1 Conclusions and Recommendations

1.1 Introduction

The rapid advancement in spatial data capture technologies such as the Global Positioning System (GPS), satellite imaging and total stations, has made the capture of digital spatial data a relatively quick and easy process. This has meant that in the last decade or so the amount of digital spatial data in existence has increased significantly. It has also meant that in recent times there has been a greater focus on how to use the spatial data that are collected and stored in the expansive "electronic silos" (Gore 1998) to their full potential. Virtual databases created by the use of metadata engines are one way in which the spatial data that is already in existence can be utilised more effectively by spatial data users.

The metadata engine that makes the virtual database possible was the topic of this thesis. The primary objective of the thesis was to develop a prototype metadata engine that is capable of undertaking distributed processing. The prototype was developed by modifying the source code of the public domain software Isite to make it act as a metadata engine. In order for the prototype to be developed effectively the following tasks had to be completed:

1) Investigate the concepts of SDIs, Data Warehouses, Clearinghouses, Interoperability, etc;

2) Investigate the current developments world wide in the areas of spatial data and metadata;

3) Investigate the options that are available in the development of a metadata engine;

4) Develop the metadata engine; and

5) Investigate the limitations and problems of the developed prototype.

This chapter summarises the findings of this research, forms conclusions based on these findings and makes recommendations arising from this research.

1.2 Summary and Conclusions of Thesis

The concepts of metadata and metadata engines were discussed in Chapter 1. It was noted that there is a distinct difference between a metadata engine and a metadata system. A metadata engine allows the user to not only search for datasets on a network, but also to concurrently display the datasets that result from the search. A metadata system on the other hand only allows the user to search for the datasets, it does not allow for users to display the dataset.

Chapter 1 also discussed that in recent years the dominant computer paradigm is moving away from closed systems to open systems. Open systems allow for application environments to interoperate with each other to provide more flexibility for the user. The days where decisions are made using a stand-alone GIS are numbered. Applications that interoperate with many datasets located over a large network will become more prevalent in the future in the decision making process.

Chapter 2 investigated concepts that have recently emerged in the more efficient handling of data, not just spatial data. Concepts that were investigated included SDIs, data warehouses, data marts, interoperability, clearinghouses and data mining.

SDIs are essentially the technology, data, communication, standards, procedures and partnerships that have been put in place that allow the dissemination of spatial data across a network, whether that network be electronic or not. Interoperability is a subsection of SDIs. It is the standards and procedures that are put in place that allow for spatial data to be transferred and accessed across an electronic network.

Data warehouses and data marts are essentially the same concept, except that data marts have a more departmental focus as compared with the data warehouse overall view. Data warehousing and data marts involve the aggregation of all of an organisations datasets into the one database. Data mining is the analysis of spatial data that is done from within the data warehouse and data mart environments and as such is reliant upon those two technologies.

A Clearinghouse is an advanced form of search engine designed for the dissemination of spatial data. They are part of the overall spatial data infrastructure and allow a person with access to a network to search that network and find what spatial data

exist, what are the characteristics of the dataset, and how they can get hold of the data if a direct link to the data does not exist.

Chapter 3 outlined some of the projects and policies that are being, or have been, implemented at the state, national and international levels. All of the projects and policies that were outlined in the chapter had a goal of improving the use of spatial data throughout the organisation that implemented them, as well as to the general community. This is a similar goal to that of the topic being researched in this thesis.

Victoria's Geospatial Information Strategy and the DOI GIS Strategy are two policy documents that were developed by Victorian government departments with a view to improving the use of spatial data within the departments as well as improving the use of it in the general community. Issues such as pricing policies, data custodianships, metadata collection and distribution, data transfer standards, access infrastructure and education are all covered within these documents. Both policies are of particular relevance to the concepts being investigated within this thesis.

Land Channel and GI Connections are two online Victorian government initiatives that have a primary aim of disseminating information and services dealing with spatial datasets across the web. Land Channel deals primarily with the delivery of spatial data services that are already available over a counter in a government department. It is part of the ESD project of the current government. GI Connections is a searchable directory of material that is in some way related to spatial information for the state. Both of these projects are essentially first steps in the process of achieving the objectives of this thesis. Both of these projects allow you to search for spatial data, perhaps even buy it, they do not allow for all the results of a users search to viewed concurrently online, which was the intention of the thesis.

The Australian Spatial Data Directory is a searchable listing of Australian public and private spatial datasets. Each of the datasets that are listed within the directory have a metadata record associated with it that conforms to the ANZLIC metadata guidelines. This project is extremely relevant to the concepts investigated in this thesis as it is the first step in developing a virtual database. The Australian Spatial Data Directory allows the user to search for certain datasets and returns the corresponding metadata

records as the result. The concept investigated in this thesis goes one step further and also allows the actual datasets to be viewed and queried.

BLIN was an initiative that was undertaken by the Department of Natural Resources in Queensland. It allows single point access to a variety of departmental databases concurrently. The project essentially achieves what was intended in this thesis. It allows for the distributed processing of spatial data across a network. It does however have a few differences. Firstly the application is not available on the Internet. Secondly, the application is not easily extendable due to the fact that all the functions, locations of datasets, etc are all hard coded into the application. If new datasets are to be added to the system a lot of code will have to be changed. Thirdly, each of the datasets are stored in the same proprietary format. This thesis intended that the datasets being accessed could be stored in any proprietary format. Finally, all the datasets are located on the local network. It was intended in this thesis that the datasets could be located anywhere around the world.

The New Brunswick Real Property Internet Information Service is a web based application that allows users to access three of its databases simultaneously. The user does not actually access the individual datasets as such, but instead a data warehouse that contains all the data from each of the individual datasets. The concepts in the New Brunswick Real Property Internet Information Service are similar to the concepts investigated in this thesis in that it allows access to several datasets at once. The difference is that the approach taken in this thesis was to keep the datasets separate.

Chapter 4 outlined the concepts that were involved in the development of a metadata engine that was capable of allowing a virtual database to be developed. For a metadata engine to allow for users with different access privileges to access datasets differently, the view that the user has of each of the datasets could be classified into one of five types:

1) Imported data;
2) Local public access data;
3) Local private data;
4) Remote modifiable data; and

5) Local remotely modifiable data.

Different users will have differing views of the same datasets depending on what type of user they are. Certain datasets may be remote modifiable for some users and yet they are only imported data for others.

When implementing a metadata engine there are five basic models that could be used to develop the system. Each of the models had its own advantages and disadvantages and each of them varies in difficulty as far as implementation is concerned. The five basic models differ in where the metadata and the metadata engine are located. The models are:

1) Metadata and metadata engine are located on the users own machine;
2) Metadata is located on the data custodians machine and the metadata engine is located on the users own machine;
3) Metadata and the metadata engine are located on one central server;
4) Metadata is located on the data custodians machine and the metadata engine is located on an independent central server; and
5) A combination of the 3rd and 4th options.

Any of these options could be used to implement the metadata engine. It is simply a matter of which of them best suits the needs of the user group that is to use the engine.

The chapter also outlined two methods for storing access metadata that was required for the remote user to gain access to the individual datasets. This access metadata was stored at the same location as the ANZLIC compliant metadata records that they belong to. The two options have their own advantages and disadvantages and are:

1) A pages approach, where the access metadata is simply located at page levels 1,2 etc in the ANZLIC metadata records; and
2) A separate file approach, where all the access metadata for the server is located in the one file in a standard format.

The first approach is probably more logical and would be easier to maintain, however the second approach is much easier to code and implement.

Chapter 5 outlined both the choices that were made in developing the prototype and the modifications that were made to the Isite source code in order to allow it to act as a metadata engine. Of the data models that were described in chapter four, option 5 was the one that was chosen to implement the metadata engine. Option 5 had the metadata engine located on a central server, and the metadata located on the data custodian's server, or on the central server. There are several reasons as to why this model was adopted:

1) There is a minimal wastage of resources. In several of the other options metadata was duplicated on two or more machines.

2) Data custodians remain in control of their metadata. This not only helps to alleviate the minds of the data custodians concerned about losing control of their metadata, it also ensures that the metadata that the engine uses is up to date.

3) Adding new servers to the metadata engine is straight forward. In the situation where there are many copies of the metadata engine located in many, sometimes unknown, locations updating the metadata engine to search a new server for metadata is a major problem.

4) The data custodian has the opportunity to have their metadata stored on the central server if they wish. This is useful where the data custodian does not have the ability to serve the metadata to the system themselves.

Of the two options that were outlined in chapter four detailing how to handle the access metadata it was the second option, the separate file approach, that was chosen as the appropriate method. The lack of a standard format for the lower pages in the ANZLIC metadata guidelines and ease of implementation of the second option were two of the reasons for undertaking this approach.

The separate file that was developed to hold all the access metadata was called “directory.txt”. It is located in the same directory as the zserver executable and is set up using a standard format that has a marker before each of the metadata elements to identify them. When a metadata record is found as a result of a user's search, the metadata file is consulted to find the access metadata.

The prototype was developed by modifying the software package called Isite. Isite is a public domain package developed in the USA that uses the z39.50 ANSI/NSIO standard communications tools to access metadata databases across a network. It allows for a web based interface to be developed that can search multiple metadata databases located in multiple locations across a network.

The prototype metadata engine works in the following fashion:

1) A http to z39.50 stateful gateway is established between the users machine and the metadata engine is located.

2) Search parameters are entered into a search page that appears as a result of establishing the http to z39.50 stateful gateway.

3) The search parameters are sent to each of the remote metadata databases by Zserver. The metadata records at each of the servers are searched to see if they satisfy the search parameters.

4) All metadata records that satisfy the search parameters have a hyperlink to the metadata record and its corresponding access metadata encoded onto a return string and passed back to the metadata engine.

5) The return strings are decoded upon after being received by the metadata engine and the decoded information is written to the relevant web pages and files.

6) By clicking the relevant hyperlink on the results web page the metadata records can now be viewed. The datasets can be viewed using the spatial data viewer that the web page has loaded.

There were many modifications that were made to the Isite source code to enable it to act as a metadata engine. These modifications can be classified into three stages:

1) Adapting the Isite source code to run as originally intended by CNIDR on two different servers.

2) Modifying the Isite source code to return the access metadata.

3) Linking the developed metadata engine to a spatial data viewer.

Chapter 6 outlined the limitations that the developed prototype had along with some of the improvements that could be made to eliminate or improve these limitations. The prototypes limitations could be classified into policy and coding limitations. Policy limitations can, in general, only be solved by government, whereas the coding limitations could be solved by a developer who had time to spend developing a commercial system.

The policy limitations included:

1) Accuracy and integration. Very few, if any, datasets are completely accurate. This results in datasets shown concurrently by the spatial data viewer not fitting together properly.

2) Charging. The prototype has no ability to charge the users for the data that they access.

3) Privacy. The creation of new information through the concurrent viewing of several datasets can result in privacy concerns for the general community.

4) Speed. Poor network speed is a hindrance to the effective installation of a virtual database.

5) Metadata divergence. If metadata standards in Australia diverge to far away from those the adopted standards in the USA and Europe software may be difficult to obtain.

The coding limitations include:

1) Limited querying. The spatial data viewer is very limited in the GIS functionality that it is able to perform. Extra code needs to be added to the spatial data viewer to give it extra querying capability.

2) Limited access. Since the access metadata is always written to the "Directory_Andrew.txt" file the metadata engine can only accommodate one user at a time.

3) Cryptic search criteria. The search criteria for each of the datasets is the ID for a particular line or point in the dataset. This ID has no relationship to its location in space and hence it is very difficult to search the datasets.

4) Hard coding of source file locations. Installation of the metadata engine on different servers is made difficult as the names and locations of the files that contain the access metadata are hard coded into the source code.

5) Points and lines display only. The spatial data viewer has the ability to display points and lines only, it does not have the ability to display polygons and attribute data.

6) Firewalls. Many organisations have the datasets located behind firewalls in order to protect them from outsiders.

7) Data currency. A result of having a mirror copy of a dataset that is located behind a firewall is that the currency of the dataset that the accessing are using has to be known.

1.3 Recommendations

The use of metadata engines that allow the development of large virtual databases will become more prevalent in the not too distant future. There ability to allow as much data to be incorporated into the decision making process will be seen as an increasingly valuable asset by decision makers in the years to come.

Based on the outcomes of this research the following recommendations have been formulated:

1) A metadata engine should be implemented that incorporates all government spatial data at the very least. Government departments should have full access to the system to encourage more efficient use of the government's spatial datasets. The general public should have access to a cut down version of the metadata engine that allows access to non sensitive data.

2) More work needs to be undertaken on the metadata engine to incorporate the developments coming out of the OpenGIS corporation in order gain better interoperability within the metadata engine.

3) More work needs to be undertaken on eliminating the current limitations that exist with the metadata engine prototype. Ideally a company with more man hours and money to throw at a metadata engine could do this.