Geo-Visualization for Geo-Science Education

By

Birgit Aagaard Woods, B.Sc. (Hons.)(SSP).

A thesis submitted to
the Faculty of Graduate Studies
in partial fulfillment of
the requirements of the degree of

Master of Science

Carleton University

Ottawa, Ontario

© September 2005 Birgit A. Woods

Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 0-494-10159-8

Our file Notre référence

ISBN: 0-494-10159-8

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Abstract

This thesis provides a study of terrain-rendering techniques and technology for the purpose of conveying geo-scientific information in the form of an educational, interactive, and scientifically representative geo-visualization. Specific techniques investigated include Level-Of-Detail (LOD), fractal, and voxel-based terrain rendering. The case study involves the use of high resolution LIDAR data of the McMurdo Dry Valleys of Antarctica to test the techniques used by specific software packages. These include Virtual Terrain Project, with LOD capabilities, NGRain, with voxel-based rendering, and the Virtual Reality Modelling Language (VRML), a programming language for creating three-dimensional objects and environments using LOD.

Through this research it has been found that three steps are common to all practices: 1, Data conversion and processing; 2, Terrain population with information and objects; and 3, Terrain Run-Time Viewing. The common limitation is the apparent trade-off between interactive quality and visual quality.

Conclusions are that LOD and Fractal based rendering are most suitable to maintain a balance between interactive and visual quality. In addition it was found that the proper use of landmarks and metaphor provide the best elements of design for Virtual Learning Environments (VLEs). VRML offers the best means to place landmarks, incorporate multi-media elements, and links to other sources of information while encouraging learning through exploration. VRML was therefore used in creation of the McMurdo Learning Module; a VLE teaching geomorphology using the high resolution LIDAR data.

Acknowledgements

Many people warrant recognition towards my experience as a Master's student and creation of this thesis. My supervisor, D.R. Fraser Taylor merits a special thank you for selecting me to join the talented group of researchers in the Cybercartography and the New Economy project. My experience provided me with a unique opportunity to gain technical and practical knowledge, leadership skills, as well as the opportunity to attend and present at international and local conferences, all of which go far beyond what I expected at the Master's level. These experiences will allow me to move forward with a strong education and will always represent a positive and significant time in my life. Funding was provided by the Social Sciences and Humanities Research Council of Canada (SSHRC), which should be acknowledged as well.

I would like to thank the two members of my thesis committee for their support as well. Sebastien Caquard, who provided help with the content, structure, and organization of the thesis, and in particular, offered insight into the artistic dimensions, which helped make this thesis more than a technical paper but a thoughtful paper covering many theories and topics. Naomi Short of Natural Resources Canada, who not only provided much editorial advice but offered scientific insight into the use of visualizations from a glacial scientist's perspective. Naomi also prevented the final version of the thesis from being written as 'waffley' as it was in the draft versions.

The thesis could not have been accomplished without the technical assistance and access to software provided for me. A thank you therefore is due for Amos Hayes, technical manager at the Geomatics and Cartographic Research Centre, who managed the messy licensing issues with ESRI. Thank you as well to Dr. Chris Herdman, head of the

Centre for Applied Cognitive Research who provided me with a much appreciated work area within his lab, and Murray Gamble, Director of Modeling & Simulation for the Aviation and Cognitive Engineering (ACE) laboratory, who provided me with technical support and access to the NGRain and Creator Terrain Studio software.

All members of the Cybercartography project provided me with a rewarding experience including Peter Pulsifer, whom I have learned a lot from about content development for the Atlas, members of the game cluster group who taught me much about team-work and the fun and work in games, members of the Human Oriented Technology Lab who tested my VRML visualization, Lisa Hagen and Jon Wade of the ACE lab who met with me to discuss the use of landmarks in virtual environments, Wataru Watanabe who gave very useful advise regarding the aesthetics of the McMurdo Learning Module, and the Cybercartographic Atlas of Antarctica cluster group, who taught me a great deal about project development, implementation, leadership and the creation of learning modules.

On a personal note, I would like to thank Mike Swan for his on-going support for success in my studies, and my parents Drs. Gurli and Howard Woods for their continued encouragement through these two years. In particular, I would like to thank my mother, for if she had not received the Carleton Newsletter in her email with a description of Dr. Taylor's project, and if she had not thought to forward that to me while I was still studying geology at Queen's, I likely would not have applied and joined the Cybercartography project, and would therefore have missed this great opportunity entirely.

Thank you all for a wonderful experience.

Table of Contents

Abstract.....	ii
Acknowledgements.....	iii
Table of Contents.....	v
List of Tables.....	viii
List of Figures.....	ix
List of Acronyms.....	xiii
INTRODUCTION.....	1
CHAPTER 1: VISUALIZATION AND GEO-SCIENCE.....	5
1.1 INTRODUCTION.....	5
1.2 GEO-VISUALIZATION.....	5
1.3 COMPUTER GENERATED IMMERSIVE VIRTUAL ENVIRONMENTS.....	8
1.4 VISUALIZATION AND COMPUTER GAMES.....	10
1.5 VISUALIZATION TECHNOLOGY.....	11
1.6 ISSUES IN VISUALIZATION.....	12
1.6.1 USER PERCEPTION AND UNDERSTANDING.....	17
1.6.2 ORIENTATION AND NAVIGATION WITHIN VIRTUAL SPACES.....	19
1.7 CONCLUSION.....	21
CHAPTER 2: VISUALIZATION FOR EDUCATIONI AND CONTENT DESIGN FOR SERENDIPITOUS EXPLORATION OF VIRTUAL ENVIRONMENTS.....	23
2.1 INTRODUCTION.....	23
2.2 VISUALIZATION FOR GEO-SCIENCE EDUCATION.....	23
2.2.1 COMPUTER-BASED LEARNING IN GEO-SCIENCE.....	24
2.2.2 VIRTUAL LEARNING ENVIRONMENTS.....	26
2.2.3 VIRTUAL FIELD TRIPS.....	28
2.3 CONTENT DESIGN FOR VIRTUAL ENVIRONMENTS TO FACILITATE EDUCATION THROUGH SERENDIPETOUS EXPLORATION.....	32
2.3.1 CONTENT DEVELOPMENT AND VIRTUAL ENVIRONMENT DESIGN.....	33
2.3.2 THE USE OF LANDMARKS IN VIRTUAL ENVIRONMENTS.....	34
2.3.3 SERENDIPETOUS EXPLORATION.....	36
2.3.4 THE USE OF METAPHOR IN VIRTUAL ENVIRONMENTS.....	38

2.4 THESIS CASE STUDY – VISUALIZING THE MCMURDO DRY VALLEYS OF ANTARCTICA.....	39
2.4.1 THE MCMURDO LEARNING MODULE: GEOMORPHOLOGY OF THE MCMURDO DRY VALLEYS OF ANTARCTICA.....	39
2.5 CONCLUSION.....	42
 CHAPTER 3: LOD, FRACTAL, AND VOXEL-BASED TERRAIN RENDERING TECHNIQUES	44
3.1 INTRODUCTION.....	44
3.1.1 BACKGROUND TO TERRAIN RENDERING.....	44
3.2 LEVEL-OF-DETAIL (LOD) TERRAIN RENDERING.....	47
3.3 FRACTAL-BASED TERRAIN RENDERING.....	61
3.4 VOLUME-BASED TERRAIN RENDERING.....	69
3.5 CONCLUSION.....	71
 CHAPTER 4: THE TOOLS OF VISUALIZATION.....	75
4.1 INTRODUCTION.....	75
4.2 USGS LIDAR DATA FOR THE VISUALIZATION OF THE MCMURDO DRY VALLEYS OF ANTARCTICA.....	75
4.3 VIRTUAL TERRAIN PROJECT [HTTP://WWW.VTERRAIN.ORG/].....	76
4.4 VIRTUAL REALITY MODELLING LANGUAGE (VRML).....	80
4.5 NGRAIN [WWW.NGRAIN.COM].....	84
4.5.1 NGRAIN TECHNIQUE OF THREE-DIMENSIONAL OBJECT RENDERING.....	87
4.6 CONCLUSION.....	92
 CHAPTER 5: CASE STUDY AND METHODS OF EVALUATION.....	94
5.1 INTRODUCTION.....	94
5.2 DISCUSSION AND EVALUATION.....	94
5.2.1 DISCUSSION OF VIRTUAL TERRRAIN PROJECT.....	95
5.2.2 EVALUATION OF VIRTUAL TERRAIN PROJECT.....	98
5.2.3 DISCUSSION OF NGRAIN.....	103
5.2.4 EVALUATION OF NGRAIN.....	108
5.2.5 DISCUSSION OF VRML.....	110
5.2.6 EVALUATION OF VRML.....	113
5.2.7 ADDITIONAL WORK WITH VRML TOWARDS CREATING THE MLM.....	113
5.3 OVERALL CONCLUSIONS.....	120
 CHAPTER 6: DISCUSSION AND FUTURE DIRECTIONS.....	122

6.1 DISCUSSION.....	122
6.2 FUTURE DIRECTIONS.....	128
6.2.1 VISUALIZATION FOR EXPERT KNOWLEDGE DISCOVERY AND COLLABORATION.....	128
6.2.2 GEO-VISUALIZATION AND KNOWLEDGE DISCOVERY IN DATABASES (KDDs).....	129
6.2.3 GEO-VISUALIZATION AND COLLABORATIVE VIRTUAL ENVIRONMENTS.....	132
 CONCLUSION.....	 134
 REFERENCES.....	 138
 APPENDIX I.....	 144
APPENDIX II.....	147
APPENDIX III.....	148
APPENDIX IV.....	149
APPENDIX V.....	150
APPENDIX VI.....	151

List of Tables

2.1	The advantages and disadvantages of VFTs from the user's perspective. (Qui and Hubble, 2002:77).	30
3.1	A summary of the generalized characteristics of using LOD, Fractal, and Volume-based terrain rendering.	73
4.1	A sample of a VRML (.wrl) file created using elevation data that was exported from ESRI ArcScene 9.0.	82
4.2	Summary table of comparison between VTP, VRML, and NGRain.	93
5.1	The extents of the clipped raster files of the McMurdo Dry Valleys of Antarctica. Clips resulted in an approximate 57% reduction in file size for both files.	95
5.2	File size reduction and export to VRML using ESRI ArcScene 9.0.	104
5.3	Conversion capabilities and reduction in file size by NGRain.	109
5.4	A sample of the 'anchor' node within VRML to make the coloured polygons interactive and link to the url provided.	111
5.5	Summary of comparison of ease of use between the different packages.	121

List of Figures

1.1	The use of visualization by experts through the process of scientific investigation (Adapted from DiBiase, 1990).	6
1.2	The cartographic visualization process (Kraak, 2002).	7
1.3	Visualization of an ore-body as an example of visualization in early stages of exploration in the private realm (a), to presentation and communication of results in the public realm (b) (Xu and Dowd, 2003).	7
1.4	CAVE virtual reality projection system (Watson et al., 2000).	9
1.5	Scene from the game Halo (XBOX, 2003).	10
1.6	Image and graphic as an illustration of low to high level of abstraction (MacEachren et al., 1992).	16
1.7	The process of knowledge through visualization, from 'real' or raw data to the brain's perception and visual interpretation (Gahegan, 1999).	17
1.8	Combining VR and GIS is best done using 2, 2.5, and three-dimensional visualization concurrently (Kraak, 2002).	20
2.1	Comparing the information travel paths of users through a virtual environment with serendipity (left), where the user misses key information, and Engineered Serendipity (right), where the user is alerted before leaving if they have missed any information (Cartwright, 2004).	37
2.2	The MapShop with curator, and user avatar in view (Cartwright, 2005).	38
3.1	A coarser mesh, as seen in a, reduces the resolution of the terrain in the visualization. More polygons, as seen in b can create more realistic terrain, but demands more computer processing power (Coddington, 1996).	46
3.2	Regular grid DEM data structure (USGS, 2002).	46
3.3	a) is the wire frame TIN structure of terrain. b) is the Rendered Model of the same terrain (Hurni, 1997).	46
3.4	Two-level hierarchical tree structure. The shaded triangle in a) corresponds to the shaded node in b) (Modified from de Berg and Dobrindt, 1998).	49
3.5	Delaunay Triangulation Method. Modified from (de Berg and Dobrindt, 1998).	50
3.6	Result of the ExtractCombination Algorithm for the changing level of detail over a virtual terrain surface with highest LOD closer to the	53

	viewer, decreasing with distance. Modified from (de Bergand Dobrindt, 1998).	
3.7	a), a grid of 4 tiles where the bordering triangles are not at a high enough resolution to accommodate the fine LOD within the tiles. b), a grid of 4 tiles where coarse LOD within the tiles is not balanced by the smaller triangles in the borders. Polygon budget is spent on the borders instead of in the tiles making reduced LOD over large areas (Terrex, 2004).	54
3.8	This figure illustrates how the SmartMesh concept where the LOD of the tiles change with movement of the viewer, and how the boundary is able to accommodate the changes while still upholding the polygon budget (Terrex, 2004).	55
3.9	Terrex Screenshots. a) Viewing the terrain as TIN reveals six complete sections of triangles. b) The terrain image is activated for comparison with a). The coloured shapes in a) correspond to those shown in c) in the next page. c) Change in eyepoint creates a change in TIN rendering. The boundary between LOD sections is visible. d) With the image activated, the boundary between low and high levels of resolution is visible. The rectangle and circle correspond to the same shapes in c).	59-60
3.10	Demonstration of Midpoint Displacement in One Dimension algorithm. Modified from (Martz, 1997)	62
3.11	Each step in the diamond-square algorithm. Modified from (Martz, 1997).	63
3.12	The second iteration of the diamond-squares algorithm. Modified from (Martz, 1997).	64
3.13	Screen shot downloaded from Blueberry 3D website illustrating varying levels of complexity within the virtual scene (Blueberry 3D, 2004).	67
3.14	Generation of a road within the virtual terrain. Left, sequence of points describing the shape of the road. Middle, a cross-section of the surface of the road. Right, the final rendered image. Modified from (Blueberry3D, 2004b).	69
3.15	TIN visualization of a discontinuous surface like a building (Stoker, 2004).	70
4.1	Example of the top-down (low to high resolution) data structure for triangulation algorithm used in VTP. a) The arrows indicate parent-child relationships between nodes in the structure. b) Illustrates the same data structure indicating the triangulation fanning process where gaps in adjacent areas of different resolution are avoided by not splitting vertices that are borders to different subdivisions. The x marks skipped vertices (Röttger et al., 1998).	78
4.2	McNally or ROAM algorithm using equally split triangles (Binary Triangle Trees) over three generations of children.	79

4.3	Terrain generated at a) low, b) optimal, and c) high variance settings. Modified from (Turner, 2000).	79
4.4	Components of an Elevation Grid within VRML. (Carey et al., 1997)	84
4.5	An illustration of the concept of Knowledge Objects in relation to 3D Knowledge Objects, and Reusable Learning Objects which are derived from the concept of Learning Objects. (Simpson, 2004).	87
4.6	4x4x4 voxel cube with three occupied voxels (Woo and Halmshaw, 2004).	88
4.7	Diagram illustrating the trade-off between Computation Speed and Visual Quality. A balance needs to be maintained to ensure a proper level of interactive quality for the viewer. The circle illustrates that with improved visual quality, the user experiences a compromised level of interactive quality.	93
5.1	Settings inputted into the Merge and Resample Elevation Tool within VTBuilder.	96
5.2	Layers incorporated into the image layer including a coastline (green), glacier flow-lines (blue), and glacier moraines (purple).	97
5.3	Clipped LIDAR file clipelev30m.tif loaded into VTBuilder.	99
5.4	Top-Down view using the Röttger Algorithm: Triangle Count 10,000 and increasing detail in Enviro.	99
5.5	Röttger Algorithm Screenshots of the Upper Wright Glacier. Notice as the view-point changes, so does the number and size of the triangles, illustrating the effects of the algorithm.	101
5.6	McNally Algorithm Screenshots. Notice as the viewpoint changes, so does the number of triangles, illustrating the effects of the algorithm. Triangle Count: 10,000. Fan method activated	102
5.7	Terrain loaded into NGRAIN Knowledge Module with link windows open.	106
5.8	NGRAIN with the McMurdo terrain loaded into the Mobilizer application with X-RAY view activated.	107
5.9	The NGRAIN Mobilizer application embedded into an HTML page as an interactive learning environment for students to learn about the geomorphology of the McMurdo Dry Valleys of Antarctica.	110
5.10	The VRML version of the McMurdo terrain as seen in prototype of the Cybercartographic Atlas of Antarctica Project.	112
5.11	Improved VRML version of the MLM.	115
5.12	The MLM with grid lines marking paths within the virtual environment following naturally occurring valleys. All landmarks	116

appear within the paths and are lined up with the grid structure. Those paths that deviate from the main grid structure appear in green.

5.13	Figure 5.13. Screenshots of the MLM with focus on the Upper-Wright Valley, to show the helicopter and live video landmark [3], a billboard of a mummified seal [2], and signs marking the outlet glaciers [1]. The resolution over the Upper-Wright Valley and the Outlet Glaciers is increased in these areas.	117
5.14	a) shows the extents of the sound bubble where upon hitting the boundary, the user hears running water. b) shows the extents of the clickable feature, leading the user to the Don Juan Pond.	118
5.15	Black and white representation of the MLM in wireframe view where a) shows areas of high resolution with more triangles representing the terrain in those areas. b) is a close-up view of the Outlet Glaciers. Movement from a) to b) to c) reveals changes in the LOD, with increasing numbers of triangles as the viewpoint moves closer to the terrain of high resolution. No attempts are made to make the boundaries appear seamless. d) reveals the result in plain view.	119
6.1	The breakdown of stages of the scientific method and scientific inference for each stage through the process of KDD, or knowledge construction (Gahegan et al., 2001).	130
6.2	Methods of exploratory visualization. Modified from Gahegan et al., 2001.	131
7.1	An illustration depicting the overall success of the three products through the three stages of development. VRML leads in many areas followed by NGRain and VTP for the creation of the McMurdo Learning Module.	135

List of Acronyms

.bt.....	Binary Terrain file format
CAVE.....	CAVE Automatic Virtual Environment
CD-ROM.....	Compact Disk Read Only Memory
CVE.....	Collaborative Virtual Environment
DEM.....	Digital Elevation Models
ESRI.....	Environmental Systems Research Institute
GIF.....	Graphics Interchange Format
GIS.....	Geographic Information Systems
HTML.....	Hyper Text Markup Language
JPEG (JPG) (.jpg).....	Joint Photographic Experts Group
KDD.....	Knowledge Discovery in Databases
LIDAR.....	Light Detection and Ranging Remotes Sensing
LOD.....	Level-Of-Detail Terrain Rendering
MLM.....	McMurdo Learning Module
.ngw.....	NGRAIN file format
.obj (OBJ).....	Object file format
PC.....	Personal Computer
.prj.....	External Projection file format
RAM.....	Random Access Memory
RIO.....	Reusable Information Object
TIF(F) (.tif).....	Tagged Image File Format
TIN.....	Triangulated Irregular Network
URL.....	Universal Resource Locator
USGS.....	United States Geological Survey
VE.....	Virtual Environment
VLE.....	Virtual Learning Environment
VFT.....	Virtual Field Trip
.vol.....	Volume file format
VR.....	Virtual Reality
VRML.....	Virtual Reality Modelling Language
VTP.....	Virtual Terrain Project
.wrl.....	VRML file format
XML.....	Extensible Markup Language
2D.....	Two-Dimensional
3D.....	Three-Dimensional
3DKO.....	3D Knowledge Object

Introduction

Within the disciplines of geography and earth science, hereafter referred to as geo-science, there is a need to visualize earth structures in order to understand their formation. In the past, structural phenomena were visualized using drawings on paper, or using foam, plastic, and wood to make physical three-dimensional models (Schmidt and Götze, 1998). These forms of visualization are effective since with a physical model, a person can feel the structure and look at it from different perspectives. Computer technologies expand on this concept, and computerized visualization presents innovative opportunities to study and interact with geographic and scientific information.

Visualization is important because our ability to form mental pictures based on indiscriminate information helps to support recognition, communication, and interpretation of complex information (Ottoson, 2003). Similarly, visualization is often necessary to turn large amounts of heterogeneous data, into manageable information, and consequently, into knowledge (MacEachren and Kraak, 2001).

Geo-scientists and students in geo-science can benefit from powerful terrain visualizations to observe structural phenomena within an environmental context. Such visualizations can enable interaction and exploration of geo-scientific information in a manner that is not possible through traditional methods of using maps and air-photographs, or even through visiting the physical location. There is no doubt that these methods are valuable to research, however they may be improved by what computerized visualization can offer. The purpose, therefore, of the visualization is not to mimic the natural environment, but to enhance upon it. Depending on the technique of computerized visualization used, features in the terrain can be represented differently, causing change

in one's perception of the space as the environment is viewed and explored from different perspectives. Not all computer visualization technologies are created equal, however. Some are better equipped with memory and processing abilities to handle the large amounts of data required for detailed rendering of visualizations. In fact one of the greatest limitations of computer visualization technology is the trade-off between interactive quality and visual quality. Much research has gone into producing visualizations that attempt to accomplish both, with varying levels of success.

With these principles in mind, the thesis will address the following research questions:

Which terrain rendering techniques and software are most suitable for generating a geo-visualization for use in communicating geo-scientific information while maintaining a successful balance between interactive quality and visual quality?

How can different terrain representations be used and virtual environments designed to enhance geo-science education for students?

The thesis aims to answer these questions through studying techniques in terrain rendering and exploring different software packages available to generate virtual environments. The thesis explores the use of these rendering techniques to provide a platform from which students can explore the terrain prior to, or as an alternative to, field investigation for the discovery of scientific information related to geographic space. Three-dimensional terrain rendering may prove beneficial for geo-science education by teaching geomorphology in an alternative way and in conjunction with traditional methods. It is hoped that through this research, the creation of virtual yet scientifically representative environments can improve communication and understanding of geo-scientific information to students.

The thesis begins with background information on the topic of visualization science and its applications and uses within geo-science. This section addresses the theories of computer generated immersive virtual environments and their implications for geo-scientific visualization. Leadership by the gaming industry in visualization is discussed and issues in visualization technology are examined with respect to interactivity, perception and understanding, orientation and navigation, among others.

Chapter 2 focuses on addressing the uses of visualizations as educational tools. The benefits and limitations are discussed, as well as implementation and design principles. These are put into practice as part of the thesis case study which involved creating an interactive learning module using high-resolution data of the Dry Valleys of Antarctica.

Chapter 3 covers geo-visualization technology and techniques with the focus placed on three specific visualization practices: Level-Of-Detail (LOD), Fractal, and Voxel based terrain rendering. Study of these techniques provides the foundation for knowledge in terrain rendering practices that enable analysis of specific software that uses similar methods.

Chapter 4 provides a discussion of three specific software packages and their methods for rendering three-dimensional terrain. These packages include: Virtual Terrain Project (VTP), which uses continuous LOD methods of terrain rendering; NGRain, which uses Voxel technology; and the Virtual Reality Modelling Language (VRML), which is a programming language used to build three-dimensional environments with multi-media and other interactive elements for visualization over the Internet.

Chapter 5 outlines the use of the afore mentioned packages from the perspective of a terrain developer. It provides a discussion of the experience of using the product and an evaluation of the product in its effective delivery of the visualization, its ability to handle the high-resolution data, and its ability to meet the design criteria discussed in Chapter 2. A CD is included in the thesis to illustrate the technology discussed and allow interaction with the 3D terrain models. Instructions on using the CD are provided in the CD folder, and are also explained in appropriate sections within the text of the thesis.

Chapter 6 outlines future directions for research including the use of geo-visualization as an expert's tool for pattern recognition in large databases, and as a collaborative research tool.

A conclusion is then offered which brings all elements together from previous Chapters, thereby providing answers to the thesis research questions.

Chapter One: Visualization and Geo-Science

1.1 Introduction

The visual representation of physical objects, abstract notions, or complex data and information are what define visualization. Maps are abstract cartographic visualizations of our world, that help us better understand our environment (Ottoson, 2003). Geographic visualizations, specifically, try to provide a more realistic, rather than abstract, representation of landscape and terrain.

With an estimated 80% of available digital data being geo-spatially referenced, through geographic coordinates, addresses, and postal codes, integrating this immense resource of data presents a challenge for information scientists; how to “transform these data into information, and subsequently into knowledge” (MacEachren and Kraak, 2001: 4).

Although many advances in technology have improved visualization in science, from medical imaging to molecular chemistry, visualization is still not used to its full potential using geo-spatial data (MacEachren and Kraak, 2001). This chapter is a literature review that explores the concept of visualization within a geo-science context giving an overview of the discipline and technology, its uses in computer generated virtual environments, its place in the gaming industry, and identifies several technical and interpretational challenges associated with visualization.

1.2 Geo-Visualization

In geo-science, research often involves intimate knowledge of the physical environment being studied. Computerised visualization can therefore be useful. However,

in order to be fully functional, the visualizations must satisfy several requirements. At a minimum they must be geographically accurate, use appropriate data projections and scales, and should incorporate effective use of legends and symbology (Kraak, 2003). Visualizations with such geographical components are referred to as geo-visualizations, which are also defined as:

“The use of visual geo-spatial displays to explore data and through that exploration to generate hypotheses, develop problem solutions and construct knowledge” (Kraak, 2003: 398).

Geo-visualization integrates concepts from scientific visualization, cartography, information visualization, image analysis, exploratory data analysis, and Geographic

Information Systems (GIS), all of which provide the theory, methods and tools for investigation, study, synthesis, and presentation of geo-spatial data (Kraak, 2003).

DiBiase (1990) proposes a framework through which scientists can benefit from visualizations in all stages of scientific research. Figure 1.1 illustrates this concept, where visuals are important, and remain important during initial exploration, through to presentation of results. In DiBiase's view, visualizations contribute to private visual thinking during the early stages of research where they function as reasoning tools for experts in the private realm. As the research progresses, the map contributes to public visual communication, where a wider audience can understand and interpret the results. Visualization can therefore be thought

of as a research tool creating a connection between visual thinking in the private realm and visual communication in the public realm (MacEachren, 1994).

Figure 1.2. The cartographic visualization process (Kraak, 2002).

Kraak (2002) created a similar illustration for cartographic visualizations using abstract maps. Comparing Figure 1.1 with Figure 1.2, the similarities in process are clear beginning with exploration of geo-spatial datasets,

including data available from the World Wide Web, through to visual thinking involving the production of maps that visually display geo-spatial data, both of which feed into the presentation of geo-spatial datasets and contribute to the visual communication process of cartographic visualization.

Fig. 1.3. Visualization of an ore-body as an example of visualization in early stages of exploration in the private realm (a), to presentation and communication of results in the public realm (b) (Xu and Dowd, 2003).

In practice, scientific investigations by geo-scientists often require interpretation of sparse data samples, especially when mapping under-ground geological structures. Figure 1.3a

shows an example of drill core samples taken from an ore body, with the drill columns shown in red. The drill cores are used to determine the shape and structure of an ore body, shown in yellow. Experts determine where the contacts occur between the ore body and the surrounding bedrock by analysis of rock layers within the drill cores (Xu and Dowd, 2003). Visualization at this stage is within the private realm of research, and helps

to enhance visual thinking for experts during exploration and analysis of preliminary data. Based on the analyses, visualization tools interpolate the structure of the ore-body, relying on accurate portrayal of the drill columns including their geographic location, length, size, and orientation. The final result is shown in Figure 1.3b as a high level approximation of the structure of the ore body with vertical exaggeration to highlight the structural features along with the overlying topography. In this image, the drill columns have been removed, and focus is placed on final presentation of the results that an expert could submit in a journal article, lecture, or conference presentation in order to visually communicate the results of their findings.

Computer visualization technology was useful for the above visualization of the ore body. The following section describes the use of computers for the creation of virtual environments.

1.3 Computer Generated Immersive Virtual Environments

Three-dimensional visualization and virtual reality technology are advancing and have the potential to be used more often in scientific visualization. Virtual reality is a form of visualization that has the capacity to partially or fully immerse a person into a simulated environment. The contrast between abstract versus realism becomes apparent as virtual reality visualizations tend to focus on realism. The degree to which a viewer feels immersion in a virtual reality setting depends on the number of senses stimulated (MacEachren et al., 1999). 100% immersion has not yet been achieved (Ottoson, 2003), however, at a minimum, the user should have the ability to manipulate the imagery, and change her/his point of view (Haklay, 2002).

True three-dimensional computer visualization has not yet been achieved either. On a computer, the screen is two-dimensional. When interacting with a three-dimensional object on a computer, there is inherent ambiguity with unavoidable, yet varying degrees of distortion (Schmidt and Götze, 1998). However, virtual reality programming tools available today can offer an effective means of creating three-dimensional settings, where users can experience at least a sense of partial immersion. For example, the Virtual Reality Modelling Language (VRML), has been available since 1997 (Hay, 2003), and has been proven successful in many applications where three-dimensional objects can be interacted with using ordinary desktop computers and over the Internet. GeoVRML, a subset of VRML, was created to represent geographic information and create interactive three-dimensional terrain models.

While VRML is considered a common desktop variety of immersive virtual reality visualization, the CAVE (CAVE Automatic Virtual Environment), offers full-body immersion, (MacEachren et al., 1999) through the use of a large cube of projection screens that completely surround the viewer. This technology was developed in 1991, and has been tested and used in a variety of applications, from studying archaeology

Figure 1.4. CAVE virtual reality projection system (Watson et al., 2000).

(Acevedo, 2001), to understanding complex forestry data (Watson et al., 2000). Figure 1.4 shows the projection system for the CAVE using LIDAR (Light Detection and Ranging) remote sensing forestry data. The CAVE allows spatial data exploration individually or with a small group.

When researchers use virtual reality for visualization of geo-scientific data, according to Ching-Rong et al. (2002), the data become more informative and

understandable, allowing geo-scientists to look at more data simultaneously. This makes collaboration more effective as a result. Virtual reality facilitates data preview, communication of results, and to a lesser extent verification and interpretation of geo-scientific data (Ching-Rong et al., 2002). Therefore, looking back at Dibiase's diagram in Figure 1.1, virtual reality enhances both stages of visual thinking (with an emphasis on exploration), and the presentation stage of visual communication.

1.4 Visualization and Computer Games

Figure 1.5. Scene from the game Halo (XBOX, 2003).

Three-dimensional gaming technology available today inspires improvements within visualization science because of the advanced graphics technology, navigation, and interface designs (Germanchis and Cartwright, 2003).

Consumers are continuously upgrading their graphics cards for computer game play, and scientific visualization researchers can reap those benefits by catering to an audience likely to have their computers optimized for superior graphics rendering (Rhyne, 2002). Therefore, scientific visualization design and structure is largely influenced by computer game market dynamics (Rhyne, 2002).

There is, however, a conflict of interest between computer game developers and visualization scientists. In a computer game, there is little concern about data reliability and real-world performance (Rhyne, 2002). The market for computer games is also based on "short release cycles" which can allow incomplete and unstable graphics drivers that go unnoticed by most users. This kind of short-cut decreases the quality of the product in a way that is unacceptable to scientific visualization.

1.5 Visualization Technology

According to Ottoson (2003), there are three components to visualization technology. These are data, graphics, and interfaces. Data provide the content for the visualization, therefore it is crucial that the data be relevant and accurate. Graphics are what allow the user to see the visualization on a computer screen, and the interface makes the graphic interactive often involving keyboards, mice, and joysticks. The interface also includes what is visible on the screen, such as icons, windows, and toolbars, which help the user to perform various tasks.

Further, Ottoson (2003) identifies three mechanisms to visualization technology. These are: data acquisition, data management, and computation. Acquiring data is the fundamental step. With increasing complexity in technology and improved forms of data acquisition, data management has also become very important. Data are usually organized and managed in databases. With the onset of three-dimensional visualization, the databases have become especially large. Proper organization and metadata standards become crucial in order to maintain comprehensive databases. Effective computation, the third mechanism, is necessary to retrieve relevant data and information from the database for analysis and to produce representative visualizations.

1.6 Issues in Visualization

As previously mentioned, the purpose of visualization technology is to turn data into information and thereafter, into knowledge (MacEachren and Kraak, 2001). To accomplish this, major issues in visualization technology must be addressed. These issues include the following five elements that affect both the developer and the viewer. Each is discussed in greater detail below.

1. Computer rendering speed
2. Integration and compatibility
3. User interaction
4. Data and information
5. Visual quality and level of abstraction

1. Rendering Speed

Rendering speed is the amount of time needed to generate the visualization and the ability of a computer processor to deliver graphics for the visualization (Gahegan, 1999). There are four main reasons for limitations in rendering performance. These are: the graphics rendering speed of the computer, the size of the graphic or scene to be visualized, the amount of available Random Access Memory (RAM) within the computer, and the computer's ability to retrieve data (Gahegan, 1999). Solutions to improving rendering speed may include generating visualizations offline, reducing the resolution, or assembling the visualization as a movie, which would eliminate interactivity.

The volume of data required for geo-visualization is immense and can have significant repercussions for rendering speed and quality of interaction. The frame rate, the number of frames rendered within one second, usually defines the quality of interaction. An acceptable minimum is 10 frames per second, although, for a truly interactive experience, 24 frames per second are deemed necessary (Gahegan, 1999).

2. Integration and Compatibility

Integration presents the issue of compatibility with other computer technologies. The more a user needs to adapt to visualization technology, the more the developer of the visualization risks losing their audience. VRML for example works within existing Internet browsers that are commonly used by everyone. For VRML to remain useful, it

needs to continue to be compatible with new browsers and graphics cards, as well as older models (Hibbard, 1999).

This issue also raises the importance of interoperability in software, where data of different types and formats are compatible across multiple platforms, keeping conversion of datasets to a minimum.

3. User Interactions

Computer interactions today usually involve the use of a keyboard and mouse. In the future, voice and gestures may be used more often (Hibbard, 1999). The interface, which acts as the interactive medium between the user and the computer, should be straightforward to use and should not interfere with the main view.

A virtual reality interface can be difficult to learn, and a user may become irritated if required to click virtual menus and markers too often. Although “natural” interaction within a virtual environment is desirable, the menu system is currently what is normally used. Auditory menus may eventually prove to be most effective in these applications (Ching-Rong et al., 2000).

Currently, visualization tools are largely intended for individual use, and it is thought that user interaction could improve by developing more support for group work (MacEachren and Kraak, 2001). Improved interfaces and telecommunications technology may help facilitate this. The CAVE described in section 1.3, is an example of such an environment.

4. Data and Information

The complexity of a visualization is influenced by the complexity of the original data and the amount of information the developer wishes to display. For example, when

higher numbers of dimensions are introduced, the complexity of the visualization also increases. Linear information is simpler than two-dimensional information, which in turn is less complex than three-dimensional information. In geo-science, the fourth-dimension, time, is of great significance often with very long timescales which may only be surpassed by the field of Astronomy. It is very difficult to model geo-spatial data that is time-dependent, and furthermore, it is difficult to manage large volumes of data that are required for the more sophisticated forms of visualization (Goodchild, 1992).

According to MacEachren and Kraak (2001), current geo-spatial visualization methods cannot measure effectively, analyze, and display large amounts of available data. One solution is to have computing power that can extract patterns from large databases that work in conjunction with human vision to decipher meaningful patterns. This is further supported by Van Dam et al. (2002), who believe that humans and computers should work together as partners so that trends, anomalies, and patterns in data can be identified more effectively.

5. Visual Quality and Abstraction

Through examination of visual transitions within geo-visualizations, Ottoson (2003) has observed that in many cases artefacts tend to “pop up” when flying through a terrain, or zooming in at different scales. This affects the interactive and visual quality of the visualization for the user. Realism with the visualization is thus affected, and according to Hibbard (1999), the level of realism influences the reliability of the visualization for the viewer.

Although visualizations can represent reality, they are not meant to imitate or mimic reality according to MacEachren et al. (1999). Much as an air photo does not

provide as much useful information for navigation as an abstract map, an exceptionally realistic visualization does not in itself offer the best means for exploring geo-spatial information (MacEachren et al., 1999). Increasing the level of abstraction allows the viewer to derive more relevant information, making abstraction potentially more beneficial.

Abstractions are inherent in any form of visualization, and it remains controversial whether “abstraction is essential for achieving insight”, or if “realism is the ideal” (MacEachren and Kraak, 2001:9). According to Hibbard (1999), complex phenomena may require higher levels of abstraction in order to present the information in a clear and meaningful way. Abstraction however creates generalizations, which is generally avoided in scientific study.

In Figure 1.6 MacEachren et al. (1992) use a Landsat image as a demonstration of low level abstraction, while a distance decay diagram, common to geographers, indicates a graphic with a high level of abstraction. This figure also illustrates that from low to high levels of abstraction the visualization moves from being an image to a graphic. In a Landsat image, the markings are symbolic, and in order to understand them, training is needed. On the other hand, a graphic is relatively explicit, and can often be interpreted with little experience (MacEachren et al., 1992). The advantage of an image with low abstraction is that information can be presented in a direct way, and no information is lost through selection or in processing the image. The advantage of a graphic with a high level of abstraction is that it does not require as much knowledge to extract information. Understanding the audience for whom the visualization is intended is therefore important to determine which level of abstraction is suitable.

Figure 1.6. Image and graphic as an illustration of low to high level of abstraction (MacEachren et al., 1992).

6.1 User Perception and Understanding

“Visualization is unable to provide any statistical or quantitative analysis of the data, but instead relies on creating an impression of data properties within the mind of the observer”.

(Gahegan, 1999: 289)

This statement is significant, because it is saying that the purpose of visualization is not to mathematically analyze data, but to provide a sense of the data characteristics and relationships for the viewer. A viewer’s perception of a visualization is largely based on the elements that make up the visual stimulus, and depends largely on the experience of an individual and when the stimulation occurs (Gahegan, 1999). Figure 1.7 shows the progression of information through visual expression by beginning with the elements of raw data, to representing the data in a visual form, to displaying the information through a visual display, to finally decoding the information through an individual’s own perception and visual interpretation.

It is also known that visual properties of objects can interfere with each other in many

ways. For example, the apparent size of an object can be influenced by the colour of the object. The colour of surrounding objects or scenery can have an affect on the apparent colour of the object. As well, the distribution of a colour may be affected by the apparent colour shade or transparency of the object, and vice versa. The apparent distance of the viewer from an object can also be influenced by the colour, since objects tend to appear bluer and less saturated at a distance (Rheingans and Landreth, 1995). In an interactive immersive virtual environment these issues of perception become pronounced since the

viewpoint is always changing. Therefore, evaluation of the effectiveness of a scene is difficult since user perception is largely individualistic (Gahegan, 1999).

The way we perceive and understand a visualization has a lot to do with how the visualization is presented. In a visualization system or model, there are always visual and concealed aspects (Ottoson, 2003). Concealed aspects include the raw data, its acquisition, and its management. Most users have little interest in seeing the database itself, with large tables of records, since information in this form is not very meaningful. The visual aspects include data that have been transformed in some way that makes them more meaningful to the user. The data at this point are represented by an image in the form of a graph, illustration, or model. The creator of the visualization is therefore responsible for assigning information to visual or concealed categories, which can have significant political and ethical implications (Ottoson, 2003).

The way the mind perceives information in virtual environments brings a cognitive dimension to the design of visualizations. The mind's ability to cope with the virtual environment is not intuitive. Motor skills, for example, are different from reality to virtual reality (Munro et al., 2002); therefore it is important to understand the demands on the mind. Munro et al. (2002), found four basic knowledge paradigms the user must manage within virtual spaces. These are: location, structure, behaviour, and procedural knowledge.

Location knowledge involves the viewer's ability to determine their relative position within the virtual space. This is aided using visual cues through the use of maps and pictures, although experience and practice are the main influences on a viewer's capability. Personal navigation has been found to increase and be more effective in three-

dimensional virtual environments, than in two-dimensions. Being able to change a viewer's perspective, or manipulate an object adds to location knowledge within the virtual space, emphasizing the benefits of three-dimensional visualization (Munro et al., 2002). Structural knowledge involves the viewer's awareness of relationships among objects within a scene. For example the user shows structural knowledge by identifying that object A can support object B, or if an object within another object can be manipulated (Munro et al., 2002). Structural knowledge should be intuitive for the user as much as possible so that s/he can interact with objects with ease. Behavioural knowledge challenges the viewer's familiarity with how objects and participants interact with each other within the virtual space. This includes behaviours of cause and effect, stimulus and reaction, and functions of objects, which depends largely on the viewer's ability to produce changes of state within the environment (Munro et al., 2002). Procedural knowledge is gained through understanding the tasks, goals and actions required within the visualization or virtual environment.

1.6.2 Orientation and Navigation within Virtual Spaces

There are several issues in user navigation and orientation within a virtual space that are identified by Gahegan (1999). For example, exploring an interactive virtual space presents unfamiliar experiences for many users, and unfortunately, some become disoriented, and as a result, feel nauseous (Fisher and Unwin, 2002). As well, some users find it difficult to associate visual representations, within a virtual environment, with the objects these visualizations are actually meant to represent. Research suggests that using maps in virtual buildings with, for example, pictures on the virtual walls present in the actual buildings, greatly helps the viewer to orient and navigate the space (Fisher and

Unwin, 2002). However, while moving through a virtual space, it is entirely likely that a viewer will inadvertently position her/himself away from an object of interest.

Gahegan (1999) proposes two solutions to this problem. The first involves setting up tasks within the virtual space, increasing the level of difficulty as the user becomes more familiar with the space. This gives the user a chance to adapt to the environment and eventually understand complex visuals. The second solution is to have icons describing relations among concepts within the visualizations that the user can refer to for help or clarification, but a disadvantage to this is that the user then needs to learn

Figure 1.8. Combining VR and GIS is best done using 2, 2.5, and three-dimensional visualization concurrently (Kraak, 2002).

another visual scene and interface. It is, therefore, unclear whether this form of aid is actually effective in easing the user's learning within the virtual environment.

To improve navigation and orientation within a virtual environment, Kraak (2002) suggests combining Virtual Reality (VR) with Geographic Information Systems (GIS) as shown in Figure 1.8. This provides the user with three possible concurrent views; a plan-view, which presents a map in two-dimensions, a model-view, which displays the VR world from a bird's eye perspective in two-and a half dimensions, and the world-view, which is a three-dimensional view that allows the user to feel immersion within the visualization.

Interestingly, Gahegan (1999) has experienced that, in general, geographers and geophysicists are able to orient themselves within a visualization more efficiently than computer scientists. This may be because geographers and geophysicists are more familiar with graphic representation in abstract forms.

1.7 Conclusion

While the goal of any visualization is to turn information into knowledge, geo-visualization incorporates geo-spatial data allowing a viewer to interpret and understand a geographic space based on a representation of that reality. This chapter related geo-visualization to visualization in geo-science, showing that visualizations are present through all stages of scientific investigation, providing a scientist with insight during the early stages of preliminary exploration, to effectively communicating results to a wider audience. As well, virtual reality technology was introduced as a useful means of delivering information in an immersive context, providing the viewer with a three-dimensional setting to a three-dimensional geographic space. In addition, the influence of computer gaming graphics technology on advancements in scientific visualization was discussed, with a conflict of interest existing between geographic accuracy and level of engagement, which can negatively affect scientific credibility. The components and mechanisms behind a visualization represent the building blocks of visualization technology, and their recognition is important for thoughtful design and implementation of a visualization.

The bulk of the chapter, focused on the major issues of visualization identified by research to date including their use in the private and public realms of communication, use within virtual environments and the influence of computer gaming technology. As well issues in computer rendering speed, data interoperability were introduced along with issues for users including interaction, perception, understanding, orientation, and navigation. Issues related to realism versus abstraction, and the implications of visual and concealed aspects to a visualization were also discussed. The goal of this thesis is to work

with these issues, and examine them in the context of creating visualizations using different software that apply different terrain visualization techniques.

Chapter 2: Visualization for Education and Content Design for Serendipitous Exploration of Virtual Environments

2.1 Introduction

Many forms of computer visualization exist to assist in collaboration for experts and enhance education for students. In the field of geo-science, experts may benefit from a visualization that conveys geo-spatial information, or current research, through exploring geo-specific locations and virtual representations of those spaces. Students on the other hand may benefit from a visualization to explore environments they would otherwise not be able to physically visit, pulling information about the environment from the virtual representation and exploring it freely.

Geo-visualization, as discussed in Chapter 1, can be used to convey different information for various purposes. For scientific investigation, it was found that visualization is primarily useful during the exploration and presentation stages of research. In this Chapter, the use of visualization for education is discussed with regard to the dissemination of information in a geo-spatial context. Issues in content development and design of virtual spaces are also introduced where the importance of serendipitous exploration becomes evident. The final section explores these concepts in the context of the thesis case study; using Antarctic LIDAR data for exploration of geomorphology of the Dry Valleys of Antarctica and the development of the McMurdo Learning Module.

2.2 Visualization for Geo-science Education

Visualization in the form of pictures, diagrams, animations, and movies can effectively stimulate learning for students by conveying elements of their course material in informative and engaging ways, as Peterson describes in creating animated

cartographic visualizations (Peterson, 1999). In geo-science education, these visualizations are commonly used to convey earth system processes and dynamics, however the use of computer-based learning has greatly transformed the way the visualizations are presented, incorporating aspects of virtual reality to build virtual environments and send students on virtual field trips.

2.2.1 Computer-Based Learning in Geo-science

Computer-based tools for instruction of geo-science has been used and tested in a variety of formats. In 1999, researchers at Penn State University evaluated the use of computer-based tools for an introductory course in geomorphology (Wentz et al., 1999). Instructional material on Compact Disk Read Only Memory (CD-ROM) was compared with Geographic Information Systems (GIS) based learning exercises in order to determine their educational value when used along with traditional teaching methods for the topic, which included the use of textbooks, lectures, and laboratory exercises. In the laboratory, maps were interpreted for landform identification, slides of landforms were presented and discussed, and landform modelling with clay was also performed. One common objective for the study of geomorphology was for students to evaluate the landforms they see and hypothesize about how they are formed and evolve over time (Wentz et al., 1999).

The CD-ROM provided descriptions of topics and showed photographs, diagrams, and animations of physical processes not available in the textbook. The GIS-based exercises involved student production of a digital image using geographical data that described concepts of fluvial geomorphology through the additional use of a photograph and their own written descriptions (Wentz et al., 1999).

When evaluating the ideal text format, most students supported the use of both textbook and CD-ROM. The 'usefulness' of the CD-ROM was rated highly by students and they found the GIS interface useful and easy to use (Wentz et al., 1999).

The overall conclusions of the study were that "computer-based instruction is a valuable addition to the introductory geomorphology course when used in combination with traditional formats" (Wentz et al., 1999: 117). Therefore, the CD-ROM did not actually provide a better method of understanding over traditional methods, rather it was deemed more useful by the students because of the resemblance of content to the format of classroom tests, thus making it a useful study aid. Similarly, the GIS-exercise was considered useful because students recognized the importance of acquiring skills in GIS for their future careers, not because the material was presented in a better way (Wentz et al., 1999). Therefore, these basic computer-based learning tools did not provide students with a significant advantage over traditional learning methods.

In another study it was found that when comparing static maps with dynamic animated maps, users more rapidly perceive spatio-temporal phenomena, although not more correctly (Caquard, 2001). In certain cases, dynamic representations facilitate the speed of comprehension of cartographic information, not necessarily improving their understanding.

The question therefore arises if the different forms of media, textbook vs. CD-ROM vs. GIS, are comparable technologies. Fabrikant (2005) for example identifies that research in cognitive science has been unsuccessful in identifying the advantages of using dynamic versus static displays. This is largely due to a general in-equivalence in content and experimentation between the two forms of media. This may be the case for the above

comparisons, and research to solve this using eye-movement tracking of participants is currently under investigation by Fabrikant's research team (Fabrikant, 2005).

2.2.2 Virtual Learning Environments

A Virtual Learning Environment (VLE) has the potential to provide many advantages over traditional methods of teaching, by encouraging learning through discovery and self-paced exploration (Dean et al., 2000). 'Virtual' is meant as a 'digital alternative representation of reality' (Stainfield et al., 2000), and a VLE can provide students with a venue from which to safely explore environments they would otherwise not physically be able to visit. In addition, students can experience augmented reality in such a venue, where interaction with objects can be done in ways not possible in the real world (Jelfs and Whitelock, 2000).

Establishing a sense of presence within a VLE, or a sense of 'being there', can be highly beneficial (Woods et al., 2005). Landmarks, interactive objects, and auditory stimulation contribute to the richness of the scenery providing an increased sense of presence and help to stir emotional reactions for the viewer. However, perfect representation of the environment is not always feasible, nor beneficial. Research shows that students benefit highly from conceptual and abstract representations of the real world (Whitelock et al., 2000). There is also the opinion that the more intricate the visual design of the VLE, the less ideal the environment is for learning. This notion is further supported by Cartwright (2004: 2-3) who states that when recording geo-spatial information into a map using a pencil and piece of paper, "if a very hard pencil is used a very precise product results" and the map therefore "cannot depict the vagaries and overlaps that are characteristic of real-world geographical information", therefore

affecting interpretation of the map for the user. “If a very soft pencil is used, a more flowing and interpretive map, or a more artistic approach to information recording would result... Areas of vagueness can be illustrated as such, interpretations can be made and impressionistic drawings produced...” (Cartwright, 2004: 2-3).

When teaching abstract concepts, it has been found that in order for a VLE to be most conducive to learning, prior introductory knowledge of the topic is needed before these concepts can be understood by the student (Moshell and Hughes, 2002). Therefore, VLEs in conjunction with external learning materials and class-room guidance seem to benefit student learning more. However, similar to the computer-based learning tools mentioned in the previous section, it has been found that when it comes to the effectiveness of VLEs in education, there is no marked difference in the learning achieved by students compared to those who learned through traditional methods (Moshell and Hughes, 2002). Reasons may include: 1, the apparent novelty of the VLEs, which tends to divert attention away from the learning objective; 2, perhaps current evaluation criteria, which are focused on the amount of factual knowledge gained, are not suitable for evaluating VLEs where the acquirement of procedural knowledge and cognitive strategies are more common; and 3, the technical knowledge required by the instructors to add and design content is often insufficient, making it a complex and daunting and exercise (Moshell and Hughes, 2002). A fourth may be the invalid comparison between these two forms of media, as was discussed previously.

It is also likely that the tools available to make VLEs may not adequately meet the standards that students expect from these worlds given their increasing exposure to computer games, which can have production costs of over \$1 million per game (Moshell

and Hughes, 2002). The computer gaming industry was worth \$22.3 billion in 2003 and is expected to double to \$55.6 billion by 2008 (Hutton and Ray, 2004), making it difficult for educators to compete with such standards of interactivity, engagement, and entertainment. However, the growth of computer game technology will likely mean that high performance computer graphics will become more readily available for VLE designers to capitalize on.

While the effectiveness of VLEs for learning is debatable, concern arises with respect to VLEs taking the place of hands-on lab exercises. It is thought that students are prone to reduced information retention as a result of VLEs since less physical effort on their part is required (Moshell and Hughes, 2002).

2.2.3 Virtual Field Trips

While virtual reality has a long way to go before total immersion is achieved, education in Virtual Field Trip (VFT) settings has been in development and used in the classroom for quite some time. The combination of information and communication technology as a supportive tool for fieldwork gave rise to the concept of the VFT (Stainfield et al., 2000).

In geo-science, fieldwork is a necessary component of research and proper education in the discipline. Fieldwork instruction has changed considerably since the 1950s and 1960s, when students passively observed their instructors (Stainfield et al., 2000). In the 1970s and 1980s, fieldwork was problem-oriented and focused more on the refinement of skills. By the 1990s, increase in enrolment in the geo-sciences, especially in the U.K., had logistical implications for the effective instruction of fieldwork, and

contributed to damaging the environment through field excursions involving growing numbers of participants (Stainfield et al., 2000).

Through the VFTs and the Internet, students are provided with a sense of ‘visiting’ distant or dangerous places all over the world. The many advantages to using VFTs as opposed to handbooks of a similar nature according to Stainfield et al. (2000) include: the ease of updating material, the ability to add links to other relevant sources, no limit to the size or breadth of material, the cost of production is lower, losing the material is not possible, they are accessible anywhere with an Internet connection, they can act as a repository for student-made material, and skills and knowledge can be tested interactively (Stainfield et al., 2000). Handbooks on the other hand are more portable, but with CDs and portable computers becoming more common, this disadvantage to VFTs will likely dissipate. Other advantages include the ability to simultaneously present data at multiple scales and different points of view, and allow students to see a greater variety of landforms than would be possible on a field excursion (Qiu and Hubble, 2002). These and several others are outlined in Table 2.1.

The main disadvantages of VFTs are their inability to effectively demonstrate and develop fieldwork skills, or provide the true sense of a field trip experience. A solution to this is to see VFTs as a tool of enhancement to be used in addition to field experience, rather than a replacement of it (Qiu and Hubble, 2002). It should be understood that in order for students to effectively learn from them, they must have some form of actual field experience. In the opinion of Qui and Hubble (2002: 78): “VFTs also lack the serendipitous nature of discovery, which is one of the most attractive things on real field trips”. The issue of serendipitous learning is discussed further in section 2.3.3.

There are many existing VFTs currently available on the Internet. In fact in October 2002, a Google keyword search, for 'virtual field trips' yielded 307,000 websites (Qiu and Hubble, 2002). One large repository of VFTs is provided by the Virtual Geography Department Project run through the Department of Geography and the University of Colorado – Boulder (Howard, 2004). At the website, <http://www.colorado.edu/geography/virtdept/contents.html>, instructors in geography contribute to an online community of curriculum materials and laboratory exercises. Each

Table 2.1: The advantages and disadvantages of VFTs from the user's perspective. (Qui and Hubble, 2002:77).

Features of VFTs	Advantages of VFTs	Disadvantages of VFTs
Use digital and computer visualization techniques	<ul style="list-style-type: none"> • Integrate diverse types of data in instantly available ways • Present images from a variety of viewpoints and at many different scales • Display non-visual data (geochemistry, etc.) • Helpful for presenting trips to inaccessible areas • Provide an alternative of fieldwork, when time, expenses, and/or logistics are real issues • Enable presentation of extensive field trips and great variety of landform diversity • Enhance and expand students' experience 	<ul style="list-style-type: none"> • Do not convey the true three-dimensional nature of objects • Do not convey the non-visual and aural feelings of touch, smell, etc. • Less beneficial than really being in the field • Lack the serendipitous nature of discovery
Based on the personal computer and Internet	<ul style="list-style-type: none"> • Enable flexibility of access (time and place) • Provides a repeatable experience which can be used to reinforce concepts in class • Provides an easily experienced preview or review of real field trips 	<ul style="list-style-type: none"> • Having limited interaction with a computer • Not interacting with people in a flexible manner
Multiple styles of access e.g. CD-ROM and websites	<ul style="list-style-type: none"> • CD-ROMs are convenient to acquire and use • Information rich 	<ul style="list-style-type: none"> • CD-ROMs can only provide a finite limited amount of information • Visiting a website can be difficult and depends on many factors, such as availability of computers, load on the network, number of connections, reliability of service provision, etc.
Wide variety available on the Internet	<ul style="list-style-type: none"> • Hold abundant materials and information • Offer rich resources of learning and teaching 	<ul style="list-style-type: none"> • Easy for students to get lost among lots of websites • Many websites are ephemeral rather than permanent
Variable quality	<ul style="list-style-type: none"> • Available for users of different levels and demands 	<ul style="list-style-type: none"> • Often difficult to find a suitable one for teaching and learning • The abundant websites are not quality controlled
Designed to be interactive like computer games	<ul style="list-style-type: none"> • Interesting and attractive to students and an alternative experience for users 	<ul style="list-style-type: none"> • It is easy for students to wallow, or obsess over particular sites, which raises the problem of time management

area of research within geography is headed by a Working Group, and authors contribute to the discipline by submitting modules that are reviewed by the Working Group and classroom tested before being made publicly available through the Virtual Geography Department. VFTs within this website are divided by region and subject. The modules under geomorphology include a VFT of Glacier National Park, Montana (Lemke, 1999), and a VFT of Indian Peaks, Colorado Front Range (Ritter, 1997). Both modules and others in the project are written in an inviting and colloquial manner in an attempt to motivate students through the material.

The VFT of Glacier National Park provides the student with overall learning objectives and begins with an interactive map of the park showing the hiking trails the students will follow on their virtual journey. When visiting sites of interest on the hike, students are presented with high quality digital images and descriptions of the landforms they 'pass by'. Activities, laboratory instructions, and questions are given at each stop. Some ask the student to find a cabin they missed along the way, others provide a topographic map and graph paper for the student to print-out and using these they construct cross-sectional profiles of the area; a common laboratory exercise in geomorphology. Overall the VFT is very well organized, easy to follow, and all the images and maps are of high quality and clearly labelled. The activities and exercises are appropriate to undergraduate level geomorphology and cover a lot of interesting material.

The VFT of Indian Peaks works in a very similar manner. Stops on a virtual field excursion are given and students are presented with images and journal entries for their trip. At the end of the trip, students are to construct a graph of elevation versus temperature, and illustrate the vegetation found at each point in the graph to determine

the relationship between all three. While there are not as many activities in this VFT as in the Glacier National Park module, it is well organized and well within the capabilities of students in geomorphology.

While both of these sites are engaging and highly informative, the progression through the 'trip' is stop-by-stop through links in web pages with images and sometimes animations of places and processes. It may then be more appropriate to call them Virtual Tours as suggested by Qiu and Hubble (2002).

While VFTs provide an interesting and engaging venue from which to learn about physical geography practices in the field, they cannot and should not replicate the full experience of work in the field (Stainfield et al., 2000). Surveys have shown that students do not wish VFTs to replace the field component of their instruction, but do enjoy interacting with them (Qiu and Hubble, 2002; Spicer and Stratford, 2001).

In Chapter 1, it was found that scientists use visualizations predominantly in the exploration and presentation stages of scientific investigation. In the case of VFTs, it has been found that as they are useful in instruction of the geo-sciences, they are particularly good for pre-study or preparation for a field excursion, and review of material after the excursion (Qiu and Hubble, 2002). Since they provide students with a method of preparing for an actual field trip or lab exercise, VFTs can improve student productivity in the field and increase time efficiency (Stainfield et al., 2000).

2.3 Content Design for Virtual Environments to Facilitate Education through Serendipitous Exploration

“..content for VEs is not just scene graphs, nodes, objects, or audio assets. Content is about stimulating perceptions, which gives rise to meanings in the mind of the user and stimulates action, which in turn leads to further meaningful insights” (Isdale et al., 2002: 524).

2.3.1 Content Development and Virtual Environment Design

Thoughtful content development and design of Virtual Environments (VEs) help to ensure satisfaction and success for the VE. In this regard, much can be learned from the design of computer games and theme parks (Isdale et al., 2002).

Computer game designers produce the largest and most intricate VEs available today. Some key aspects to computer game design are: establishing a purpose with emotional objectives, ensuring spatial consistency, providing an intuitive interface, allowing for several outcomes or solutions to problems, and providing the gamer with an experience that is rewarding yet challenging (Isdale et al., 2002).

Theme parks are designed for visitors to move through a physical space at liberty, and theme park designers commonly build a physical space that follows a story or theme. Some of the key elements of theme park design that can be applied to VE design include: building the space that conveys a story, establishing rules of the world that cannot be broken, providing navigational aids like arrows and pathways as guides, adding landmarks or objects as anchors that accompany the story, and being sure to not clutter the space with too much detail or provide too many choices to the visitors (Isdale et al., 2002).

On that last note, Isdale et al. (2002: 525) remark contradictorily that “people are used to the real world being complex and cluttered, so it helps if the virtual world is as well”. There is perhaps a fine line between what appears as clutter, and what sparks visual interest and encourages the viewer/visitor to explore.

When designing content to be used for education, and specifically in the realm of VFTs, there are three important questions to consider when designing high quality VFTs (Qiu and Hubble, 2002): 1, what are the learning objectives for the student? This question depends on course material and requirements for the degree; 2, how best can the material be presented for student learning? This depends on the learning strategies of the students, and the skills the instructor would like the students to develop; 3, how can a VFT be best designed? This depends largely on the technical availability and ability of the instructor. In general: “A good Virtual field trip should have high quality images and will offer as many chances for users to investigate in detail the location or phenomenon they are interested in” (Qiu and Hubble, 2002: 78).

2.3.2 The use of Landmarks in Virtual Environments

Fundamental to the success and design of a VE is the method used to help users navigate the space. The use of landmarks within VEs can greatly improve user navigation by adding distinctive features to the landscape that serve as reference points (Vinson, 1999). When creating a VE that represents a city, landmarks such as statues and buildings can be helpful in user orientation, navigation, and for providing spatial awareness to a user. For natural environments on the other hand, where human artefacts in the terrain are uncommon, the use of landmarks can be challenging, but can be solved by incorporating artificial landmarks (Vinson, 1999). Vinson (1999) has determined 13 guidelines for the use of landmarks in VE. They are described as follows:

1. **It is important that several landmarks be present in a VE.**
2. **The landmarks chosen for the VE should include the following elements:** **Paths** in the form of streets and canals for example provide a channel of movement for the user. **Edges** including fences and rivers can

provide limits or borders to the areas. **Landmarks** such as statues provide other reference points.

3. **Landmarks should be distinctive.**
4. **Landmarks should be concrete rather than abstract in nature.** The use of three-dimensional familiar objects improves user navigability compared to using abstract objects. When navigating a natural environment, the use of human-made structures such as roads, sheds, and fences are preferable. Contour features of the landscape including hills, slopes, and cliffs are often used as well as water features like lakes, streams, and rivers when human-made structures are unavailable. It has been found that in practice, little reliance is made on the use of vegetation as landmarks since it tends to change over-time.
5. **Each Landmark should be visible at all scales.** No matter what the extent of the VE, landmarks should be visible and reflect the scale the users are navigating in whether they zoom in or out of an area.
6. **Landmarks should be easily distinguishable from other objects and landmarks nearby.**
7. **The sides should be unique to each landmark.**
8. **The distinctiveness of a landmark can improve if other objects are placed nearby.**
9. **Landmarks should be made distinguishable from data objects in the VE.**
10. **Landmarks should be placed on major paths or junctions.** Also, using paths in a VE minimizes the number of landmarks needed, which enhances interactivity and supports user navigation.
11. **Paths and edges are two elements of point of view mentioned in #2, and should be placed in a grid structure in the VE.**
12. **Landmarks should align with the above grid of paths and edges.**
13. **The landmarks should align with each other.** The above 3 guidelines are meant to minimize distortion. When a user creates a 'cognitive map' or mental map of the area, the error in direction and distance is reduced when grids alignment of paths and landmarks is used, preventing the user from misjudging their position.

While landmarks may be useful for orientation and navigation for the user, their use as tools of spatial awareness and indicating points of interest are of particular importance. While the use of paths may be beneficial to help guide the user from place to place, the concept of exploring the environment serendipitously without the use of those elements needs further consideration.

2.3.3 Serendipitous Exploration

The issue of serendipity has been brought up several times in the previous sections. Referring to the opinion of Qui and Hubble (2002: 78), VFTs “lack the serendipitous nature of discovery, which is one of the most attractive things on real field trips”. Interestingly, Dean et al. (2000) state that VLEs encourage learning through discovery and self-paced exploration. Qui and Hubble (2002) view VEs as a hindrance to the exploratory nature of field work, yet Dean et al. (2000), contend that VEs can in fact encourage such elements of discovery. On this note, Cartwright (2004) argues that serendipitous exploration of geo-spatial virtual environments is possible, and should be strived for.

Whether designing a VLE, VFT or any other virtual environment, the degree of freedom for the user needs to be decided. VEs can be designed to either keep the user’s movements tightly controlled, making sure they visit all important material, or to allow for free movement, where information is then encountered by chance (Cartwright, 2004). Furthermore, a VE can be designed to be passive, where the user is guided through the experience, or reactive, where the environment responds to interactions from the user. Information discovery is an asset to a VE, and as stated above by Qui and Hubble (2002), is important to the field trip experience and scientific discovery (Cartwright, 2004; Fine

and Deegun, 1996). The difficulty is providing a high degree of freedom for the user to discover information, yet ensuring that they have not missed any important information along the way. Cartwright's (2004) method of 'Engineered Serendipity' attempts to accomplish this.

Figure 2.1 compares two cases of serendipitous exploration through a virtual environment. In the completely serendipitous case (shown left), the user travels through information spaces, but misses Essential Information #3 before they Exit. In the case of Engineered Serendipity (shown right), the user is free to travel to any information node they wish, in any order. Before they exit, however, a checklist is provided which lists the information nodes they missed along the way, and encourages the user to visit them.

Figure 2.1: Comparing the information travel paths of users through a virtual environment with serendipity (left), where the user misses key information, and Engineered Serendipity (right), where the user is alerted before leaving if they have missed any information (Cartwright, 2004).

In the words of Isdale et al. (2002: 524), the overall goal for a VE is to provide “users with carefully structured opportunities to allow them to explore, strategize, formulate and solve problems, and plan for and attain goals”. VEs should be designed to accomplish all of the above goals.

2.3.4 The Use of Metaphor in Virtual Environments

Cartwright (2005) suggests the use of metaphors in VE to convey information. For example, in his GeoExploratorium, he designed a VE where users can navigate, browse, and explore geographical information. The metaphor set for his GeoExploratorium concept includes that of a: Storyteller, Navigator, Guide, Sage

Figure 2.2: The MapShop with curator, and user avatar in view (Cartwright, 2005).

(expert), Data Store, Facts Book, Gameplayer, Theatre, and Toolbox. Based on this set of metaphors, a VE developer can decide which is best suited to conveying their information. If, for example, the purpose is to convey a geographical story, the Storyteller metaphor might be most useful. In Cartwright's (2005) MapShop, an interactive VE

built to help users browse for maps and geo-spatial information, the Game metaphor is implemented where users navigate a three-dimensional model of The MapShop in much the same way that users explore and interact with a game space (Cartwright, 2005). A screenshot of the Map shop with a central column which lists information, a human form acting as the MapShop curator, and drawers of maps are seen in Figure 2.2. The character in the suit outside the shop is the avatar representing the motions of the user as they walk through and explore the MapShop.

The following section introduces the thesis case study, which involves the creation of a visualization for an interactive learning environment with implementation based on the above design and content guidelines.

2.4 Thesis Case Study - Visualizing the McMurdo Dry Valleys of Antarctica

This thesis explores the case of developing an interactive VE of the McMurdo Dry Valleys of Antarctica for education in geomorphology. The guidelines and lessons learned that were described in the sections above are taken into consideration in the design of the McMurdo Learning Module (MLM) which is discussed in the sections that follow.

2.4.1 The McMurdo Learning Module: Geomorphology of the McMurdo Dry Valleys of Antarctica.

As a brief introduction, the geomorphology of the McMurdo Dry Valleys is such that most of the landscape is carved through glacial activity, therefore the MLM more specifically discusses glacial-morphology, and identifies the McMurdo Dry Valleys as a unique and interesting place to study it. This area is often referred as the oasis of Antarctica since the environment is considered a polar desert where the climate is very cold and very dry (May, 1989). The temperature of the Dry Valleys is often below -20 degrees Celsius, and as little as 10mm of precipitation falls over the span of one year. Glaciers are able to form despite this, although because there is so little snow accumulation, the glaciers move very slowly. Several valleys and glaciers characterize the Dry Valleys landscape, contributing to the high-relief nature of this environment, making it an excellent place for students to learn about geomorphology. Refer to Appendix I for a map of the McMurdo Dry Valleys of Antarctica.

The purpose of the McMurdo Learning Module (MLM) is to convey elements of geomorphology considering the issues and guidelines raised in the previous sections of this chapter. The goal is to combine the successful elements of computer-based learning

tools, virtual learning environments (VLEs), and virtual field trips (VFTs) discussed in Section 2, and create an interactive and engaging learning module for students to learn about the unique environment of the McMurdo Dry Valleys.

As outlined by Qiu and Hubble (2002), the effective design of a learning environment is accomplished through first identifying the learning objectives, the method for presenting the material, and the best method for design. The learning objectives for a student using the MLM are to: 1, become familiar with geomorphological terminology; 2, identify geomorphological features including different types of glaciers; and 3, to appreciate the unique environment of the McMurdo Dry Valleys. The best method for presenting the material is probably in a three-dimensional format since education in geomorphology is typically done using stereoscopes, air photographs, and topographic maps. The MLM can be best designed therefore by taking advantage of computer-visualization technology to create an interactive three-dimensional representation of the McMurdo Dry Valleys for students to explore as an addition to the traditional study methods mentioned above.

In section 2.1, focus was placed on the use of VLEs in education providing students with the opportunity to ‘virtually’ visit a distant location without danger or logistical constraints. The MLM is such an example, where students can explore the unique landscape of the McMurdo Dry Valleys of Antarctica, a place they are unlikely to visit in reality. In addition, VLEs have the advantage of providing augmented reality which is implemented in the MLM in the form of interactive objects, such as signs and billboards, that do not exist in the physical environment. They also behave differently from what is possible in the real world. For example, in the MLM, a user can ‘walk’ up to

a sign and click on the landmarks for further information with animations and panoramic views. Signs in reality can convey information, but not with the same level interactivity.

The educational goals for the MLM are to provide students at the undergraduate level with a basic understanding of the topic of geomorphology for a glaciated landscape. It will not follow a particular curriculum, but will allow students to explore the terrain and read accompanying text to learn about glacial features and how the landscape is carved through glacial processes as well as by water and wind. Through studying the specific landscape of the McMurdo Dry Valleys of Antarctica, students will learn about that environment, but the information present is relevant to other areas of the earth where similar geomorphologic characteristics exist.

The MLM is therefore designed to give a sense of presence to the user through the use of high quality LIDAR remote sensing data for the area, interactive objects and landmarks that act as areas of interest for the student to access information through links, and by providing sound effects that convey the nature of the environment and human activity. Information on the LIDAR data is provided in Chapter 4, and is illustrated in Appendix I.

A game-like quality is intended for the MLM to increase engagement on the part of the student. The MLM therefore utilizes the game metaphor like that of Cartwright's (2005) MapShop. Discovery through serendipitous exploration is the intent for the MLM as well, allowing the student to benefit from this form of learning which is similar to the experience of an actual field trip. The effort therefore is not to create a tour of the McMurdo Dry Valleys but an exploratory VE. Navigation and orientation are still important however, making the effective use of landmarks crucial to the success.

Students need to be able to determine where the interesting areas are, and how to derive information about features. In terms of providing appropriate landmarks and placement of landmarks as discussed in section 2.3.1, the guidelines deemed most appropriate for the MLM were that they should be distinctive, concrete, and visible at appropriate scales, align within a grid structure, align with each other, and follow paths. With these guidelines the landmarks chosen and their position are shown in Figure 5.12 along with a greater detailed discussion in Chapter 5. Because of the serendipitous nature to be achieved in this design, paths leading from one landmark to another are seen as the valley walls themselves as the student can ‘walk’ within the confines of the valleys from landmark to landmark if they choose.

The result is an experience where the student learns about the environment directly from a representation of the environment in an exploratory and serendipitous manner.

2.5 Conclusion

This chapter introduced the educational experiences and successes for geo-visualization through the use of computer-based learning tools, virtual learning environments, and virtual field trips. Many issues in content design and development of a learning tool were discussed relating to the use of landmarks as navigational aids, the use of metaphor for describing the method of content delivery, and the theory of serendipity and its place in knowledge discovery both in the field and within a virtual setting. Many of the design principles and guidelines conveyed in the above sections are acknowledged in the design of the thesis case study’s McMurdo Learning Module which is meant to

function as a virtual learning environment for students to learn about the geomorphology of the McMurdo Dry Valleys.

In Chapter 1, the issues of perception and visualization were discussed and the differences in perspective, experience, and understanding of the material have a lot to do with the way that it is presented, where the technology used is an important element for this purpose. The following chapter therefore looks at the technology behind the creation of detailed terrain visualizations for the intended purpose of creating an interactive learning module.

Chapter 3: LOD, Fractal, and Voxel-Based Terrain Rendering Techniques

3.1 Introduction

In Chapter 1, a broad overview of the theory, and technology of geo-visualization, and the issues and challenges faced within the discipline for conveying geo-scientific information were discussed. Of particular interest are the apparent trade-off between interactive quality and visual quality for virtual environments (VEs), and issues of user perception and understanding. While Chapter 2 focused on design and content issues for the thesis case study and VEs in general, in this Chapter focus is placed on terrain rendering techniques for the purpose of providing a knowledge base for evaluating the techniques used in later chapters towards the creation of the McMurdo Learning Module (MLM). The rendering techniques studied include Level-Of-Detail (LOD), Fractal, and Voxel-Based terrain rendering. First, a background to the technology of terrain rendering is provided below.

3.1.1 Background to Terrain Rendering

A crude definition of terrain is: “the graph of a continuous function that assigns to every point on the plane an elevation” (de Berg and Dobrindt, 1998: 1). This is true considering that a virtual terrain is generally made up of a grid of polygons or triangles for which elevation values are given throughout. However in it is important to recognize that: 1, objects remain static because it is only the viewer that moves; 2, the land surface is a single connected object; and 3, terrain is generally a two-dimensional surface where only the elevation enters into the third-dimension (Coddington, 1996). Therefore, terrain

visualizations are said to be 2.5 dimensional representations, as opposed to fully three-dimensional.

In order to create a virtual terrain, a large amount of data is needed. The computer must handle large volumes of data in order to produce a virtual terrain and maintain interactive quality. Rendering speed has to do with the amount of time needed to generate an image, and the ability of a computer processor to deliver graphics for visualizing the terrain (Gahegan, 1999). High-performance computers with advanced graphics hardware are therefore desirable when performing terrain rendering and other forms of visualization (Coddington, 1996). Using a scaling algorithm, terrain visualization can be accomplished on computers with less computer power where a small reduction in graphics quality can improve the interactive quality. For a truly realistic visualization, one that provides animation in real time, performance capabilities and quality are severely reduced with increases in resolution (Gahegan, 1999).

Virtual representations of terrain usually involve creating a mesh of polygons that vary in size for different levels of resolution. Larger polygons can be used to represent areas of terrain that are far from the viewer or to represent flat regions like lakes or plains. Meshes of smaller polygons are best suited for representing regions that are closer to the viewer or representing features such as mountains and cliffs (Coddington, 1996). The size of the mesh greatly affects the quality of graphic for terrain visualization. Figure 3.1 contrasts the effects that different size polygons can have on the resolution of a surface. Figure 3.1a made with larger polygons creates a coarse mesh, which has poorer visual quality and resolution compared to Figure b), which uses much smaller polygons to represent the same area (Coddington, 1996).

There are two basic data structures for the generation of virtual terrain. These are Digital Elevation Models (DEMs) and Triangulated Irregular Networks (TINs), both of which are described briefly below.

Digital Elevation Models

A DEM is a data structure, which consists of a matrix of elevation values (Kumler, 1994). Figure 3.2 illustrates a DEM data structure, where the terrain is evenly covered with a regular grid of squares all representing the same value for area, with differing values of elevation. The smaller the squares are, the higher the resolution, and the more accurate the terrain representation.

Triangulated Irregular Networks

A TIN is a data model based on irregularly spaced points that are linked within a system of non-overlapping triangles (Kumler, 1994). A TIN can be considered a data model as opposed to a data structure

because the data within a TIN describe the whole surface of the terrain, as opposed to having elevation and topology data for selected points in the terrain as is the case with a DEM. However, a

TIN does have a structure consisting of points, triangles, and topological data (Kumler, 1994). Figure 3.3 presents a TIN as a data structure, a), and a rendered data model, b).

As discussed in previous chapters, most improvement in graphics technology has been a result of greater demands for higher graphics and interactive quality in computer game technology. However, since the focus of a computer game is to provide the player with realistic interactivity, computer game developers use a process called polygon reduction, which reduces the number of polygons in a terrain so that the level of detail is significantly reduced (Rhyne, 2002). Therefore, the trade-off between interactive-ability and visual quality is recognized, and has to do with data capacity, data storage, and the computer processing speed, which is compromised when the computer has to deal with large amounts of data.

Good quality terrain visualization over the Internet is especially difficult to achieve because the amount of information that can be transferred over the Internet is dependant on the bandwidth. Serious performance problems arise when dealing with large amounts of data over the Internet (Ottoson, 2003), which affect the interactive quality of the visualization.

The following sections describe three terrain rendering techniques: Level-Of-Detail, Fractal, and Voxel-based terrain rendering.

3.2 Level-Of-Detail (LOD) Terrain Rendering

“Photorealistic rendering, modelling of minute details, and exact physical simulation may be unnecessary for many purposes” (Isdale et al, 2002: 320). The notion of “selective fidelity” was therefore introduced where only specific portions of the VE

would be rendered at high quality. The concept of LOD terrain rendering is based on the idea that different areas of terrain can be represented at different levels of detail within the same image. Areas of the environment that are close to the viewer for example, will be at a higher resolution or level of detail compared to areas that are farther away (de Berg and Dobrindt, 1998). Millions of triangles as part of a TIN, are needed for this method of terrain rendering.

While the concept is intriguing, an important factor to consider when using this method is the switching between different LODs through zooming or while the user is moving. “Jumps” in the image should not occur while interacting with and exploring the VE (de Berg and Dobrindt, 1998). This is also known as “vertex popping” where detail suddenly emerges (Röttger et al., 1998). It is difficult to avoid in a terrain where the LOD is continuously changing. When implementing the LOD technique with objects in the terrain such as buildings or vehicles, the entire object can be replaced instantly with different LODs (Terrex, 2004) which is noticeable to the viewer and unwanted when the surface needs to appear continuous.

The LOD technique is usually accomplished by first dividing the terrain into a grid composed of tiles (Terrex, 2004). Within each tile, multiple LODs can be represented depending on the tiling scheme. There are two traditional approaches for LOD terrain rendering which are discussed below. The first is the hierarchical tree-structure method; the second is the Delaunay triangulation method.

Hierarchical Tree-Structure Method

Using the hierarchical tree-structure method, a coarse representation of the terrain is first generated using only a small subset of data points (de Berg and Dobrindt, 1998).

Through the connection of these data points, the space is divided into triangles. These triangles are then refined by incorporating new data points within them, and creating new triangles using these new points. Each triangle is thus replaced with several smaller triangles. This procedure continues until all of the data points for the image are included. This hierarchical form of terrain rendering can be represented in the form of a tree

structure, which is illustrated in Figure 3.4.

Figure a) illustrates the terrain divided into a series of triangles. The bold lines represent the coarse triangles generated in the first step of the process using only a small subset of data points. In the corresponding tree b),

the first set of nodes represents these coarse triangles. When new data points are added and new triangles are drawn, the nodes are divided by the number of new triangles generated. The grey triangle of Figure 3.4 a) corresponds to the grey node in the tree b). The following six nodes represent the six triangles created during the second phase of the procedure, contained entirely by the parent triangle. The figure thus represents a two-level hierarchical tree-structure.

One problem with this method of triangulation is that the triangles generated at the higher levels appear 'skinny'. This is even noticeable within the second level of the hierarchical divide seen in Figure 3.4. The effects of this can be seen in the image where features appear crude and raw, improperly representing features in the terrain (de Berg and Dobrindt, 1998). The Delauney triangulation method reduces these problems.

Delaunay Triangulation Method

The Delaunay triangulation method is in itself a hierarchical method, however the minimum angle of the triangles are maximized in this method in order to alleviate the problems previously identified for the hierarchical tree-structure described above. What makes the Delaunay triangulation method unique is that an arc can be drawn from

triangles of one level in the hierarchy to triangles in the next level, even if there is an intersection between the triangles. This means that triangles are drawn independently for each

data set. This is illustrated in Figure 3.5. Both a) and b) show the same area of representation. All of the data points are shown where the first level of data points appear in bold, and those of the second level appear smaller. Figure 3.5a has connected the first sub-sample of data points into large triangles. The red triangle in Figure 3.5b is added to illustrate how the triangles of the higher level intersect the triangles of the first level.

Much like Figure 3.4, the nodes in the tree correspond to the triangles in the image. Figure 3.5 c), therefore, shows how intersection can occur between the children and parent triangles.

One problem encountered with the Delaunay triangulation method is that the different levels of triangles cannot be incorporated into one representation or visualization (de Berg and Dobrindt, 1998). As can be seen in Figure 3.5, both a) and b) are needed to represent the two levels of detail. This was not necessary in the previous example as shown in Figure 3.4. However, because the triangles do not appear “skinny”

this method proves to be more useful since triangulation involves connecting the points based on proximity. As well, and more importantly, different levels of hierarchy can be extracted from different parts of the terrain and connected to form a continuous terrain representation with different sections being represented by different hierarchical levels of detail (de Berg and Dobrindt, 1998).

The process of combining these different LODs begins with the coarsest level. For each subsequent level, the appropriate triangles are extracted and combined to fit the model. If the coarser triangles are thought of as the parent triangles, and the triangles at the higher level of detail as the child triangles, then the process can be thought of as the parent triangles being replaced in certain areas with their children triangles. When extraction occurs, the triangles become disjointed. Adding an intermediate level can help alleviate this by creating a union between the disjointed parent and child triangles (de Berg and Dobrindt, 1998).

The ExtractCombination algorithm below describes the steps involved in

ALGORITHM *ExtractCombination*().

1. Initialize an array $A[1:k]$ whose elements are lists of triangles, where k is the number of levels in the hierarchy, and initialize an empty list L .
2. Add all triangles of the coarsest triangulation
3. **for** $i := k$ **downto** 1
4. **do while** $A[i]$ is not empty
5. **do** Remove a triangle t from $A[i]$, together with all other triangles in $A[i]$ and L that are in the same parent polygon as t . Let G_t be the resulting collection of triangles.
6. **if** *Accept* (G_t)
7. **then** append all triangles of G_t to L .
8. **else** add each triangle t_0 of the child polygon to which t is linked through an intermediate node to the list $A[j_0]$, where j_0 is the finest level in the hierarchy where t_0 is present.
9. **return** L

Modified from (de Berg and Dobrindt, 1998)

combining different LODs into one terrain representation using the Delaunay triangulation method where “ i ” is the level of hierarchy for the triangle, and “ k ” is the total number of levels in the hierarchy.

The algorithm begins at the coarsest level, and runs a series

of tests on groups of triangles to see if they should be accepted as they are or refined (de Berg and Dobrindt, 1998).

Figure 3.6 shows the results of the algorithm in generating terrain. Figure 3.6 a) is a representation at the highest level of detail for the whole surface of terrain. The same terrain is shown in b) using a quarter of the number points of a). The combination of the two based on the ExtractCombination algorithm is given in c). It is composed of the same number of triangles as a), but has a higher LOD closer to the viewer (de Berg and Dobrindt, 1998).

There are limitations associated with the LOD method of terrain rendering. Bordering areas of different resolution risk appearing inconsistent and, as previously mentioned, vertex-popping can also occur (Röttger et al., 1998). A technique called geomorphing is used to minimize the popping effect where adjacent LODs are amalgamated, but this is often not effective because it assumes a constant distance to the viewpoint (Röttger et al., 1998). This is discussed further in Chapter 4.

With reference to the first problem, the tiles, which assemble the terrain, share common borders with adjacent tiles. Shared edges must match in order for the surface to appear continuous (Terrex, 2004). Leaving the borders unchanged at a fixed LOD, the tiles can be joined along common borders. If these borders never change, then they will always line up with the neighbouring tiles producing a continuous grid. This comes at a price to the visual quality of the terrain, however, because the LOD of the borders will not correspond to coarser or finer LODs within the tiles. Figure 3.7 illustrates two examples with tiles of constant LOD within its borders. Figure a) shows the tiles at a very fine level of detail. The boundaries have triangles that are too large to accommodate the finer detail of the tiles. The rendered visualization will show troughs between these tiles as a result. Figure 3.7 b) shows tiles of coarse detail where too many triangles define the boundary. If the terrain has a polygon budget, a certain number of triangles and points

allowed within a representation, then the polygon budget will be spent on the borders of the tiles rather than within the larger areas, resulting in reduced LOD over large portions of the terrain. Therefore, in the final rendering, attention will be drawn to the boundaries as a result of the irregular distribution of points

(Terrex, 2004).

The concept of SmartMesh developed by Terrex, Terrain Experts Inc. attempts to alleviate the border problem, and is described in the following section.

SmartMesh and the Terrex, Terrain Experts Inc. Terrain Rendering Product

Terrex introduced a unique form of triangulation. It is able to automatically smoothen out cracks or seams that can appear between the tile boundaries while changing LOD within a virtual scene (Terrex, 2004). The concept is based on an adaptive boundary

that can keep up with the changing LOD within the tiles as the viewer moves while still upholding the polygon budget. Figure 3.8 illustrates the concept of SmartMesh. Figure 3.8 a) shows the eyepoint of the viewer off centre to the grid with a constant LOD throughout. The boundary is cut out to illustrate where it appears. In b) the changing LOD is shown as the viewer moves across the surface. The detail is finer, with smaller triangles and a larger number of points closer to the viewer, and becomes increasingly coarse with larger polygons

characterizing the surface farther away. As well, it can be seen that the left boundary moulds seamlessly into the LOD of the tile, while the right boundary has a slightly coarser LOD. As the viewer moves directly over the centre as seen in c), a high LOD is maintained over the entire surface, and the process reverses as the viewpoint moves across to the right side of the terrain as seen in d). SmartMesh is able to stay within the polygon budget even at the highest LOD. This process thus attempts to solve the boundary problem and greatly enhances the visual quality of a virtual terrain.

The Terrex Software

From the Terrex website, a demonstration version of terrain representing Los Angeles U.S.A. was downloaded for an exploration of the SmartMesh technique. Los Angeles satellite imagery was derived from Satellite acquired from ortho-rectified true-colour TIFF files, which are ideal for applications requiring a high level of positional accuracy (INTEC, 2004). In the case of this demo', the LOD transitions are said to be shorter so that they may be seen through the interaction, making the SmartMesh technique more visible (Terrex, 2004).

Once having loaded the Los Angeles demo', a window opened with a simple interface and several icons for exploring the scene. The viewer is presented with an oblique view of the terrain.

The Figures 3.9 a), b), c), and d) represent screenshots of the Los Angeles demo' data within the Terrex viewer and demonstrate the SmartMesh Technique within the context of this data. All four screenshots represent the same area of terrain whereby, with movement of the viewer, the LOD is changed.

While flying over the terrain, the triangular mesh responds automatically to the viewer, becoming coarse and fine depending on the viewer's movements. Figure 3.9 a) presents the view of an area where the viewer is looking directly down to the terrain. From this perspective and at this scale, the triangle mesh was toggled on as is seen clearly in the figure. A bold grid was added to highlight the noticeable grid tiles present on the rendered terrain. Six whole grid tiles are visible in this view. Two distinctive triangle configurations are outlined in yellow and green, and a red circle outlines an area of interest. What is interesting to note, is that the triangles are fairly large, generating a

coarse LOD in the terrain. The borders are visible, yet the triangles match well over the grid borders without compromising the LOD of the bordering triangles. Figure 3.9 b) is the same view with the orth-rectified image draped over. Shadows and fog were removed from the image to enhance clarity. The grid outline from Figure 3.9 a) are shown again in the same place and it can be seen that the grid lines do not extend into the terrain at all. It is also interesting to notice that larger triangles are used to represent the area of urbanization and the river valley, while smaller triangles characterize the rough terrain in the mountainous region.

The concept of SmartMesh really emerges as a technique in the figures that follow. In these figures, the viewer has moved slightly to the right and zoomed in as well. The results are seen in the triangle mesh of Figure 3.9 c). In this figure the same distinctive triangle configurations are outlined again in yellow and green. These correspond exactly to those seen in Figure 3.9 a). Upon the viewer's movement, the grid representing those triangles had not changed, but had become larger. The red circle outlines the same lower right grid boundary as seen in Figure 3.9 a). The boundary surrounding this grid has remained unchanged and is outlined in the screen capture.

What is of dramatic difference is the area of smaller triangles, which consists of two rows of triangles along the grid boundary. The SmartMesh has adapted to the new view and made a much higher LOD within these grids. The resulting image is seen in Figure 3.9 d), where the change in LOD is noticeable. The resolution of the image has greatly improved but is most clear in the center of the image. The red circle from Figure 3.9 c) is also visible in this figure representing the exact location. It can be seen that within the circle the resolution is not very high. Directly below the circle, where the

rivers intersect, the resolution of the image is significantly greater. Within the turquoise rectangle in Figure 3.9 c), the border is clearly seen dividing the two LODs. In Figure 3.9 d), the turquoise rectangle corresponds exactly to that of Figure 3.9 c), and it can be seen that there is a marked divide between the two LODs. The left portion of the rectangle has a significantly lower resolution compared to the resolution in the right portion of the rectangle.

These observations illustrate how difficult it can be to maintain smooth seams with changing LOD within a virtual environment. Even with the SmartMesh technique, the differences along the borders are evident but still may be to a lesser extent than other comparable techniques. While flying through the terrain, the differences in resolution are barely noticeable. The process of analysing and isolating the static images may create a bias and bring out the worst perception of the interactive experience, but this careful analysis does help to illustrate the strengths of this technique and its limitations.

Figure 3.9. Terrex Screenshots. a) Viewing the terrain as TIN reveals six complete sections of triangles. b) The terrain image is activated for comparison with a). The coloured shapes in a) correspond to those shown in c) in the next page.

Figure 3.9. Terrex Screenshots. c) Change in eyepoint creates a change in TIN rendering. The boundary between LOD sections is visible. d) With the image activated, the boundary between low and high levels of resolution is visible. The rectangle and circle correspond to the same shapes in c).

3.3 Fractal-Based Terrain Rendering

Fractals represent the concept of self-similarity, where magnified divisions appear identical to the whole (Martz, 1997). An example of this is the branching system within the human circulatory system being the same at different scales, from arteries, to veins, to capillaries. Fractal-based terrain rendering builds on this concept, whereby a ridge or fault line in the terrain in the distance is similar to the uneven, rough edge of a broken rock. Therefore, with this technique, the rendered terrain will always look like terrain, no matter what the scale (Martz, 1997).

The generation of fractal terrains begins with a process called spatial subdivision, where after splitting squares into a grid, the vertices are subsequently moved randomly through each iteration of a fractal program (Bourke, 1991). This concept leads to the idea of fractal image compression, where the computer saves the instructions on how to build the terrain, using less computer memory than storing the image itself (Martz, 1997). Many algorithms and procedures exist for producing fractal terrains. Two will be discussed in this section: 1, midpoint displacement in one dimension; and 2, the diamond-square algorithm; followed by procedural geometry in the next section.

Midpoint Displacement in One Dimension

This form of fractal algorithm is particularly useful for representing mountainous horizons in the distance (Martz, 1997). Basically, there are two loops within the algorithm to generate the feature. First, a single line has its midpoint established and is moved randomly within a chosen range. This results in Figure 3.10 a). Following this, the procedure is repeated where the midpoints of the two generated lines are moved within a

range that is half of the original range (Martz, 1997). For example, if the range was set to be within -1.0 to $+1.0$ in the first step, generating a), then the range for the generation of b) will be -0.5 to $+0.5$. The four lines that are then generated have their midpoints displaced within another reduced range of -0.25 to $+0.25$ creating c). The midpoint of a line is found and displaced by a random amount. Reducing the range of random numbers can determine the roughness of the generated terrain, where more reductions result in smoother ridgelines.

This simple algorithm can be repeated as many times as necessary. If it is not necessary for the image to appear the same each time the terrain is rendered, then using fractal-based terrain rendering can benefit from fractal image compression, whereby recursive instructions are stored for making the image, which uses less computer memory than storing the image itself (Martz, 1997).

The Diamond-Square Algorithm

The diamond-square algorithm brings the midpoint-displacement in one dimension algorithm into the second dimension and thus into the 2.5 dimension where height values are manipulated (Martz, 1997). The grids in Figure 3.11 represent a 5×5 array that follows the diamond-square algorithm through one iteration as shown from a) to c). The algorithm begins with a two-dimensional array, or grid. The four corner points are in bold as shown in a).

From there, the diamond step begins by creating a random value for the midpoint of the square, creating a new point, which is in bold in b) (Martz, 1997). This random value is the average height value of the four corners plus a random number. This creates a pattern of four diamonds, which extend beyond the grid, one of which is illustrated in b). The other diamonds extend from the grid to the right, left, and top of the grid. Following this, the square algorithm takes the center of the four diamonds in b), and generates a value for the new points based on the average values of the corners of the diamonds and adding a random number. The new points are shown in bold in c). Four small squares are then the result. A wireframe mesh based on c) is shown to the right, in d). The numbered points of d) correspond to those of c) (Martz, 1997).

Through the second iteration of this algorithm, the centers of the squares are generated from Figure 3.11 c), and produce new points seen in Figure 3.12 a). The range for generating random numbers in this iteration is reduced by half as discussed previously, so the values will not create peaks above those generated through the first iteration. This creates twelve diamonds for which the centers need to be calculated. The result is seen in Figure 3.12 b). This wireframe that results is shown in c), where more

surface detail can be seen compared to that of the first iteration. Figure d) shows the result after five iterations following the same procedures (Martz, 1997).

Fractal rendering of terrain has many advantages, yet fractal rendering introduces a random component, which may not produce the desired terrain for applications where an accurate depiction of the environment is needed. Therefore, it is important to note that some of the vertices can be given fixed values so that valleys and mountains can be generated as they appear. However, using the diamond squares technique for filling in the areas of less known detail can be quite effective.

Blueberry3d Software: Fractal Based Procedural Geometry for Real-Time Terrain Visualization

Blueberry3d is a software package developed by Sjöland and Thyselius Virtual Reality Systems based in Stockholm, Sweden.

The Terrex software previously discussed uses aerial photos draped over their terrain height models to generate the terrain. The problems with Terrex's approach are the limitations in resolution for the data, even using the Smart Mesh technique (Blueberry3D, 2004b). At minimum distances, the terrain looks flat and uninteresting.

Using Blueberry3d it is possible to add content such as trees, rocks and bushes, but like Terrex, it can be limited in resolution if it needs to be pre-generated.

Fractal mathematics is the basis for creating very detailed virtual environments using Blueberry3D software (Thyselius, 2003). The fractal-based terrain rendering software allows for the generation of virtual terrain using the same database of information for views at both high and low altitudes (Blueberry, 2004a). The concept behind Blueberry3d, is that terrain polygons as well as natural objects such as trees, shrubs, and rocks are generated while the viewer is interacting with the virtual environment (Amand, 2003). This means that features in the terrain are only created and built when needed. For example, if a spherical object were to be drawn, the computer is provided with the center and the radius (Blueberry3D, 2004b). From that, the computer will build the sphere itself based on this information. The idea is that the user supplies the system with very little information so that the system can implement the details (Blueberry3D, 2004b). This is accomplished in real-time producing realistic interactivity with the virtual environment (Amand, 2003).

Procedural Geometry

A process called Procedural Geometry is used within Blueberry3d to generate the terrain. Procedural Geometry works on the premise that as objects in a scene become closer, they also increase in complexity, from trees as a whole, to the branches and leaves attached to the branches as can be seen in Figure 3.13 (Blueberry3d, 2004a). Each object within the scene can be viewed in close proximity at frame rates that are consistent with real-time, with the views only being generated when needed. There are no limits to the level of detail that can be introduced to the image (Blueberry3D, 2004b). In addition, if

limited data are available to generate an image, Blueberry3D uses fractal algorithms to add natural objects such as trees, bushes etc, where none of these objects are created equally, meaning no tree is exactly like another.

There are many benefits to using Procedural Geometry for generating terrain (Blueberry3D, 2004b). First, the resolution can be unlimited. Fractals can be used to create views with very small, fine detail. As well, while the random component can be controlled, the terrain remains interesting where repetition between objects does not occur. Each tree is therefore unique and independently generated. These benefits combine to provide a natural representation of the terrain. The design time required to produce the fractal terrains is also much less than other methods because finer detail can be generated by the Procedural Geometry system with controlled random values. The size of the database is also reduced because only the instructions for building the terrain are saved in memory (Blueberry3D, 2004b). Another interesting feature of Blueberry3D is that it takes into account the varying computer terrain rendering power available on different machines. In other words, Blueberry3d has the ability to adjust scene complexity depending on the available hardware within the computer system (Blueberry3D, 2004a).

Problems do however exist with Procedural Geometry. The computing power necessary to generate the terrains is quite high. However, most personal computers are now able to work with rich terrains using this technique. Another problem is the memory that is required. Although the size of the database can be reduced, in order to provide the finest levels of detail within the image, more Random Access Memory (RAM) is needed for the use of Procedural Geometry within the terrain. In addition, like the Terrex software, Blueberry3D has advanced LOD handling capabilities (Blueberry3D, 2004a)

where objects closer to the viewer appear in greater detail and complexity than objects in the distance. Figure 3.14 is a screen capture of one of the virtual environments produced by Blueberry3D. The terrain within the red box has slightly higher resolution than the terrain far in the distance shown by the green box.

Unfortunately, the problems identified with changing LOD in virtual terrain discussed for the Terrex software are still an issue in Blueberry3D, where smooth transitions between levels of detail are difficult to accomplish without objects popping up (Blueberry3D, 2004b).

Creating Fractal Terrain using Procedural Geometry

Of particular interest are the properties of material that can be stored within the Blueberry3D database. The material in the terrain can be classified such that soil types, density, and features of erosion can be stored within the database (Blueberry3D, 2004a).

The structure of the ground in the virtual terrain is accomplished using ground layers such as soil, rock, and sand (Blueberry3D, 2004b). The properties of erosion, coarseness, and fertility are set as parameters for the ground layers, and the ground is created based on these parameters. By configuring the ground layers, objects such as rocks can appear exposed through the ground, or waves can wash away soil and grass.

With respect to vegetation distribution, clusters of vegetation are distributed to simulate natural distribution within an environment. Individual trees are created using a geometry generator and Blueberry3D's interactive Tree Editor, which uses controlled random numbers (Blueberry3D, 2004b).

Curved surfaces such as roads, paths, and rivers can be implemented in such a way that the user simply creates a set of points for the curved feature to follow and a profile showing a cross-section of the surface. For example, Figure 3.14 shows how a road can be generated within the terrain with the terrain developer only providing a minimal amount of information. From this, the system blends the road into the terrain by changing the textures, smoothing the adjacent terrain, and removing some vegetation and other features. Because the road is generated in real-time, the road will appear smooth as the user moves in to view the area at a closer range.

3.4 Volume-Based Terrain Rendering

Although volume graphics do have a history dating back to the early 1980s, there is a certain stigma attached to volume or voxel graphic rendering including slow computer performance, low image quality, drains on computer memory, and unmanageable file sizes (Woo and Halmshaw, 2004). Visualization developers are therefore more inclined to use surface representations for their applications, although voxel rendering has improved and does provide many advantages.

Volume graphic rendering incorporates the use of voxels, a unit of area represented by a 3D pixel to build a 3D image. Voxels can be used to visualize multiple-return LIDAR data.

LIDAR stands for Light Detection and Ranging. It is a type of remote sensing where a laser beam is directed towards the ground from an airplane, helicopter, or satellite, and a measurement of the distance between the sensor and the target is taken and converted into high-resolution elevation data that is markedly accurate (Stoker, 2004). The distance is determined by keeping track of the time taken for the laser to be sent, and

for time for the signal to be reflected back to the sensor. One major application for LIDAR remote sensing is for mapping of the uncovered earth for land-use surveys. For mapping the Dry Valleys of Antarctica, LIDAR was ideal considering there are no trees or man-made objects to obscure the laser trajectory.

When geographers create 3D visualizations of surfaces, they most often use raster

grids or Triangulated Irregular Networks (TINs).

These methods are not generally useful for representing surfaces that were scanned using LIDAR technology, especially those surfaces laden with vegetation due to multiple-return effects caused when the laser signals are reflected back from branches and leaves on a tree, rather

than the ground. Visualization of discontinuous surfaces, for example buildings in an urban setting, is equally ineffective using raster grids and TINs as can be seen in Figure 3.15, where the TIN overshadows the actual footprint of a building.

Volume rendering presents certain advantages over raster grids and TINs when LIDAR data is retrieved over surfaces with discontinuous elevation. Buildings for example represent a discontinuous surface. Breaklines are commonly used to identify the building to ground boundaries, using secondary imagery or through ground surveys. TINs have the disadvantage that if LIDAR sensors collect sidewall information in the process of scanning, this data will become lost using surface rendering, but remains important in volume rendering.

Multiple attributes can be tagged to individual voxels providing a greater scene complexity than would be possible using a surface ‘drape’. Surface rendering of bare earth may produce satisfactory results, however, when changes in elevation are frequent within the scene, volume rendering is better suited for accurate visualization. The deterrent is that in many cases the added complexity creates higher computing power demand. More efficient storage and rendering strategies now exist for improved visualization using this method.

3.5 Conclusion

Having investigated the LOD, fractal, and voxel-based terrain rendering techniques, many advantages and disadvantages can be associated with each. The concept of LOD seems important for maintaining a high level of interactive quality within a virtual environment. If the system is able to selectively lower or increase the LOD depending on the movement of the viewer, computing power can be used more efficiently and effectively, concentrating its efforts closer to the viewer.

The LOD technique would be most suited for applications where a broad view of the terrain is adequate, for example during flight simulation, or the exploration of an image that has been classified based on land-cover types through remote sensing. Basically, applications where the user does not need to feel immersed within the environment would prove to be the best use of LOD terrain rendering.

The ‘jumping’ or voxel-popping that can occur as a result of changing LOD is still an issue that needs to be resolved. Although the concept of SmartMesh has improved the appearance of a continuous land surface with the progressive mesh borders to its tiles, a problem still exists with the resolution, which is significantly different across

the borders as was seen in Figure 3.9. Work still needs to be done in order to appropriately deal with these problems. With these improvements, the LOD technique could be used to its full potential as an efficient means of terrain rendering.

Fractal-based terrain rendering builds terrain on an entirely different concept, yet still embodies the same principle of objects closer to the user appearing in greater detail. But fractal-based terrain rendering goes beyond this by actually building terrain on the fly as the viewer explores. Trees are built and ground characteristics are implemented based on a few set parameters with the fractal system filling in the gaps with controlled random values. It is an idea that easily integrates objects within a scene increasing the interactive experience where the viewer feels more immersed in the environment. Applications where this can be useful include Virtual Field Trips (VFTs), where a student can feel immersed in the environment, which they may or may not study in reality. As well, fractal-based terrain rendering using Blueberry3d could be useful for an earth scientist where soil properties can be entered in the database and geological features within rocks and cliffs could be investigated at any scale. The random variables would then have to be adequately controlled for such an application, to maintain scientific validity.

Problems do need to be addressed, however, much like the LOD technique. Popping objects and jumping backgrounds, large data storage, and computing power are common to both techniques of terrain rendering.

Although both techniques provide their own set of advantages and disadvantages, merits and limitations, they are able to provide any viewer with a good sense of spatial awareness within a virtual environment. The display and exploration of geographic data can be enhanced, allowing the viewer to investigate the data from different perspectives.

The data used for creation of the McMurdo Learning Module (MLM) is LIDAR data of the McMurdo Dry Valleys of Antarctica. With regards to rendering the image using voxel-based methods, it may be possible to compare the effect of LIDAR on the representation of urban structures with the representation of glaciers and discontinuous elevation, such is the case of the Dry Valleys. The surface boundary between the tip of a glacier may be influenced the same way as the surface boundary of buildings. Surface rendering in the form of LOD and Fractal may not then account for the discontinuous changes in elevation the way that voxel-based rendering can.

In comparing the volume-based rendering to surface rendering Woo and Halmshaw (2004) have identified several issues. In general the quality of the image during volume rendering is poor by comparison, and the memory required for voxel-rendering far exceeds that of surface rendering. As well, the interactive quality is generally poorer and the conversion of three-dimensional models and objects can be difficult in many cases. Volume rendering does however tend to be effective for specialized visualizations: for high-end scientific applications, for example. Table 3.1 gives a summary of the above comparisons.

Table 3.1 A summary of the generalized characteristics of using LOD, Fractal, and Volume-based terrain rendering.					
	Image Quality	Interactivity	Computer Efficiency	Major Issues	Most Suitable For:
LOD	Good	Good	Fair	- Continuous Land Surface - Jumping Backgrounds	Broad Terrain Views
Fractal	Good	Good	Fair	- Random Variables - Jumping Backgrounds	Immersive Environments
Volume	Poor	Poor	Poor	- Image quality is poor - Data conversion is difficult	- Specialized Visualizations - High-end scientific applications

In the next chapter the software packages used for the creation of the MLM are discussed in detail. These are Virtual Terrain Project, an open-source program with applications for terrain rendering using LOD; Virtual Reality Modelling Language (VRML), which is a common programming language for generating three-dimensional objects and models for use over the Internet also using LOD; and NGRain, a sophisticated voxel-based rendering engine for the creation of three-dimensional knowledge objects.

After careful consideration and assessment of financial resources available for the thesis, fractal-based rendering software packages were unattainable. In programs such as Blueberry3D, fractal-based vegetation rendering is the primary focus and because the McMurdo Dry Valleys have no vegetation in addition to the unavailability of Blueberry3d for this thesis, the fractal-based rendering must be left as an area for future research.

Chapter 4: The Tools of Visualization

4.1 Introduction

Three rendering packages are used for the thesis case study which involves creating the McMurdo Learning Module (MLM) using the rendering technique specific to those packages. Virtual Terrain Project (VTP) is a freely available, open-source software package, which is used to reflect the LOD method of terrain rendering. VRML (Virtual Reality Modelling Language) also reflects LOD, and is a standard three-dimensional modelling language for use over the Internet with the added capability to include elements of multimedia and other forms of interactivity. NGRain is a voxel-based software package used for rendering three-dimensional objects into three-dimensional knowledge objects.

What these three programs have in common is an ability to produce three-dimensional interactive models. While all three use the same LIDAR terrain data for the McMurdo Dry Valleys, each has its own capabilities, strengths, and weaknesses which are discussed in detail following a description of the LIDAR data used for the geo-visualizations provided below.

4.2 USGS LIDAR Data for the Visualization of the McMurdo Dry Valleys of Antarctica

LIDAR remote sensing is ideal for places like the McMurdo Dry Valleys, because there is no vegetation or buildings to block the laser light from reaching the ground, allowing for a very accurate representation of the terrain. In addition, the LIDAR scanning was taken at 30 meter spacing. This results in a very high resolution of the

terrain, being accurate to 30 meters over a large area. For a map of the McMurdo Dry Valleys, refer to Appendix I.

The LIDAR data was acquired from the United States Geological Survey Polar Program funded by the National Science Foundation with similar data available free to download from the website: <http://usarc.usgs.gov/antarctic%5Fatlas/>. The data are comprised of two Tagged Image File Formats (TIF). One is an elevation file (30m_elev.tif), and the other is an image file (dry_valleys.tif). With the imagery draped over the elevation data, a high-resolution three-dimensional terrain model results with a top-down view seen in Appendix I. Appendix II outlines the specifications for the file including the original map projection and file sizes.

The technology and capabilities of the terrain rendering packages mentioned in the introduction are each discussed in the sections that follow.

4.3 Virtual Terrain Project [<http://www.vterrain.org/>]

Virtual Terrain Project (VTP) is an open-source terrain rendering software package that focuses on the LOD method of terrain rendering while offering terrain developers the ability to add custom built algorithms within the VTP platform.

There are four applications within the VTP package. These are BExtractor, CManager, Enviro, and VTBuilder. BExtractor extracts the locations of buildings from specific data files. CManager stands for Content Manager and is a tool for preparing the three-dimensional models to be used in VTP. VTBuilder allows for viewing and processing of geo-spatial data, and Enviro is the run-time environment which provides interactive navigation of the terrain for the user.

Once the data is processed using BExtractor, CManager, and particularly VTBuilder, the terrain data is viewed in the Enviro application, which offers a few LOD rendering methods to choose from for the geo-visualization. Because VTP is an open-source package, members of the VTP community can create custom algorithms that take advantage of the LOD performance that VTP offers, but adds specific functionality to accommodate their needs and to test novel approaches.

The standard package uses the ‘Continuous LOD on Regular Grids’ method of terrain rendering which strives to provide the user with “visually continuous change from one resolution to another” as they fly over the terrain (Lindstrom et al, 1996:2). This technique uses a hierarchical tree structure, similar to what was described in Chapter 3, where the terrain is generated depending on the distance to the viewer (Röttger et al., 1998).

The challenge with Continuous LOD, as touched on in Chapter 3, is to keep the transitions between LODs as unnoticeable as possible, with a minimal amount of ‘vertex popping’ (Röttger et al., 1998). There are many variations on the Continuous LOD technique and three variations on the method are offered by VTP. These are 1, Röttger; 2, TopoVista; and 3, McNally. A discussion of each is provided below.

4.3.1 Röttger Continuous LOD Algorithm

With the Röttger technique, the terrain developer selects a maximum number of triangles ‘Triangle Count’ to render in the visualization. This allows the developer to control the resolution of the visualization as well as computer processing requirements. The Röttger technique of Continuous LOD implements a ‘geomorphing algorithm’ that uses a top-down quadtree data structure (Röttger et al., 1998), shown in Figure 4.1, as

opposed to a bottom-up method. The number of vertices needed to be visited for removal in the LOD method is significantly reduced because in a top-down approach, the vertices are looked at from the coarsest to the finest polygons, or from the lowest to the highest resolution, as opposed to the vice versa (Röttger et al., 1998).

Triangulation fanning is an approach used to eliminate gaps between varying resolutions as a result of the LOD (Röttger et al., 1998). It works by determining if a node has an adjacent neighbour with a different resolution from itself. If they do vary in resolution, the process prevents them from becoming disjointed by skipping over, and preventing the subdivision of these vertices that act as borders between different areas of resolution. In Figure 4.1 b), x marks the vertices that are skipped during the triangulation fanning process (Röttger et al., 1998).

4.3.2 TopoVista Continuous LOD Algorithm

When TopoVista is selected as the LOD rendering algorithm, the terrain developer is no longer able to select a Triangle Count to limit the size of the visualization, but instead is offered a Pixel Error Bound.

If the three corners of a planar polygonal triangle are stapled flat over a terrain surface, there will be some places where the terrain appears above or below that fixed triangle. The vertical distance between the terrain surface and the triangle surface

determines the Pixel Error Bound (Evans and Townsend, 2003). By setting this value, the algorithm generates more triangles as needed to keep the distance within the boundary value. For example, when the algorithm encounters an area where the error is larger than the Pixel Error Bound, TopoVista cuts the triangle in half to reduce the error and therefore creates a better approximation of the topography (Evans and Townsend, 2003).

4.3.3 McNally Continuous LOD Algorithm

The McNally algorithm is also known as the ROAM (Real-time Optimally Adapting Mesh) algorithm and uses a LOD process called Binary Triangle Trees which cuts the landscape

into triangular sections as seen in Figure 3.4 (Turner, 2000). When a triangle splits, it is split equally to produce children of the original triangle. With each increase in generation of children, the terrain reaches a higher resolution. More splitting thus occurs where highest LODs are needed, depending on the position of

the view-point (Turner, 2000). Once the desired LOD has been reached, the triangles are rendered on-screen during a second pass at the data. This two-pass system prevents the need for storing per-vertex data which significantly reduces the amount of memory needed (Turner, 2000).

A feature called the “Variance Tree” works with the Binary Triangle Tree described above to determine how many children should be created from parent triangles. This is similar to the Pixel Error Bound of the TopoVista algorithm, except that the error is called variance in this

case, and based on the variance, the triangles split into equally sized children, thus covering the same area (Turner, 2000). With each iteration of the

algorithm, the variance drops roughly by half until the desired smoothness is reached as shown in Figure 4.3.

4.4 Virtual Reality Modelling Language (VRML)

The Virtual Reality Modelling Language (VRML) is a common programming language used for generating three-dimensional objects intended for interaction over the Internet. The release of VRML was on August 4, 1996 (Roehl et al., 1997). Because VRML is not a software package like Virtual Terrain Project or NGRain, all that is needed to make a virtual world or objects is a standard text file with correct VRML programming syntax, saved as a .wrl file rather than .txt, and a VRML viewer of which many are free to download, for example ParallelGraphics Cortona.

For this thesis project the LIDAR data were converted to VRML in one step using ESRI ArcScene 9.0. While VRML programming and editing of the generated VRML file was possible using Microsoft Notepad, ParallelGraphics' VRMLPad 2.1 was purchased and installed because of its ability to make editing the large file more manageable

VRML was the program of choice of Cartwright (2005) in the implementation of The MapShop and GeoExploratorium. As was mentioned in Chapter 2, the metaphor for the virtual environment (VE) was games, and he argues that VRML is the tool of choice to create a VE with the look and feel of a computer game.

The ability of VRML to make highly interactive scenes and objects are what make VRML a good candidate for creating Virtual Learning Environments where links to images and other multi-media can be accomplished. This makes it an excellent candidate for creating the McMurdo Learning Module (MLM) despite it not having sophisticated terrain rendering capabilities.

A VRML file is very simple. Within a text editor it appears as a set of programming nodes that informs the browser how to construct the visualization. In Table 4.1 a VRML programming file is shown. Text superseded with a '#' represents a comment within the code, '{' and '}' represent the beginning and end of a node, and '---skip---' represents segments of skipped code within the Table.

The WorldInfo node contains a title as well as additional information about the production of the file. The Background node specifies the colour of the sky. The sky colour in this case is set to white (1,1,1). The DirectionalLight node specifies the light source and direction of light within the visualization. The ambient intensity is set to 1, meaning 100% illumination of the environment, and the direction of the light is defined here and is set to a default value. The Navigational node determines if there is a headlight, meaning if the user emits light from their viewpoint. In this case it is set to TRUE. As well, the default speed of navigation is also set in this node, and in this case it is left as default to be 6237.000093 meters per second. The Viewpoint node describes the initial view point for the viewer as s/he enters the environment in terms of field of view, which is set to 30%, orientation, and position. Beyond these primary nodes of the .wrl file, the script increases in complexity and are generated by ESRI ArcScene 9.0 in the export to VRML.

Table 4.1 A sample of a VRML (.wrl) file created using elevation data that was exported from ESRI ArcScene 9.0.

#VRML V2.0 utf8	Group
	{
	children
	[
WorldInfo	---skip---
{	Transform
title "ArcScene Document"	{
info "Generated by ArcScene"	translation -45470.000000 0.000000 -62370.000000
}	children
	[
Background	LOD
{	{
skyColor 1 1 1	center 45470.000000 1641.719984
}	62370.000000
	range [498960.000000 997920.000000
DirectionalLight	1995840.000000 3991680.000000]
{	level
ambientIntensity 1.0	[
direction 0.612372 -0.500000 0.612372	Shape
}	{
	appearance Appearance
NavigationInfo	{
{	texture ImageTexture
headlight FALSE	{
speed 6237.000093	url "McMurdoScene000.jpg"
}	}
	material Material
Viewpoint	{
{	---skip---
fieldOfView 0.3	}
orientation 1 0 0 -0.7853980	
position 0.000000 251121.718750	
249480.000000	
}	} # end appearance
	geometry ElevationGrid
	{
	ccw FALSE
	solid FALSE
	xDimension 187
	zDimension 256
	xSpacing 488.924731
	zSpacing 489.176471
	height
	[
	1836.928833 1834.829224 1832.698120
	1881.672974 1911.295410 1850.134155
	1824.303589 1822.198730 1820.089111
	1817.977417 ...
	...etc...

The Group node represents a group of objects, and the ‘children’ are objects within the Group. The Transform node gives the translation parameters, where the objects occurs in Cartesian space, and rotation if necessary, together indicating the placement of objects for the children of the Group. Beyond this, the rendering method becomes apparent. First, the Level-Of-Detail is indicated by the LOD node, where a center, range, and level are given. The center provides the center of the LOD node for calculating the distance, and the range specifies the distance from the viewer where the LOD should switch (Carey et al., 1997). The level provides a list of nodes in order of highest to lowest LOD to the browser so that objects can be represented at varying levels of detail depending on the distance from the viewer (Carey et al., 1997). In the example of Table 4.1, only one level is shown with the object being a Shape with an image texture called *McMurdoScene000.jpg*, one of the images generated by ArcScene in creation of the VRML file. The VRML browser calls this image using the url given. The *ElevationGrid* is then defined giving dimension to the Shape in the form of a height field and geometry settings, resulting in a 3D visualization. The ‘ccw’ field provides the order of vertex coordinates of a geometry, giving the normals of each surface (Carey et al., 1997). With the ccw set to FALSE, the normals are oriented in a clockwise order. The ‘solid’ field indicates how each polygon should be displayed. With it set to FALSE all polygons in the visualization are visible from all viewing directions. The fields ‘xDimension’ and ‘zDimension’ provide the extents of the grid height array, in this case with 187 elements in the X direction, and 256 in the Z direction. The ‘xSpacing’ (488) and ‘zSpacing’(489) provide the distance between vertices, and the height field provides a long list of numbers that represent height values for each vertex. Figure 4.4 illustrates the components for a

VRML GridElevation node with a 4x5 rectangular grid representing the height array. The

x and z Spacings are defined, and the elevation grid is shown hovering above with vertices placed at different heights and with normals set by the ccw field.

4.5 NGRAIN [www.ngrain.com]

While the NGRAIN products are designed for aerospace, defence, and automotive industries, NGRAIN also provides the framework necessary for this research, being able to visualize terrain using voxel technology.

There are three components to the software: NGRAIN Transformer, NGRAIN Knowledge Module, and NGRAIN Mobilizer. NGRAIN Transformer is first in the line of NGRAIN products used by a developer and allows existing three-dimensional data in a variety of formats to be imported and compressed to the NGRAIN file format .ngw with a subdivision of the model defined as follows:

Non-NGRAIN Model -> Converted to:

NGRAIN Model (.ngw format) -> Composed of:

Model Parts -> Composed of:

Voxels (three-dimensional pixels)

Therefore, the voxels are the building blocks of the model parts, and these parts can be modified, edited, combined, and individualized as needed within the NGRAIN Transformer package. There are four main purposes of NGRAIN Transformer (NGWTrans, 2004): 1, to **import and compress** already existent three-dimensional data into the .ngw NGRAIN format; 2, to **synchronize and edit** model parts ensuring that the parts are properly identified and well represented; 3, to **manage** model parts; and finally 4, to **prepare** the three-dimensional models for use in multiple applications.

Once the model is prepared within NGRAIN Transformer, the model can then enter into the second phase of development within NGRAIN Knowledge Module. This has many of the same capabilities as NGRAIN Transformer including the ability to create cross sections, auto-spin, zoom, pan, assemble and disassemble, manage parts and labels, and create XML documentation. However, once the three-dimensional model has been refined and edited in NGRAIN Transformer, NGRAIN Knowledge Module is used to enhance the model by adding links, logic, and animation. This is done without using programming or scripting codes making this platform a form of visual programming.

Creating hyperlinks within the model can be very useful for adding reference material. Hyperlinks can link a part to another NGRAIN object, WebPages, HTML files, multimedia files etc.

Because NGRAIN can be used to visualize complex machinery and equipment, disassembly and assembly logic are necessary for proper education about these objects. Assembly logic dictates the order in which parts can be attached or detached relative to other parts in the model. Custom error messages can accompany these procedures.

Another interesting feature of the NGRain Knowledge Module is the ability to create and publish multiple configurations of the same three-dimensional model. This means that the developer can customise the model, links, logic, animations, and interactive capabilities for different users. Publishing the Knowledge Object configuration into a standard HTML page is done instantly. As well, when saving the model in the NGRain Knowledge Module or NGRain Transformer, both platforms update the same .ngw model file.

NGRAIN Mobilizer is the only NGRain application that is free to download from the NGRain website. It is a run-time viewer plugin required to view .ngw files outside of NGRain Transformer and NGRain Knowledge Module. It can be installed as a stand-alone application, but could also be embedded into Microsoft Word documents, PowerPoint slides, or HTML pages (NGRAIN Products, 2005).

The capabilities and interactions possible within NGRain Mobilizer depend on the configurations set by the developer in the NGRain Knowledge Module. If all the icons are active within the configuration, the user can assemble and disassemble parts, rotate the model, zoom in and out, pan, and create cross section. The user can also draw using a simple 'pencil' tool, and view the model in X-Ray mode. The viewer can run animations from NGRain Mobilizer, and hyperlink to reference material.

Most voxel sets that represent surfaces in NGRain are over 90% empty, given that the surfaces are hollow, representing empty voxels, even in cases of parts within parts. The data structure therefore places no memory costs on empty voxels, thus saving on memory space (Woo and Halmshaw, 2004). Maintaining small file sizes is important for interactivity, and web applications.

4.5.1 NGRAIN Technique of Three-Dimensional Object Rendering

The NGRAIN products convert three-dimensional models into three-dimensional knowledge objects. These knowledge objects are built using voxels, and rendered using anti-aliasing techniques, discussed further below. The following section describes the three elements of an NGRAIN model.

Three-Dimensional Knowledge Objects

Three-Dimensional Knowledge Objects (3DKOs) are built on other existing terms for virtual objects that carry information. These are Learning Objects, Reusable Learning and Information Objects, and Knowledge Objects (Simpson, 2004). See Figure 4.5.

Learning Objects are defined by Wayne Hodgins as distinct and portable amounts of information, which are functional among various computer-based training courses (Simpson, 2004). Reusable Learning and Information Objects are a derivative of the Learning Object and are based on a task that

includes several Reusable Information Objects (RIOs) which are pieces of information designed to tackle one learning objective. Knowledge Objects are different from the two mentioned above because the information is replaced with 'subject matter expertise' that are then placed in learning and reference material composed of Learning Objects, Reusable Learning Objects, and Reusable Information Objects. Three-Dimensional Knowledge Objects 3DKOs are therefore just an extension of the Knowledge Objects

which incorporate traits of Reusable Learning Objects and Learning Objects. 3DKOs are three-dimensional models or scenes that contain knowledge about the object being represented in a way that is visually communicated (Simpson, 2004).

NGRAIN Rendering Technology using Voxels

While NGRAIN technology includes the ability to convert, modify and manipulate 3DKOs, the technology behind the rendering of the objects is what is particularly relevant for this research. A volumetric data structure is used to represent the three-dimensional model using voxel technology, and polygonal surface rendering technology, which are the two most common forms of producing three-dimensional representations.

A voxel is a three-dimensional pixel. A volume graphics object consists of voxels as the basic element of the model. They are based and positioned within a three-

dimensional grid of X, Y and Z coordinates making a voxel cube space. Every voxel within the grid is the same size and same distance from surrounding voxels, all being cube shaped. Figure 4.6 shows a 4x4x4 voxel cube, with 64 places to be filled, three of which are seen as occupied

voxels (Woo and Halmshaw, 2004).

The sandcastle analogy best describes the use of voxels to represent or build a three-dimensional object. Much like individual grains of sand together contribute to the overall structure of a sandcastle, voxels stacked in the same manner produce virtual three-dimensional objects (Kaas, 2004). Not all voxels need to be perfect cube shapes as seen in Figure 4.6, but can be other shapes within irregular grids (Woo and Halmshaw, 2004).

Using voxels, or volume graphics, presents an advantage over polygonal representation such that voxels can be used in characterization of both surface and internal aspects of an object, making the representation more precise (Kaas, 2004). A disadvantage, however, is that the computational demand is significantly higher than with polygonal representation. Producing high quality, interactive graphics requires a lot of memory, which has restricted the use of voxels, making polygonal representation the choice in many applications that supply a ‘realistic approximation of the surface of three-dimensional objects’ (Kaas, 2004).

Recent developments in technology have encouraged the use of, and efficiency for volume graphics rendering on common PCs without special technical needs (Kaas, 2004). Both volumetric and polygonal graphics rendering present their own sets of advantages and disadvantages. NGRain has the goal of overcoming many of the disadvantages present in both (Kaas, 2004).

NGRAIN Data Structure

Although volume data and rendering traditionally require extensive memory and computational power, NGRain’s data structure works to eliminate these hindrances to produce interactive graphics of high quality. These include the ability to have a high data compression rate, data access speed, and voxel attributes (Kaas, 2004).

High data compression rate refers to NGRain’s ability to store only the data within the volume grid that occupy voxels. In many programs, data is inserted into unoccupied spaces, which causes an inefficient use of storage space. NGRain therefore has a ‘small memory footprint’. NGRain models are continuously updated as new data

is added or changed. ‘Selective data loading’ is another method of ensuring that memory requirements are minimized, where only necessary data are loaded.

Data access speed is significantly enhanced by NGRAIN’s ability to randomly access voxels within the compressed data structure, facilitating model manipulation and alteration. Content streaming is another method for ensuring fast data access, where a rough shape of the model is provided before the entire model is loaded.

Voxel attributes contribute to the photo-realistic rendering capable within NGRAIN. Geometry information defines the physical shapes within the model (points, lines, curves, angles, surfaces, solids, and relative positions) (Kaas, 2004). Associated data includes property information for each voxel such as colour, surface normal, intensity, and layer (Kaas, 2004). Customizing the attributes can allow the designer to assign a density to certain voxels. It is then possible to design algorithms using the voxel attribute data in, for example, simulating the spread of a fire (Kaas, 2004).

NGRAIN File Format

An NGRAIN file contains voxel attribute information, light, transformations, and material settings as well as custom attributes (Kaas, 2004). With XML, user-specified metadata can be imposed on the model and read in various other applications (Kaas, 2004).

As discussed previously, data compression is a significant factor for NGRAIN and is accomplished using lossless compression techniques. This involves only storing data in occupied voxels with attributes saved in different adjacent memory cells. Volume data compression works the same way as the 2D pixel compression seen in JPEG, GIF, and TIFF file formats.

Any type of existing three-dimensional object can be visualized with voxels within the NGRain platform. Accuracy within NGRain models is defined by the voxel resolution, which is specified using a unit-less value.

When a polygon model is converted into NGRain, the triangles are drawn into a volume cube. The number of voxels representing a polygon within the model needs to be high enough that the results are of high quality. Many three-dimensional models are intrinsically volumetric including geo-spatial elevation models.

Rendering the Three-Dimensional Object in NGRain

Aliasing of objects is a common rendering problem for volumetric 3D visualization, where edges appear jagged and textures appear unintentionally wavy or shiny (Zwicker et al., 2001). NGRain uses an anti-aliasing effect called “splatting” which allows for better representations of curves on the object incorporating contiguous memory access, and quick exit conditions applied to voxels not currently in view.

Anti-aliasing is a form of rendering typically used for 3D volumetric visualization, where the algorithm constructs a continuous 3D function, transmits the function onto a screen, and changes the opacity of the visualization depending on changes in the line-of-sight (Zwicker et al., 2001). The ‘splatting’ method improves on this concept by representing volume data as particles that absorb and emit light (Zwicker et al., 2001).

With the anti-aliasing method, each voxel is projected onto the screen, accumulating to produce the visual representation (Mueller et al, 1998). Each voxel is visited in a back-to-front or front-to-back approach so that the farther voxels from the view are overwritten by the closer voxels (Mueller et al., 1998). An image kernel

constructs the voxels into an image using a splat or footprint function, whereby parallel rays of information from the kernel reconstruct the image (Mueller et al., 1998).

NGRAIN has a fast rendering speed because voxels not in view are rejected, along with attribute data for the voxel in the rendering processes, which significantly reduces time.

4.6 Conclusion

This Chapter focused on the specific forms of rendering three-dimensional terrain and objects used by three separate software packages. Virtual Terrain Project uses the Continuous Level-of-Detail approach, which is also present with VRML. Using a mesh of polygons is the most common form of surface visualization according to Kaas (2004) where triangles are used to characterize the surface. The more triangles are used, the more curvature and features can be better represented. Increasing the number of polygons increases the resolution and accuracy for the representation. Because each polygon required in a visualization requires computations to determine the correct shading etc. This puts a lot of demand on computer power. There is a trade-off therefore between visualization quality and computation speed, which determines the level of interactivity that can be experienced by the user. This trade-off has limited the amount of inexpensive visualizations for use on PCs, and reduces the ability of using the same model in other applications (Kaas, 2004). This concept is illustrated in Figure 4.7.

Polygonal rendering is only effective for visualising surface or exterior characteristics of an object. Characteristics such as mass, temperature, elasticity, and texture cannot be represented accurately using polygonal rendering and also limit the user's ability to explore internal portions of the three-dimensional model (Kaas, 2004).

Table 4.2 provides a summary of the software comparisons discussed in this Chapter.

Table 4.2 Summary table of comparison between VTP, VRML, and NGRAIN.			
	Availability	Rendering Technique	Useful for Visualizing:
VTP	Free upon request	Continuous LOD (surface)	- the surface or exterior
VRML	Free	LOD (surface)	
NGRAIN	Free Evaluation License	- Voxel (volumetric) - 3D Knowledge Objects	- the internal portions of objects. - mass, temperature, elasticity, texture characteristics

The following Chapter describes the use of these packages in the creation of the McMurdo Learning Module from the terrain developer's perspective and provides an evaluation of the rendering methods used by them.

Chapter 5: Case Study and Methods of Evaluation

5.1 Introduction

In this chapter, all of the principles of design, interactivity, and implementation of a geo-visualization as determined through the previous chapters are discussed with specific focus on the use of the McMurdo Dry Valleys LIDAR data towards the creation of the McMurdo Learning Module (MLM). The chapter begins with a discussion of the evaluation process followed by a detailed evaluation of the specific software and techniques used in representing the geomorphology of the McMurdo Dry Valleys. This includes how the products function from the terrain developer's perspective and what results are generated.

5.2 Discussion and Evaluation

Having used the visualization packages described in Chapter 4, there are three recurrent steps to terrain visualization that are common for all three. These are:

1. Data conversion and processing
2. Terrain population with information and objects
3. Terrain run-time viewing

In Appendix VI, the system requirements needed to run each packages' terrain run-time viewing environment is provided. The following sections describe the use of the packages through these three steps from a terrain developer's perspective. An evaluation of each package, in terms of representing geomorphology in the geo-visualization and use towards the creation of the McMurdo Learning Module (MLM) follows each discussion.

5.2.1 Discussion of Virtual Terrain Project

As mentioned in Chapter 4, Virtual Terrain Project (VTP) is an open-source terrain visualization program that functions based on the Continuous Level-Of-Detail (LOD) approach to terrain rendering. VTP is free to download and provides three separate algorithms for rendering while staying within the Continuous LOD rendering platform.

Of the four applications available through VTP, the VTBuilder and Enviro applications are of particular use for visualizing the McMurdo Dry Valleys LIDAR data. The original data and file formats are provided in Appendix I and II respectively. The three for terrain visualization using VTP are discussed below.

Data Conversion and Processing

Before the data could be viewed in VTBuilder, the original raster data was clipped

Table 5.1: The extents of the clipped raster files of the McMurdo Dry Valleys of Antarctica. Clips resulted in an approximate 57% reduction in file size for both files.	
<p>Extents for both clipped files:</p> <p>Left: -45136.313457 m Top: 121235.961603 m Right: 45808.579835 m Bottom: -3501.423212 m</p> <p>Projection: Lambert Conformal Conic. (See Appendix II for more details)</p>	<p>File: clippedraster.tif from dry_valleys.tif Size: 108.18 MB down from 251.77 MB Reduction of: 143.59 MB or 57.03% Exported map with layers converted into JPG image: 440KB</p> <p>File: clipelev30m.tif from 30m_elev.tif Size: 48.08 MB down from 112.08 MB Reduction of: 64 MB or 57.10%</p> <p>Saved to .bt and .prj clipelev30m.bt: 65.569 MB clipelev30m.prj: 1KB</p>

using ESRI ArcMap 9.0. This was done to limit the file size to the immediate area of

interest for the visualization and thus reduce the file size for both the image and elevation .tif files. The dimensions of the clip and resultant file sizes are given in Table 5.1. The clipping resulted in a reduction of approximately 57% for both .tif files.

Once the files were clipped, the clipelev30m.tif elevation file was imported into VTBuilder as shown in Figure 5.1, and the extent was set with the Area Tool to be equal the extent of the elevation layer which was: 90930 x 124740 pixels. Using the Merge and Resample Elevation command, the elevation file was converted into two separate .bt (Binary Terrain) and .prj (External Projection) file formats.

For Continuous LOD, rendering the

Grid Size for the terrain needed to be set with a constraint of power of 2 plus 1. The Grid Size can therefore either be 2x2, 3x3, 5x5, 9x9, 17x17, 33x33, 65x65, 129x129, 257x257, 513x513, 1025x1025, 2049x2049, 4097x4097 and so on. Depending on the Grid Size chosen for the LOD, the Grid Spacing is affected as shown below.

Grid Size: 2049 x 2049 Grid Spacing: **44.3994140625** x **60.908203125**
 Grid Size: 4097 x 4097 Grid Spacing: **22.19970703125** x **30.4541015625**
 Grid Size: 8193 x 8193 Grid Spacing: **11.099853515625** x **15.22705078125**

The original Grid Spacing for the elevation file is:
30.00990099009901 x **30.0072167428434**

Therefore a Grid Size of 4097x4097 is determined to be the most ideal for this representation. Saved as DryVallElev.bt, the generated file has no gaps and has 4097 lines of pixels for each side of the square area. Once saved, the two files created are: DryVallElev.bt (65,569 KB) and DryVallElev.prj (1 KB). Converting to .bt therefore increased the size of the clipped elevation file from 48.08 MB to 65.569 MB, an increase of 26.67%.

Terrain Population with Information and Objects

Figure 5.2. Layers incorporated into the image layer including a coastline (green), glacier flow-lines (blue), and glacier moraines (purple).

Within VTP it is possible to add layers in VTBuilder, and objects such as park benches, fences and trees within the Enviro application. As well, by combining layers into the image file, those layers can be incorporated into the environment as seen in Figure 5.2. This results in poor visual quality however in the run-time environment. When using VTBuilder to import layers, they often do not load properly, resulting in errors.

Terrain Run-Time Viewing

The final visualization is viewed using the Enviro application. Before viewing the terrain, the files need to be placed into the correct folders. Therefore, the clipelev30m .bt and .prj files were moved into the C:\Program Files\VTP\Data\Elevation directory, and

the ClippedRaster.jpg were moved into the C:\Program Files\VTP\Data\GeoSpecific directory. These directories are created when installing VTP.

Upon loading the application a Startup menu opens with options for viewing an Earth View to start, or to jump right in to the terrain visualization. Under Terrain View, “Generic Terrain” is selected for visualization where under Edit Properties the user selects the elevation file (clipelev.bt) and the texture file (ClippedRaster.jpg) to generate a custom terrain. The three functioning choices for LOD algorithms are the Röttger, TopoVista, and McNally algorithms which were discussed in Chapter 4. There are other settings as well including adding culture, sky and ocean parameters, abstract layers, and miscellaneous options to the view.

Once the data is loaded into Enviro, the user can fly through the terrain, take ground measurements for distance, and camera snapshots of the terrain. The system requirements for the VTP Enviro application are provided in Appendix VI.

5.2.2 Evaluation of Virtual Terrain Project

Because VTP is an open-source package, the documentation for the product is limited, and applications do not run as smoothly as the more sophisticated commercial products, Terrex or NGRain. For example, the data processing is fairly involved and the placement of files in specific directories would be automatically done using a more sophisticated package. Nonetheless, VTP did provide valuable insight into the use of Continuous LOD in the rendering of terrain.

Figure 5.3 shows the imported clipelev30m.tif file into VTBuilder. With the original projection of the file, it is properly placed within the Antarctic Continent as seen in a), and is magnified in b). Also, when loading the files in Enviro, the TopoVista

algorithm caused a program crash, therefore leaving only the Röttger and McNally algorithms for comparison.

Using the Röttger algorithm, the triangle count was set to 10,000 and the resultant terrain can be seen in 5.4 with the wire-frame view switched on. Within Enviro there is an ‘Increase Detail’ icon which adds triangles to the view making the terrain, including

the Upper Wright Glacier Valley, circled in red, appear at higher resolution. This algorithm is the default algorithm for VTP, and from the wireframe view, the grid structure divided into triangles is clearly visible. It appears different from the wireframe view of Terrex discussed in Chapter 3, because the triangles appear as right angle triangles within a defined grid structure without visible boundaries between areas of differing LOD.

Figure 5.5 demonstrates how the triangles change or multiply as the viewpoint changes. In a) the viewpoint has just moved over a ridge, and while there are many triangles in the foreground, they multiply as the viewpoint continues over the terrain.

Figure 5.6 shows a series of three screenshots using the McNally Algorithm in nearly the same location, overlooking the Wright Valley. Like the Röttger algorithm, the triangle count is set to 10,000 and uses fans for the rendering as described in Chapter 4. The number of triangles over the ridge is significant as seen in a), and as the view-point changes the triangles multiply very quickly and become very small as seen in b), and furthermore in c).

In the accompanying CD, there are two videos which may help to illustrate what is shown in Figures 5.5 and 5.6. Click on the VTP Folder and select Rottger.avi for a demonstration of the Rottger algorithm, and McNally.avi for a demonstration of the McNally algorithm.

In both cases the frame rate fluctuates between 62.5 and 66.7 frames per second (fps) and the movement over the terrain appears very similar without much noticeable difference between the two. With the frame rate averaging at 64.6 fps, this is deemed an acceptable value according to Gahegan (1999) who suggests that 24 fps is a desired minimum.

Using VTP for creating the McMurdo Learning Module (MLM) remains questionable. The addition of objects with links to information does not seem to be possible. It is therefore a useful tool for viewing the terrain and objects but not for

interacting with them. As well, the Enviro run-time environment requires that the user place the terrain files into the appropriate folders and make the correct selections before running the view. This is a problem when the package requires that multiple files be stored in different directories. Saving the terrain in one file with all objects and textures would be ideal so that a generic plugin could be used to view the file. This would open the visualization for viewing with free distribution and allow for embedding the run-time viewer into WebPages for example.

5.2.3 Discussion of NGRAIN

There are three applications within the NGRAIN package. These are: NGRAIN Transformer 2.1, NGRAIN Knowledge Module 2.2, and NGRAIN Mobilizer 2.2. One advantage of the software not being open-source is the amount of documentation provided including many tutorials. The three steps for creating the visualization in NGRAIN are provided below.

Data Conversion and Processing

The process of visualization using NGRAIN begins with the Transformer application. On its own, Transformer can convert .vol (Volume), .obj (Object), and PointCloud file formats into the .ngw file format specific to NGRAIN. The terrain data however is not in either format. To use other file types, NGRAIN recommends the OKINO PolyTrans package. The demo conversion tool called NuGraf was therefore installed with the ability to convert many file formats, excluding the .tif format of the McMurdo LIDAR raster files. NuGraf did however accept VRML or .wrl files for conversion. Only the demo version was free to download which presented several limitations, with the ones of relevance to this study being:

- Exported three-dimensional files will have every 5th polygon removed,
- A small watermark bitmap will appear in the lower-left corner of all rendered images,
- Texture files used for rendering or re-export will have black lines added to them,

Using ESRI ArcScene 9.0, a clipped version of the McMurdo Dry Valleys Terrain was created including simple polygons representing areas of interest which were exported as a three-dimensional VRML file. The original .tif files were clipped down in size, which reduced the file size by 11.49% as seen in Table 5.2.

Table 5.2: File size reduction and export to VRML using ESRI ArcScene 9.0.
File: Clip.tif 225.83 MB down from 251.77 MB of the original. Reduction of: 25.94 MB or 11.49%
Extent: Top: 107106.727749 Bottom: 11576.727748999998 Left: -60582.679340000002 Right: 63357.320659999998

Once the terrain was in .wrl format, it could be read by the NuGraf demo and converted into a .obj file. Once this was done, the .obj file looked compromised based on the limitations of the NuGraf demo, which were outlined above.

The import feature of NGRain Transformer allowed for the .obj file to be converted into NGRain's .ngw file format. The Advanced Importing options allowed for orienting the model to optimize rendering speed, with maximum model dimensions set to 1024 voxels, representing the maximum number of voxels for the longest dimension.

There was an initial error in the conversion such that: the "Swap File Size Might Be Too High. Reduce." This required a change in the Virtual Memory settings of the computer. The Virtual Memory was initially set to 2046 MB Max Size: 4092 MB, and

therefore reduced to 500 MB Max Size: 4092 MB. Another error then occurred: “Can’t be saved in the specified folder”. It saved only 430KB in the first attempt using the original Virtual Memory settings. During the second attempt, the Maximum Model Dimensions were set to the highest level of 1024 voxels, resulting in 1.99MB of the model data being saved. This was the most NGRain Transformer could handle of the clipped LIDAR data. The conversion was successful therefore only up to the latter specifications.

Once the file was imported into the Transformer Application, three-dimensional parts were generated automatically by the program based on the source file. With an .obj file, all objects within the file are considered different parts by NGRain. The polygons therefore appear as separate parts to the three-dimensional model. Appendix III illustrates the terrain model within Transformer and provides the system requirements for NGRain. The file size is 2.1 MB. The dimensions are x: 790 y: 1026 z: 127 with a total of 968.355 voxels. Fortunately as well, the demo limitations imposed by NuGraf did not appear to be transferred into the NGRain Transformer Application.

Terrain Population with Information and Objects

Once the model was loaded into Transformer and the parts identified and refined, knowledge was added to the terrain using the Knowledge Module application. With this application, all of the parts, which represented areas of interest, were given labels and links to important information. There are 11 parts in the scene representing areas of interest for the user to visit. The red polygons represent piedmont glaciers, yellow polygons represent outlet glaciers, green polygons represent alpine glaciers, and blue polygons represent frozen glacial lakes as can be seen in Appendix III. In Figure 5.7 the

model is opened in Knowledge Module, with the Webb Glacier polygon linked to the corresponding HTML file with information about the Webb Glacier, and with a setting to target the HTML pop-up information in a new window.

No assembly logic was added to the model since the parts were not meant to detach or reattach in any particular order.

Figure 5.7. Terrain loaded into NGRain Knowledge Module with link windows open.

Terrain Run-Time Viewing

Viewing the three-dimensional model with the links set is done using NGRain's Mobilizer application. This viewer is free to download from the NGRain website, and can be embedded as an object in an HTML file, as well as Microsoft PowerPoint and

Figure 5.8 NGRain with the McMurdo terrain loaded into the Mobilizer application with X-RAY view activated and the image brightened.

Word. Appendix VI lists the system requirements for the Mobilizer application, and Figure 5.8 provides a screenshot of the terrain loaded into Mobilizer. The image to the right shows the image through the 'XRAY' view, which is quite different from the wireframe views of Terrex and VTP. The XRAY view is based on the density of voxels

within the model. For a terrain, the density image looks very similar to a contour map, which is understandable considering the voxels are densest in areas of greater relief in the topography. The image does not change with changing viewpoint, however. The visualization and ‘contour lines’ stay the same with movement of the user, however there is a refresh period where the image appears blurry for a few seconds, before resolving itself very quickly. There is a maximum zoom level that the user can reach, and the resolution of the image is quite good considering the full file was not able to be converted. From the accompanying CD, install the NGRain Mobilizer application by clicking the Mobilizer2.2.exe file and following instructions. Then open the file NGRain.html using Internet Explorer to see a live version of this application.

5.2.4 Evaluation of NGRain

The data conversion process was the most challenging aspect of using NGRain for the visualization of the terrain data. Even with all of the limitations set by the demo version of the NuGraf conversion tool, and although the image at its highest resolution was not able to import fully within NGRain, the result was acceptable, providing an interesting approach to geo-visualization through the conversion of three-dimensional terrain into a three-dimensional Knowledge Object (3DKO). Once the files were converted into the .ngw format of NGRain, the same file worked with the three different NGRain applications, making managing changes efficient and easy.

Table 5.3 illustrates the conversion capabilities of NGRain using the Jet Fighter model described in their documentation, compared to the McMurdo Terrain Model used for the case study. With the jet fighter, there was a decrease in file size of 215MB, amounting to a reduction of 97% from the .obj file. For the McMurdo terrain file, the

decrease in file size of 107 MB amounted to a reduction of 98% of the .obj file. The reductions in sizes are therefore comparable to that advertised by NGRain. The reduction in file size, however, may be due to NGRain's inability to import the entire model into NGRain Transformer.

A reduction in the file size is a common goal for most rendering packages because generally, the smaller the file size, the less computing power is necessary to render the visualization. The goal of this thesis includes looking at how the different packages handle the file sizes to produce a 3D representation the terrain.

Table 5.3 Conversion capabilities and reduction in file size by NGRain.						
Model	#Polygons (for surface rendering)	# Voxels	# Parts	Before Conversion. Object File Size. Wavefront OBJ	Before Conversion. Object File Size. 3D Studio Max	After Conversion NGRAIN File Size
Jet Fighter	2.5 million	1.8 million	5,792	220 MB	27 MB	5.0 MB
McMurdo Terrain	?	968,355 0.9 million	11	109 MB	N/A	1.99 MB

NGRAIN serves as an excellent candidate for the creation of the McMurdo Learning Module since it is easily embedded within a Webpage as seen in Figure 5.9, and the parts provide links to separate Webpages of information, pictures, and other multi-media sources. As well, the functionality with NGRain Mobilizer is such that students can create cross-sections through all axes of the model providing for unique perspectives of the elevation data. By opening NGRain.html in the accompanying CD, the page appears as it does in Figure 5.9.

Figure 5.9. The NGRain Mobilizer application embedded into an HTML page as an interactive learning environment for students to learn about the geomorphology of the McMurdo Dry Valleys of Antarctica.

5.2.5 Discussion of VRML

The Virtual Reality Modelling Language (VRML) offers many benefits for creating a functional, interactive, and engaging learning environment for students. As with the others packages discussed above, the use of VRML for modeling terrain involves the three basic steps to visualization.

Data Conversion and Processing

Creating the VRML involved exporting the terrain from ESRI ArcScene 9.0 with a significantly reduced resolution since the intent was to minimize the file size for optimal rendering over the Internet. The file created in ESRI ArcScene was called

cutv3.tif and the specifications and a screenshot of the file before conversion is given in Appendix IV. It can be seen from the figure that the rendering resolution is set to below 50% to reduce the file size of the VRML model.

The rendered VRML was named LowResVRML which consisted of a .wrl file that was 14.9MB in size, and 36 accompanying .jpg image files that the VRML calls up to generate the scene. A capture of the files associated with the VRML folder is given in Appendix V, which shows that 16 of the 36 .jpg files contain pure terrain textures. The other 20 correspond to the coloured polygons that represent areas of interest, much like the ones used for NGRain.

A small file size is essential to maintain with a VRML especially when it is intended to be placed on a Web Page and run over the Internet. Testing with different resolutions of the McMurdo Dry Valleys exported to VRML using ESRI ArcScene 9.0, it was found that VRMLs made using the highest resolution resulted in a .wrl file that was 8,644 KB in size with 980 accompanying .jpg images. At the lowest resolution, it resulted in a .wrl file 1,716 KB in size with a single 20 KB JPG image.

Terrain Population with Information and Objects

Table 5.4 A sample of the 'anchor' node within VRML to make the coloured polygons interactive and link to the url provided.

```

}
  ]
    description "This is the Taylor Glacier"
    url "../HTML/TaylorGlacier.html"
  } #end anchor
]
}

```

Adding content and interactivity to the scene was made easier using the VRML Pad 2.0 editing package. Using this, the coloured polygons were made into 'anchors', which linked to outside

sources of information, in this case .html files. A sample of the VRML code for the

creation of anchors is provided in Table 5.4. This line of code would fit within a Shape node in a file like that shown in Chapter 4: Table 4.1.

Terrain Run-Time Viewing

In order to view VRML, a plugin must be downloaded and installed into the

user's computer. Of the several options available, the Cortona VRML plugin was used to view the LowRes.wrl file. Much like NGRain's Mobilizer discussed in Section 5.2.3, the link to a VRML file can be embedded within an HTML document and viewed in a standard web page. The anchors therefore link to other html files, pictures, and animations as can be seen in Figure 5.10. The VRML and text were incorporated into the prototype version of the Cybercartographic Atlas of Antarctica Project (Taylor and Pulsifer, 2004).

5.2.6 Evaluation of VRML

VRML, like NGRain makes an ideal platform for creating the McMurdo Learning Environment as can be seen at the above URL. The most challenging thing about working with VRML is determining the correct resolution for the terrain to improve the interactive quality. Through earlier trials, standard desktops were not able to handle the imagery and the resultant fly-throughs were ‘jumpy’ and ‘clunky’. Reducing the file size to the above specifications and reducing the resolution greatly helped in increasing the interactive quality, at the expense of visual quality, however.

Another difficulty in evaluation is the number of choices for VRML plugins available. Many, including Cortona offer an interface that is not intuitive to use and can be confusing to inexperienced users. The use of VRML and the Cortona plugin does however give a game-like quality to the environment which is striven for, given the game metaphor established in Chapter 2. The system requirements for Cortona are provided in Appendix VI. Additional work to improve the VRML has been done and is discussed below.

5.2.7 Additional Work with VRML towards creating the MLM

Since VRML provided the most straight-forward approach to terrain representation suitable for creation of the MLM, effort was made to make the terrain model more engaging with more interactive quality and increasing the image quality. The result is seen in Figure 5.11. In this figure the valley names are labeled in bright yellow text and geomorphologic features within the valleys such as a Rock Fall Cliff, Alpine Glacier, Glacier Lake, Melt Water Labyrinth, Outlet Glaciers, Valley Glacier, and Glacier Canyons are marked with large colourful banners. Specific Features are marked with a

sign including the Webb Glacier, Lake Vida, Upper-Write Glacier, Taylor Glacier, Lake Bonney, Canada Glacier, Commonwealth Glacier, and Ferrar Glacier. Two large billboards are present as well, giving view to Gargoyle Ridge, and a mummified seal. This VRML is included in the CD provided with the thesis. Click on cortvrml.exe to install the Cortona plugin, and then click VRML.html to view the MLM. By clicking the objects and labels listed above, information pops up about what they are.

Landmarks

The use and placement of landmarks was discussed in Chapter 2, and it was determined that they should be distinctive, concrete, visible at appropriate scales, align within a grid structure, align with each other, and follow paths. Figure 5.12 illustrates the placement and appearance of landmarks within the VRML. The grid structure appears as yellow lines that approximately follow the paths between landmarks, where the paths are the valleys themselves. The sections of the grid where the paths appear slightly askew to the main grid structure are shown in green instead of the yellow.

Figure 5.11. Improved VRML version of the MLM.

Landmarks include three-dimensional sign posts, billboards, and 3D objects as shown in Figure 5.13. In Figure 5.13 [1], a 3D sign post was made to resemble the rustic wooden signs seen in Antarctica. In the example provided, the student can click on the sign labelled 'Canada Glacier' and learn about that glacier as an example of an outlet glacier. Another landmark is the billboard shown in [2], where the actual image of a mummified seal is provided and placed within the VRML as an active link to more information about the mummified seal carcasses scattered within the Dry Valleys. What is

even more eye-catching is the use of the helicopter and active video link shown in [3]. Instead of a still image over the Wright Valley, a video plays live within the VRML using a rectangle with a movie texture (movie draped over the object), which in itself acts as a link to a full size version of the video showing a helicopter flying through a labyrinth of deep channels and gullies caused by catastrophic meltwater activity. The helicopter object was downloaded and is referenced as DCSEC, 2005.

Figure 5.13. Screenshots of the MLM with focus on the Upper-Wright Valley, to show the helicopter and live video landmark [3], a billboard of a mummified seal [2], and signs marking the outlet glaciers [1]. The resolution over the Upper-Wright Valley and the Outlet Glaciers is increased in these areas.

The other addition for the encouragement of discovery through serendipitous exploration of the terrain is through the use of sound. VRML provides the capability of incorporating a sound bubble within the VRML scene that is invisible. In the MLM a sound bubble surrounds the Don Juan Pond of the Wright Valley shown in Figure 5.14 where the salinity of the pond is so great that even at temperatures of -20 degrees Celsius,

the pond water remains liquid and flows. The sound bubble links to a sound file of running water, and as the user hits the boundary, they are led through sound to the landmark of the Don Juan Pond and thereby ‘discover’ the interesting phenomenon.

Interactive Quality versus Image Quality

As mentioned previously, the interactive quality of the VRML is significantly reduced with increases in resolution. One method of combating this is by incorporating both high and low resolution imagery into the same file. This is accomplished by creating an overall low resolution .wrl file that is 1,716 KB in size, with 4 JPG images; 20, 19, 20, 14 KB in size each. Three other separate VRML files were created using clipped sections of specific areas of terrain with the highest resolution possible for their representation. By overlapping and positioning the high resolution areas over the corresponding low resolution areas, the areas of particular interest to the study of geomorphology within the Dry Valleys can be seen at the highest resolution possible. The results of this endeavour

are visible in Figure 5.15. The three areas of high resolution show the Wright Valley, the Taylor Glacier, and the Canada and Commonwealth Outlet Glaciers. The effort then was to make a terrain visualization that has high interactive quality yet maintains a high visual quality by limiting the high resolution to specific areas.

5.3 Overall Conclusions

The use of all three forms of rendering platforms provides valuable insight into the nature of terrain rendering and the ability to create an interactive learning module with those tools. While VTP did provide some interesting visual representations of the LIDAR imagery, it was difficult to use, mainly because of the different files that needed to be kept track of and placed in the correct directories. One reason why NGRain was so easy to use, was that for all three applications involved in producing the final visualization, only one file, the .ngw file of 1.99MB was of concern for updating and modifying. The difficulty in converting data files and its not being specifically designed to handle terrain data are what made the use of NGRain a disadvantage. As well the Mobilizer plugin is not well known and unfamiliar to many people.

While VRML did not provide a sophisticated method for rendering the terrain, it provided the most productive environment for the creation of the MLM. Populating the VRML terrain with interactive multi-media objects is straight-forward, and there are many VRML plugins available.

Table 5.5 provides a summary of the differences between the products looking at ease of use, disadvantages, and advantages.

Table 5.5: Summary of comparison of ease of use between the different packages.			
	Ease of Use	Disadvantages	Advantages
VTP	Poor	-cumbersome file placements - lots of errors and bugs	- Open-source - Able to Customize Algorithms - Fully free
VRML	Good	- not sophisticated - common plugins	- Many common VRML Plug-ins.
NGRAIN	Fair	- unfamiliar plugin - Difficult data conversions	- One file for modification, used in all 3 applications. - Many tutorials and thorough documentation

Chapter 6: Discussion and Future Directions

6.1 Discussion

In this Chapter, elements from all chapters are brought together to illustrate how the guidelines and principles of design were used in implementing the geo-visualizations and creating the McMurdo Learning Module (MLM). This chapter also discusses directions for future research in the field of geo-visualization for expert users.

Visualization and Geo-science

Chapter 1 captured many of the issues and challenges of visualization in geo-science and the many technological components to visualization. Ottoson (2003) mentioned the use of data, graphics, and interfaces as essential components to visualization. Data provides the content and in the case of the thesis, the data were of high quality and high resolution LIDAR data from the McMurdo Dry Valleys. The graphics generated from the data were dependent on the rendering technology used by the specific applications and the desired quality of interaction. The interface depended on the specific package used as well. The interface for the Virtual Terrain Project, Enviro, was the least straight-forward to use, because management of the data within specific folders and directories was cumbersome, and appropriate selections on start-up needed to be made by the user. The NGRain Mobilizer interface is more straight forward with helpful instructions on how to navigate and interact with the three-dimensional model, however Mobilizer is a plugin specific to the NGRain Product, which is not widely known and may deter users from installing it. VRML is not limited to any specific interface. There are many freely available plugins to choose from, and VRML is a more well known form

of three-dimensional visualization compared to NGRain, making it more likely that a user will have experience operating VRML plugin interfaces and navigating through VRML worlds.

The three mechanisms to visualization identified by Ottoson (2003) are data acquisition, data management, and computation. The data for the thesis were acquired by the United States Geological Survey Polar Program and the data management involved was not too complex since there were only the two .tif files representing elevation and the image texture. The data were large however making it necessary to 'clip' them into more manageable sizes and lowering their resolution in the case of VRML. Computation was limited in the thesis since links to a database of information was not necessary in any of the cases.

Several issues in visualization technology were introduced including the five outlined by MacEachren and Kraak (2001).

1. **Computer Rendering Speed:** The computer processing for the visualizations was performed on two separate systems. But it was discovered that it did not matter which computer was used to make the visualization, it was the computer using the run-time viewer applications that was of concern. Computer Rendering Speed using the VRML visualization was severely compromised with moderate to high levels of resolution using a standard desktop.
2. **Integration and compatibility:** As alluded to in Chapter 5, the pre-processing and data conversions were the most challenging aspects to visualization using the specific packages and programs. The integration of the .tif files within the systems was not straight forward since issues of projection and file type compatibility

were of significant concern for Virtual Terrain Project, and NGRain, and to a lesser extent, VRML.

3. **User Interaction:** Although it was beyond the scope of the thesis to do formal user testing, as part of the Cybercartographic Atlas of Antarctica project a usability study was carried out by project affiliates in the Human Oriented Technology (HOT) Research Lab which included a study of the McMurdo Learning Module as it was seen in the first prototype of the atlas. This learning module was created using VRML, and the study found that: 1, it was not clear that the polygons used on the VRML to represent areas of interest were links to information; 2, a legend for the polygons rather than a text explanation would be more beneficial; 3, yellow polygons were difficult to distinguish; and 4, users had difficulty using the Cortona VRML plugin provided for them (Parush et al., 2005). The tool was not intuitive making it difficult to navigate and explore the map correctly. The use of different landmarks to replace the polygons was suggested, and implemented in the final version of the MLM. With respect to the interface provided with Cortona, little that can be done about that other than changing the settings, or finding a VRML plugin that is easier to use, for example the Bitmanagement 3D viewer, which is available at cost.
4. **Data and Information:** The complexity of data for the visualizations was manageable since there were only two files for the original data and visualization into the fourth-dimension was not performed.
5. **Visual Quality and Abstraction:** An important question was raised by MacEachren and Kraak (2001) of whether abstraction or realism is the ideal for

achieving insight through visualizations, where the level of abstraction depends on the intended audience (MacEachren et al., 1992). For the purposes of this thesis a medium level of abstraction was used because a certain level of realism is desired when looking at the characteristics of the ground to learn about geomorphology, however elements are added and the resolution modified to ease navigation, improve interactive quality, and stimulate interest for the student. The original LIDAR data of the McMurdo Dry Valleys was significantly clipped and reduced in resolution, especially in the creation of the VRML, to reduce the area of interaction and increase interactive capabilities for standard desktop computers. The low resolution of the VRML is appropriate for the student audience, where high-resolution areas were superimposed only over areas of particular interest. The issue of concealed aspects (Ottoson, 2003) arises through decisions of high versus low resolution areas where the decision rests on the terrain developer as to where those areas should be. The conflict of interest between visual representation and level of interactivity are what can undermine the scientific credibility of a visualization.

Location, Structure, Behaviour, and Procedural Knowledge for users within the virtual space, as outlined by Munro et al. (2002), were met through the use of metaphor, landmarks, paths, and goals set for the user. They all allow for maintaining a knowledge about her/his relative position, the relationship with objects, the interactivity with objects, and the understanding of the overall objective, which was to learn about the geomorphology of the Dry Valleys. Although design implementations were made to reflect the needs of the user, as stated by Gahegan (1999), user perception is largely

individualistic making it very difficult to evaluate the effectiveness of a virtual scene in carrying out those objectives.

Visualization for Education and Content Design for Serendipitous Exploration of Virtual Environments

Computer-based learning within geo-science in the form of Virtual Learning Environments (VLEs) and Virtual Field Trips (VFTs) provides the ability for students to ‘visit’ areas of the earth or beyond that would not be possible in reality. The McMurdo Dry Valleys provide a rich opportunity for students to learn about geomorphology, making the creation of a VLE or VFT for this area particularly beneficial. In Chapter two, it was observed that the computer-based methods did not improve the education for students, but did enhance the experience for students in a positive manner. Creating a VLE or VFT with a sense of presence in the environment is therefore important. One of the major concerns about VFTs according to Qui and Hubble (2002) is the loss of the serendipitous nature of discovery inherent to actual field trips. The design of the MLM was meant to encourage serendipitous exploration by not limiting the movement of user through the visualization, but providing landmarks and a ‘path’ to the landmarks in the form of the valley walls themselves, and through the use of the game metaphor. In addition, by incorporating a hidden element of sound, water running through the Wright Valley, students are attracted to the sound and ‘discover’ the Don Juan pond in a serendipitous manner.

LOD, Fractal, and Voxel-Based Terrain Rendering Techniques

A review of the technology behind visualization in the form of Level-of-Detail, Fractal, and Voxel-based terrain rendering provided the educational base for evaluation

of techniques used by the specific packages described in Chapter 4, and implemented in Chapter 5 for the creation of the MLM. The evaluation of LOD using Terrex demo data and Fractal-based rendering using Blueberry3d demo data proved an interesting analysis of the two methods for specific packages. Although neither were used in the creation of the MLM, they provided insight into the evaluation of terrain-rendering applications.

Discussion and evaluation of the Tools of Visualization

The three rendering packages used for the creation of the MLM were Virtual Terrain Project, VRML, and NGRain. VTP proved to be an effective choice in illustrating the method of Continuous LOD rendering, VRML for including interactive elements and representing terrain through elevation grids while also using LOD, and NGRain for presenting the terrain as a three-dimensional Knowledge Object using voxels.

One significant factor for the effective use of the packages is that of file management. VTP requires that several files be made and managed by placing them in specific folders and directories. The NGRain package provided the conversion of the three-dimensional model into a single .ngw file that is updated through all of NGRain's applications, making file management easy to handle. VRML is similar, although the texture files and objects do need to be in a specified folder for the .wrl file to call during rendering. File management is therefore less cumbersome in VRML, compared to VTP, with NGRain being the overall best of the three.

6.2 Future Directions

The intention of the thesis was to link techniques of terrain rendering to the design of a geo-visualization that accurately and interactively expressed the geomorphology of the McMurdo Dry Valleys in the form of a VFT or VLE. In doing so, other applications of geo-visualization became apparent for use by experts in collaborative decision-making and research environments or as a means of pattern recognition in large geo-databases. These topics were recognized as interesting and potentially viable directions for future research of geo-visualization. The following section provides a brief discussion for future research in those areas.

6.2.1 Visualization for Expert Knowledge Discovery and Collaboration

Visualizations were not always incorporated, nor accepted, within documents of the scientific community. In fact, pictures and diagrams that were included in scientific documentation were frowned upon, contributing to the obscure and specialized nature of the language of science (Heller, 2003). Although graphics were seen as “not necessary for those versed in analysis” (Heller, 2003), scientists today have largely changed their view, and use visualization as an effective tool of communication. Research has suggested that “we are on the cusp of a substantial increase in the role of maps, images, and computer graphics as mediators of collaboration – in a range of contexts including ... scientific inquiry, and education” (Brewer et al., 2000: 137). The following sections describe the use of geo-visualization as communication and knowledge discovery aids for experts to facilitate scientific inquiry.

6.2.2 Geo-visualization and Knowledge Discovery in Databases (KDDs)

The technologies of geo-visualization, Knowledge Discovery in Databases (KDD), and geo-computation have been largely disassociated (Gahegan et al., 2001). However, geo-visualization for use in KDDs, can act as a highly effective method of organizing large amounts of spatial data (MacEachren et al., 1999). Environmental data largely incorporates geo-referencing with both temporal and spatial components, but with the overwhelming amount of data available, they largely remain unexplored and the means in which they are stored, eventually become outdated (MacEachren et al., 1999). Geo-spatial data is rich in structure and patterns, and the role of a KDD is then to find structure and relations in large datasets to help establish meaning (Gahegan et al., 2001). This is accomplished in five stages including: data selection, pre-processing, transformation, data mining (i.e. pattern recognition), and interpretation (Gahegan et al., 2001).

In the search for patterns and relations among geo-referenced data, the process given as the four stages of scientific investigation outlined by DiBiase, and discussed in Chapter 1, is one of the first to provide a visual perspective on the process (MacEachren et al., 1999). The KDD process refines large datasets into manageable information through adding structure and relations among the data (Gahagan et al., 2001). As well, KDDs can simplify general stages into concrete tasks. For example, much like DiBiase's figure of scientific progression, Figure 6.1 shows striking similarity by illustrating the stages of progression in the scientific method along with scientific reasoning that are made at each stage including abduction (inference without pre-existing knowledge),

induction (inferences learned from examples), and deduction (reasoning through recognizing patterns and structure) (Gahegan et al., 2001).

The process begins with data exploration, to development of a hypotheses and the construction of knowledge about objects, rules, and categories so that analysis and presentation of final results can be realized (Gahegan et al., 2001). However, it is important to recognize that knowledge construction is not a process for determining the end result for the user, but rather a process for describing the operations and tools used for knowledge construction by the user (Gahegan et al., 2001).

The application of data exploration techniques in a geographic setting is termed by geographers as 'Exploratory Spatial Data Analysis (ESDA)' (Gahegan et al., 2001), and while in Chapter 1 it was discovered that scientists predominantly use visualizations in the exploration and presentation stages of scientific investigation, the use and

combination of visualization methods for data analysis and the interpretation of spatial features is shown in the following example.

Figure 6.2 illustrates visualization techniques for exploratory visualization using a) an elevation model with 3D data symbols, b) a visualization indicating relations between layers, c) a scatter plot, d) an intersection of two surface models, and e) a parallel coordinate plot indicating the distribution of values over multiple dimensions (Gahegan et al., 2001).

An important addition to this conclusion is that the perceived patterns observed by the user, and understanding of the data are a direct consequence of the exploratory visualization technique used, and how the visualization is constructed and rendered (Gahegan et al., 2001). This therefore has a profound effect on how data are interpreted, and the technique used should reflect the nature of the task or problem at hand.

6.2.3 Geo-visualization and Collaborative Virtual Environments

It is common for work using geo-spatial data to be completed through group activity, therefore providing experts with a means of communicating about the data through a ‘same-time/different-place collaborative geo-visualization environment’ (CVE) may prove useful (Brewer et al., 2000: 137). In terms of science the term ‘collaboratory’, established by William Wulf in 1993, is used to describe a work-place where the locations of the participants in group-work would not impede “interacting with colleagues, accessing instrumentation, sharing data and computational resources, and accessing information in digital libraries” (MacEachren, 2001: 435. From Cerf, 1993).

The importance of maps for geo-spatial discussion is emphasized as well by MacEachren, where he states that maps act as houses for information and facilitate communication (MacEachren, 2001). Three types of maps for CVEs include: 1, Annotation maps, which allow comments from users; 2, Argumentation maps, which facilitate participant negotiation and discussion; and 3, Alternative maps, which can illustrate alternative outcomes to problems. Discussion of problems using these forms of maps can be done by participants at different times (asynchronously). It has been found that asynchronous collaboration is preformed largely by geographers. Interestingly, synchronous collaboration is dominated by other disciplines. However, synchronous distant collaboration is important for geo-spatial discussion groups which support decision making and science (MacEachren, 2001).

Other work involving CVE includes collaborative work towards Disaster Situation Management where emergency situations depend on effective collaboration

among distant individuals to manage the situation, deliver the appropriate resources, and restore the situation to a non-emergency state (Baraghimian and Young, 2001).

Conclusion

Chapter 5 provided a detailed view of the use of the three specific packages for visualizing the McMurdo Dry Valleys LIDAR data. In all three cases, data conversion and processing, terrain population with information and objects, and terrain run-time viewing were common steps to the generation of terrain using those packages.

Figure 7.1 gives an overall impression of the three packages and their effective use through the three stages of development towards the creation of the McMurdo Learning Module (MLM). At the centre of the Figure, a picture of the LIDAR .tif image is shown. The surrounding ellipse describes the first stage of development of data conversion and processing. NGRain required the most effort, followed by VTP and then VRML. The population of terrain with objects and information is step two, and the figure shows that landmarks and metaphor were accomplished using NGRain and VRML, with VRML being ahead of NGRain in comparison. The third step, using a run-time viewing environment is seen to be most straightforward using the VRML application with several plugins to choose from that are free to download. NGRain follows with the freely available Mobilizer application, and VTP comes in last with the freely available Enviro application which is difficult to manage and cannot be incorporated or linked to HTML files. The Figure therefore shows that there is significant room for improvement within the VTP application to make a useful Virtual Learning Environment (VLE), followed by NGRain. VRML is shown to have the most potential for creating a VLE and is therefore deemed the most appropriate for creating the MLM. Room for growth is apparent

however considering the difficulty of use during run-time viewing for users in the usability study referenced in Chapter 6.

With regards to answering the thesis research questions initially posed in the

Introduction:

Which terrain rendering techniques and software are most suitable for generating a geo-visualization for use in communicating geo-scientific information while maintaining a successful balance between interactive quality and image quality?

How can different terrain representations be used and virtual environments designed to enhance geo-science education for students?

In response to the first question, the use of LOD, Fractal, and Voxel based rendering each offered their own advantages to the communication of geoscientific information. LOD, and Fractal-based rendering are the techniques that most suitably maintain a balance between interactive and visual quality. The successful balance between interactive quality and image quality were achieved in Virtual Terrain Project, by setting the triangle count and pixel error boundaries, in NGRAIN, by importing high resolution models, and in VRML, by incorporating different levels of resolution for specific areas. In comparison, VRML provided the most straight-forward method of incorporating specific areas of high resolution where the detail is considered conducive to learning about the geomorphology of those areas and for the McMurdo Dry Valleys as a whole.

In response to the second question, it was found that the proper use of landmarks and metaphor provided the best elements of design of VLEs and VFTs, and the VRML offered the best means in which to place interesting landmarks that incorporate multi-media objects into the scene with links to other sources of information, while encouraging serendipitous exploration through unrestricted movement through the scene and the use of sound to stimulate interest.

Although VRML was considered the best suited to create the MLM, the limitations to VRML cannot be overlooked. In particular, the plugins tend to be difficult to use and testing the resolution settings is necessary to ensure a high level of interactive quality since the trade-off between interactive and visual quality are pronounced using VRML.

Overall the thesis provided a detailed look into the theory and technology behind geo-visualization and offered insight into the creation of VLEs and VFTs using different packages with specific rendering capabilities. Future research may involve a change of user base, from students to expert users in the use of geo-visualization as a tool for collaboration and knowledge discovery in databases.

References

- Acevedo, Daniel., Vote, Eileen., Laidlaw, David H., and Joukowsky, Martha S. (2001). "Archaeological Data Visualization in VR: Analysis of Lamp Finds at the Great Temple of Petra, a Case Study". In proceedings of IEEE Visualization, October 2001. San Diego, California.
- Amand, Michelene K. St. (2003). "MÄK stealth demonstrates stealth support of Blueberry3d module for dynamically generated rich, detailed terrain". [http://www.blueberry3d.com/bb3d2002/newpages/downloads/BB3D_MPI_MAK.pdf]. (December 1, 2003). Viewed April 6, 2004.
- Baraghimian, Tony., and Young, Mark. (2001). "GeospacesTM – A Virtual Collaborative Software Environment for Interactive Analysis and Visualization of Geo-spatial Information". General Dynamics Electronic Systems. IEEE.
- Blueberry3D. (2004 a). Website [<http://www.blueberry3d.com/Blueberry3D/index.htm>]. Sjöland and Thyselius Virtual Reality Systems AB. Viewed April 6, 2004
- Blueberry3D. (2004 b). Whitepapers. "Introduction to the Technology of Blueberry3d". Blueberry3d Whitepapers. Sjöland and Thyselius Virtual Reality Systems AB.
- Bourke, Paul. (1991). "Fractal Landscapes". Swinburne University of Technology, Australia. [<http://astronomy.swin.edu.au/~pbourke/terrain/frachill/>]. Viewed March 18, 2004).
- Brewer, Isaac., MacEachren, Alan M., Abdo, Hadi., Gundrum, Jack., and Otto, George. (2000). "Collaborative Geographic Visualization: Enabling shared understanding of environmental processes". Information Visualization. IEEE Symposium. October, 2000. Pages:137–141.
- Caquard, Sébastien. (2001). "Des Cartes Multimedias Dans Le Debat Public: Pour Une Nouvelle Conception de la Cartographie Appliquee a la Gestion de L'eau". These Ph.D. Université Jean Monnet de Saint-Etienne. U.F.R. de Géographie. 19 Décembre 2001.
- Carey, Rikk., Bell, Gavin., and Marrin, Chris. (1997). "ISO/IEC 14772-1:1997, Virtual Reality Modeling Language, (VRML97)". The VRML Consortium Incorporated. [<http://www.vrml.org/Specifications/VRML97>]. Viewed August 1, 2005.
- Cartwright, W. (2004) "Engineered Serendipity: Thoughts on the Design of Conglomerate GIS and Geographical New Media Artefacts". Transactions in GIS. 8(1):1-12.

- Cartwright, W. (2005) "Exploring the use of a Virtual Map Shop as an Interface for Accessing Geographical Information". First International Workshop on Geographic Hypermedia. Denver, Colorado.
- Cerf, V.G., and committee. (1993). "National collaboratories: applying information technology for scientific research". National Academy Press. Washington, DC.
- Ching-Rong, Lin., Loftin, R.B., and Nelson, H.R. Jr. (2000). "Interaction with Geoscience data in an immersive environment". Virtual Reality. Proceedings. IEEE, March 2000, 18(22). Pages 55-62.
- Coddington, Paul. (1996) "Terrain Rendering". Terrain Rendering for a Geographical Information System. Northeast Parallel Architectures Center at Syracuse University.
[http://www.npac.syr.edu/users/paulc/lectures/terrain/p_lecture.html]
Viewed January 23, 2003.
- DCSEC. (2005). Helicopter VEML Object.
[http://www.cse.cuhk.edu.hk/~csc5460/task/prevWork/2001/USA_Gym/helicopter.rwl]. The Department of Computer Science and Engineering at the Chinese University of Hong Kong.
- Dean, K. L., Asay-Davis, X. S., Finn, E. M., Friesner, J. A., Imai, Y., Naylor, B. J., Wustner, S. R., Fisher, S. S., and Wilson, K. R. (2000). "Virtual Explorer: Interactive virtual environment for education". Presence:Teleoperators and Virtual Environments, 9(6): 505-523.
- de Berg, Mark., and Dobrindt, Katrin T. G. (1998) "On Levels of Detail in Terrains". Graphical Models and Image Processing. 60(1): 1-12.
- DiBiase, D. (1990). "Visualization in the Earth Sciences". Earth and Mineral Sciences. Bulletin of the College of Earth and Mineral Sciences, Penn State University 59(2): 13-18.
- Evans, Will., and Townsend, Gregg. (2003). "TopoVista Viewer". Source Code. Documentation. University of Arizona. Dept. of Computer Science.
[<http://www.cs.arizona.edu/topovista/>]. October 28, 2003. Viewed June 17, 2005.
- Fabrikant, S.I. (2005). "Towards and Understanding of Geovisualization with Dynamic Displays". Proceedings, American Association for Artificial Intelligence (AAAI). 2005 Spring Symposium Series: Reasoning with Mental and External Diagrams: Computational Modeling and Spatial Assistance. Stanford University, Stanford, CA. Mar. 21-23, 2005.
- Fine, G., and Deegan, J. (1996). "Three principles of Serendip: Insight, chance, and

- discovery in qualitative research.” *Qualitative Studies in Education*.
[<http://www.ul.ie/~philos/vol2/deegan.html>]. 9(4):15
- Fisher, Peter., and Unwin, David. (2002). “Virtual reality in geography: An introduction”. *Virtual Reality in Geography*, edited by Fisher, Peter and Unwin, David. Taylor and Francis, London and New York, 2002. ISBN: 0-7484-0905-X. Pages1-4.
- Gahegan, Mark. (1999). “Four barriers to the development of effective exploratory visualization tools for the geosciences”. *Int. J. Geographical Information Science*. 13(4): 289-309.
- Gahegan, M., Wachowicz, M., Harrower, M., and Rhyne, T.-M. (2001) “The integration of geographic visualization with knowledge discovery in databases and geocomputation”. *Cartography and Geographic Information Science* 28 (1): 29–44.
- Germanchis, T., and Cartwright, W. (2003). “The Potential to use Games Engines and Games Software to Develop Interactive, Three-Dimensional Visualizations of Geography”. *Proceedings of the 21st International Cartographic Conference*. August 2003. Durban, South Africa. ISBN: 0-958-46093-0
- Goodchild, M. (1992). “Geographical Information Science”. *Int. J. Geographical Information Systems*. 6(1): 31-45.
- Haklay, Mordechai E. (2002) “Virtual Reality and GIS: Applications, trends and directions”. *Virtual Reality in Geography*, edited by Fisher, Peter and Unwin, David. Taylor and Francis, London and New York, 2002. ISBN: 0-7484-0905-X. Pages 47-57.
- Hay, R. (August 2003). “Visualization and Presentation of Three Dimensional Geoscience Information”. *Proceedings of the 21st International Cartographic Conference*. August 2003. Durban, South Africa. ISBN: 0-958-46093-0.
- Heller, Eric. (2003) "The Power of the Image to Promote Science". From: Mike Silver and Diana Balmor. *Mapping in the age of Digital Media: The Yale Symposium*. Published in Great Britain. John Wiley & Sons Ltd. ISBN: 0470-85076-0. Pages 56-63.
- Hibbard, B. (1999). “Top Ten Visualization Problems”. *SIGGRAPH Computer Graphics Newsletter-VisFiles*. May 1999. [www.siggraph.org], viewed Oct. 2003.
- Howard, Anita. (Created 6 February 1995. Last Revised: 2004/11/30) “The Virtual Geography Department: Linking the discipline worldwide using the Internet and Worldwide Web”. University of Colorado – Boulder.
[<http://www.colorado.edu/geography/virtdept/contents.html>]. Viewed May 12, 2005.

- Hurni, L. (1997). "Interactive Vector-Based Topographic 3D maps". Institute of Cartography (IKA), ETH Zurich, Switzerland.
[<http://www.geod.ethz.ch/fe97/ika/ika.html>]. Viewed March 18, 2004.
- Hutton, Desmond., and Ray, Tiernan. (2004). "Duke it Out". Tech Weekly. The Ottawa Citizen. December 2, 2004. Section F.
- INTEC Americas Corp. (unknown). "Ortho-rectified Ikonos Imagery Now Available". [<http://www.intecameras.com/techbulletinIkonosOrtho.htm>]. Viewed April 4, 2004.
- Isdale, Jerry., Fencott, Clive., Heim, Michael., and Daly, Leonard. (2002). "Content Design for Virtual Environments". From: Stanney, Kay M. Handbook of Virtual Environments: Design, Implementation, and Applications. Lawrence Erlbaum Associates. Mahwah, New Jersey, London. Pages 519-532.
- Jelfs, A., and Whitelock, D. (2000). "The Notion of Presence in Virtual Learning Environments: What Makes the Environment 'Real'". British Journal of Educational Technology. 31(2):145-152.
- Kaas, Erik. (2004). "The NGRain Technology Difference Explained: A whitepaper for technical evaluators of visualization and simulation technologies". NGRain Corporation. [<http://www.ngrain.com/geo-spatial/index.html>]. Viewed June 17, 2005.
- Kraak, Menno-Jan. (2002) "Virtual Reality and GIS: Applications, trends and directions". Virtual Reality in Geography, edited by Fisher, Peter. Unwin, David. Taylor and Francis, London and New York. ISBN: 0-7484-0905-X. Pages 58-67.
- Kraak, Menno-Jan. (2003). "Geovisualization Illustrated". ISPRS Journal of Photogrammetry & Remote Sensing. Elsevier Science. 57: 390-399. Accompanying website: <http://www.itc.nl/personal/kraak/1812/>
- Kumler, Mark P. (1994). "An Intensive Comparison of Triangulated Irregular Networks (TINs) and Digital Elevation Models (DEMs)". Cartographica. 32(2): Monography 45.
- Lemke, Karen A. (1999). "Glacier National Park". University of Wisconsin-Stevens Point. The Virtual Geography Department Project.
[http://www.uwsp.edu/geo/faculty/lemke/gnp_vft/home.html]. Viewed May 12, 2005.
- Lindstrom, Peter., Koller, David., Ribarsky, William., Hodges, Larry F., Faust, Nick., and

- Turner, Gregory A. (1996). "Real-Time, Continuous Level of Detail Rendering of Height Fields". Siggraph.
- MacEachren, Alan M. (1994). "Visualization in Modern Cartography: Setting the Agenda". Visualization in Modern Cartography, edited by MacEachren and Taylor. Pergamon Press, UK. ISBN 0080424155. Chapter 1.
- MacEachren, Alan M. (2001). "Cartography and GIS: extending collaborative tools to support virtual teams". *Progress in Human Geography*. 25(3):431-444
- MacEachren, Alan M., Battenfield, Barbara., Campbell, Jim., DiBiase, David., and Monmonier, Mark. (1992). "Visualization". Geography's Inner Worlds, edited by Abler, R.F., Marcus, M.G. and J. Olson. Rutgers University Press. New Brunswick, N.J.. Chapter 6.
- MacEachren, A.M., Edsall, R., Haug, D., Baxter, R., Otto, G., Masters, R., Fuhrmann, S., and Qian, L. (1999). "Virtual environments for geographic visualization: Potential and challenges". *Proceedings of the ACM Workshop on New Paradigms in Information Visualization and Manipulation*, Kansas City. Pages 35-40.
- MacEachren, Alan M., and Kraak, Menno-Jan. (2001). "Research Challenges in Geovisualization". *Cartography and Geographic Information Science*. 28(1):3-12.
- MacEachren, Alan M., Wachowicz, Monica., Edsall, Robert., and Haug, Daniel. (1999). "Constructing knowledge from multivariate spatiotemporal data: integrating geographical visualization with knowledge discovery in database methods". *International Journal of Geographical Information Science*. 13(4):311-334.
- Martz, Paul. (1997) "Generating Random Fractal Terrain". Robert C. Pendleton. [<http://www.gameprogrammer.com/fractal.html>]. Viewed March 18, 2004.
- May, John. (1989). The Greenpeace Book of Antarctica: A new view of the seventh continent. Dorling Kindersley Limited, London. ISBN: 0-7715-9648-0. Pages 48-51.
- Moshell, Michael J., and Hughes, Charles E. (2002). "Virtual Environments as a Tool for Academic Learning". From: Stanney, Kay M. Handbook of Virtual Environments: Design, Implementation, and Applications. Lawrence Erlbaum Associates. Mahwah, New Jersey, London. Pages 893-909.
- Mueller, Klaus., Möller. , Torsten Möller, Swan, J Edward., Swan., Crawfis, Roger., Shareef, Naeem., and Yagel, Roni. Yagel. (1998). "Splatting Errors and Antialiasing". *IEEE Transactions on Visualization and Computer Graphics*. April-June 1998. 4:2.
- Munro, Allen., Breaux, Robert., Patrey, Jim., and Sheldon, Beth. (2002). "Cognitive

- Aspects of Virtual Environments Design". Stanney. Handbook of Virtual Environments Design, Implementation, and Applications, edited by Kay M. Stanney. Lawrence Erlbaum Associates, Publishers. Mahwah, New Jersey. Chapter 20. Pages 415-434.
- NGWTrans. (2004). "NGRAIN Transformer 2.1 Datasheet". NGRAIN Corporation. [<http://www.ngrain.com/solutions/pkm/products.html>.] Viewed June 17, 2005.
- Ottoson, P. (2003). "Three-Dimensional Visualization on the Internet". Maps and the Internet, edited by Peterson, Michael P. Amsterdam. London. Elsevier. ISBN 0080442013. Chapter 15. Pages: 247-269.
- ParallelGraphics. (2005). "Cortona Release Notes". ParallelGraphics. [<http://www.parallelgraphics.com/products/cortona/notes/>]. Viewed September 4, 2005.
- Parush, Avi., Khan, Shamima., Narasimhan, Sheila., Gauthier, Michelle., and Philip, Karen. (2005). "Overall Recommendations for the Cybercartographic Atlas of Antarctica: Usability Test Results". Human Oriented Technology Lab (HOTLab). Internal Project Report. June 19, 2005.
- Peterson, Michael P. (1999). "Active Legends for Interactive Cartographic Animation". *Int. J. Geographical Information Science*. 13 (4):375-383.
- Qiu, Weili., and Hubble, Tom. (2002). "The Advantages and Disadvantages of Virtual Field Trips in Geoscience Education". *The China Papers: Tertiary Science and Mathematics Teaching for the 21st Century*. 1:75-79.
- Rheingans, P., and Landreth, C. (1995). "Perceptual principles for effective visualisations". Perceptual Issues in Visualisation, edited by G. Grinstein, and H. Levkowitz. Berlin: Springer-Verlag. Pages 59-69.
- Rhyne, T.M. (2002). "Computer Games and Scientific Visualization". *Communications of the ACM*. 45(7): 41-44.
- Ritter, Michael. (1997). "Virtual Field Trip to the Indian Peaks, Colorado, Front Range, U.S.A.". The University of Wisconsin-Stevens Point. The Virtual Geography Department Project. [<http://www.colorado.edu/geography/virtdept/contents.html>.] Viewed May 12, 2005.
- Roehl, Bernie., Couch, Justin., Reed-Ballreich, Cindy., Rohaly, Tim., and Brown, Geoff. (1997). Late Night VRML 1.0 with Java: Unleash the power of VRML and Java. Ziff-Davis Press. Macmillan Computer Publishing USA. Emeryville, California. ISBN: 1-56276-504-3.
- Röttger, S., Heidrich, W., Slussallek, P., and Seidel, H-P. (1998). "Real-Time Generation

of Continuous Levels of Detail for Height Fields”. Proceedings of the 6th International Conference in Central Europe on Computer Graphics and Visualization. Pages 315–322.

- Schmidt, S., and Götze, HJ. (1998). “Interactive Visualization and Modification of 3D-Models using GIS-Functions”. *Phys. Chem. Earth*. Elsevier Science Ltd. 23(3): 289-295.
- Simpson, Josie. (2004). “An Introduction to three-dimensional Knowledge Objects White Paper”. A Product Knowledge Management White Paper. NGRain Corporation. [<http://www.ngrain.com/solutions/pkm/products.html>] Viewed June 17, 2005.
- Spicer, J. I., and Stratford, J. (2001). “Student perceptions of a virtual field trip to replace a real field trip”. *Journal of Computer Assisted Learning*. 17(4): 345-354.
- Stainfield, John., Fisher, Peter., Ford, Bob., and Solem, Michael. (2000). “International Virtual Field Trips: a new direction?” *Journal of Geography in Higher Education*. 24(2):225-262.
- Stoker, Jason. (May, 2004). “Voxels as a Representation of Multiple-Return LIDAR data”. ASPRS Annual Conference Proceedings. Denver, Colorado.
- Taylor, D.R.F., and Pulsifer, P. (Nov. 26, 2004). “The Cybercartographic Atlas of Antarctica Project”. Carleton University. [<http://www.carleton.ca/gcrc/caap/>]. Viewed June 18, 2005
- Terrex. “SmartMesh” Terrain Experts, INC. [<http://www.terrex.com/www/pages/technology/SmartMeshpage.htm>]. Viewed March 18, 2004.
- Thyselius, Rune. (2003). “Sjoland and Thyselius Virtual Reality Systems (STVTS) Announces Blueberry3d Terrain Editor v1.1 for Creator Terrain Studio and Blueberry3d Development Environment v.1.1 for Vega Prime”. Sjoland and Thyselius Virtual Reality Systems AB. [http://www.blueberry3d.com/bb3d2002/newpages/downloads/BB3D_v1_1_release.pdf]. Viewed April 6, 2004.
- Turner, Bryan. (2000). “Real-Time Dynamic Level of Detail Terrain Rendering with ROAM”. Gamasutra. CMP Game Group. [<http://www.gamasutra.com>]. Viewed June 13, 2005.
- USGS. (2002). “What Do Maps Show?” United States Geological Survey. [<http://interactive2.usgs.gov/learningweb/teachers/mapsshow.htm>]. Viewed March 18, 2004.

- Van Dam, Andries., Laidlaw, David H., and Simpson, Rosemary Michelle. (2002). "Experiments in Immersive Virtual Reality for Scientific Visualization". Computers and Graphics. Elsevier Science Limited. 26: 535-555.
- Vinson, Norman G. (1999). "Design Guidelines for Landmarks to Support Navigation in Virtual Environments". Proceedings of CHI'99, Pittsburg, PA. National Research Council of Canada. Institute for Information Technology.
- Watson, Margaret., Eggleston, Neil., Irby, Derek., Moorhead, Robert., and Evans, David. (2000). "A Virtual Reality Interface for Analyzing Remotely Sensed Forestry Data". SIGGRAPH 2000 Conference Abstracts and Applications, Catalog and CD-ROM, Sketches and Applications.
[<http://www.erc.msstate.edu/vail/projects/rstc/forestry/forest.htm>], viewed December 9, 2004.
- Wentz, Elizabeth A., Vender, Joann C., and Brewer, Cynthia A. (1999). "An Evaluation of Teaching Introductory Geomorphology using Computer-based Tools". Journal of Geography in Higher Education. 23(2):167-179
- Whitelock, D., Romano, D., Jelfs, A., and Brna, P. (2000). "Perfect Presence: What Does This Mean for the Design of Virtual Learning Environments?" Education and Information Technologies. 5(4):277-289.
- Woo, Andrew., and Halmshaw, Paul. (2004). "Myths and Truths of Interactive Volume Graphics". Interservice/Industry Training, Simulation, and Education Conference (IITSEC). Paper No. 1755.
- Woods, Birgit., Whitworth, Elizabeth., Hadziomerovic, Aida., Fiset, J.P., Dormann, Claire., Caquard, Sebastien., Hayes, Amos., and Biddle, Robert. (June 2005). "Repurposing a Computer Role Playing Game for Engaging Learning". ED-MEDIA– World Conference on Educational Multimedia, Hypermedia & Telecommunications.
- XBOX. [<http://www.emulationgalaxy.co.yu/xbox.htm>]. Viewed October, 2003.
- Xu, C., and Dowd, P.A. (2003). "Optimal construction and visualization of geological Structures". Computers and Geosciences. Elsevier Science Ltd. 29 (6): 761-773.
- Zwicker, Matthias., Pfister, Hanspeter., van Baar, Jeroen., and Gross, Markus. (2001). "EWA Volume Splatting". Proceedings of IEEE Visualization.

Appendix I

Original LIDAR Data of the McMurdo Dry Valleys of Antarctica

Appendix II

Projection and File specifications for the original USGS LIDAR data for the McMurdo Dry Valleys of Antarctica.	
Projection Information for dry_valleys.tif and 30m_elev.tif Projection: Stereographic_South_Pole Parameters: False_Easting: 0.000000 False_Northing: 0.000000 Central_Meridian: 0.000000 Standard_Parallel_1: -71.000000 Linear Unit: Meter (1.000000) Geographic Coordinate System: Name: GCS_WGS_1984 Angular Unit: Degree (0.017453292519943299) Prime Meridian: Greenwich (0.000000000000000000) Datum: D_WGS_1984 Spheroid: WGS_1984 Semimajor Axis: 6378137.000000000000000000 Semiminor Axis: 6356752.314245179300000000 Inverse Flattening: 298.257223563000030000	File Sizes Image File dry_valleys.tif: 253 MB (265,414,876 bytes) Elevation File 30m_elev.tif: 113 MB (118,761,028 bytes) Total: 366 MB

Appendix III

NGRAIN with the McMurdo terrain loaded into the Transformer application with Parts Information on the right, and polygons representing parts on the left.

System Requirements for NGRAIN (NGWTrans, 2004):

Random Access Memory (RAM): 512 MB

Processor: 1.3 GHz Intel Pentium IV or equivalent x86 code-compatible processor, or higher

Operating System: Microsoft Windows 2000 with SP4, or XP with SP1a

Other: Microsoft DirectX 8.1 or 9.0 Microsoft .NET Framework 1.0 SP2

Display: Screen area of 1024x768 or higher with 16-bit or 32-bit color.

Appendix IV

The file cutv3.tif as seen in ESRI ArcScene 9.0 before conversion into VRML. The resolution is set below 50% to reduce the file size for the terrain and make rendering over the Internet more manageable.

Specifications for the file cutv3.tif made for extraction in ESRI ArcScene 9.0 into the VRML format.

File: cutv3.tif.

File Size: 102.60 MB down from 251.77 for the original image file.

Extents are:

Top: 84841

Left: -40360

Right: 40760

Bottom: 18531

Appendix V

Thumbnails of the LowRes.wrl file and the associated .jpg images for VRML construction of the terrain.

Appendix VI

System Requirements for:

1. Virtual Terrain Project – ENVIRO run-time viewing application - [www.vterrain.org]

- Win32 machines with 3-D acceleration.

2. VRML – ParallelGraphics Cortona run-time viewing application

Current Cortona[®] VRML Client version is 4.2 (release 93) (ParallelGraphics, 2005)

- Operating System: Microsoft Windows 98/ME/2000/XP or Windows NT 4.0
- Internet Browser: Internet Explorer 4.0 or later version, Netscape Navigator 4.0 or a later version, Mozilla 1.0 or a later version, or Opera 7.0 or a later version.
- Computer: Pentium 90 MHz or better
- Random Access Memory (RAM): minimum of 16MB.
- Free Disk Space: 6MB of hard disk space for program files.
- Display: SVGA. The 800x600, high colour mode and higher.
- Input Devices: Mouse and keyboard, joystick optional.
- Sound Card: optional.

4. NGRAIN – Mobilizer run-time viewing application – [www.ngrain.org]

- Processor: 700 MHz Intel[®] Pentium[®] III or equivalent x86 codec compatible.
- Processor Memory: 128 MB RAM.
- Operating System: Microsoft[®] Windows 2000 Service Pack 4 or Windows XP Professional Service Pack 1a or Windows XP Tablet Edition Service Pack 1
- Display: 800x600 screen area; 16 or 32 bit colour depth.
- Browser: Microsoft[®] Internet Explorer 6.0 Service Pack 1
- Other: Microsoft[®] DirectX[®] 8.1 Microsoft[®] .NET Framework 1.1 Service Pack 1.