of interest (e.g. parked cars, people, signs, etc.).

#### 4.3.7 Juxtaposing GIS Coverages or Databases With Spatial Metadata

Since spatial metadata is stored as a shapefile with a related DBF database file. It is possible to treat a spatial metadata layer of shapes as another standard map layer in a GIS. Likewise, it is possible to make queries on the DBF portion of the shapefile using a variety of relational databases and SQL (Structured Query Language).

For example, imagine a situation where our favorite MMPSS using planner, Beth, wants to find some references to townhouse designs. Assuming that she now has access to

an MMPSS designed along lines similar to the MapVision prototype she could begin by using a keyword search on “townhouse” or “rowhouse”. Since she is interested in getting results from any part of the city she does not really need to use the spatial metadata. Or does she? After thinking about what she wants a bit more, Beth realizes that what she really wants are examples of townhouse projects that have been developed in or near wealthier neighborhoods. Since she is now using an MMPSS integrated with GIS and database systems it is now possible for her to formulate a sequence of queries that will: 1) let her define wealth based on available demographics (e.g. census data); 2) identify residential areas based on community information (e.g. zoning maps); 3) visualize these areas as maps (e.g. GIS thematic maps); and 4) perform an intersection between these areas and the spatial metadata enabled multimedia objects. Beth quickly isolates those parts of the city that she has defined as being suitably wealthy and are residential areas. She can now perform a quick series of region queries for relevant multimedia data.

What the above example shows is an example of how tight integration with GIS/databases can allow a user to quickly target a topic of interest. In essence the GIS/database support can provide some powerful tools for helping to narrow the search domain. Some work was begun in this area during the development of the MapVision server that yielded interesting results. However, it was decided that the prototype should be devoted to demonstrating those techniques that had not been commonly used to retrieve multimedia data. SQL querying of images has been demonstrated in the field of multimedia databases on several high end database engines. The example above could have also been implemented with nearly any good desktop GIS package (provided that the multimedia database with properly constructed spatial metadata exists). Therefore, it was deemed redundant to engineer such functionality into a prototype given the short span of time available for development. A fully functional MMPSS would give serious consideration to this powerful aspect of working with multimedia objects tied to spatial metadata.

#### **4.4 Extending Spatial Metadata**

This chapter presented some of the most straight forward and useful image retrieval techniques that can be applied to images indexed with spatial metadata. These are only the possibilities that have been developed using a very simple spatial metadata structure. It is possible that more sophisticated techniques can be developed by increasing the descriptiveness of the spatial metadata. One such additional descriptor may be the addition of a bearing field to record the absolute direction that the person was facing when an image was taken. How this might be used is described in Chapter 7 on advanced MMPSS development.

The lessons that have been learned in developing the spatial metadata system for MapVision can be summarized in three points. These points are worth considering when extending or developing the spatial metadata format:

1. What is being represented by the spatial metadata? In this thesis, the metadata represented the field of view of a camera. Are there other qualities about multimedia objects that can be represented spatially?
2. How does the metadata element enhance the understanding or use of multimedia objects in the MMPSS? The techniques described in this chapter allow for retrieval of images in ways that are valuable to people who need to visualize the urban environment. If the needs are different, perhaps the spatial metadata should be changed to best serve those needs.
3. What is the overhead (time and knowledge) required to create the spatial metadata? There needs to be a balance between trying to provide comprehensive integration and retrieval for every possible contingency and providing a usable service within an acceptable time frame. This is especially true for organizations with limited funds, human resources, and time to build a multimedia database.

Finally, thinking about spatial metadata development should not be limited to static images either, but can be developed for digital video and possibly audio. For videos where the camera does not move, the techniques described in this chapter are equally applicable. However, both digital video and audio have the added dimension of communication over time. Is it possible to encapsulate some of this information using spatial metadata? How can a moving camera or sound source be indexed in a meaningful way that facilitates retrieval? These are just some of the questions that future work in MMPSS can explore.





## **Chapter 5**

# **Review of the MapVision MMPSS Prototype**

This chapter contains a tour of the MMPSS prototype developed as part of the thesis research. It will begin with a look at the functions available to the client for working with multimedia objects and then turn to the features found in the server to support those functions. The chapter will conclude with observations about the design and operation of the MapVision system.

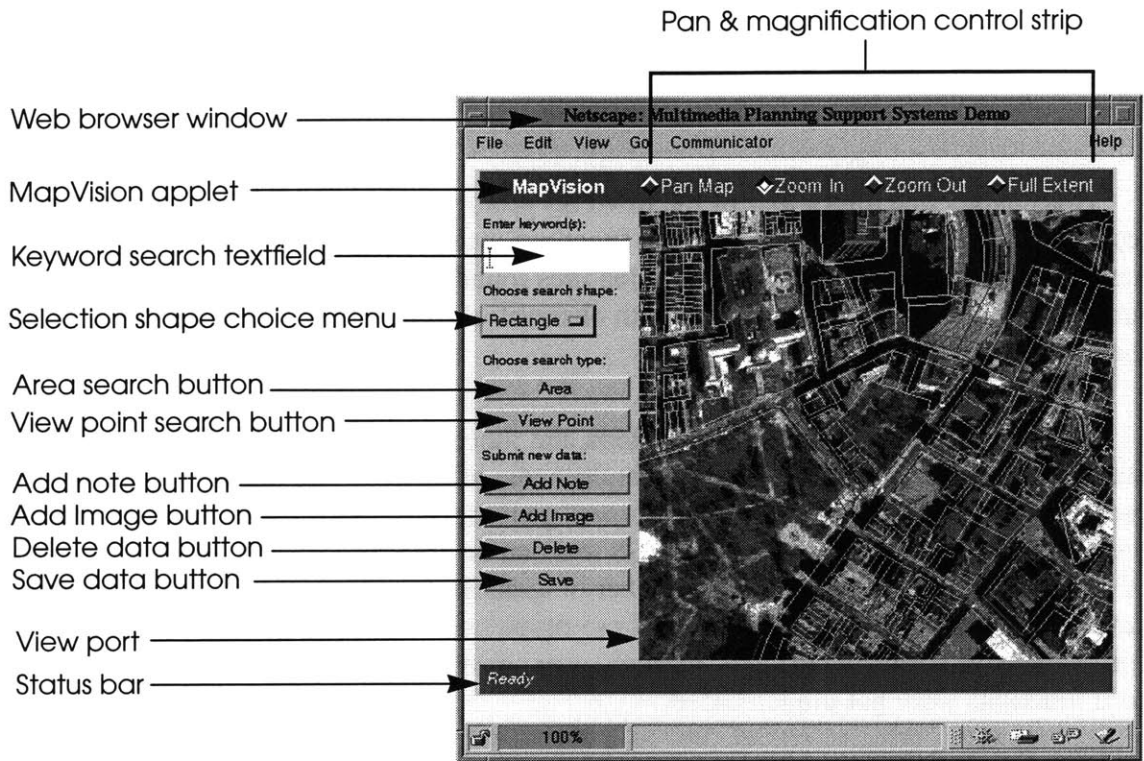
It is important to remember that the look and functionality of the MapVision system is but one possible implementation of the multimedia integration and retrieval strategies presented in this document. The development goal for MapVision is to provide a working prototype for a more generalized MMPSS that can be used to store and retrieve a variety of multimedia objects for a diverse number of applications. Furthermore, MapVision was developed to provide a proof-of-concept prototype rather than to engineer a comprehensive software package. As such, user conveniences beyond those necessary for core functions were not implemented.

### **5.1 MapVision Client**

The user interacts with MapVision and its information base through a world wide web accessible client. The client is composed of two parts: a Java applet that provides client-side data handling and an interactive interface for MapVision; and a Java capable web browser that hosts the MapVision client applet. For the purposes of this chapter's discussion, the term client refers to the MapVision Java applet unless otherwise noted. Figure 5.1 (next page) points out the various components in the MapVision client described in the next few sections. The communication between MapVision's client and server will be discussed in greater detail in Chapter 6.

#### **5.1.1 Basic Navigation Functions**

The MapVision client is built upon a set of Java classes included with ESRI's MapObjects Internet Map Server product. These classes provide the core visual navigation functions that users expect such as panning, zooming, viewport graphics refreshing, and interactive construction of basic geometric shapes. The navigation functions are available to the user at all times. By using an active content client (as opposed to regular web pages), MapVision can provide a high level of user feedback during navigation (e.g. real time panning, drawing of bounding boxes for zooming).



**Figure 5.1:** MapVision Java Client

*The MapVision Java client can be embedded within a web page and viewed in any Java capable browser such as Netscape Communicator.*

### 5.1.2 Keyword Searches

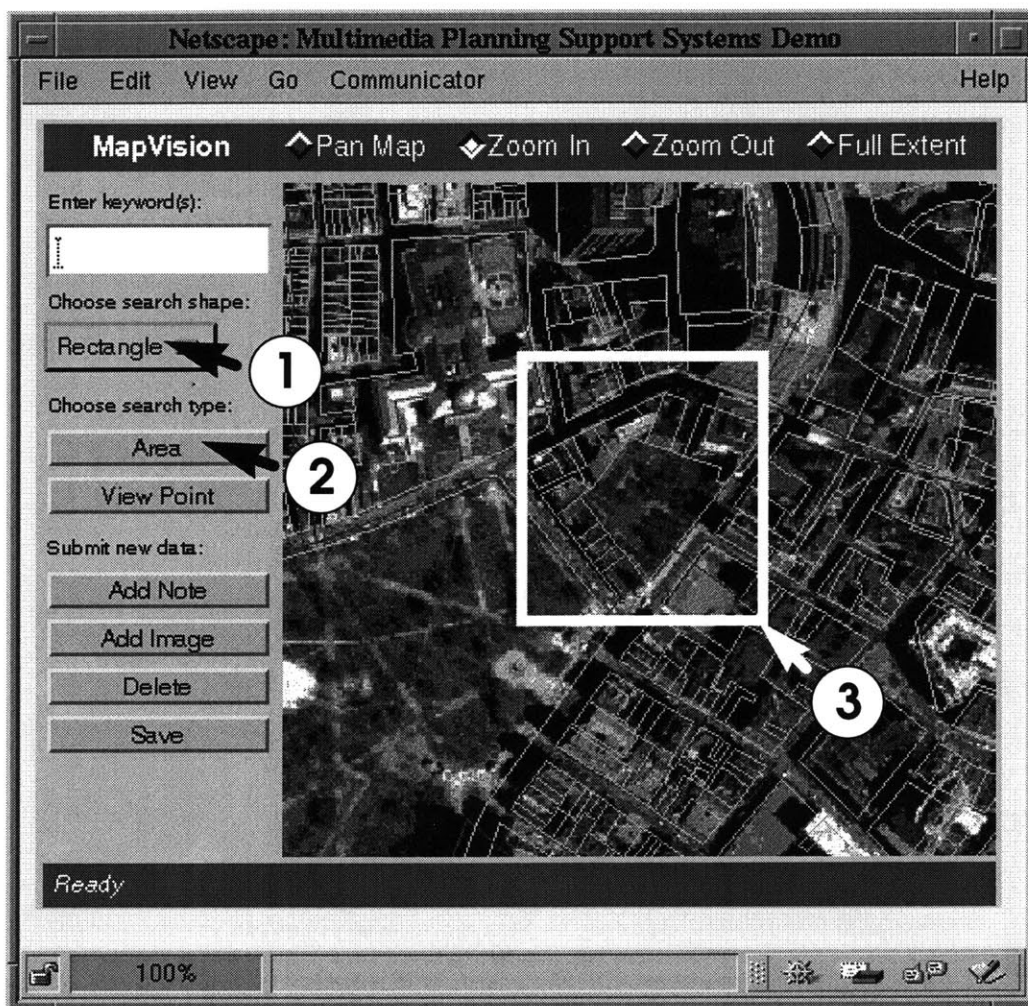
Users can look for multimedia objects in the MapVision Server's database of references by means of keyword searches. To execute the keyword search, the user simply types in the words or phrases describing the type of material he or she wishes to search for in the keyword search textfield. Pressing <Enter> executes the keyword search. Optionally, the user can enter keyword(s) into the textfield and use them in conjunction with the area search described below. In this case, only those objects matching the keyword(s) AND are situated within the specified area will be returned.

### 5.1.3 Area Searches

MapVision allows users to delimit certain geographical areas of interest and then search for relevant multimedia objects within these areas. This function is invoked by choosing a selection shape (i.e. point, rectangle, circle, or polygon) from the selection

shape choice menu. The user then clicks on the *Area* search button and draws the selected shape in the viewport to delimit a search area. Once the area has been defined the client automatically sends a query to the server to find multimedia objects in the defined area. Figure 5.2 shows the steps involved. Keywords specified in the keyword textfield will act as further constraints to what is returned by the server.

Area searches are useful for returning collections of multimedia objects related to a place through the use of just one interaction as opposed to repeatedly clicking on large numbers of hyperlinks. Notice that the viewport does not need show the actual locations of any of the multimedia objects that may be referenced to the outlined area. This makes the map and image in the viewport easy to read.

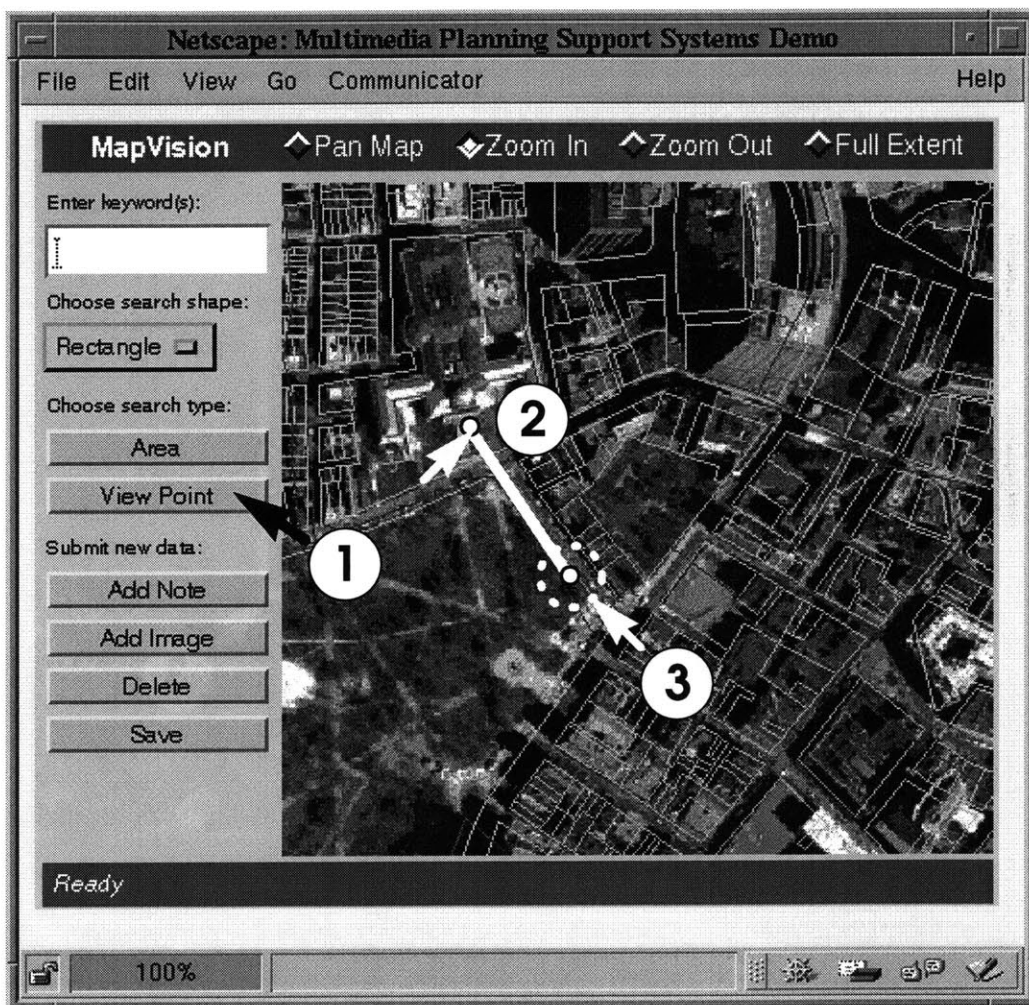


**Figure 5.2:** MapVision Implementation of Area Searches

*Users can specify specific areas in the viewport to search for multimedia objects. The process begins by (1) selecting a search shape from a choice menu; (2) clicking on the Area search button (2); and then (3) drawing the shape in the viewport.*

### 5.1.4 View Point Searches

By making use of spatial metadata, MapVision can help retrieve images of a point that the user is interested in looking *at* in relation to a point where the user wants to be looking *from*. Specifying this search is relatively simple in MapVision. The user starts by clicking on the *View Point* button to activate the search. In the viewport, the user clicks and holds on a point that he wishes to look *at* and then drags a line to a spot where he wants to look at that object *from*. After letting go of the mouse button at the point to view from, a bounding circle is formed. The bounding circle is used to specify the search radius to look for the 'from' point since it is generally not possible for the user to be able to specify a 'from' point that will precisely match a point in the database (see Section 4.3.4 for more information about why this is so). This process is illustrated in Figure 5.3.

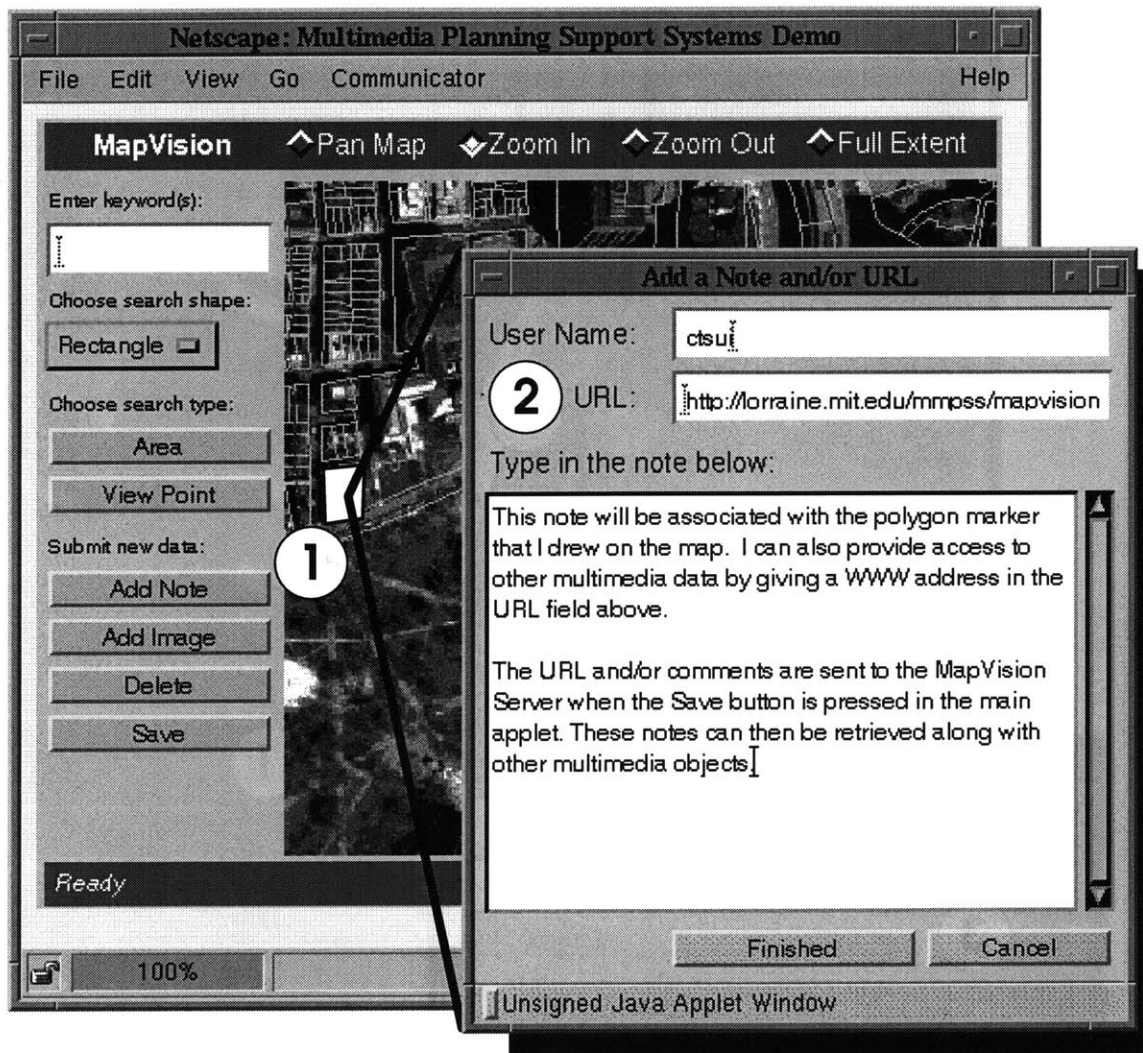


**Figure 5.3:** MapVision Implementation of View Point Searches

*Users can specify both what they want to see and where they want to see from. The search begins by (1) pressing the View Point button; (2) clicking on the point of interest and dragging out a sight line; and (3) drawing a search radius to locate the 'from' point.*

## 5.1.5 Submitting Notes

MapVision also allows for the limited submission of georeferenced multimedia data. In the context of the MapVision client, a note is the most basic unit of annotation. It consists of a URL to a web resource and/or ASCII text. The user can choose to annotate any location shown in the viewport by clicking on the Add Note button and then drawing a polygon outlining the region to which the note applies. Once a polygon marker has been drawn on the map, a form appears where the user can supply a user name, a URL, or a text description. The note is not considered finished unless a user name is supplied and a URL and/or text is entered. Figure 5.4 shows the data entry form and client. Notes are not available for querying until the data is saved to the server.



**Figure 5.4:** Submitting Notes in MapVision

*Notes are the basic unit of annotation in MapVision. It consists of a polygon marker (1) showing where the note applies; and some information (2) such as the user's name and the URL and/or text message.*

### 5.1.6 Submitting Images

The MapVision note described in the previous section is a generic annotation mechanism suitable for a broad range of applications. By providing a URL reference to a web resource, any multimedia object that can be delivered via the HTTP protocol can be accessed. However, as discussed in Chapter 3 (and further elaborated on in Chapter 4), the use of spatial metadata to reference images can greatly improve the range and accuracy of queries. The MapVision client provides for the submission of images based on the spatial metadata standard described in this document. Figure 5.5 shows the MapVision implementation for submitting images via the client.



**Figure 5.5:** Submitting Images in MapVision

*Image submission is based on the definition of view geometry. The user begins by: (1) defining the origin point (near #1 in this case) and (2) the field of view of the image. The URL and associated keywords (3) are then entered in a separate form.*

The process for submitting an image is similar to that described for submitting a note except that the constructed polygon has some very specific meanings associated with it. First, the initial point of the polygon is interpreted by MapVision as being the origin point of the image (e.g. where the person was standing when they took the picture). Second, the polygon should be drawn to define the area that can be seen in the picture. This polygon represents the view geometry of the image as described in Section 3.2.3.

Once the polygon defining the view geometry has been created in the viewport, the user can specify the URL to where the image is stored and any associated keywords. As in the case for submitting notes, the submitted image is not available for querying by other MapVision clients until the data is saved to the MapVision server.

### 5.1.7 Saving and Deleting Data

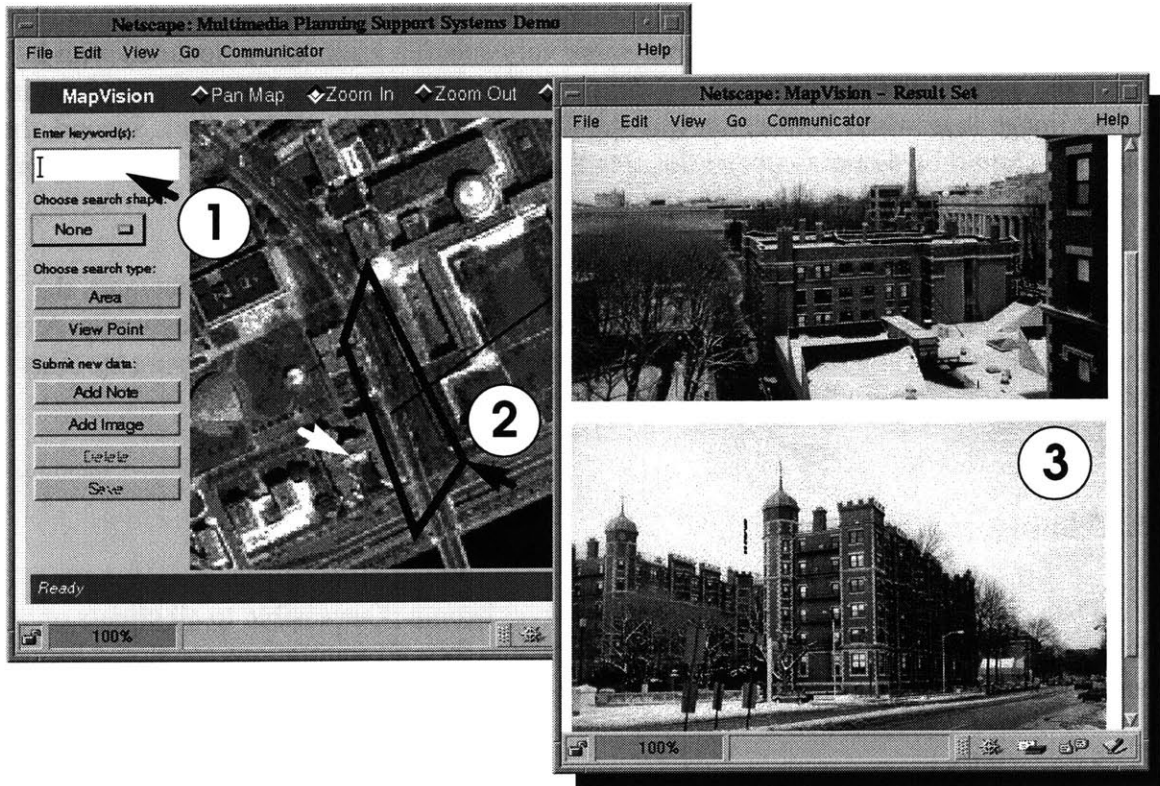
Data that is meant for submission to the server becomes accessible to all clients only when it is saved by pressing the Save button. Prior to this, all annotations are stored locally and will be lost if the user terminates the session before saving.

Users also have the ability to delete annotations (notes and images) that they have generated in during their current session. At this time, no provision has been made to allow any client to delete data from the MapVision server in order to avoid complex user authentication and security problems that are beyond the scope of this investigation.

### 5.1.8 Displaying Retrieved Multimedia Objects

Up until this point, no mention has been made of how multimedia objects matching the search criteria are returned to the user. While there are potentially many different ways of handling this, the MapVision client displays retrieved multimedia objects by spawning a separate browser window under the client's control. This window then receives dynamically generated HTML pages from the MapVision server that contain either the requested objects or the hyperlinks to the objects. Figure 5.6 (next page) illustrates what may be seen from the user's perspective after MapVision provides images in response to a query.

This display scheme is both simple and powerful. It allows the results of a user's session with MapVision to be saved as collection of HTML pages and recalled for later viewing. Transmitting result sets by way of dynamically generated HTML allows for maximum flexibility in customizing format of the response. By using the web browser to view the multimedia objects, MapVision takes advantage of the web browser's advanced multimedia content handlers. It also avoids security problems associated with saving data from Java applets to a local computer.



**Figure 5.6:** Displaying Retrieved Multimedia Objects in MapVision

*MapVision displays any retrieved multimedia objects in a client controlled web browser window. In the example above, the user has formulated a combined area and keyword search for images of Ashdown House at MIT along Massachusetts Avenue (indicated by white arrow). The query starts with: (1) specifying "Ashdown House" in the keyword search textfield; (2) drawing a polygon enclosing the region of interest. The results (3) are displayed in a separate web browser window. Note that the upper image in the result window is actually an image taken from Ashdown and does not show the building itself. It was returned because MIT's main entrance can be seen and that was part of the area search.*

## 5.2 MapVision Server

Although the client gathers the user's input and displays any matching results it has only limited computational capabilities. The work of interpreting the client's directives, storing multimedia object references, and performing queries, and delivery of output is performed by the MapVision server. The server is a Microsoft Visual Basic application composed of MapObjects GIS components and networked to a Netscape FastTrack web server using the MapObjects Internet Map Server control. All references to the server in this chapter refer to the MapObjects Internet Map Server unless otherwise noted.

The MapVision server can actually perform many of the functions described in the client section including: all navigation functions, area searches, submission of images, and

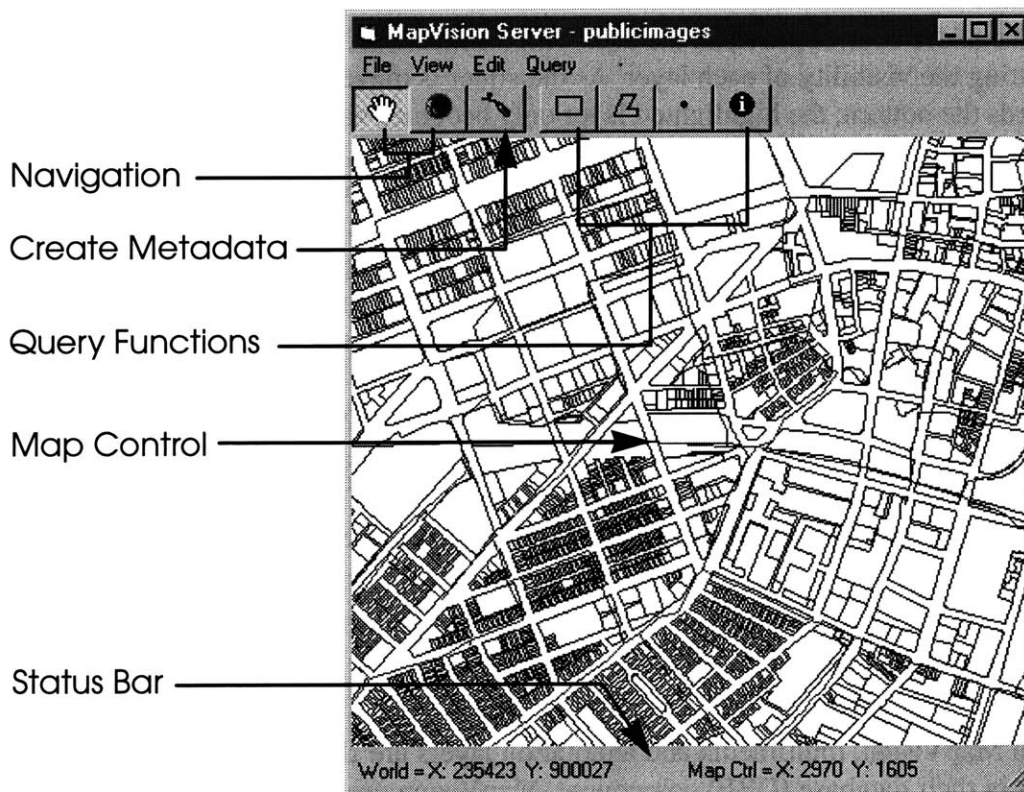


the capability to query multimedia objects by data found in other map layers that was not implemented in the client due to a lack of time.

This section will not describe any of the functions that have already been explained in the client section, but will concentrate on highlighting some server specific features. It is also beyond the scope of this thesis to provide detailed explanations of specific programming techniques or algorithms developed to implement the functions found in the MapVision Server. Instead the reader is referred to the documentation for MapObjects (*Building Applications With MapObjects*), MapObjects IMS (*Getting Started With MapObjects Internet Map Server*), and VisualBasic documentation (*VisualBasic Programmer's Guide and VisualBasic Language Reference*) for a starting point.

### 5.2.1 The Map Window

Figure 5.7 shows the MapVision server's main window and interface. The server has the standard navigation controls as well as tools for performing simple queries. All of the toolbar functions are also accessible through the menu bar. Note that the image that is seen in the map control is the image that will be sent to the client.



**Figure 5.7:** MapVision Server Map Window

*Some basic functions can be found in the MapVision server interface.*

The map window's primary function is to establish the dimensions of the map control. This is very important because the dimensions of the map control mirror those of the viewport of the MapVision client and define the affine transformation between the real world coordinates and the pixel coordinates. Without proper coordination of dimensions between the client viewport and the server's map control accurate spatial queries cannot be made nor could accurate georeferenced shapes be drawn on the client.

### 5.2.2 Layer Control Panel

The MapVision server provides a layer control panel for managing the various GIS map, multimedia, and annotation layers that can be accessed through the MapVision system. This is shown in Figure 5.8. The layer control window is where the ordering and removal of various layers can be controlled with those layers appearing at the top of the list being the ones at the top of the stack. MapObjects provides support for ESRI's shapefiles, Arc/Info coverages, and SDE layers although only shapefiles are supported in the MapVision server. Layers can be inserted into the system through a menu item.

The slider control provides the user with a convenient way of setting the visibility of each layer. As the slider is moved towards the bottom, the highlighted layer only becomes visible at increasing magnification levels. At the top of the slider, a layer is always on; conversely, moving the slider to the bottom turns a layer off. This is useful for controlling what layers are visible to the user at different magnifications. By allowing for detailed layers to become visible only when magnification is high, the client's viewport can be less cluttered. This can also translate into savings in network transmission times since simpler rasters of the map control result in smaller file sizes.



**Figure 5.8:** Layer Control Panel

*Ordering and visibility of each layer can be assigned through this control panel*

Annotation and multimedia layers are invisible by default in order to eliminate the clutter associated with displaying large numbers of annotation areas and view geometries.

### 5.2.3 Multimedia and Annotation Layer Creation

In MapVision, multimedia and annotation layers are shapefiles with specially defined fields in their database (DBF) component. MapVision uses these fields to distinguish between multimedia and annotation layers and other GIS map layers. The MapVision server provides menu items that create empty multimedia and annotation shapefiles complete with the appropriate DBF fields. Shapes and data can then be added to these files

through either the server or the client. Since the server relies on the existence and ordering of the DBF fields to identify multimedia and annotation layers, the sequence and naming of fields in other shapefiles should avoid duplicating those in the multimedia and annotation shapefiles.

The field specifications for multimedia layers is presented in Table 3.1. The annotation layer field specifications are given in the table below.

**Table 5.1: Annotation Layer Database Fields**

Shape	SEQID	USERNAME	DATE	URL	PATH

Associated with the actual shapes are: the SEQID (a unique identification number), USERNAME (author of the note), DATE (day the note was created), URL (link to web resource associated with this note), and PATH (the path to the text file that contains the annotation; this may either be a local path or another URL).

#### 5.2.4 Storage of Notes From Client

MapVision requires very little file handling outside of working with the custom shapefiles associated with multimedia and annotation layers. This is because files are referenced by URLs. The only exception is the text from notes submitted through the MapVision client. This is handled in the following manner:

1. The client transfers all data related to a note to the server EXCEPT for any entered text the user entered in the comment box via the standard HTTP GET. This is because a GET request can only contain a limited amount of data and any substantial comment (e.g. greater than 500 characters in length) can cause the request to fail. In testing, such failed requests often caused the server to crash.
2. The MapVision server parses the data and makes the appropriate entries into the annotation layer's DBF file. In doing so, the server generates a unique file name for the user's comments.
3. The file name is transferred back to the MapVision client. If no comment needs to be stored the sequence ends here.

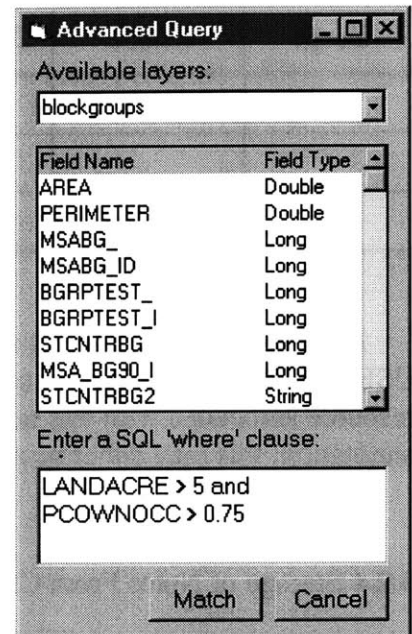
4. If a comment needs to be transferred to the server then an HTTP POST request is made directly to the *web server* where an awaiting CGI script writes the file using the file name generated in step #2.

This four step process requires more communication between the client, the MapVision server, and the web server, but enables the system to preserve its structure of keeping actual files (e.g. the comment text) separate from the references to the files (e.g. the annotation layer).

### 5.2.5 Query By Other Data Sources

Although unimplemented in the MapVision client, the MapVision server has the capability to allow users to search for multimedia objects based on attributes in other databases or map layers. This was successfully engineered into the server, but was not integrated into the client due to time constraints.

The MapObjects component software provide all the necessary commands and data structures for managing such queries. Applied in the context of the multimedia planning support system, such a capability helps fulfill the objective of providing tighter integration with database sources. Figure 5.9 shows the form constructed to accommodate queries that utilize information from other layers to select multimedia objects. The usefulness of this type of query is further elaborated upon in Section 4.3.7.



**Figure 5.9:** Multimedia Queries Utilizing Other Data Sources

*Form that allows for SQL querying of multimedia layers.*

## 5.3 Observations About The System

This section provides a number of observations about the MapVision prototype as it was being developed and experimented with. An evaluation of the prototype's ability to meet the established objectives of the thesis can be found in Section 7.1.

1. It was difficult to accurately define the view geometry associated with some images even though the user was familiar with the area. This was more problematic with images that contained long distance shots. In some situations the MapVision client could not display a large enough area in its viewport to accurately define the view geometry of the image.

2. A highly accurate map transformation between the server and client proved difficult to obtain. This made the reliable client side georeferencing of images questionable. All images for the test database were entered on the server side to counter this effect.
3. The use of HTTP GETs as a communication protocol proved to be effective enough for the prototype, but may prove to be a serious limitation if the system is to expand in its capabilities. A client/server model that maintained state may prove to be more effective.
4. Some serious thought needs to be devoted to the issue of how to update a database of URL references in the event that a multimedia object storage server changes domain names. Perhaps system diagnostic tools could be developed to automatically probe for dead links or update URLs that have been redirected to other servers.
5. The collaborative aspect of the MapVision system remains limited. How open should the system be to the public for adding images? It is almost certain that metadata creation will not be consistently handled in an uncontrolled environment. This would degrade the usefulness of the entire system.
6. MapVision proved that an effective Internet distributed client could be developed and deployed as an MMPSS interface, but also proved that substantial programming experience is still needed to do this using 1998 software. The emergence of JavaBeans and other reusable Internet software components should allow for easier development of such clients in the future.
7. It is uncertain how the system would perform in a large scale implementation since no stress tests were performed. This, however, was deemed less critical in a system developed primarily to prove concepts.
8. It is difficult to record the exact location of where a picture was taken without the ability to reference a detailed site plan or an ortho photo in the field. This may be partially mitigated with the use of global positioning systems technology.
9. Work still needs to be done in providing an effective and attractive display for the result sets sent to the client. There should be user definable options that would allow for how multimedia objects are shown. (e.g. ordering of multimedia objects, providing links vs. images, formatting for easy visual comprehension, highlighting of view geometries in the viewport to show what multimedia object relates to what location, etc.)



# Chapter 6

## MapVision System Architecture

MapVision is an Internet based client/server application constructed using as much industry standard software as possible. Even though specific software packages have been used in designing the MapVision system the underlying concepts are easily transferable to other related designs and applications. This chapter describes the general system architecture employed to design the prototype including notes about technology choices.

### 6.1 Server Design

The heart of the MapVision MMPSS is a custom server application built using Microsoft's Visual Basic and ESRI's MapObjects executing on a PC with a Windows operating system. MapObjects is a component GIS package consisting of OLE controls (OCX) that provide many standard GIS functions including: basic display and navigation functionality, SQL support, I/O for standard GIS data formats, thematic mapping, address matching, labeling and symbolizing, and statistics to name just a few. These GIS components are tied together using the Visual Basic RAD (rapid application development) environment to provide all the functionality needed to demonstrate the multimedia integration and retrieval techniques discussed in the thesis. The advantages of using a general programming language to develop an MMPSS are stated in an explanation by Moreno-Sanchez et al, 1997):

“a) It can be used to develop other necessary pieces of software to perform functions such as interface development, systems integration, database management and client/server applications, b) it would not force the professional end users to learn a proprietary multimedia authoring tool to be able to add new information and content to the system, c) new capabilities are constantly and rapidly added through language development, and especially, through custom controls (for examples of the wide range of functionality that can be added to VB through these controls visit [www.vbxtras.com](http://www.vbxtras.com) on the World Wide Web). These factors improve the return on the investment of time and resources to learn a development tool.”

Through the MapObjects component GIS design, the MapVision MMPSS has the ability to integrate GIS and database functionalities with multimedia. Additional GIS functionalities can be incorporated into the server application as needed. Furthermore,

since the sever is developed using reusable software components, the MMPSS has the potential to grow and adopt beyond the functionalities provided by any single software vendor.

A GIS based server was chosen for several reasons:

1. The GIS is an integral component of future planning support systems. Recall that one of the goals of this thesis is to not build a stand-alone MMPSS, but a system that is integrated with other planning support systems.
2. GIS provides many algorithms that can be applied to the spatial metadata system described in the previous chapter.
3. MapObjects provides a very convenient way to construct dynamic displays that are free of the imagemap limitations. The base display can be easily changed from project to project, or from organization to organization, simply by changing the map content.
4. Links to relational databases can be established via either the MapObject components or through Visual Basic to provide access to more data.

## **6.2 Client Design**

To realize the goal of an open distributed architecture, the MapVision client is an active content environment based on the WWW infrastructure. This allows the MapVision server to be easily accessible between organizations and even the public. Active content is general term that refers to the ability of a web based application to provide dynamic interactions at the client level such as performing calculations or providing animated visual feedback. Java, ActiveX, and JavaScript are among the most popular ways of providing active content for web pages at the time of this writing.

The MapVision client is a Java applet (based on the JDK1.1.x specifications) capable of being deployed on all modern web browsers on the major operating systems (Macintosh, Windows, and many flavors of Unix). The choice of using Java over standard HTML is absolutely necessary in order to deliver the appropriate level of visual feedback while the user manipulates the elements on the interface. For example, the Java client allows for real-time “rubberbanding” operations to occur during zooms, polygon construction and area selection.

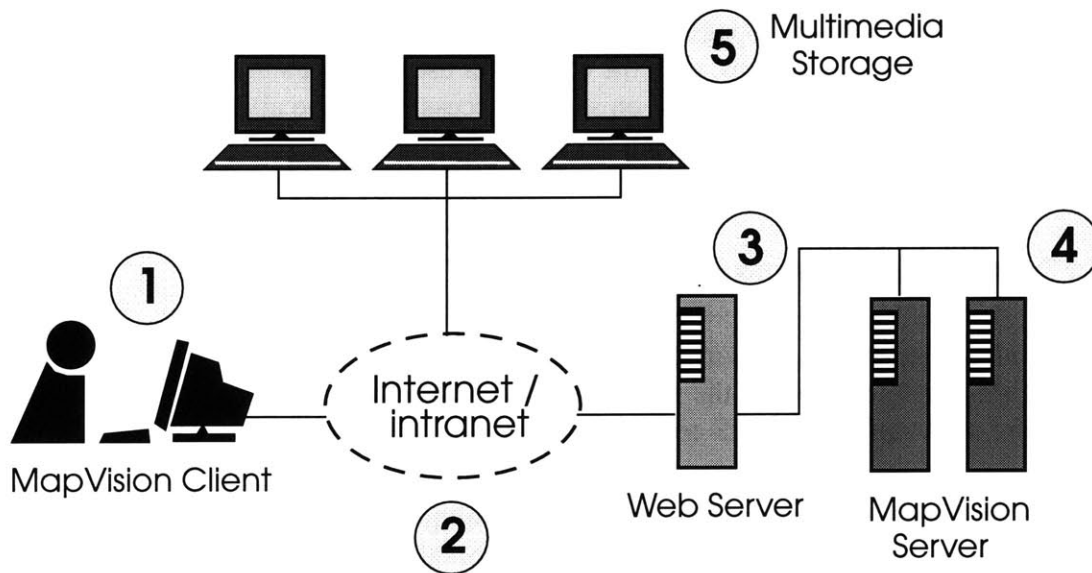
A client based solely on HTML, while capable of catering to a broader audience, would also be of significantly reduced functionality. Java, being a full programming language, provides the necessary functions and also allows for a broad number of paths for



upgrading functionality. Chapter 5 shows pictures of the prototype containing many features and conveniences expected in today's professional applications. These features can only be provided effectively in an active content client environment.

### 6.3 Overview of Client / Server Functionality

The MapVision client/server architecture is based upon the HTTP (HyperText Transmission Protocol) communication standard used to serve pages on the WWW. Much of the work of constructing a custom Internet accessible application is handled by the MapObjects IMS (Internet Map Server) OLE control. The resulting system architecture for MapVision is therefore one that complies to the standard IMS model as developed by ESRI Inc. A diagram of the general system architecture can be seen in the figure below.



**Figure 6.1:** MapVision System Architecture

*The MapVision system architecture is based upon the MapObjects IMS design (ESRI Inc., 1996) with the one addition being the call to multimedia servers elsewhere on the network.*

1. The user interacts with MapVision through a user interface designed as a Java applet embedded into a standard web browser (client). From the client, the user can perform tasks such as studying maps, defining queries for images, or adding annotation. These actions cause the client to send an instruction request to the MapVision server.

2. The request travels through the Internet (or within an intranet) using HTTP to reach the web server that initiated the client's session (i.e. provided the client's web browser with the page containing the Java applet).
3. The request is identified by the web server as instructions from the MapVision client requesting a service of the MapVision server. The web server then forwards the request to the MapVision server.
4. Once the request reaches the MapVision server, it is parsed for a specific service demanded of the server. Provided that it is a valid service, the server performs the necessary tasks before packaging a reply to the client. A reply can consist of messages, screen updates, or multimedia data depending on what the client requested. The reply is sent back up the chain to the web server which in turn sends the reply to the client.
5. If the reply contains a page with URL references to multimedia objects (e.g. images, other web pages, etc.) the client will use the URLs to retrieve the objects. These objects must be located on servers accessible to the client (as shown in the diagram).

### 6.3.1 System Design Strengths and Weaknesses

The advantage to using this model is that a tested networking infrastructure can be used to form the foundation of the system. This was also the only model available for integrating the MapObjects' GIS components with the Internet at the time of development.

The disadvantage is that the infrastructure is built primarily to accommodate the serving of maps over the Internet and not as well suited to tasks requiring more advanced interactions such as in the MapVision project. For example, the MapObjects IMS use of HTTP GETs as the only means of transferring information between the client and server can be a constraint to transporting larger amounts of information (e.g. several paragraphs of annotation text) from the client to server (1024 character limit).

## 6.4 Detailed Description of Client/Server Interactions

This section provides a closer look at the interactions that occur between the client and the server. The basic concept to understand when looking at the transaction tables to follow is that the MapVision clients communicate with the web and MapVision server using a *stateless* connection. Very simply, this means that the server knows nothing more about a particular client than what it is specified within each request issued by the client. Furthermore, once a request is fulfilled by the server the connection with the client is

closed. This is the model that HTTP is built upon and is what the MapObjects IMS provides. By contrast a connection that maintains state allows a client and server to continue communicating with one another until a disconnect is requested by either the client or the server.

The stateless model of client/server communication is easier to manage and can handle requests from many clients, but is less suitable for complex transactions such as those that require continuous coordinated communication between client and server (e.g. chat servers) or the frequent transmission of complex data types (e.g. binary, objects). In order to better understand the order of communication between the client and server, and to understand the processes that are invoked by the client and fulfilled by the server, a series of transaction tables have been provided below. These are ordered for reading from the top to the bottom (following the Step # column) and from left to right. Table 6.1 shows what happens when the user first tries to access the MapVision system:

**Table 6.1: Client/Server Transactions for Initializing the MapVision System**

Step #	Web Browser / MapVision Client	Web Server	MapVision Server
1	User types in the URL of the web site hosting the MapVision system into the web browser.	Sends the HTML page back to the client containing the MapVision Java client.	
2	MapVision Java client establishes the initial map extents and finds the URL of the MapVision Server through applet <PARAM> tags. Transmits a URL containing this data to the web server.	esrimap.dll recognizes the URL as a request to the MapVision Server and passes the string along to the MapVision Server.	Reads the extent coordinates from the URL and rasterizes the appropriate area of the map as a Jpeg file. Jpeg image is then sent back to the client.
3	MapVision client is now initialized and ready for other interactions.		

A key concept worth noting from above is that the MapObjects IMS model relies on the transmission of a rasterized image to present a map display. This is a common method of transporting GIS data through the WWW. It sacrifices the user's ability to manipulate individual map layers for the simplicity of transporting graphics with a single raster.

Table 6.2 outlines the transactions that take place when the user of the client performs simple navigation actions such as panning, changing magnification, and returning to the full extents of the map. After the initial transmission of the MapVision client to the user's web browser has occurred (in the initialization stage) the web server is simply passing requests to the MapVision server.

**Table 6.2: Client/Server Transactions for Navigation Functions**

Step #	MapVision Client	Web Server	MapVision Server
1	User selects a pan, zoom, or full extent action. For pans and zooms, the user interactively defines the new extents. Full extent calls set the display to the initial extents.		
2	A URL is constructed containing the new map extents and transmitted to the web server.	esrimap.dll recognizes the URL as a request to the MapVision Server and passes the string along to the MapVision Server.	Reads the extent coordinates from the URL and rasterizes the appropriate area of the map as a Jpeg file. Jpeg image is then sent back to the client.
3	Display is updated with the image from the MapVision Server and is now ready for another user action.		

Table 6.3 (next page) shows the general sequence that needs to be followed in order to fulfill a client's query. The steps are familiar, but in the case of queries the MapVision server is required to do more. Instead of returning an image to the client, the MapVision server queries the layers of spatial metadata to determine which (if any) multimedia objects should be returned to the client. The server will search through all layers that are currently loaded into the server application and identified as multimedia layers (e.g. ones containing spatial metadata). Administrators on the server side can therefore easily control what is available to the clients at any time.

Once searching has been completed, the MapVision server is also responsible for generating a response page. This is a pure HTML page generated with the appropriate headers and formatting to include all the links (found in the URL field of the multimedia

layers) to images that met with the user's search criteria. This response is shown in a separate window on the client side. The two window approach is important since it allows the results to be shown without disturbing the window that contains the MapVision client (see Section 5.1.8 for a specific example).

**Table 6.3: Client/Server Transactions for Image Queries**

Step #	MapVision Client	Web Server(s)	MapVision Server
1	User selects a query and enters keywords and/or defines search points and areas using the client's functions.		
2	The search type, the coordinates of the defined regions (or point), and/or keyword(s) are encoded into a URL and transmitted to the web server.	The MapVision web server's esrimap.dll recognizes the URL as a request to the MapVision Server and passes the string along to the MapVision Server.	The query string is parsed to determine the search strategy that should be used (e.g. region search, location-to-location search, etc.)
3			The appropriate search is carried out with the user's defined point/region and/or keyword(s). Every map layer loaded in the server designated as a multi-media layer is searched.
4			The search results are parsed into an HTML page that contains the objects (or their links). The page is sent back to the client.
5	The page generated in step 4 is displayed in a separate browser window. Links to images are retrieved by requests to the web servers on which they reside.	Web server(s) try to locate the images in the HTML results page. These are not necessarily the same web server(s) that deliver the MapVision system.	

**Table 6.4: Client/Server Transactions for Adding a Note**

Step #	MapVision Client	Web Server	MapVision Server
1	The user draws a shape indicating the note's physical location on the map. A user name plus a text message entered in the client and/or a URL is associated with this shape.		
2	Once the Save button is pushed A URL is constructed containing the points that make up the shape, the user name, and or the URL. Any text message is held in the client's memory.	esrimap.dll recognizes the URL as a request to the MapVision Server and passes the string along to the MapVision Server.	The URL is parsed for the shape, user name, and/or the URL. The shape is re-assembled from its points.
3			The shape, user name, date, and/or URL is written to a publicly accessible multimedia layer. If necessary, a new file name is reserved by the server for the possible writing of the text message still to come.
4			The filename is sent to the client for its use in saving the text file if needed.
5	If a message is in the client's memory it is sent to the web server via an HTTP POST.	A CGI script receives the note and writes it to a file on the web server.	

Table 6.4 explains how the user can make a geo-referenced annotation to the MapVision system. Note that the above sequence is only triggered when the save button on the client is pushed. This allows the user to add and delete notes as needed and saves them only when he is sure no more annotations will be made during the session. A CGI script is needed in step 5 to handle the HTTP POST request since the MapObjects IMS

communication structure only supports the more limited GET channel that is quickly consumed by items such as the shape point data, user name, and URL. The POST method is necessary in order to allow for the submission of any note of non-trivial size. This also acts to transfer the load of file writing to the web server instead passing down the request one more level for the MapVision server to handle.

**Table 6.5: Client/Server Transactions for Adding an Image**

Step #	MapVision Client	Web Server	MapVision Server
1	The user draws a shape indicating the view geometry of the image and provides a URL and keywords.		
2	A URL is constructed containing the view geometry's shape, the URL, and the keywords. This is sent to the web server when the Save button is pushed.	esrimap.dll recognizes the URL as a request to the MapVision Server and passes the string along to the MapVision Server.	The URL is parsed for the shape, the URL, and the keywords. The shape is re-assembled from its points.
3			The origin point, and area are computed based on the submitted shape. All of the data is saved to a publicly accessible multimedia layer.
4			The client is signaled of the completion of the save.

The process for adding an image to the MapVision system, shown in Table 6.5, is less complicated than adding a note because less information needs to be transmitted to the MapVision Server from the client. Again, no data is transmitted to the server until the user pushes the save button on the client. This allows users to submit multiple images at once as well as giving the user to refine the view geometry before it is saved.





## **Chapter 7**

# **Advancing Multimedia Planning Support Systems**

The MapVision prototype was able to demonstrate the feasibility of many of the integration and retrieval strategies documented in this thesis. This, however, is only one advantage of constructing a working prototype. Another advantage (or disadvantage as the case may be) is that it forces the designer to consider many aspects of the product's performance along with the user's requirements. What works in theory may sometimes prove unmanageable or too difficult in practice. Working with a real product also provides insight into bettering the design in ways that would otherwise not be possible.

This chapter reflects on the issues that have arisen in the course of designing and constructing the MapVision system. This chapter begins by discussing some of the areas that still need more thought and effort, offers some examples of possible applications, and describes the author's vision for the advancement of multimedia planning support systems.

### **7.1 Evaluation of Prototype**

Overall, the prototype MMPSS proved to be valuable for experimenting with many different ideas of how an MMPSS could be enhanced. MapVision followed many of the system design principles that are likely to be required of information systems in the future (see Table 2.1) by:

- Being a networked application
- Having a platform independent client
- Using industry standard software and data formats
- Allowing for distributed data access

By using the concept of spatial metadata, MapVision also succeeded in developing several improved techniques for retrieving multimedia objects. These retrieval techniques addressed the problems of navigating and working with large sets of multimedia data (and images in particular).

MapVision was able to take advantage of existing component GIS software to build a modular MMPSS. This allowed for GIS and database functions to be accessible without having to spend time rebuilding either. Using these software building blocks, the MMPSS can be expanded as new functions are needed. While this still requires an experienced computer programmer to do the work, the use of component software considerably shortens the development time. Even with these success, there are still aspects of the

system that could benefit from further refinement and research. These are discussed in the following sections.

## **7.2 Distributed Data Submission and Collaboration**

One of the original objectives of the MapVision system was to allow users to contribute multimedia objects to the database of the MMPSS. This is represented in the metadata creation and saving features available in the MapVision Java client. The client allowed users to create spatial metadata, keyword references, and specify the location of the object and save these to the MapVision server. Once saved, anyone accessing MapVision would be able to use the submitted data. This feature might be useful in two ways. First, as a method for cooperatively building a database, such features help promote the continuous change and growth of the MMPSS database so that content has a chance to remain fresh and relevant. Second, the availability of methods for distributed interaction might also be beneficial to promoting public dialogue about planning processes and projects. The public could theoretically retrieve and add information to the MMPSS. This is in direct contrast to the model of the institutionalized information system that is solely the tool of the expert. Unfortunately, there are significant problems associated with implementing such distributed access.

Distributed data submission for the purpose of constructing a knowledge base or for public collaboration are ideas worthy of exploration. In the context of the MapVision project, this could allow many organizations to submit images referenced by keywords and spatial metadata. The decentralized input of information into the knowledge base means that no single organization (or person) would need to be responsible for providing the content for the entire database. Gathering multimedia data can be a potentially time consuming and expensive process so the decentralization of this process among many interested parties would be a benefit to the system as a whole.

However, the decentralized model also raises some serious questions. For example, who should have access to the system? Should everybody be allowed to submit data? Should everybody be allowed to retrieve data? What parts of the data? Given that the proper indexing of multimedia data is important for its later retrieval, what assurances are there that everyone with permission to submit data is doing it properly? How will these people be instructed? What are the policies for organizing the data, for deleting data, for archiving data? What policies are needed for governing the types of content that can be submitted?

The above questions need to be addressed in the implementation of any information system, but they are particularly important in an Internet-based system. Many of the questions also revolve around network security and file management issues which were deemed to be beyond the scope of this thesis to deal with in actual implementation.

The integration and retrieval strategies as presented in this thesis are best suited for a model that promotes public access to retrieval of the information but private construction of the database. Using the MapVision system as an example, the following table lists features that should be available to public users (anyone with access to the client) and private users (the maintainers of the database). Naturally, the private users will have access to all of the functions available to the private users.

**Table 7.1: Recommended Access Privileges for Public and Private Users**

Public Users	Private Users
Browse maps and data layers	Define metadata standards
Make queries for multimedia data	Increase functionality of MMPSS
Limited data submission capabilities (e.g. annotation)	Full privileges to add, delete, and edit records to database

Under this proposed system, the information in the MMPSS reaches the broadest audience possible by providing a WWW accessible client to the general public. The ability of the client to add to the MMPSS knowledge base is limited. Under this access scheme, only simple data types can be submitted by the public user to the MMPSS (e.g. text for annotation purposes, or a simple URL). While this limits the ability for different groups to participate in the construction of the MMPSS knowledge base, it does reduce the number of potential problems by minimizing system complexity and limiting access. Furthermore, it is unknown at this point whether unstructured participation is something that benefits an MMPSS or if it is a feature that would be demanded by users.

More complex data types (e.g. spatial metadata) can only be created by a private user with access to the MapVision server / desktop application. Private users also have the responsibility for maintenance of the MMPSS knowledge base and adding functionality to the MMPSS as necessary. This would also include any extensions to the metadata format used to integrate the multimedia objects.

Additional work in MMPSS that addresses distributed data submission and collaboration will be needed if a more decentralized model is desired. The study of real time collaborative systems such as those found in commercial groupware products may provide further insight into the problems. More importantly, the designers of an MMPSS will need to consider the advantages and disadvantages of: i) providing an unmanaged collection of objects open totally to the public; and, ii) providing a system oriented for limited access or in-house use.

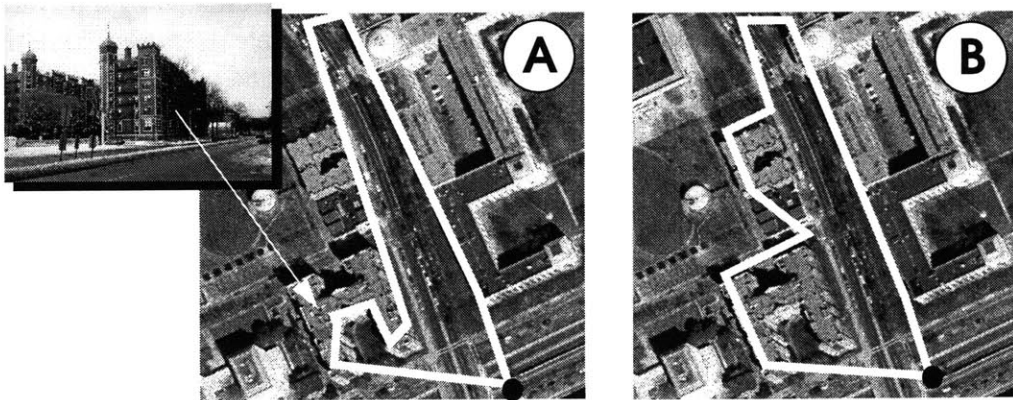
## 7.3 Methods for Generating View Geometry

Chapter 4 described many strategies for retrieving digital images (and possibly video) by using spatial queries. These queries rely on the availability of spatial metadata descriptors (specifically the view geometry) to provide computable attributes. Therefore, the accurate creation of view geometry is critical to an MMPSS dependent on spatial metadata for multimedia object retrieval.

### 7.3.1 Manually Sketching View Geometry

MapVision made use of the simplest possible method to generate view geometries by allowing a user to sketch an image's field of view based on reference to the image itself. This can be a highly accurate way of defining a field of view since the user can intelligently determine what is and is not visible. This method is also simple to understand and not computationally intensive. There are, however, some disadvantages:

1. Sketching the view geometry requires that the person be familiar with what is visible since the sketching process is often based on familiar landmarks. This can be a problem when the person taking the picture and the person entering the image into the database are not the same and the latter is unfamiliar with the surroundings. It is also hard to judge where to terminate the view geometry for any image that can see far into the horizon.
2. Subjectivity is introduced into the process when different people are sketching the view geometries. Figure 7.1 shows variations on how view geometry can be recorded for an image.



**Figure 7.1:** Variations in the Recording of View Geometry

*The building shown in the upper left is the building indicated by the white arrow in the aerial photograph. The black dots in "A" and "B" indicate the location where the photo was taken. In "A" the view geometry (white line) outlines the various building surfaces that are shown. In "B" the view geometry encompasses the buildings that are shown in their entirety.*

Ultimately, the “right” methodology is probably less important than maintaining a consistent approach throughout the entire database. Consistency is of utmost importance when indexing a series of images with subtly different fields of view (e.g. a 180 degree panoramic taken with several images).

3. Manually sketching the view geometries can be time consuming. This would be especially true for large volumes of images that need to be integrated quickly.

Methods for automatically generating the view geometry for images would not only save time, but would increase the consistency of the index enabling the system to handle queries with greater fidelity.

### 7.3.2 Calculating View Geometries Through Visibility Analysis

One possibility is to take advantage of visibility analysis algorithms found in high end GIS software. Arc/Info from ESRI, Inc. has the ability to perform a visibility analyses using lattices. A lattice in Arc/Info is a type of three dimensional digital terrain model based on a “mesh” of points. Using the visibility command it is possible to determine what regions on the terrain model are visible from any other point on the model. The visible regions are returned as a coverage of polygons. Without delving further into the specifics of visibility analysis it is safe to assume that such polygon coverages could ultimately be used for determining the view geometry of an image.

Visibility analysis in Arc/Info is not without its own drawbacks. The process is computationally intense and requires the availability of a lattice for the region being observed (constructing a lattice requires elevation data as well). The accuracy of the results depend on the granularity of the lattice and on the person being able to place themselves at the proper location and elevation on the earth. It is also difficult (but not impossible) to take into account obstructions such as vegetation and buildings. Experience with Arc/Info’s visibility analysis has shown that it is a better tool for determining the visibility of landscapes with larger viewsheds. Larger viewsheds are also more tolerant of errors in determining visibility. In the urban environment it is very easy to tell what is and is not visible in a picture due to the presence of distinct structures.

### 7.3.3 Calculating View Geometries Using Three Dimensional Space

One possible approach to determining view geometries automatically is to construct a representation of the real world environment in three dimensional space including terrain elevation and building heights. The coordinates of the model are calibrated to a real world coordinate system matching that of other data being used in the MMPSS. A ‘virtual camera’ is then inserted into the 3D scene at the same location and bearing that the real photograph was taken at. The system then records the real world coordinates of all the

objects that are visible to the virtual camera. These coordinates are then systematically connected to form a polygon (or polygons) that represents the view geometry for the image.

The construction of the model relies on some additional data that is relatively easy to obtain: a coverage showing building footprints with attributes that indicates its height. An digital terrain model might also be useful for urban environments with many significant grade changes. A basic 3D model can be constructed by simply extruding the building footprint polygons to the height specified in the associated height attribute. Terrain elevation can optionally be used to position individual buildings on the z-axis of the 3D environment.

It is important to note that the 3D environment is used only to situate the virtual camera so that it can capture what is visible. The user never sees the 3D model since its use is strictly for the generation of the appropriate view geometry. Theoretically, this means that the complex computations required to render the 3D model are not needed and the view geometry should be calculable in very little time even on a standard office PC.

Skeptics may be quick to point out that, "If you are willing to construct a 3D environment to determine view geometry, then why not just render the 3D model?" While it may be useful to also render the 3D model (e.g. for a new navigation or query interface) it will be crude and much less representative of the real world. Some may also argue that such an approach would merely be a poor parallel to the objective of photorealistic 3D rendering of a built environment. Indeed, there has been work on distributed virtual GIS (Preston, Clayton, Wells, 1997) that follows this precise line of thinking. However, the quest for a full 3D city environment faces an even more challenging problem of constructing and then rendering hundreds or thousands of detailed models. Modeling the real world in this way can also never capture real world phenomenon as it changes over time (e.g. pedestrian traffic studies).

In the end, the debate is not whether photographic representations are better than computer generated representations, but rather what techniques can enhance the effectiveness of MMPSS as a whole. These last few sections have identified some additional areas of research along with possible directions for exploration.

## **7.4 Applications**

Although this thesis has promoted a very different approach to building and using MMPSS, all of the traditional uses of multimedia applications in planning still apply to the MapVision system. These are applications in public information provision, education, simulation, and collaborative systems. This section covers some application areas that have been emphasized throughout the development of the prototype.

### 7.4.1 Visualization

The development of MapVision was guided in large part by interest in finding techniques to assist in visualizing the urban environment. Therefore, it should come as no surprise that visualization may be a leading application of the research described in this paper. Over the years there have been many different methods to computer based visualization ranging from manual alterations of digital images, to 3-D modeling and animation, to fully immersive 'virtual reality' experiences. Given these many different techniques, why is a system such as MapVision necessary?

Computer based environmental visualization (and particularly 3-D visualization) has at times been criticized for its ability to deceive professionals and/or the public about the visual impact of a project. This can occur through missing features, dimensional errors, and inconsistent details (Alley, 1993). Apart from the biases of the parties involved, the use of 3D models, animations, or digital composites is often an artistic endeavor. The skill of the model builder, artist, or cinematographer has a great impact on the simulation. An accessible image database of the urban environment as it truly exists can be very helpful for properly interpreting computer based visualizations. Since images are referenced by a spatial metadata standard the objectivity of the data presentation is increased.

However, the true role of an MMPSS is to serve as a framework for storing multimedia data regardless of how it was generated. Urban visualization through 3-D modeling and other digital techniques are typically employed to predict how a place will look after some change occurs to the environment. The data generated from such exercises are still about places on earth and therefore can be integrated within the MMPSS just as easily as digital photos. The provision of a structured technique for storing and retrieving multimedia about places is perhaps the strongest argument for deploying such a system. Without such a system, valuable multimedia data may be used once and become lost or neglected. Text reports are usually properly filed and referenced for future projects in any planning organization. Why not multimedia objects?

### 7.4.2 Collaborative Repository of Multimedia Data

The flexible and open system architecture of the MapVision design allows for many departments or organizations to collaborate on a multimedia repository. Instead of the Transportation Department's video of traffic flows being kept separate from the Public Works Department's inventory of road signs it is now possible to place both sets of multimedia data together. Although much work still needs to be done in designing a system that can truly support large scale access to all forms of multimedia data, the future of both planning and MMPSS lies in enhancing collaboration.

In the past, multimedia applications frequently faced the problem of how to update its information base. Since it is no longer necessary to construct multimedia applications

using proprietary multimedia authoring systems it is now easier to integrate multimedia data into one broadly used data format (e.g. shape files, dxf files, etc.). Maintenance of the MMPSS is now much closer to updating a database than redesigning a program. Although much work still needs to be done in designing a system that can truly support large scale access to all forms of multimedia data, the future of both planning and MMPSS lies in enhancing collaboration.

#### **7.4.3 Added Functionality to Existing Planning Support Systems**

At the very minimum, the methodology developed in this thesis provides some degree of multimedia data access to existing planning support systems. As stated early on, this allows planners to leverage the rich qualitative descriptions of visual and/or audio information towards planning issues.

By using powerful back end systems such as relational databases and GIS, a methodology has been provided for the scalable integration of multimedia data into planning support systems. This basic framework should prove many times more capable in terms of retrieving and delivering information than a traditional multimedia application.

### **7.5 Multimedia Information Planning Systems**

One of the primary arguments of this thesis is that multimedia applications in planning should move towards being a set of modular tools that can be integrated with other planning information systems. In this way, planners can better utilize multimedia for broader purposes than current stand-alone systems can provide. The MapVision prototype demonstrated one possible method of integrating multimedia with other planning support systems, but what happens next?

There will always be aimed for better indexing schemes, or retrieval mechanisms, or interface improvements. One day the sheer computation power of desktop computers alone may automatically solve problems associated with real time performance for collaboration. There will also be no end to the amount of multimedia peripherals and software that will be affordable to a planning organization. These innovations will continually force those involved in MMPSS to re-evaluate their approach to applying technology to planning. But what might be an underlying vision for those interested in applying new technologies to planning? Consider one such vision statement in the paragraphs below:

Whether a multimedia application for planning serves to provide a resource to assist in analysis, provide a tool for collaboration, or simply provides information to the public, the objective of the application is to tell a story. Stories about places and how they used to be,



are today, or may be tomorrow. Stories about people's lifestyles, their living arrangements, how they play, how they work, and where they go. In many ways, the act of planning can be analogous to editing an ongoing story authored by communities. Communities tell their stories to the planners who in turn must guide its development towards the best possible copy without infringing on the authors' vision. If we accept that this is one possible metaphor for planning then one possible role for multimedia information technology becomes clear. The technology should assist planners and communities alike in telling and writing their stories.

The story directive provides a powerful guiding vision for truly advanced multimedia planning support systems. We should move beyond merely having systems with the capability of providing the information we request of it. Is merely having copious amounts of data enough for understanding? The interpretation of information is even more important than having access to information. Multimedia planning support systems should move towards intelligently organizing and presenting the information embedded in multimedia objects.

The advanced MMPSS of the future will not merely retrieve a multimedia objects and present them to the user "as is", but will try to organize the objects around related themes to tell a story about a place. The planner who needs to review the background of a neighborhood will be able to receive an organized presentation of facts rich in images of the past to present, videos of events, recordings of meetings, and references to relevant documents. The advanced MMPSS will have intelligently sequenced the information for maximum impact and relevance and not just thrown at a user at the click of the mouse.

The MMPSS will allow professionals and general public alike to participate in an interactive process that will inform all through the revelation of individual stories and aspirations. It will have the capability to find and inform those who share similar perspectives or goals. The MMPSS will take on the simultaneous roles of a collaborative plan authoring system and information distribution system. There will then be better records of the act of planning instead of just records of produced plans.

Connected to the immense resources of the Internet, the advanced MMPSS will hunt for similar records by similar systems in other communities; possibly through the use of software agents that watch as other communities plan. These interconnected systems will call the planners' and public's attention to the case studies of other communities in the country or even the world. The advanced MMPSS will transcend being an information storage and retrieval system and move towards being an information planning system.

## **7.6 Conclusion**

This thesis has made a concerted effort to scrutinize the use and development of multimedia planning support systems. While many demonstration multimedia

applications have shown the potential of multimedia data for helping to describe and understand planning issues, the wide spread adoption of multimedia planning support systems has yet to occur. This thesis has concentrated on some of the technical issues that may be limiting the growth of MMPSS while touching on some organizational issues. The topic of integrating and retrieving digital images about the urban environment was pursued throughout this work in order to provide a focus to the research. The leading technical challenges were identified as:

- i) providing a scalable solution for integrating multimedia data with traditional databases and GIS tools;
- ii) investigating and developing new image retrieval strategies for planning and planning related purposes;
- iii) designing effective user interfaces for dealing with larger amounts of multimedia data than has typically been integrated into MMPSS thus far;
- iv) distributing the system to as broad an audience as possible.

Experimentation with the MapVision prototype proved that the integration and retrieval concepts documented in this thesis were implementable and effective. Each of the above challenges were addressed to some degree and possible solutions to the leading unresolved issues surrounding this implementation of an MMPSS were explored.

This work has sought to provide an alternative MMPSS design strategy. It is an approach that promotes component software technologies, distributed access, and intelligent multimedia retrieval strategies. It has tried to inspire thinking about how to better organize and deliver multimedia content to support planning through the use of a new metadata scheme.

Organizing and retrieving data is only the beginning. It is also worth considering how multimedia planning support systems might better sequence and present the information in its knowledge base to relate stories about places. How might multimedia planning support systems act to help record and relate the stories of communities involved in a planning process? How can networked multimedia planning support systems communicate between each other so that planners and public alike can have access to rich multimedia data from many different cities? The answer to these and other questions may be found through further research and development of multimedia planning support systems that creatively apply new technologies. In this way, better tools can be provided to help planners and citizens alike to learn and communicate about their communities.

## References

- Alexander, John P., and Valerie J. Warwick. "Writing GIS Applications for the WWW." In Proceedings of 1997 ESRI User Conference, 1997. <<http://www.esri.com/base/common/userconf/proc97/PROC97/TO550/PAP543/P543.HTM>> (1 May 1998).
- Alley, John. "Using 3-D CAD Perspective Images as Evidence in Planning Hearings." Plan Canada July 1993: 27-30.
- Batty, Michael, and Paul J. Densham. "Decision support, GIS, and urban planning." Systema Terra 1 (1996) : 72-76.
- Christiansson, P. "Building a city advisor in a 'hypermedia' environment." Environment and Planning B: Planning and Design 18 (1991) : 39-50.
- Davenport, Glorianna, and Michael Murtaugh. "ConText Towards the Evolving Documentary." In Multimedia '95, by Association for Computing Machinery, San Francisco, CA: Association for Computing Machinery, 1995. 381-389.
- Engen, David. "Using MapObjects with Java to Internet Enable GIS Applications." In Proceedings of 1997 ESRI User Conference, 1997. <<http://www.esri.com/base/common/userconf/proc97/PROC97/TO250/PAP207/P207.HTM>> (1 May 1998).
- Environmental Systems Research Institute, Inc. Building Applications with MapObjects. Redlands, CA: Environmental Systems Research Institute, 1996.
- Environmental Systems Research Institute, Inc. Getting Started with MapObjects Internet Map Server. Redlands, CA: Environmental Systems Research Institute, 1996.
- Federal Geographic Data Committee. "Geospatial Data Clearinghouse Activity." <<http://www.fgdc.gov/Clearinghouse/Clearinghouse.html>> (17-May-1998).
- Federal Geographic Data Committee. "The National Spatial Data Infrastructure." <<http://www.fgdc.gov/NSDI/Nsdi.html>> (17-May-1998).
- Flickner, Myron et al. "Query by Image and Video Content: The QBIC System." Intelligent Multimedia Information Retrieval. Ed. Mark T. Maybury. Menlo Park, CA: American Association for Artificial Intelligence, 1997. 7-22.

- Gallagher, Jim, and Seth Asante. "Distributed Congestion Management System Information Using the World Wide Web." Paper presented at Sixth TRB Conference on the Application of Transportation Planning Methods, Boston, May 1997.
- George, R. Varkki. "HyperSpace: Communicating Ideas About The Quality of Urban Spaces." Journal of Planning Education and Research 17 (1997) : 63-70.
- Gouveia, Maria Cristina. Augmenting Public Participation with Information Technology in a Portugese Environmental Assessment Context. Cambridge, MA: Massachusetts Institute of Technology Department of Urban Studies and Planning, 1996.
- Gudivada, Venkat N., Vijay V. Raghavan, and Kanonluk Vanapipat. "A Unified Approach to Data Modeling and Retrieval for a Class of Image Database Applications." In Multimedia Database Systems: Issues and Research Directions. edited by V.S. Subrahmanian and Sushil Jajodia, 37-78. Berlin: Springer-Verlag, 1996.
- Hartman, Robert A., and Larry N. Sanford. "GIS Access and Marketing with Multimedia on the World Wide Web." In Proceedings of 1995 ESRI User Conference, 1995. <<http://www.esri.com/base/common/userconf/proc95/to150/p116.html>> (1 May 1998).
- Horacek, Helmut. "Empirical Evidence of the Need for Intelligent Methods in Multimedia Information Retrieval." Intelligent Multimedia Information Retrieval. Ed. Mark T. Maybury. Menlo Park, CA: American Association for Artificial Intelligence, 1997. 427-448.
- Hughes, John R. "Technology Trends Mark Multimedia Advancements." GIS World November 1996: 40-43
- Huxhold, William E. An Introduction to Urban Geographic Information Systems. Oxford: Oxford University Press, Inc., 1991.
- Kashyap, Vipul, Kshitij Shah, and Amit Seth. "Metadata for Building the Multimedia Patch Quilt." In Multimedia Database Systems: Issues and Research Directions. edited by V.S. Subrahmanian and Sushil Jajodia, 297-319. Berlin: Springer-Verlag, 1996.
- Khoshafian, Setrag, and A. Brad Baker. Multimedia and Imaging Databases. San Francisco: Morgan Kaufmann Publishers, Inc., 1996.
- Kindelberger, Charles P. "HyperMedia for State and Local Government." Journal of the Urban and Regional Information Systems Association 6, no. 3, (1990): 52-55.

- Klosterman, Richard E. "Planning Support Systems: A New Perspective on Computer-Aided Planning." Journal of Planning Education and Research 17 (1997) : 45-54.
- Lang, Laura. "GIS Comes to Life." Computer Graphics World October 1992 : 27-29, 35-36.
- Laurel, Brenda, Tim Oren, and Abbe Don. "Issues in Multimedia Interface Design: Media Integration and Interface Agents." In Human Factors in Computing Systems: CHI'90 Conference Proceedings, by ACM Special Interest Group: Computer-Human Interaction, New York: Association for Computing Machinery, 1990.
- Limp, W. Frederick. "Weave Maps Across the Web." GIS World September 1997: 46-55.
- Lu, Guojun. Communication and Computing for Distributed Multimedia Systems. Norwood, MA: Artech House, Inc., 1996.
- McComb, Michael S. "Sound and vision: an introduction to multimedia AM/FM/GIS." In Proceedings of AM/FM Conference XVII. Aurora: AM/FM International, 1994: 29-40.
- Moreno-Sanchez R, J Malczewski and L A Bojorquez-Tapia "Design and Development Strategy for Multi-media GIS to Support Environmental Negotiation, Administration, and Monitoring at the Regional Level." Transactions in GIS 1(3) (1997): 161-175.
- Prabhakaran, B. Multimedia Database Management Systems. Norwell, MA: Kluwer Academic Publishers, 1997.
- Preston, Michael, Peter Clayton, and George Wells. "Distributed Virtual GIS." In Proceedings of Teletraffic '97, 1997. <<http://cs.ru.ac.za/homes/g93p4978/preston.html>> (1 May 1998).
- Rossel, Gregory A. Technical Augmentation of Visual Environmental Review in the Planning Process. Cambridge, MA: Massachusetts Institute of Technology Department of Urban Studies and Planning, 1994.
- Shiffer, M. J. "Environmental review with hypermedia systems." Environment and Planning B: Planning and Design 22 (1995) : 359-372.

- Shiffer, M. J. "Interactive Multimedia Planning Support: Moving from Stand-alone Systems to the World Wide Web." Environment and Planning B: Planning and Design 22 (1995) : 649-664.
- Shiffer, M. J. "Managing Public Discourse: Towards the Augmentation of GIS with Multimedia." In Geographic Information Systems: Principles, Techniques, Management and Applications. edited by P. Longley, M. Goodchild, D. Maguire, D. Rhind, New York: Wiley, 1997.
- Subrahmanian, V.S. Principles of Multimedia Database Systems. San Francisco, CA: Morgan Kaufmann Publishers, Inc., 1998.
- Tsui, Carleton. Multimedia Applications in Planning, Waterloo, Ontario: University of Waterloo School of Urban and Regional Planning, 1993.
- University of California, Santa Barbara. "The Alexandria Digital Library." Rev. 30-Mar-1998. <<http://www.alexandria.ucsb.edu/>> (17-May-1998).
- Wiggins, Lyna L. and Michael J. Shiffer. "Planning with Hypermedia: Combining Text, Graphics, Sound, and Video." Journal of the American Planning Association. (Spring 1990): 226-235.
- Worboys, Michael F. GIS: A Computing Perspective. Bristol, PA: Taylor & Francis, 1995.