International Conference on Discrete Global Grids

Introduction

This conference, organized by the U.S. National Center for Geographic Information and Analysis (NCGIA) brought researchers together from many different disciplines to share advances in the rapidly developing fields of discrete global grids, global coordinate systems, and global georeferencing, and their applications. Participants were asked to submit position papers along with their application. The conference was held in the Radisson Hotel on the Santa Barbara waterfront from March 26-28, 2000.

This document contains:

- The original conference announcement
- The participant list
- All submitted abstracts (and full position papers where applicable)
- The conference agenda

Original Announcement

The ability to specify geographic location is fundamental to many areas of science and many human activities. In addition to latitude/longitude, horizontal/vertical datums and rectangular coordinate systems such as UTM, new approaches to tessellating planets -- often involving triangulations, some of which are hierarchical -- have been developed by researchers in a variety of domains to satisfy a range of scientific objectives. Papers describing some of these efforts have appeared, and conference sessions have been organized to discuss them, but no international conference has yet been convened to specifically address discrete global grids (DGG) either theoretically or from perspectives of applications they can serve. The meeting described in this Call for Papers is intended to fill this void, and all workers in DGG-related areas are strongly encouraged to participate. Many disciplines are interested in methods for gridding the curved surface of the Earth. They include:

- Statistics: sampling schemes and statistical models over the Earth's surface;
- Geographic information systems and science: georeferencing, data structures and indexing schemes for global data, global visualization, and Digital Earth;
- Remote sensing: consistent schemes for global imagery;
- Environmental modeling: finite difference and finite element schemes for solution of partial differential equations in global system modeling;
- Digital libraries: methods to support search, assessment, and retrieval of global geospatial and georeferenced data from distributed servers.

This conference will bring researchers together from many different disciplines to share advances in the rapidly developing fields of discrete global grids, global coordinate systems, and global georeferencing, and their applications. The program will include keynote presentations, contributed papers, demonstrations, and informal discussions.

The conference will be held in the Radisson Hotel on the Santa Barbara waterfront. It will begin with a reception and keynote presentation on Sunday evening March 26, and continue through Tuesday March 28, 2000. It is being organized by the Santa Barbara site of the U.S. National Center for Geographic

Information and Analysis, by a steering committee chaired by Michael Goodchild (UCSB), and including Noel Cressie (Ohio State), Geoffrey Dutton (Spatial Effects), Nick Faust (Georgia Tech), Ralph Kahn (JPL), Tony Olsen (EPA), Denis White (EPA), Hrvoje Lukatela (Geodyssey), and Jon Kimerling (Oregon State).

Papers are invited that address any aspect of global grids. Suitable topics might include, but are not limited to:

- Interoperability of and with global grids; compatibility issues for different grid schemes, and global grids as a medium of exchange for geospatial data.
- Data quality issues; what kinds of positional error and distortions do different schemes have, what are the consequences, and for whom and why do they matter.
- Application of discrete global grid systems in survey designs for environmental and natural resources.
- Alternative discrete global grid systems based on platonic polyhedra.
- Discrete global grids in atmospheric and ocean modeling.
- Spatial analyses using hierarchical structures in discrete global grids.
- Graphics and visualization based on discrete global grids.
- Efficient addressing schemes for hierarchical discrete global grids.

Participants

David Ashkenas
Informix Software Inc.

Mary Jo Brodzik
National Snow & Ice Data Center, CIRES
University of Colorado

Earl F. Burkholder
Surveying Engineering Dept
New Mexico State University

Keith C. Clarke
Dept of Geography
University of California, Santa Barbara

John Cloud
Dept of Geography
University of California, Santa Barbara

Peter H. Dana
Independent consultant

Geoffrey H. Dutton
Spatial Effects

Nickolas L. Faust
Electro-Optics, Environment & Materials Lab
Georgia Tech Research Institute
Georgia Institute of Technology

Ion Mateescu
Ctr. for Intl. Earth Science Information Network
Columbia University

Karen A. Mulcahy
Dept of Geography
East Carolina University

Val Noronha
Digital Geographic Research Corporation
Mississauga, ON
CANADA

Anthony R. Olsen
Natl. Health & Env. Effects Research Lab
ORD Western Ecology Division
U.S. Environmental Protection Agency

Dragan Petrovic
Environmental Systems Research Institute, Inc.

Robert G. Raskin
Jet Propulsion Laboratory
California Institute of Technology/NASA

Martin Reddy
Artificial Intelligence Center
SRI International

Michael W. Freeston
Alexandria Digital Library
University of California, Santa Barbara

Michael F. Goodchild
Dept of Geography
University of California, Santa Barbara

Matt J. Gregory
Dept of Geosciences
Oregon State University

Lee Hancock
Go2 Systems, Inc.

Jordan T. Hastings
Go2 Systems, Inc.

Linda L. Hill
Alexandria Digital Library
University of California, Santa Barbara

Ralph Kahn
Jet Propulsion Laboratory
California Institute of Technology/ NASA

A. Ross Kiester
Forestry Sciences Laboratory
U.S. Dept of Agriculture, Forest Service

A. Jon Kimerling
Dept of Geosciences
Oregon State University

Ken Knowles
National Snow & Ice Data Center
University of Colorado

Shohaku Ko
Center for Environmental Remote Sensing
Chiba University, Japan

Yvan G. Leclerc
Artificial Intelligence Center
SRI International

Hrvoje Lukatela
Geodyssey Limited

Marc Riedo
GIS Laboratory/Dept of Rural Engineering
Swiss Federal Institute of Technology

John Russell
Geodyssey Limited

Kevin M. Sahr
Dept of Computer Science
University of Oregon

Hanan J. Samet
Computer Science Dept
University of Maryland

Ashton Shortridge
Dept of Geography
University of California, Santa Barbara

Dennis R. Smith
Environmental Systems Research Institute, Inc.

Daniel L. Specht
Topographic Engineering Center
United States Army

Waldo Tobler
Dept of Geography
University of California, Santa Barbara

Tom Van Sant

Richard A. White
Earth System Science Center
Pennsylvania State University

Amy Yeh
ZipRealty.com

May Yuan
Dept of Geography
University of Oklahoma

Simon Zhang
University of California, Berkeley

Position Papers and Abstracts

Augmenting SAND with a spherical data model

By Houman Alborzi and Hanan Samet

Computer Science Department
University of Maryland
College Park, MD

SAND is a spatial database and information browser system developed at University of Maryland which supports 2D and 3D data models. In this abstract we describe our experience in adding a spherical data model to SAND which enables it to perform spatial queries on spherical data and to browse their results in an incremental manner. In particular, we focus on the issues that arise in adding a spherical data model to a database that has been built on the basis of the planar data model. Special emphasis is place on how the data structure were chosen for the task. We also describe the various geometrical primitives that were needed for modeling spherical data as well as the algorithms that were required in our implementation. As SAND is based on the quadtree representation, a natural method for adding spherical data to SAND was to find a spherical adaptation of the quadtree. We mapped the sphere into a quadtree representation by projecting the sphere onto a cube, and then constructed the quadtree on the faces of cube. Alternatively, it is also possible to map the sphere into a plane, and then simply use a planar quadtree. We discuss some of the differences between these approaches. We also describe the display models and visual query mechanisms used in spherical SAND to ease the task of navigating through the data. By adding a spherical sector primitive, we make it possible for users to locate data on the sphere which are around a great circle of the sphere. The same primitive allows users to query the database using a range of view from any point on the sphere.

..

National Snow and Ice Data Center
World Data Center for Glaciology

EASE-Grid

A Versatile Set of Equal-Area Projections and Grids

by *Mary J. Brodzik and Kenneth W. Knowles,* NSIDC

Introduction

The Equal-Area Scalable Earth Grid (EASE-Grid) comprises three equal-area projections, combined with an infinite number of possible grid definitions. It is based on a philosophy of digital mapping and gridding definitions that was developed at the National Snow and Ice Data Center (NSIDC), in Boulder, CO. This philosophy was used to implement a library of software routines, based on the assumption that a gridded data set is completely defined by two abstractions, the map projection and an overlaid lattice of grid points. The complete source code is available via ftp, and contains software to convert among many projections, but this paper will be restricted to an overview of the family of specific projections and grids that we have called the NSIDC EASE-Grid, or simply EASE-Grid.

The EASE-Grid is intended to be a versatile tool for users of global-scale gridded data, specifically remotely sensed data, although it is gaining popularity as a common gridding scheme for data from other sources as well. We begin with a short introduction to the abstractions used in NSIDC's generic mapping and gridding software, and proceed to the specific projections and grids that comprise the EASE-Grid family of grids. We include several short descriptions of the EASE-Grid used for different projects, as examples of the flexibility of the format. We conclude with an example of two EASE-Grid data sets that are used to study sea ice concentrations in Baffin Bay.

Defining a Gridded Data Set

NSIDC's mapping software is based on the assumption that a gridded data set is completely defined by the map projection and an overlaid lattice of grid points, often referred to as "cells". It is useful to think of the projection and the grid lattice as separate but related abstractions. The projection is simply a mathematical coordinate transformation of points on the curved surface of the Earth to points on a plane. The lattice of grid points can be imagined as a transparent piece of graph paper, overlaid on the plane of projection and anchored to it at a specified point.

Once the projection is chosen, any number of grid definitions can be used to describe the effect of changing the grid, or "graph paper," for the application at hand. For example, smaller graph cells can be used when a higher resolution is needed, or the size of the graph paper (number of columns/rows) can be reduced to study a subset area of the full projection.

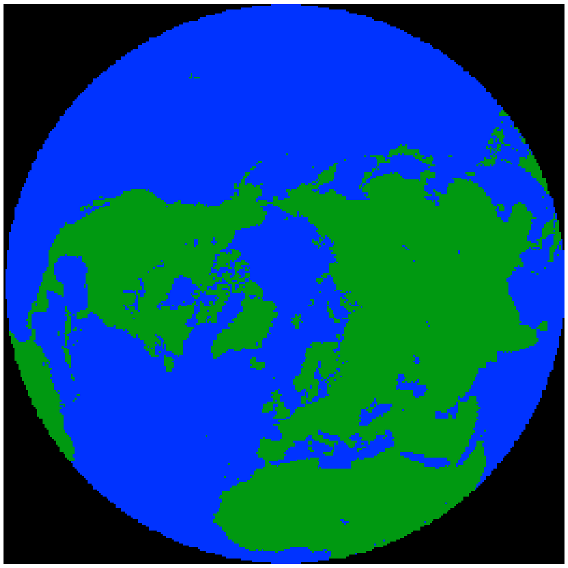
An array of gridded data, then, consists of one data element for each grid "cell" or lattice point. The user has complete flexibility to define the meaning of grid cell values, according to the most appropriate sampling technique for the data and application at hand. In many cases, particularly with remotely sensed imagery, it is important for a user to think of gridded data elements as values associated with the lattice points of the graph paper, rather than as associated with the "area of the grid cell." We have found the more general lattice point concept to be more useful when the data represent regularly sampled measurements in a continuous field, for example, passive microwave brightness temperatures, or visible wavelength radiances. In the sensor swath space, the data value is simply a sample measurement from the continuous field, and may or may not have any physical relationship to the size and shape of the eventual grid "cell" surrounding the eventual lattice point in the regularly gridded data array. For example, although the sampling interval of a 19 GHz scanning passive microwave radiometer might be 25 km, the effective field of view of the antenna might be an elliptical area, 40 km x 60 km. The eventual gridded data array might be defined on a 25 km grid, with data element values chosen from the nearest neighbor of the latest swath. In this case, the data element still represents a 40 km x 60 km brightness temperature.

It is entirely up to the data set producer to define the meaning of a gridded data element during the sampling process. The process of sampling data to the regular grid is sometimes referred to as binning. Examples of binning techniques include averaging all data that falls into a given cell, or taking a maximum, minimum, median or latest value in the cell. The producer might choose to involve the area and shape of the cell in the definition of a data element, but is certainly not required to do so. Alternative methods include nearest neighbor, bilinear interpolation, or otherwise weighted averages of surrounding data samples.

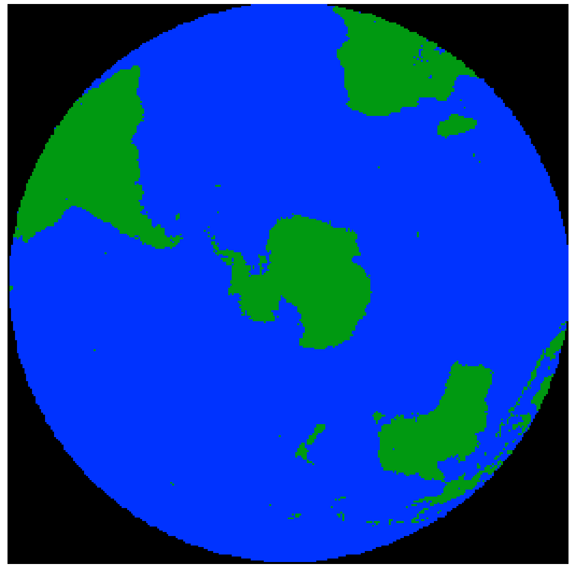
EASE-Grid Map Parameters

The EASE-Grid was originally developed at NSIDC for the data products generated by the Special Sensor Microwave Imager (SSM/I) Level 3 Pathfinder Project, which includes gridded passive microwave brightness temperature and related geophysical products derived from the brightness temperatures at a relatively coarse 25 km resolution. However, given the flexibility of an infinite number of grid definitions for the EASE-Grid projections, the format has since been adopted by a number of other projects, with grid resolutions ranging from 1.25 km to 250 km. These include the TIROS-N Operational Vertical Sounder (TOVS) and Advanced Very High Resolution Radiometer (AVHRR) Polar Pathfinders, the AARI (Arctic and Antarctic Research Institute, St. Petersburg, Russia) Sea Ice data, the Arctic Climatology Project Arctic Meteorology and Climate Atlas, and NSIDC’s EASE-Grid versions of the Global Land Cover Classification (GLCC) data and the International Permafrost Association Permafrost and Ground Ice Map.

The three EASE-Grid projections comprise two azimuthal equal-area projections, for the Northern or Southern hemisphere, respectively, and a global cylindrical equal-area projection.



Northern Hemisphere EASE-Grid projection



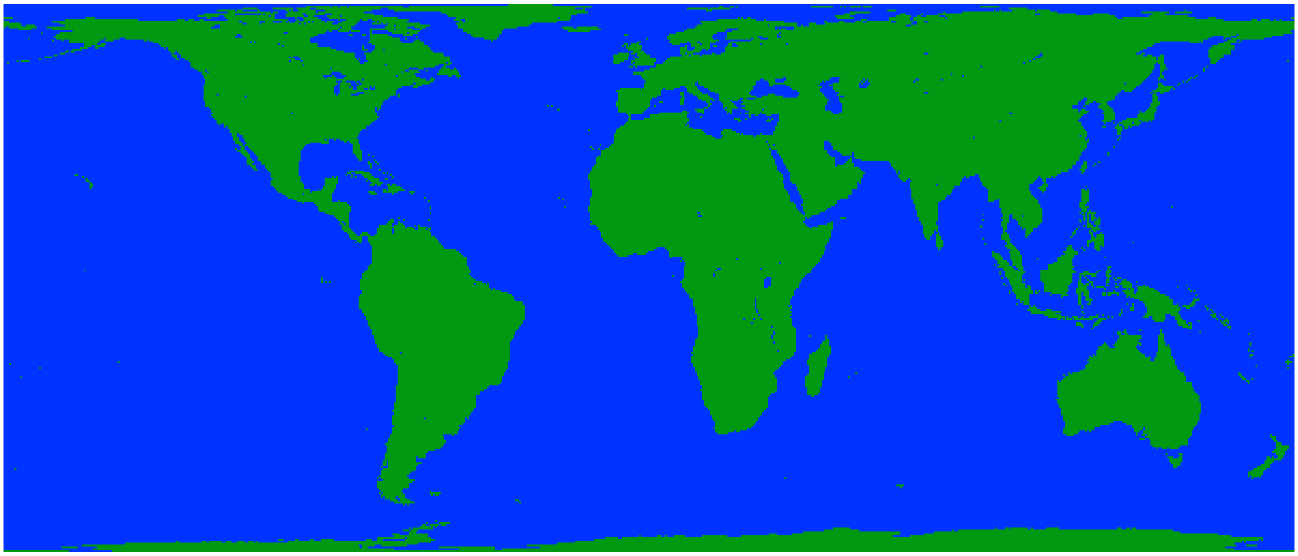
Southern Hemisphere EASE-Grid projection

The North azimuthal equal-area map is defined as

- r = 2*R/C * sin(lambda) * sin(PI/4 - phi/2) + r0
- s = 2*R/C * cos(lambda) * sin(PI/4 - phi/2) + s0
- h = cos(PI/4 - phi/2)
- k = sec(PI/4 - phi/2)

The South azimuthal equal-area map is defined as

- r = 2*R/C * sin(lambda) * cos(PI/4 - phi/2) + r0
- s = -2*R/C * cos(lambda) * cos(PI/4 - phi/2) + s0
- h = sin(PI/4 - phi/2)
- k = csc(PI/4 - phi/2)



Global EASE-Grid projection

The cylindrical equal-area map is defined as

- r = r0 + R/C * lambda * cos(30)
- s = s0 - R/C * sin(phi) / cos(30)
- h = cos(phi) / cos(30)
- k = cos(30) / cos(phi)

The cylindrical equal-area map is defined as true at 30N/S.

where:

- r = column coordinate
- s = row coordinate
- h = particular scale along meridians
- k = particular scale along parallels
- lambda = longitude in radians
- phi = latitude in radians
- R = radius of the Earth = 6371.228 km
- C = nominal cell size
- r0 = map origin column
- s0 = map origin row

Both projections are based on a spherical model of the Earth with radius R = 6371.228 km. This radius gives a sphere with the same surface area as an ellipsoid using the International Datum.

The values of C, r0 and s0 are determined by the grid that is chosen to overlay the projection.

Why "equal-area" maps?

Discussion of map projections is often unnecessarily lengthy and sidetracked by disregard for the fact that there is no one best map projection. Each projection has different properties and thus different "best" uses. Sometimes the question is raised as to why we chose equal-area projections over the other possibilities for the EASE-Grid, and the answer relies on a basic understanding of projection characteristics, in the context of our original application: we were seeking minimal distortion over hemispheric and global scales.

"Two of the most important characteristics of maps are whether they are conformal or equal-area. No map projection is both, and some are neither" (Knowles, 1993). On equal-area maps, a small circle placed anywhere on the map will always cover the same amount of area on the globe, and, at any point on the map, the product of the scale h along a meridian of longitude and the scale k along a parallel of latitude is always one. The aspect ratio k:h is a measure of shape distortion.

For the Northern and Southern hemisphere EASE-Grid projections, the aspect ratio varies from 1:1 at the pole to 1.17:1 at 45N and increases to only 2:1 at the equator. For the global EASE-Grid projection, the aspect ratio varies more widely (see details in the following table). The selection of +/-30 for the standard parallels of the cylindrical projection gives a map with minimum mean angular distortion over the continents. This projection is intended for the study of parameters in the mid- to low-latitudes.

Aspect ratios (a measure of shape distortion) of the EASE-Grid projections:

Azimuthal Equal-Area			**Cylindrical Equal-Area**	
latitude	**k/h**		**latitude**	**k/h**
90	1.00		80	24.90
75	1.02		75	11.20
60	1.07		60	3.00
45	1.17		45	1.50
30	1.33		30	1.00
15	1.59		15	0.80
0	2.00		0	0.75

In contrast, on conformal maps, angles within a small area are reproduced accurately, so a small circle on the globe will look like a small circle on the map. At any point on the map, the scale h along a meridian of longitude is equal to the scale k along a parallel of latitude, and hk - 1 is a measure of areal distortion. For example, NSIDC produces other polar gridded data products using a polar stereographic map true at 70N. The projection is a conformal map. By definition, the aspect ratio remains 1:1 everywhere, however, the areal distortion of this map varies from -6% at the pole to +29% at 45N and increases to +276% at the equator.

Areal distortion of the Polar Stereographic map true at 70N:

Polar Stereographic, (true at 70N)	
latitude	**kh - 1**
90	-6%
45	29%
0	276%

A very popular map that is neither equal-area nor conformal is the cylindrical equidistant map, also known as the "lat-lon grid." This map suffers from both areal and shape distortion, as follows:

	Shape Distortion	**Areal Distortion**
latitude	**k/h**	**kh - 1**
89	57	5630%
80	6	476%
60	2	100%
45	1.4	41%
0	1	0%

In summary, given the choices of either shape distortion or areal distortion or both, we decided in favor of the equal-area projections for the EASE-Grid because they minimized the amount of distortion over the hemispheric and global scale we were attempting to portray. One convenient side effect of this choice is that calculations of areal statistics are reduced to simply summing pixels and multiplying by a constant area per pixel, so the acronym, "EASE-" takes on a secondary meaning, as in "easy to use."

Why a Spherical Earth Model?

Another question that is sometimes raised is why we chose to use a spherical earth model over an elliptical model, and how much "error" this introduces in the gridding geolocation. The answer is that no error is introduced by this model choice.

Representation of the gridded data as a fixed array of values is accomplished with a set of equations to map from geographic coordinates (latitude, longitude) to grid coordinates (column, row). In this sense, the location (column and row) of each grid "cell" can just be considered an entry in a look-up table, i.e. a place to store the data (Brightness Temperature, albedo, time stamp, etc.) for a specific, implicitly defined, geographic location. As long as the transformation back from grid coordinates (column,row) to geographic coordinates (latitude, longitude) is performed with the inverse transformation that uses the same Earth model, there is no error introduced by using a spherical Earth model. Choice of an elliptical model would only slow down the transformation calculations, (geographic to grid and back), with no gain in accuracy.

EASE-Grid Family of Grid Definitions

A grid is always defined in relation to a specific map projection. It is essentially the parameters necessary to define a rectangular coordinate system overlaid on a flat map and anchored to it at the map origin. The following four elements completely describe a grid:

- the map projection
- the numbers of columns and rows
- the number of grid cells per map unit (the map unit is part of the projection parameters)
- the grid cell coordinates of the map's origin

The EASE-Grid family of grid definitions includes, but is not limited to, the following specific grids.

The Original SSM/I Grids

The original 25 kilometer grids were defined for the data products generated by the SSM/I Level 3 Pathfinder Project at NSIDC. These grids have a nominal cell size of 25 km x 25 km. A slightly larger actual cell size C=25.067525 km was chosen to make the full global (ML), 25 km grid exactly span the equator, and was then used for all three projections for the sake of data product consistency. Of course, few cells actually have these dimensions, but they all have the same area.

By convention, grid coordinates (r,s) start in the upper left corner, at cell (0,0), with r increasing to the right and s increasing downward. Rounding the grid coordinates up at .5 yields the grid cell number. A grid cell is centered at grid coordinates (j,i) and bounded by: $(j - .5) <= r < (j + .5)$ and $(i - .5) <= s < (i + .5)$.

The 25 km hemispheric grids for the North and South azimuthal projections (aka "NL" and "SL") are defined with 721 columns, 721 rows, and the respective pole anchored at cell (360.0,360.0). The ML grid for the cylindrical projection is defined with 1383 columns, 586 rows, and is defined with the point where the equator crosses the prime meridian at cell location (691.0,292.5).

For each 25 km grid, the set of corresponding 12.5 km grids was defined such that the grid coordinates are coincident (aka. "bore-centered") and exactly double the lower resolution grid coordinates. The ML grid is symmetrical about the prime meridian, but the MH grid is not. The 25 km ML grid exactly spans the equator, from 180 W to 180 E, with 1383 grid cells. The 12.5 km grid, (aka MH), also exactly spans the equator, with 2766 grid cells. However, since the center of the ML column 0 is coincident with the ML column 0, the western edge of the Mh grid cell in column 0 row 293 (at the equator) is slightly east of 180 W, and the eastern edge of the Mh grid cell in column 2765 is slightly east of 180 E.

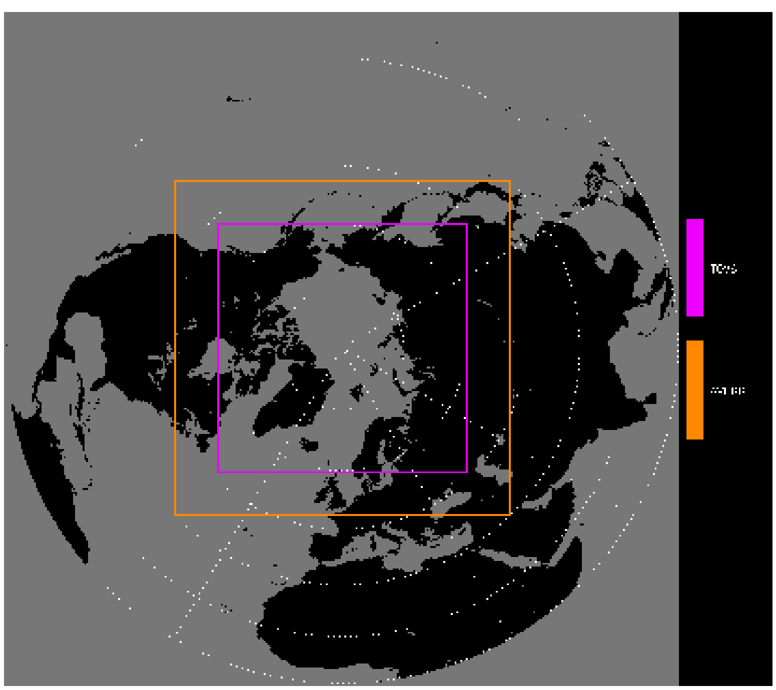
The dimensions, center, and extent of the original SSM/I grids are summarized below. It is important to remember that there is nothing specific to the SSM/I data in these definitions. If these grid definitions are considered appropriate for another data set, they can be used with no changes.

Original 25 and 12.5 km Grids									
Grid	Dimensions		Map Origin		Map Origin		Grid Extent			
Name	Width	Height	Column (r0)	Row (s0)	Latitude	Longitude	Minimum Latitude	Maximum Latitude	Minimum Longitude	Maxim Longit
ML	1383	586	691.0	292.5	0.0	0.0	86.72S	86.72N	180.00W	180.
MH	2766	1171	1382.0	585.0	0.0	0.0	85.95S	85.95N	179.93W	180.
NL	721	721	360.0	360.0	90.0N	0.0	0.34S	90.00N	180.00W	180.
NH	1441	1441	720.0	720.0	90.0N	0.0	0.26S	90.00N	180.00W	180.
SL	721	721	360.0	360.0	90.0S	0.0	90.00S	0.34N	180.00W	180.
SH	1441	1441	720.0	720.0	90.0S	0.0	90.00S	0.26N	180.00W	180.

Other Grid Definitions in the EASE-Grid Family

The Polar Pathfinders

Users of the NSIDC EASE-Grid are not limited to the grid orientation, size and resolution described above, and are free to define grids that are more appropriate to a given data set. For example, the TOVS Polar Pathfinder data were defined with the EASE-Grid Northern hemisphere map projection parameters, and a polar subset of the original hemisphere at a 100 kilometer resolution. The AVHRR Polar Pathfinder data were defined for both Northern and Southern hemisphere maps, as subsets of each, at 1.25 km, 5 km, and 25 km resolutions. The figure below shows the Northern hemisphere grid extent for SSM/I (the full hemisphere), TOVS Polar, and AVHRR Polar grids (respective subsets).



Relative Northern hemisphere grid extents of Polar Pathfinders

(SSM/I (full hemisphere), AVHRR and TOVS).

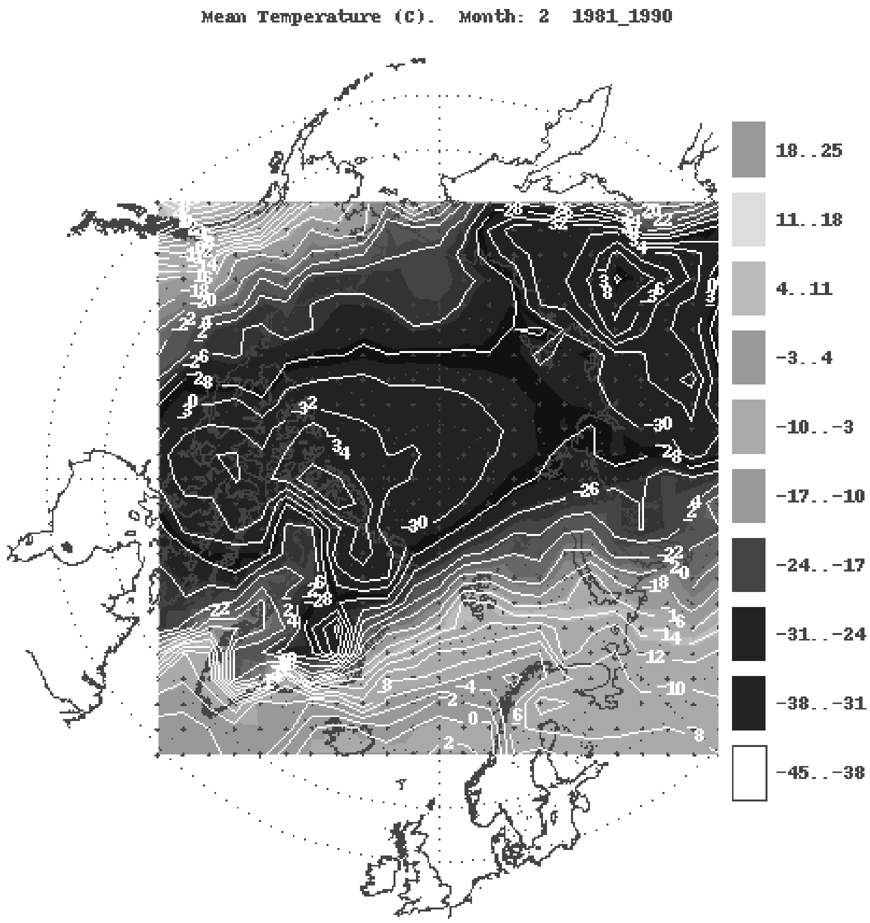
The AARI Sea Ice Data in EASE-Grid

The AARI EASE-Grid sea ice data provide another example. These data did not require hemispheric coverage, but the data set producers at NSIDC wanted to provide them in a grid that would facilitate intercomparison with sea ice data derived from SSM/I. Therefore the AARI EASE-Grid was defined to be the subset of the SSM/I Pathfinder NH grid (Northern hemisphere, 12.5 km resolution) defined by columns 360 through 1080 and rows 360 through 1080. The resulting AARI EASE-Grid is 721 columns and 721 rows. This, in turn, relates the AARI EASE-Grid definition to the 25 km AVHRR EASE-grid (aka "NA25") subset via the following simple relationship:

- AARIcolumn = 2 * NA25column
- AARIrow = 2 * NA25row

The Arctic Climatology Project Arctic Meteorology and Climate Atlas in EASE-Grid

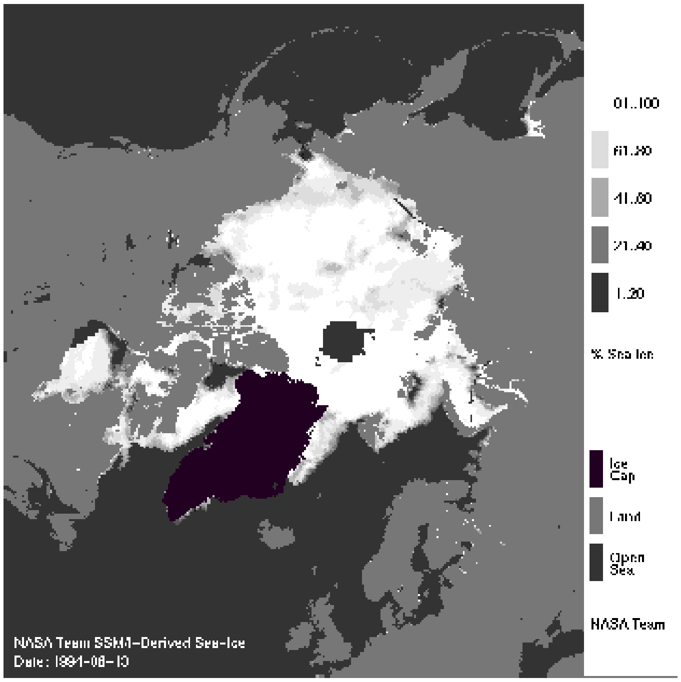
NSIDC has produced an atlas of Arctic meteorology and climatology under the auspices of the U.S.-Russian Joint Commission on Economic and Technological Cooperation’s Environmental Working Group (EWG). The gridded fields produced for this atlas are defined for a subset of the full northern hemisphere above 65N, at a 250 km grid spacing. Data arrays are 23 columns by 23 rows. An example temperature field is included below.



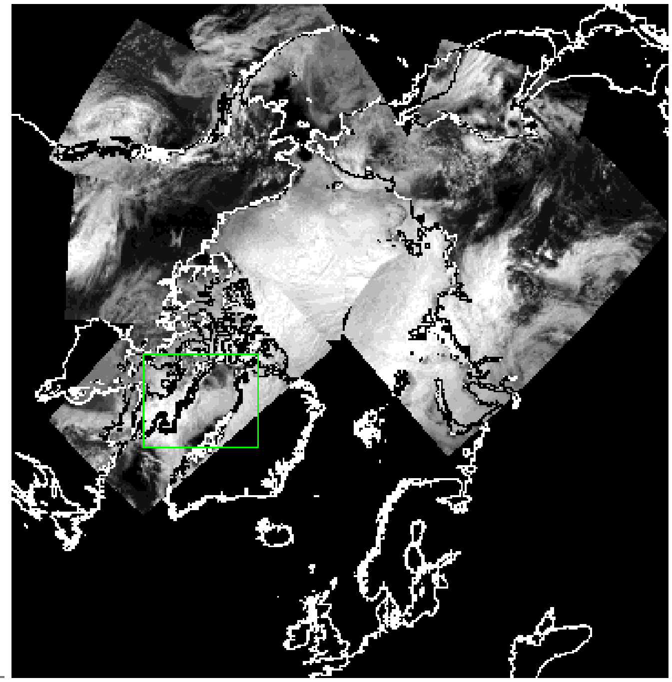
Mean air temperature (degrees Centigrade), February, 1981 - 1990.
(Dots indicate the EWG EASE-Grid lattice points.)

An Example Application

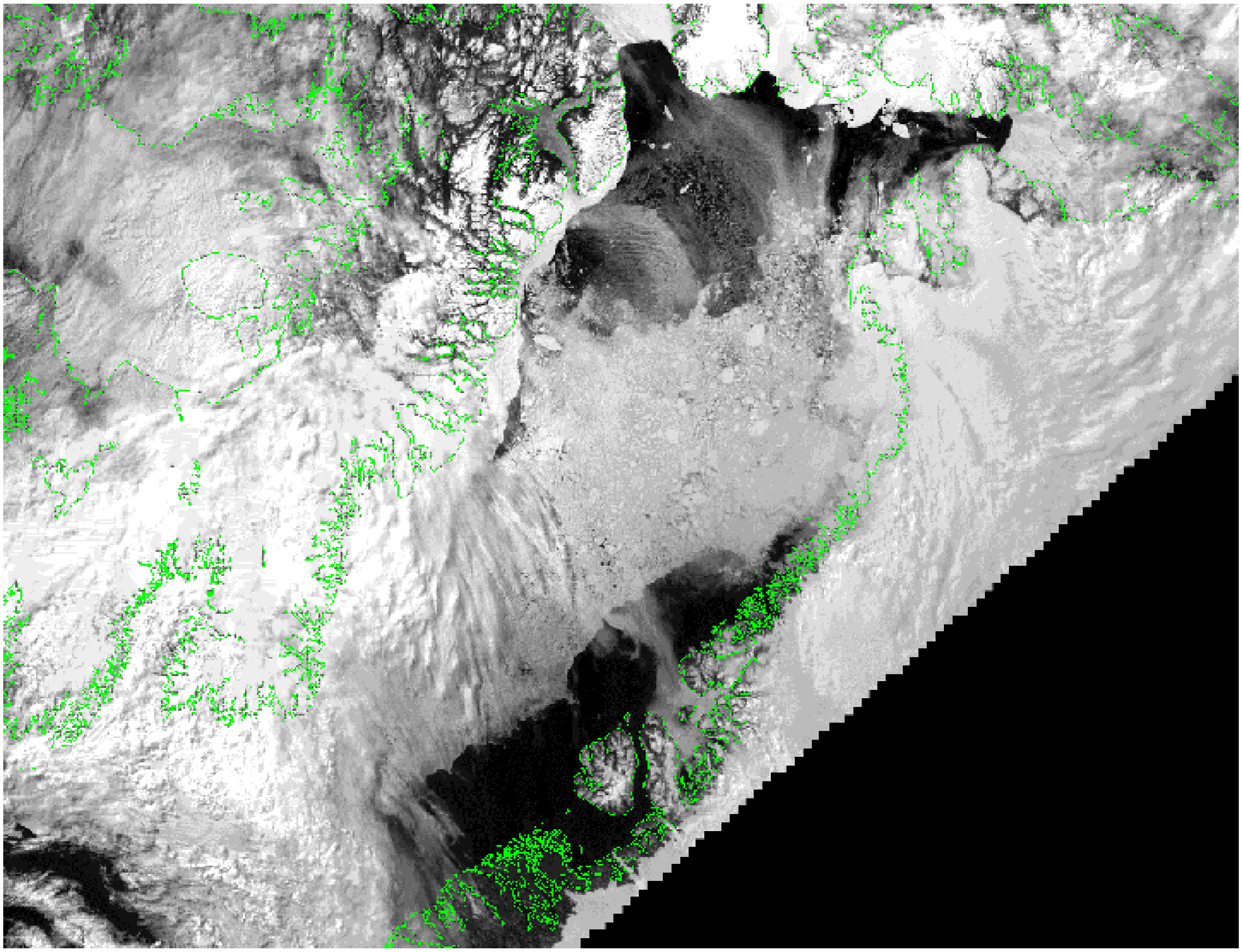
The following series of images illustrates a comparison of SSM/I-derived sea ice in Baffin Bay with AVHRR-derived albedo for June 13, 1994. Since all images are derived from various resolution grids of the same NL EASE-Grid projection, the reader can easily visually compare the ice edge. Digital comparison and analysis are likewise easily performed using the user's favorite graphical analysis package.



Arctic 25 km Sea Ice Concentration
Derived from SSM/I, June 13, 1994.



Arctic 25 km AVHRR Albedo Product
Browse Image, June 13, 1994.



Zoomed area of Baffin Bay, AVHRR Albedo Product Image, full resolution (1.25 km), June 13, 1994.

Conclusions

While originally intended for use with a single data product, the EASE-Grid has proven to be flexible and extensible to other global, gridded applications. The projection and gridding abstractions are simple and easy to apply to the requirements of a new data set. Data from diverse sources can be resampled and expressed as digital arrays of varying resolutions, which are defined in relation to one of three possible projections. Storage as a simple digital array facilitates portability and ability to be imported into a user's favorite analysis package. Users find that visualization and intercomparison operations are then greatly simplified, and that the tasks of analysis and intercomparison can be more readily accomplished.

References

- Arctic Climatology Project. 2000. Environmental Working Group ARctic meteorology and

Climate Atlas. Edited by F. Fetterer and V. Radionov. Boulder, CO: National Snow and Ice Data Center. CD-ROM.

- Armstrong, R. L. and M. J. Brodzik. 1995. An Earth-Gridded SSM/I Data Set for Cryospheric Studies and Global Change Monitoring. Adv. Space Res. 16(10):155-163.

- Armstrong, Richard, M. J. Brodzik, A. Varani, 1997. "The NSIDC EASE-Grid: Addressing the need for a common, flexible, mapping and gridding scheme." Earth System Monitor, Vol. 7, No. 4, 3 pp.

- Knowles, Kenneth W. 1993. Points, pixels, grids, and cells -- a mapping and gridding primer. <http://www-nsidc.colorado.edu/NASA/GUIDE/docs/reference_documents/ppgc.html>.

- National Snow and Ice Data Center. AARI 10-Day Arctic Ocean EASE-Grid Sea Ice Observations. <http://www-nsidc.colorado.edu/NSIDC/CATALOG/ENTRIES/nsi-0050.html>.

- National Snow and Ice Data Center. AVHRR Polar 1 km Level 1b Data Set. <http://www-nsidc.colorado.edu/NASA/GUIDE/docs/dataset_documents/advanced_very_high_res

- National Snow and Ice Data Center. NOAA/NASA Pathfinder EASE-Grid Brightness Temperatures. <http://www-nsidc.colorado.edu/NSIDC/CATALOG/ENTRIES/nsi-0032.html>.

- National Snow and Ice Data Center. TOVS Pathfinder Path-P Daily and Monthly Arctic Atmospheric Grids. <http://www-nsidc.colorado.edu/NSIDC/CATALOG/ENTRIES/nsi-0027.html>.

- National Snow and Ice Data Center. Complete source code for mapping and gridding software. <ftp://baikal.colorado.edu/pub/NSIDC/>.

The Global Spatial Data Model (GSDM)

Earl F. Burkholder, PS, PE
Surveying Engineering Department - 3SUR
New Mexico State University - Las Cruces, NM 88003

International Conference on Discrete Global Grids
U.S. National Center for Geographic Information and Analysis
Santa Barbara, CA - March 26-28, 2000

Abstract

Spatial data are 3-dimensional (3-D) and modern measurement systems collect data in a 3-D environment. Computer data bases store 3-D digital spatial data. Human perception of spatial relationships is primarily visual and, due to gravity, our natural reference for spatial experience is horizontal (2-D) and vertical (1-D). Various models are used to establish a conceptual connection between the measurements, digital spatial data, and its representation - data visualization. Digital spatial data are also used to make analog products such as maps, charts, and other hardcopy diagrams. The point is, once spatial data are put into digital form, they can be manipulated at the whim of the user. To preserve the integrity and value of spatial data, there should be a common storage format and the data manipulation processes should be well-defined, unique, bi-directional, and three-dimensional. Of course, the challenge is to identify a common format for 3-D data which is at the same time simple, rigorous, and global. The data storage format must also accommodate reliable statistical measures of spatial data accuracy.

The global spatial data model (GSDM) described in this presentation is a collection of concepts and procedures which can be used to collect, organize, store, process, and manipulate 3-D spatial data. The GSDM uses one set of solid geometry equations which are equally applicable around the world. This simple standard preserves global interoperability and each discipline, agency, corporation, or individual spatial data user is free to implement any derivative use or application.

The GSDM includes both a functional model and a stochastic model. The functional model encompasses the geometry of spatial relationships and the stochastic model defines the process for establishing, tracking, and reporting the accuracy of spatial data using existing standard mathematical procedures.

Introduction

An appropriate introduction to the global spatial data model (GSDM) might be to list several challenges facing geospatial data users. According to Vice President Al Gore (1998) in a speech titled, The Digital Earth: Understanding our planet in the 21st century, "the hard part of taking advantage of this flood of geospatial information will be making sense of it - turning raw data into understandable information." The following challenges all come under the umbrella of Gore's statement, but are listed separately for purposes of this discussion. Challenges for spatial data users include:

1. Handling vast amounts of 3-dimensional (3-D) digital spatial data efficiently.

2. Describing spatial data accuracy without ambiguity.

3. Finding the best (appropriate) combination of tools, talents, and resources to accomplish the task at hand.

The third challenge is very open ended and relies heavily on the judgement of the user but the first two challenges are addressed specifically by the GSDM and supported by an underlying BURKORD® database (see www.zianet.com/globalcogo). The functional model portion of the GSDM includes geometrical equations which permit each user to work with local rectangular (flat earth) coordinate differences while preserving true geometrical integrity on a global scale. The stochastic model portion of the GSDM includes rigorous error propagation procedures which accommodate input of measurement uncertainties and provide output of standard deviations for each 3-D coordinate position and/or other derived quantities. It has been said, "No job is difficult if you have the right tools." The goal of this paper is to examine features of the GSDM with the idea of finding better tools for handling digital spatial data. If the GSDM is an appropriate tool, and if it can be used beneficially by various disciplines, then the larger issue is becoming more familiar with the fundamental concepts, using and building on those concepts, and sharing that knowledge with others. As Al Gore concluded in his speech, "Working together, we can help solve many of the most pressing problems facing our society..."

Definition of the Global Spatial Data Model (GSDM)

The GSDM, formally defined in (Burkholder 1997), is a collection of existing mathematical concepts and procedures which can be used to manage spatial data both locally and globally. It consists of a functional model which describes the geometrical relationships and a stochastic model which describes the probabilistic characteristics--statistical qualities--of spatial data. The functional part of the model includes equations of geometrical geodesy and rules of vector algebra (solid geometry) as related to various coordinate systems, see Figure 1, Diagram Showing Relationship of Coordinate Systems. The primary coordinate system

The BURKORD™ 3-D Diagram

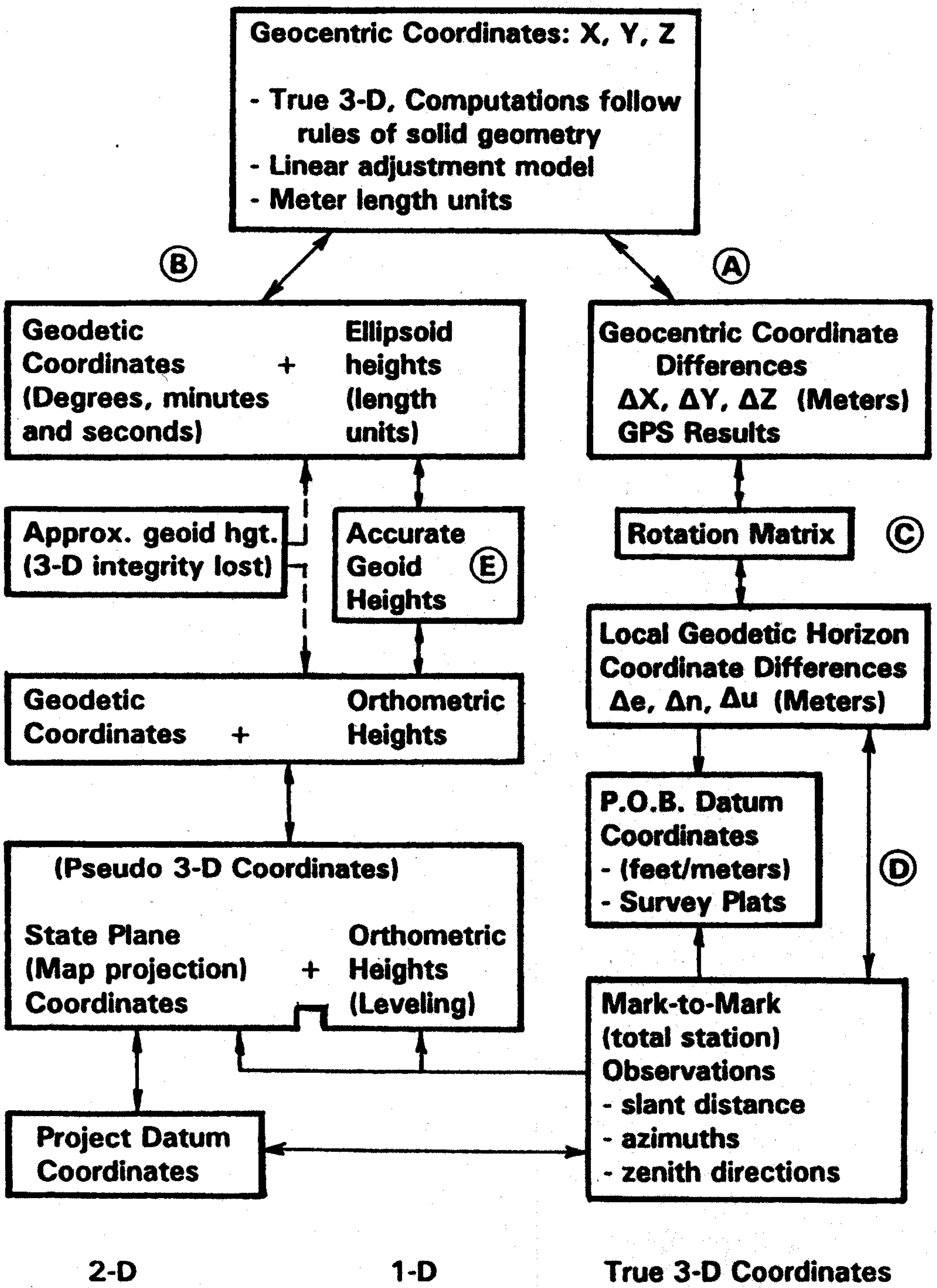


Figure 1 Diagram Showing Relationship of Coordinate Systems

used by the GSDM is the earth-centered earth-fixed (ECEF) geocentric X/Y/Z coordinate system as defined by the Department of Defense (DMA 1991). The GSDM is intended to be consistent with the 3-D Geodetic Model described by Leick (1995) with the following exception; the GSDM, being strictly spatial, does not include gravity measurements, but presumes gravity affects are accommodated before data are entered into the spatial model.

The stochastic model component of the GSDM uses fundamental error propagation concepts as described in Chapter 4 of Mikhail (1976) and Chapter 5 of Wolf/Ghilani (1997). The GSDM stores stochastic information in the variance/covariance matrix associated with each point defined by ECEF coordinates and in the correlation between point-pairs. The local perspective (e/n/u) covariance values and standard deviations need not be stored but are computed upon demand from the geocentric values. The accuracy defined by each point covariance matrix, whether geocentric or local, is called **datum accuracy**. A BURKORD® 3-D database accommodates storage of both the geocentric covariance values for each point and the point-pair correlation. That capability supports and allows one to exploit the rigorous mathematical definitions of **local accuracy, network accuracy,** and **P.O.B. accuracy** when computing the position of one point with respect to another (Burkholder 1999).

Figure 2 is a diagram of the GSDM listing the core concepts surrounded by various disciplines which, to one degree or another, use geographic information systems based upon the National Spatial Data Infrastructure (NSDI) as defined by President Clinton (1994) in Executive Order 12906 signed April 11, 1994. The GSDM fully supports the NSDI and is compatible with details of that Executive Order.

Features of the Global Spatial Data Model (GSDM)

It is not possible or feasible to describe all the features of the GSDM here but several of the more prominent features are:

1. The GSDM uses existing standard mathematical equations of solid geometry and vector algebra for defining the location of a point and for manipulating spatial data. Many complicated geometrical geodesy equations, needed when working with the mathematical ellipsoid model, can be avoided. Instead, using a rotation matrix, the GSDM provides a way to view any set of global X/Y/Z points in terms of local "flat earth" coordinate differences. Additionally, no geometrical integrity is lost (due to the model) and traditional coordinate systems (latitude/longitude) can still be used as or if desired.

2. The GSDM utilizes rigorous error propagation concepts in all three dimensions and provides a standard set of tools which can be used by various disciplines to describe spatial data accuracy with statistical reliability. Meta data are still important but, in many cases, standard deviations are more efficient because they provide a numerical filter for categorizing the accuracy of each point; horizontal, vertical, or both.

Global Spatial Data Model - GSDM

(A Universal 3-D Model for Spatial Data)

The Global Spatial Data Model provides a simple, universal 3-dimensional mathematical foundation for the National Spatial Data Infrastructure (NSDI) which supports Geographic Information System (GIS) database applications in disciplines such as:

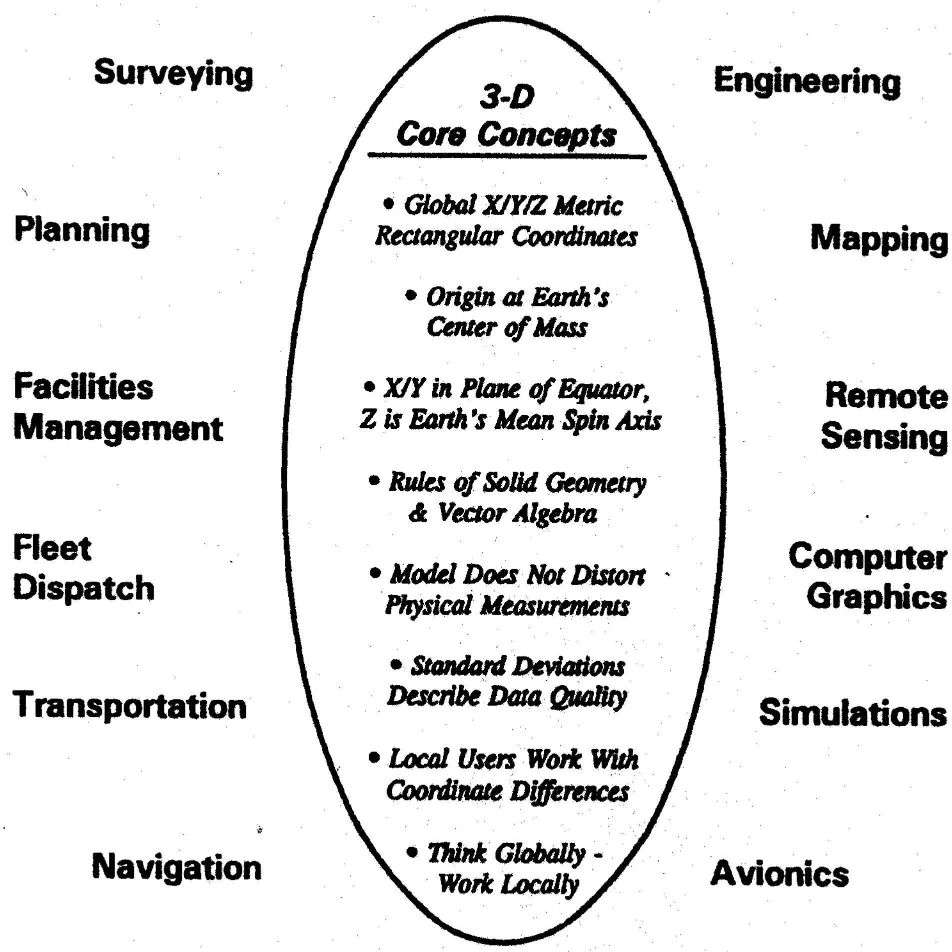


Figure 2 The Global Spatial Data Model (GSDM)

Circleville, Ohio 43113

3. In particular, the GSDM:

   a. provides a globally unique location tag for each point.
   b. uses one set of equations world-wide. No zone constants are required.
   c. preserves geometrical integrity both locally and globally.
   d. is three-dimensional and does not distort distances as does a 2-D map projection. The grid/ground distance dilemma is solved.
   e. combines horizontal and vertical into a single 3-D data base.
   f. permits each user to know the 3-D positional accuracy of each point.

Model & Perspective

The best model is one that is at the same time simple and appropriate. Current horizontal and vertical models as used in geodesy and other spatial data applications are really not very simple but, given the traditional use of horizontal and vertical datums, have been used because they are more appropriate than the flat earth model. Burkholder (1998) describes various models used for spatial data, gives background and details of the GSDM, and includes equations for using the GSDM.

With advent of modern data collection instruments, electronic storage media, and geographic information systems (GIS's) the demand for spatial data has mushroomed. Productivity in map making skyrocketed because traditional manual processes were computerized and increases in productivity continue by using automated data collection and faster computers. But the question must be asked, "Are the underlying spatial data models really appropriate?" For the way humans visualize the world, the answer might be yes. But not for automated data collection, electronic processing, and digital storage. Modern GIS's have evolved from 2-dimensional data bases (state plane coordinates) to 2.5-dimensional systems where elevation is an attribute of location to pseudo 3-D systems in which state plane coordinates are used with elevations (called pseudo because elevations are referenced to an irregularly curved surface - the geoid). The next step is to use a truly 3-dimensional model that is both simple and appropriate--the GSDM which accommodates new technology, modern practice, AND permits each user to view the world from any perspective.

Perhaps that is the most novel idea associated with the GSDM. The origin of a local rectangular coordinate system travels with the user and is placed (or moved) at will. The unique three-dimensional location of each point is uniquely stored in the data base but, given a data set of points (10, 50, 1000 or millions), the user selects any point as the Point of Beginning (POB) and all other points are brought out of the data base with respect to the chosen POB. The technology of data visualization is already in place and it is possible to walk or fly through a proposed development using virtual reality. With the GSDM, the difference is that all points in the data base represent real world points. And, the added benefit is that the GSDM provides an efficient way to describe the spatial accuracy of each point--in three dimensions.

The GSDM is already defined and in place. It supports interoperability, it is seamless, simple, and rigorous, and data quality is defined in terms of standard deviations of each component. The term GSDM is generic and all equations used in the GSDM are public domain. The term BURKORD® is the name of prototype 3-D coordinate geometry software and is an adjective describing specific design of a 3-D data base.

Applications to Current Initiatives

Many persons and organizations are doing impressive things with digital spatial data. Decentralization and the freedom to innovate is fundamental to progress, but the importance of a simple standard underlying spatial data model is the point of this paper. The GSDM is a unifying concept which provides each researcher and user the luxury of complete freedom with respect to how spatial data are used. On the other hand, complete interoperability and compatibility are assured to the extent each user is also willing to define the relationship of their data base to the GSDM. In most cases, that is already being done by default if not by specific declaration.

The following list is incomplete, but intended to be somewhat representative of large-scale efforts which could benefit from adopting the GSDM as a matter of policy. It would not restrict the prerogative of any initiative, but it would insure compatibility between disciplines, between agencies, between government and industry, and between users all over the world.

Where appropriate, a web link is given and the reader is encouraged to visit each one for more information.

1. Digital Earth Initiative by NASA - http://digitalearth.gsfc.nasa.gov

2. National Integrated Lands System - www.blm.gov/nils

3. X-Y Project - www.fgdc.gov/standards/documents/proposals/ugrsprop.html

4. SRI's Digital Earth Project - www.ai.sri.com/digital-earth

5. International Global Grids Conference www.ncgia.ucsb.edu/globalgrids/ann.html

6. NAD 2000 project - www.ngs.noaa.gov/initiatives/NewRefSys/NSRSpolicy.html

7. The Web Mapping Testbed project sponsored by the Open GIS Consortium. See Photogrammetric Engineering & Remote Sensing (PE&RS), November, 1999, pp 1239 - 1241 and December, 1999, pp 1345 - 1359.

8. The National Academy of Public Administration efforts to establish a National Spatial Data Council, see PE&RS, November, 1999, pp 1231 - 1235.

References

Burkholder, E.F., 1999, "Spatial Data Accuracy as Defined by the GSDM," *Surveying & Land Information Systems*, Vol 59, No. 1, pp 26-30.

Burkholder, E.F., 1998, "A Practical Global Spatial Data Model for the 21st Century," presented at the National Technical Meeting of the Institute of Navigation, Long Beach, CA, January, 1998.

Burkholder, E.F., 1997, "Definition and Description of a Global Spatial Data Model," U.S. Copyright Office, Washington D.C.

Clinton, W.J. (President), 1994; "Coordinating Geographic Data Acquisition and Access: The National Spatial Data Infrastructure," Executive Order 12906, signed April 11, 1994, Federal Register, vol. 59, No. 71, pp 17671-17674. (see www.fgdc.gov/publications/documents/geninfo/execord.html)

DMA (Defense Mapping Agency), 1991, "Department of Defense World Geodetic System 1984: Its Definition and Relationships With Local Geodetic Systems," Technical Report 8350.2, Defense Mapping Agency (now National Imagery and Mapping Agency), Fairfax, Virginia.

Gore, Al, (Vice President), 1998, "The Digital Earth: Understanding our planet in the 21st century," (www.digitalearth.gov/speech.html) speech given at the California Science Center, Los Angles, California, January 31, 1998.

Leick, Alfred, 1995, "*GPS Satellite Surveying*," John Wiley & Sons, NY, NY.

Mikhail, Edward, 1976; "*Observations and Least Squares*," Harper & Row, New York, NY.

Wolf, P.R. and C.D. Ghilani, 1997, "*Adjustment Computations: Statistics and Least Squares in Surveying and GIS*," John Wiley & Sons, NY, NY.

Paper prepared for:
INTERNATIONAL CONFERENCE ON DISCRETE GLOBAL GRIDS
SANTA BARBARA, CALIFORNIA, USA,
MARCH 26-28, 2000

Criteria and Measures for the Comparison of Global Geocoding Systems

Keith C. Clarke
University of California, Santa Barbara
Santa Barbara
CA 93106-4060

ABSTRACT

There is no shortage of systems for global georeferencing. Each system, however, differs to a varying extent from a hypothetical ideal system and from other systems. Factors of variation are the systems: authority, succinctness, definitiveness, degree of exhaustiveness in coverage, scaling properties, degree of hierarchical structure, uniqueness, intuitive understandability, tractability, and accuracy. These properties are defined, using examples from some existing grid systems, and are developed as criteria against which the comparison of systems is possible. For each criterion, a metric or metrics are suggested that can be taken individually or collectively for comparing global geocoding systems quantitatively. Methods for such a comparison are discussed and presented. It is suggested that user oriented criteria should be weighted more heavily in global grids for the information and g-commerce age.

INTRODUCTION

Within the last few decades, the number of global georeferencing systems available for applications such as navigation, cartography, position finding and surveying has multiplied significantly. A standard cartographic textbook (Robinson et al., 1995) lists the following systems in use in the U.S.: telephone area codes, postal zip codes, street addresses, letter-number grids, geographic coordinates, cartesian coordinates, GEOREF (or the World Geographic Reference System), Universal Transverse Mercator (both the Military Grid and the USGS civilian version), Universal Polar Stereographic, State Plane Coordinates, and the U.S. Public Lands Survey System. Quite clearly, there is no shortage of ways in which to digitally or otherwise encode position on the earth's surface. Only as recently as 1884, however, was the long used geographic coordinate system finally institutionalized, probably forming the first truly global grid.

In October 1884, at the invitation of the President of the United States, 41 delegates from 25 nations met in Washington, DC, for the International Meridian Conference. Conference resolutions 1 to 3 dealt with standardizing geographic coordinates. It was resolved that "It was desirable to adopt a single world meridian to replace the numerous one's already in existence. 2.The Meridian passing through the principal Transit Instrument at the Observatory at

Greenwich was to be the 'initial meridian' and 3.That all longitude would be calculated both east and west from this meridian up to 180°." Resolution 2, fixing the Meridian at Greenwich was passed 22-1 (San Domingo voted against), France & Brazil abstained (Howse, 1997, p. 141). Thus the Meridian conference established two criteria for acceptance of geographic coordinates as a global geocoding system. These were its universality (by determining its scope of application and origin) and authority (by the acts of resolving and voting at an international conference).

This paper examines briefly the criteria that a global geocoding system or geographical reference frame strives for. These criteria are recounted here not out of originality, but with the intent of providing a framework against which individual geocoding systems can be compared. Furthermore, since qualitative comparisons serve primarily to stimulate debate rather than good science, a set of metrics is proposed so that each existing and new system alike can be measured against the remainder and against definitive absolute metrics. To illustrate the approach, these metrics are computed for a selected small set of existing global grid systems and the results presented for discussion. It is hoped that presenting the metrics and their means of computation will stimulate analytical studies of global grid systems, to refine their applicability and characteristics, and to aid the map user in their selection for a particular purpose.

DIMENSIONS FOR COMPARISON OF GLOBAL GEOCODING METHODS

To this author's knowledge, there have been at least two prior attempts to derive criteria for comparing global grids. Goodchild (1994) listed 14 criteria, set forth in Table 1. Kimerling et al. (1999) added to, refined and reordered Goodchild's criteria to reflect their work on nested hierarchical tessellation. Kimerling et al. (1999) saw an ideal global grid as being able to summarize irregular global measurements, calculate gradients, compare time series, compare regions, compare data collected at different resolutions, improve numerical modeling and to document the precision and location of spatial data on the globe. Criteria used for evaluations of the grid systems were chosen analytically as further derivatives of the Goodchild criteria. Those of critical importance were metrics for spheroidal area, compactness and center point spacing. Some specific metrics were proposed that included the Zone Standardized Compactness. In addition, Kimerling noted that some criteria could be automatically tested, for example by point-in-polygon testing, coordinate range checking, and verifying that recursion is possible.

Table 1: Comparison of Criteria for the Assessment of Global Grids

Criteria in Goodchild (1994)	***Criteria in Kimerling et al. (1999) (Goodchild's Numbers given in parentheses)***
1. Each area contains one point	Areal cells constitute a complete tiling of the globe, exhaustively covering the globe without overlapping. (3,7)
2. Areas are equal in size	Areal cells have equal areas. This minimizes the confounding effects of area variation in analysis, and provides equal probabilities for sampling designs. (2)
3. Areas exhaustively cover the domain	Areal cells have the same topology (same number of edges and vertices). (9, 14)

4. Areas are equal in shape	Areal cells have the same shape. ideally a regular spherical polygon with edges that are great circles. (4)
5. Points form a hierarchy preserving some (undefined) property for m < n points	Areal cells are compact. (10)
6. Areas form a hierarchy preserving some (undefined) property for m < n areas	Edges of cells are straight in a projection. (8)
7. The domain is the globe (sphere, spheroid)	The midpoint of an arc connecting two adjacent cells coincides with the midpoint of the edge between the two cells.
8. Edges of areas are straight on some projection	The points and areal cells of the various resolution grids which constitute the grid system form a hierarchy which displays a high degree of regularity. (5,6)
9. Areas have the same number of edges	A single areal cell contains only one grid reference point.(1)
10. Areas are compact	Grid reference points are maximally central within areal cells. (11)
11. Points are maximally central within areas	Grid reference points are equidistant from their neighbors. (12)
12. Points are equidistant	Grid reference points and areal cells display regularities and other properties which allow them to be addressed in an efficient manner.
13. Edges are areas of equal length	The grid system has a simple relationship to latitude and longitude.
14. Addresses of points and areas are regular and reflect other properties	The grid system contains grids of any arbitrary defined spatial resolution. (5,6)

This study hopes to build upon this pioneering research. In Kimerling et al.'s rebuilding of the criteria, two factors emerged. First, that metrics are critical to implementing and using the criteria effectively. Second, that while commonalties have emerged between the two versions of the Goodchild criteria, nevertheless the Kimerling et al.'s work was oriented primarily toward hierarchical recursive global grids. An attempt is made here, therefore, to be both more specific in terms of metrics, and more general in terms of criteria. Furthermore, criteria that relate to geometry rather than topology (such as Kimerling's seventh and Goodchild's eighth) are specific to projection properties rather than grid characteristics. In many grids, a projection is assumed, and recursion takes place on the plane. The grid's relation to projection distortion is then a given. This is because Goodchild's properties 2, 4, 8, 9, 10, 12, and 13 are consequences of the projection decision and the assumption of the earth model, not necessarily the grid systematics. This is the case even with geographic coordinates.

The approach taken here is more generic, and includes more of a grid system user orientation than an algorithmic, topological, or computational geometric perspective. It is hoped that a user based grid comparison will be of use in the grid selection process, and assist users in learning grid systems. First, the dimensions of the criteria are considered, and some metrics identified. These are collected into a single framework, and some examples given for specific grids.

1. Universal

Global grids are designed to be universal. That is, in the ideal, they apply not only to all three dimensional bounded objects such as the geoid and the planets, but also to the whole earth. Clearly a first assumption affecting universality is the choice of earth model. Historically, geodesy divides chronologically into periods based on the sphere, the oblate ellipsoid and the geoid as earth models. Since the geoid is usually expressed as deviations from the best fit ellipsoid, and since the earth fits the spherical model reasonably well for highly generalized mapping applications, the universality dimension clearly reflects accuracy. At the overview level, however, universality may be thought of as the degree to which the global grid system allows location georeferencing for the whole earth or equivalent geographic object (such as the Moon, Mars, or a baseball).

Related to the universality dimension is the ability to recover the system from empirically derivable features. Examples are the poles, the equator and universal time, all of which are absolute and measurable given the right algorithm. Actual origins, however, may or may not be tied to tangible features. Perhaps the most universal of systems is geographic coordinates, which applies to all three earth models and to the whole planet. Nevertheless, its origin point (Figure 1) is recoverable only with extensive use of sophisticated navigation equipment and/or astronomical observation. A first metric of universality, therefore, might be the degree or ease of recoverability of the primary reference monuments and the linear dimensions for the system.

Figure 1. Origin Point of the Geographic Coordinate System (0 degrees North, 0 degrees East)

On the purely practical level, universality also relates to the adoption of standards. International Standards Organization (ISO) standards represent the peak of a hierarchy that moves through national standards such as ANSI, all the way to "industry" standards and de facto standards such as PostScript. Some measure of universality, therefore, should reflect the incorporation of broadly acceptable standards. This may include the use of the decimal system, use of metric units, use of Arabic numbers (0,1,2,3... etc.), and the specification of a reference ellipsoid by an international body. For example, WGS84 is broadly accepted and used as a reference ellipsoid, though its exact specification remains classified. A higher standard is the International Earth Rotation Service's International Terrestrial Reference System (ITRS). A second measure of universality therefore, might be the number of geocoding system parameters that are tied to standards, weighted for the level of the standard.

In addition, universality applies to the extent of the system's actual coverage. In UTM, for example, coverage extends from 84 degrees North to 80 degrees South, leaving the UPS for polar coverage. While UTM covers just about all of the earth's inhabited area, nevertheless the real extent is incomplete, depending on the UPS to fill the gaps. A metric corresponding to the extent of coverage might be the actual land area of the terrain "covered" by the grid, as a ratio to the total surface area of the earth model or ellipsoid. This metric is reconsidered under the dimension of exhaustiveness.

2. Authoritative

Almost anyone is capable of devising a global georeferencing system. Not all, however, are of equal credibility. Systems establish authority in two ways, by recognition and by acceptance. Recognition implies a hierarchy of acknowledgment. At one extreme, a system "exists" if it is in any way documented. So for example, an independent scholar could publish a minimal set of definitional parameters as a web page, and the system would exist but with no recognition. An example of such a system would be the geographic coordinate referencing using Washington, D.C. as the prime meridian, marked on many older American maps. Initial recognition of a system is primarily academic or highly specialized, but publication in the peer reviewed scientific literature of cartography, as in the case of Dutton's Quaternary Triangular Mesh (Dutton, 1999) is a critical mark of authority. Passing from academic research into standard teaching practice also enhances credibility, so that inclusion of a system in a cartography textbook, or its teaching at a major University in any relevant curriculum adds a further degree of credibility (and acceptance). A metric of such inclusion might be the number of references in a bibliography, the number of citations to the system, or the number of entries in catalogs or textbook indices.

Formalization into standards, such as FIPS 173, at the national level moves the system to the next level of credibility. Beyond national standards lie those of international collaboratives (e.g. scientific or professional societies, NATO), then the official recognition of International Standards bodies and the United Nations. The so-called Peters projection, for example, was less than a historical footnote until Peters convinced the World Council of Churches and the United Nations to endorse the map (Monmonier, 1995).

At the highest level of recognition lies acceptance by the International Standards Organization (ISO), and by consortia of professional societies at the international level and acceptance for extended periods of time. For example, the International Earth Rotation Service (IERS) was created in 1988 by the International Union of Geodesy and Geophysics (IUGG) and the International Astronomical Union. It replaced the Earth rotation section of the Bureau International de l'Heure, and the International Polar Motion Service. IERS is a member of the Federation of Astronomical and Geophysical Data Analysis Services and has been established since 1988 to provide to the worldwide scientific and technical community reference values for Earth orientation parameters and reference realizations of internationally accepted celestial and terrestrial reference systems. The IERS is charged to define, use and promote the International Terrestrial Reference System (ITRS) as defined by the IUGG resolution No 2 adopted in Vienna in 1991.

Another type of authority is that of broad popular acceptance and use. Map users are highly influenced by choices make by the map production agencies. The United States Geological Survey, for example, normally marks geographic coordinates, UTM, and State Plane coordinates on the collars of its topographic quadrangle maps at 1:24,000. City Street Guides commonly use one-off Alphabet-Number referencing. Neither group based the choice of systems that the map consumer must use by practical necessity on user demand assessment, although the USGS has occasionally changed grids and their depiction at the request of its fellow federal agencies such as the US Forest Service. Acceptance of systems is probably highest with street referencing, involving common use and standardization by the US Post Office, and including extensive international cooperation.

Measures of authority are necessarily subjective, since they are based primarily on trust. A simple measure would be the hierarchical level in the schema from individual to global organization. Another would be the customer acceptance, in terms of market share. A simple alternative metric would be the number of standards that form the definition of a particular grid system, or conversely, the number of standards that use a system for defining geographical referencing.

3. Succinct

The ideal grid reference results in coordinates that are terse. Typically, a grid reference handles the three dimensions separately, with each depending upon one or more items of metadata that define a zone or region. The degree to which the elements of the coordinate are embedded are important. Systems exist which contain the entire global reference in the string, use separate strings for eastings and northings (and elevations), and interleave the digits of the reference in alternating pairs.

Communication theory provides a measure of the quantity of information that flows during a transmission from sender to receiver (Shannon, 1948). Transmission flows involving coordinates are common, for example when GPS data are collected, when search and rescue operations are conducted, or when GIS coordinate based data are moved between systems or used over a network. The general formula for reducing the uncertainty of communication to zero ("computing the amount of information generated by the reduction of n possibilities (all equally likely) to 1." Dreske, 1999) or the entropy, is given by the base 2 log of n. This value, in bits, is the necessary length of a binary number that fully defines an atomic unit in the system undergoing transmission. In the case of coordinates, this is a point. We will consider only the case of two coordinates, in spite of the fact that three coordinates strictly are necessary to define location (the reference ellipsoid is assumed instead of the geoid).

A paradox of grid coordinate systems is that points located close to each other in geographic space have similar coordinates. This is a corollary of Tobler's famed "first law of geography," that everything is related to everything else, but near things are more related than distant things (Tobler, 1970). If indeed the importance of discrimination of locations is equally important as points become closer together, say for local navigation, then the redundancy within the coordinates becomes a maximum, when it would be best be minimum.

Tukey (1977) advanced the stem-and-leaf plot as a simple tool for finding the level of redundancy in transmissions. Shannon (1948) had previously devised a mathematical theory of information in communication that related the amount of information transferred to the reduction of the uncertainty at the receiver end of the transmission. Such an approach can be used with coordinates in a global grid system. For example, as a map user parses along a vector, information is sequentially extracted from strings of coordinate pairs that carry geographic information content, usually coded from left to right, but sometimes interwoven and alternating digits. As each successive digit (or bit) is traversed, more spatial information flows, and serves the function of distinguishing the current point from the previous point. Similarly, we can talk about the set of points constituting a geographic feature, whether it be a point, line, area or compound feature.

For a set of geographic coordinates, a ratio can be defined that is the measured range, standard deviation, and maximum possible range of each coordinate. For example, for the set of coordinates {632794.69 4538257.50 632948.69 4538520.00 632554.25 4538098.50 632794.69 4538257.50 632231.31 4537331.00 632554.25 4538098.50} which constitute UTM coordinates in Zone 18, Northern hemisphere, forming a small part of the outline of Long Island, we can compute the following values:

Table 2. Statistical Description of Six Coordinate pairs in UTM

Minimum (x, y)	Maximum	Range (m)	Standard Deviation (m)	Std. Dev as Proportion of Range	Zone 18, N
632231.31	632948.69	717.38	232.609114	0.32425	UTM Easting
4537331.00	4538520.00	1189.00	369.041521	0.31038	UTM Northing

These estimates of total range and variance, however, mask the variance structure as it relates to the coordinates digits themselves. Accordingly, consider that for every significant digit of the coordinate there is both an actual and an expected proportion of the coordinate count within the set. For the ten digits of decimal numbers, if over the set the actual proportional occurrence of each digit, "0", "1", "2" and so forth was the same as the expected (0.1) then the sum of the deviations would be zero. If all coordinate digits were identical in occurrence across points, then nine digits would have no occurrence (0.0 - 0.1 x 9 = -0.9) and one digit would have the whole point set (1.0 - 0.1 = 0.9), whose magnitudes sum to 1.8. This is the case for the first three digits of both the easting and the northing in the set. Thus digit variances would vary from zero (for no difference between expected and actual digit occurrence) and 1.8, when all digits are identical. The equivalent values are 1.875 for hexadecimal, 1.75 for octal and 1 for binary, given by $2(B-1)/B$ where B is the number base. For any digit n at any one significant digit location out of N possible digit values or states (10 for decimal), I is defined, where:

$$I_n = \sum_{1}^{N} \left| \frac{D}{\sum D_n} - \frac{1}{N} \right|$$

Similarly, the set of these values across all significant digits defines a function that starts at complete redundancy, drops as the information content increases, then returns to near redundancy in this case beyond the decimal point. This might be termed the *Coordinate Digit Density function*. The "area" or total divergence of this function from complete redundancy defines a value which is an entropy or information quantity value for the coordinate set which may be independent of the coordinate system and therefore of value in comparison. This value, termed *S*, for *N* significant digits and assuming number base *B* is given by:

$$S = \sum_{1}^{N}\left(I_n - -\frac{2(B-1)}{B}\right)$$

Computing *S* for the six point set above yields 2.167 for the northings and 2.407 for the eastings. Both eastings and northings have their highest maximum entropy at the third decimal place, and have five redundant digits. The Coordinate Digit Density functions are shown in figure 2.

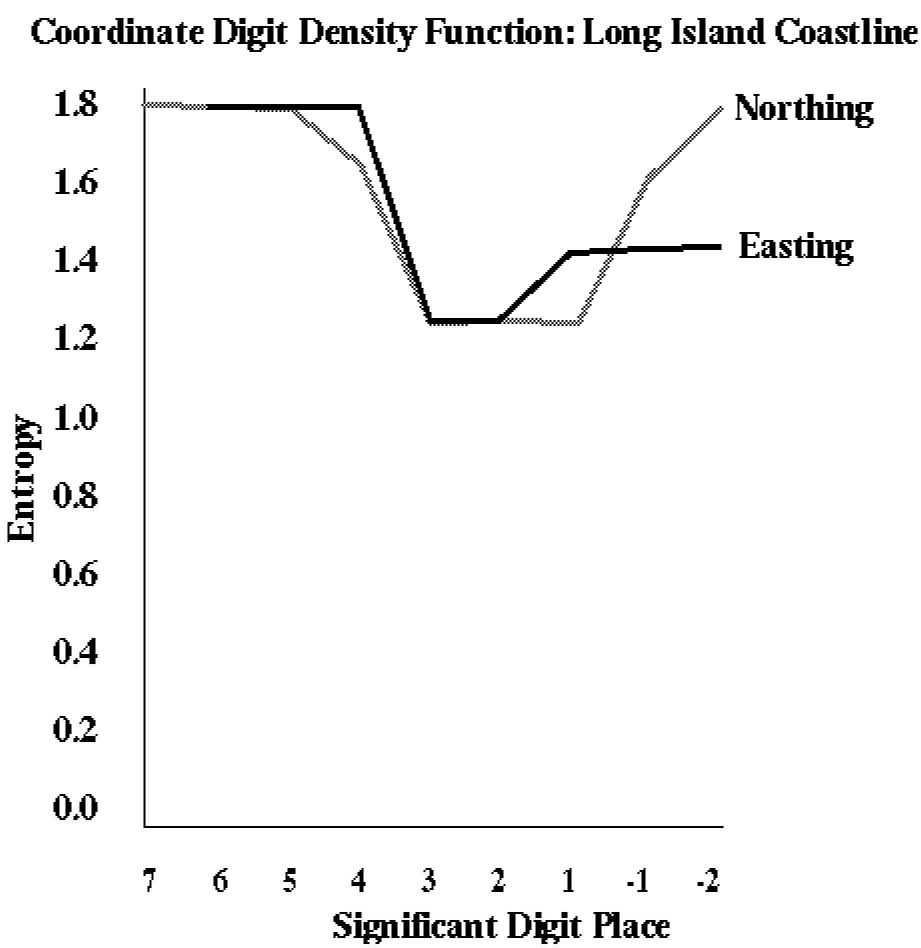


Figure 2. Coordinate Digit Density Function

The easting and northings show some differences, which for a larger point sample may be worth investigating. For example, neither reaches a value below 1.0, which might be expected at the peak information content digit near the decimal place. The first three and last digits of the northing and the first three digits of the easting are redundant. Any compression system, such as leaving off the zone number and a 100 km intersection reference and truncation of redundant decimal places would compress the data to those digits below the 1.8 line. Significantly, the abbreviation of the northing to the nearest 0.5 meter and the repetitive rounding of the northing imply that the data have been converted from other units or degrees (they were).

The measure *S*, and the Coordinate Digit Density function are proposed as useful means for the comparison of global grid systems. Since the density function depends only on frequencies, letters and other systems, such as octal, binary and hexadecimal are equally suited to the metric. Comparison is possible between and among the same exact position in different systems, set of positions in different systems, or coordinate extremes. Simple statistical description implied that the northing had a much higher variation in range and standard deviation, but a lower deviation as a proportion of the range. The digit analysis shows that the eastings have a higher information content, and that it is concentrated between the fourth digit and second decimal place, i.e. in the 1km to 1cm range.

A succinct coordinate system, therefore, is one with the most spiked coordinate digit density function, the lowest entropy associated with a single coordinate digit, and the highest S. Nevertheless, this ignores the fact that many coordinate systems are hierarchical. While the measure applies equally well to almost all systems, it is possible to score the levels of the hierarchy independently, to measure the bits required per level (using Shannon's formulae), and to estimate the total information content of a set of coordinates in different systems.

While succinctness depends heavily on assessing the total amount of information in coordinates, there are at least two other measures that may be critical for comparing systems. The first of these is retrieval time, that is the amount of time it takes to move an encrypted coordinate into a system where it can be readily interpreted. This can be considered in several ways. At the simplest, it is a simple number of algorithm steps, computations or look-ups necessary to write a coordinate into ASCII digits. So, for example, a transformation of coordinates may involve a decompression, binary to decimal conversion, bounds checks and an affine transformation. This could be quantified as a number of steps or as real CPU seconds. Secondly, use time is the opposite transformation as applied to the user of the grid system. How long would it take a novice or skilled map user to place the point onto a map, or encode a given location? This could be assessed subjectively, or measured in real seconds by testing map users.

4. Definitive

Definitiveness is the ability of a coordinate or grid system to unambiguously determine a georeference. There are at least three concerns for definitiveness. First, a single point on the earth's surface must be assigned one and only one reference. Clearly this is often not the case in many global grid systems. The geographic coordinate system assigns identical latitudes and both marginal and extreme range longitudes to a single point at the poles. UTM allows a half degree overlap between zones, a fact that becomes vital when regions of interest fall over zone boundaries. A quantitative metric of overlap definitiveness for a whole system could be the total earth surface area for which overlap is permitted, perhaps weighted for the total number of overlaps (there may be more than two).

Lack of overlap reduces redundancy, but may be an integral part of the grid system. Equally as important to the amount of overlap is the lack of confusability between higher orders of coordinates. For example, within a UTM zone (and even between systems, such as USPLS in meters) there is little to distinguish between zones. The Military Grid system recognizes this by assigning letter references and redundant discriminators to the UTM zone number, which usually has only one digit offset. Thus an effective grid system never creates coordinates for nearby points that have lower order northing and easting coordinate specifications that are similar. This argues for integrated hierarchies in coordinates, so that the confusion is explicitly eliminated. It is also supportive of those georeference systems that interleave eastings and northings.

Similarly, a global grid system should be able to support objects of different dimensions. Here the largest difference is the treatment of the coordinates themselves. In the Military Grid and the British National Grid, the reference unit is of varying area depending on the interwoven number of digits employed. While this works well for many applications, it is usually not sufficient for mixed point, line and area objects.

5. Exhaustive

Opposite to definitiveness is exhaustiveness. A grid system should cover each and every location on earth at any level of scaling, spatial resolution, or measurement precision. Complete exhaustiveness is less common than might be thought. UTM for example, covers only 80 degrees South to 84 degrees North. Just as we measured universality as a proportion of the earth's surface area, we can similarly quantify exhaustiveness as the proportion of the earth's surface covered by the system.

Exhaustiveness may or may not scale. For example, UTM is not exhaustive on a global level, even within a zone, but has redundant coverage at the edges of zones. Thus exhaustiveness should be assessed both at the global level and the local level in a hierarchical system, perhaps at every level. Similarly, grid exhaustiveness may be a function of the resolution of the atomic grid unit. Obviously as resolution becomes coarser, assigning grids to features involves overlap, redundancy and drop out. Some measures of this exhaustiveness have been quantified by Mulcahy (1999).

Finally, a grid system should be able to store sufficient precision to ensure exhaustiveness. At least, this means that atomic features (for example, bench marks, survey points, pixels on high resolution images, utility features such as power poles and manholes) must have a unique location. A better condition might be that the precision associated with the atomic unit for the grid is close to the accuracy of the measurement instruments. The latter implies an "effective resolution" that might be concisely delimited using the sampling theorem (Tobler, 2000). As a rule of thumb, the grid "spacing" or level at which precision is capable of feature discrimination should be less than half the average size of the smallest feature that the system is designed to locate. There are standard measures of precision and accuracy (Goodchild and Gopal, 1989). Relating these to the grid's effective atomic unit via the sampling theorem might be best done with a simple ratio.

6. Hierarchical

The merits of recursion and the repetition of rules and structures inherent in hierarchies are too great to be ignored with global grid systems, and almost all systems use some degree of hierarchical tessellation or tiling. Nevertheless, tiling of any sort creates a tension between core and edge. Typically tiles are reprojected using unique central meridians, points of tangency or secancy, so that the pattern of error is centered on the tile and is maximum at the edge. Examples are shown in Kimerling et al. (1999).

Joints are where tiles meet on the ground. If joints overlap, tiles or zones interleave to form zones of a lack of definitiveness and redundancy. The geometry of the overlap and joints is important for accuracy, scale, direction, area and shape on the grid. These properties have long been quantified and even cartographically symbolized on maps (Mulcahy and Clarke, 2001). A metric of jointing should simply reflect the amount and distribution of joins in the system. This is given by (1) the number of recursive tilings used to reference the atomic unit in the grid and (2) the total number and length of tile edges in the entire system. This refers to all appropriate unique hierarchical levels, and may be computed as a function of level. The Quaternary Triangular Mesh (Dutton, 1999) for example, does not vary with recursion beyond the level one partition into triangles from the globe, nor does the quad tree approach of Tobler and Chen (1986). UTM has only 60 interior and 120 exterior edges at the highest (zone) division. Thus UTM could be said to score 180 on the zone edge scale.

7. Unique

The degree of uniqueness has already been covered under definitiveness and exhaustiveness, and is part of both Goodchild and Kimerling's criteria lists. The problem of coordinate confusion is a real one, both within and between systems, and perceptual testing could be used to define uniqueness in terms of user errors in coordinate interpretation. Some notorious errors in coordinate specification, from Embassy bombings to friendly fire incidents, could be easily avoided with effective enforcement of local uniqueness of coordinates. To be unique, each entity or atomic spatial feature has only one geocode, and the geocodes are distinctive from each other. A measure may be the average number of significant digits that discriminate between features deemed to be adjacent or contiguous. Such an assessment would be possible by measuring the Coordinate Digit Density function for point pairs or point sets drawn at random from points at "near" and "far" distances from each other. A correspondence measure between two coordinate pair sets could be simply the proportion of coordinate bits that match exactly divided by the sum of the match plus the mismatch.

8. Intuitive

Anyone who has taught global grids and coordinate systems in undergraduate cartography or geography knows that many people find the grasp of the basics of grid systems an intellectual challenge at best, and a mystery at worst. There is a strong correlation between the ease of teaching a system and the system's effective use in practice, especially in applications such as field survey and navigation. Simplicity, however, is a highly subjective metric. Occam's razor tells us that if there are two acceptable theories explaining a set of facts, the simpler one is better. Such a rule can be applied also to global grids, yet with caution, since the intended function of the grid system is every bit as critical to effectiveness as simplicity of the system's rules and constants.

Measuring intuition is perhaps hardest of all of the measures proposed! One set of ways of quantifying intuition is to count the impediments to use of the system. Possible metrics are (1) the number of separate "facts" necessary to learn or explain the system; (2) The number of "magic numbers", i.e. constants, arbitrary origin points, earth radii etc. necessary to define the system or to locate a single point within it; (3) The average number of words or pages that a software manual, textbook, or help system must devote to explaining the system to a user.

Harder to quantify are metrics that define the time necessary for fluency in the system (say, to achieve making no errors per 1,000 point fixes) by book learning or experience. Among these are the level (elementary school, junior high, high school, college) at which education is possible, the reading grade level necessary to understand system documentation, the time required for explanation of the grid, and the amount of retraining required for maintenance of the knowledge.

Another important property of global grids is their memorability. This property is not the memorability of the systematics of the system, but the memorability of the georeferences themselves. For example, it is relatively easy to remember that Santa Barbara lies in UTM Zone 11, and that the boundary of Zone 10 is at the 120th meridian just West of Goleta, but only if you live in Santa Barbara! Other aspects of grid references may or may not easily commit themselves to memory, but if they do are useful for all sorts of basic fact retrieval and navigation. Effective systems promote such recall, and exploit it. Nevertheless, measurement of this property seems almost impossible without resorting to qualitative methods.

9. Tractable

Many advantages of global grids are not necessarily part of the system but are consequences of the system's properties. Central to the tractability of a global grid system is the availability of a mechanism for encoding, decoding and plotting of the systems with maps. This may imply web access, computer programs, software utility or built-in functions, and full documentation. A measure of the tractability and ease of use of a system is the programming code volume in bytes, number of logic steps, number of lines of code, or program execution time associated with deriving or plotting coordinates of different types.

Secondly, many applications of coordinates focus on their use for the extraction of successive sample locations directly from random numbers applied to the coordinates and their ranges. To be suitable, a grid system should allow the extraction of samples in random, systematic, hierarchical or other appropriate sampling methods. Resampling coordinates is often an important part of multi-scale cartography, therefore support for multiple representations or multiple display scales is desirable. If this happens as coordinates are resampled, then generalization is possible by simply weeding duplicate coordinates.

10. Accurate

Traditional metrics of accuracy involve tests against independent map sources of higher authority. A measure of a grid's accuracy, in addition to the already suggested ratio of resolution to features size, is the comparison with an independent, or original source. If a common database, such as the Digital Chart of the World, is transformed to another grid system, and then the transformation is inverted, then there should be a one-to-one correspondence on a bit-by-bit level between the original and retransformed maps (Clarke, 1995). Any disagreement, as a proportion of the original, is the omission error. Such error can be quantified in many ways.

Within a system, accuracy is defined by repeatability. A grid system should be able to return a user or navigator to the exact same location, independently of minor details such as rounding error, algorithm implementation, and pixel resolution. A measure might be to locate a set of points one thousand times each, and to quantify the average positional error involved in repetition. Even with computer algorithms differences emerge. With human interpretation and with look-up solutions, errors can be significant.

Finally, error is both a global and a local property of grids. Aggregate accuracy measures mask the extremes and spatial distribution of error. Efforts should be made to portray not just the amount of error, but also its spatial distribution. This is often possible with quite traditional cartographic methods (Clarke and Teague, 1998).

METRICS FOR COMPARISON

Several metrics have been proposed in the discussion of the dimensions of comparability. Any or all of these could be computed and used for comparison between global grid systems. In Table 3, attention has been given to the ranges and units of the metrics. Most values could be computed, of course, in several different ways.

Table 3. Summary of the Metrics for Global Grid Comparison

Dimension	**Metric**	**Value**	**Units**	**Geographic Coordinate Example**
Universality	Proportion of earth's surface covered by grid	Ratio (0.0-1.0)	None	1.0
Universality	Recoverability of grid system origin monument and standard dimensions	Boolean (2 values)	Yes/No	{0, 1} Units are ISO defined
Universality	Proportion of parameters and constants tied to International or national standard	Ratio (0.0-1.0)	None	1.0 Assuming metadata (e.g. ITRF, WGS84)
Universality	Number of International or National Standards referenced by	Greater than or equal to zero	Standards	2 (Ellipsoid, International Meridian

	specification			Conference)
Authority	Number of bibliographic references to grid system	Greater than or equal to zero	References in Snyder bibliography	351
Authority	Number of references on World Wide Web	Greater than or equal to zero	Number of hits using altavista with "geographic coordinate"on 3/21/00	1,305,095
Authority	Number of catalog entries	Greater than or equal to zero	Search of "magazine and journal articles" in all UC libraries on 3/21/00 for "geographic coordinates" under subject keywords	27
Authority	Entries in textbook index	Greater than or equal to zero	Number of pages referenced in index for Sixth Edition of Robinson et al. "Elements of Cartography", under "geographical coordinates", "latitude", and "longitude"	11
Authority	Degree of conformance to standard	Boolean	Assumed, since ISO references.	1, Yes
Authority	Number of standards that reference the system	Greater than or equal to zero	Standards	Unknown
Succinctness	Number of Digits in Geocodes	-90.0 to 90.0 for latitude, -180.0 to 180.0 for longitude	Degrees, decimal or degrees, minutes and seconds with decimals	22 (including decimals , a space, and 2 sig. figs. for DMS)
Succinctness	Number of ASCII characters per point with full geocode	Greater than or equal to zero	ASCII characters	27 (includes EOL)
Succinctness	Coordinate Digit Density Function	Graph (one value per significant digit)	None	NA
Succinctness	S (entropy measure	0-1.8	None	NA

	based on CDDF)			
Succinctness	Number of Algorithm Steps for retrieval	Greater than one	Steps	NA
Succinctness	Number of computations to ASCII conversion	Greater than one	Computations	NA
Succinctness	Number of look-ups performed	Greater than or equal to zero	Look-ups	NA
Succinctness	CPU time for conversion	Greater than zero	seconds	NA
Succinctness	User encoding and decoding time	Greater than zero	seconds	Needs human subject tests
Definitivenes s	Overlap as a proportion of total space covered by grid	Greater than or equal to zero	Ratio	0.0 for latitude approx. 0.001 for longitude
Definitivenes s	Weighted Overlap as a proportion of total space covered by grid	Greater than or equal to zero	Relative value, reflecting multiple counts	0.0 for latitude Infinity at 90N and 90S, 0.0 elsewhere
Definitivenes s	Similarity coefficient for adjacent cells/points	Ratio of bitwise mismatch to mismatch + match (0.0-1.0)	Ratio	At 2 sig. fig for seconds, 0.9091
Exhaustivene ss	Range of resolutions covered	Two values, both representative fractions or ground distances	Ratios	1:400,000,00 0 to 1:1
Exhaustivene ss	Range of precision	Significant Digits or parts per million	Digits/PPM	5 (whole degrees) to 17
Exhaustivene ss	Proportion of earth covered at finest resolution and precision	Greater than zero	Ratio	1.0
Exhaustivene ss	Compared to Geographic, pixel loss and duplication ratios	Loss 0.0-1.0 Duplication greater than zero	Ratio	NA (comparison base)
Exhaustivene ss	Ratio of atom to resolution	Greater than zero	Ratio (smallest desired resolution/precisio	1m/0.31 m = 3.23

			n)	
Hierarchy	Number of reprojections within system	Greater than or equal to zero	Different projections/central meridians, points of tangency or secancy	0
Hierarchy	Number and length of joints	Greater than or equal to zero	Count	0
Hierarchy	Number of recursions from base to atom level	Greater than one	Recursion levels	3 (for DMS)
Uniqueness	Average number of significant digits in coordinates that distinguish between adjacent cells or points	Greater than one	Digits	1
Uniqueness	Coordinate Digit Density Function for point pairs	Graph	NA	NA
Uniqueness	Match ratio for adjacent coordinates	1-(mismatch/(match + mismatch)) 0.0-1.0	None	At 2 sig. fig for seconds, 0.9091
Intuitive Understanding	Number of facts that explain system	Greater than one	Facts	8
Intuitive Understanding	Number of parameters externally defined (magic numbers)	Greater than zero	parameters	4
Intuitive Understanding	Length of explanation/documentation	Greater than zero	Words (Source: Snyder Map Projections: A Working Manual)	C. 1000
Intuitive Understanding	Time to achieve error free use	Greater than zero	days, minutes	NA
Intuitive Understanding	Educational level required	K-16	Grade Level	10
Intuitive Understanding	Time to achieve explanation	Greater than one	minutes	20
Intuitive Understanding	Frequency of retraining	Greater than zero	months	NA
Intuitive Understandin	Memory recall of common geocodes	Binary, or Human subjects	Yes/No or proportion of error	NA

g		derived error rate		
Tractable	Availability of Method	Formulae or algorithm in literature/web	Yes/No	Yes
Tractable	Size of computer program for use	Smallest available computer program	Bytes	NA
Tractable	Steps in program logic	Lines of code	Lines	NA
Tractable	Program execution time	CPU or user time	seconds/point	NA
Tractable	Supports sampling and generalization	Boolean	Yes/No	Yes
Accurate	Test against independent source of higher authority	1-(mismatch/(match + mismatch)) 0.0-1.0	None	1.0 (self)
Accurate	Forward to inverse transformation comparison	1-(mismatch/(match + mismatch)) 0.0-1.0	None	1.0 (self)
Accurate	Repeatability	Proportion in error	None	1.0 (assumed)
Accurate	RMS or other single accuracy value for whole data set, as projected	distance or standard deviation	meters	0.31 m

APPLICATION

The criteria listed above, coupled with the metrics in Table 3, are a foundation around which objective comparisons between global grid systems is possible. The illustrative values entered in the table for the geographic coordinate system are provided as a first set of estimates, and will be refined over time. Similar values for the various grid systems in use can be computed accordingly. Comparison can then take the form of a series of greater than tests, by principal components analysis of the scores, or by the computation of weighted aggregate scores. No such comparison is attempted here, but research is invited in this new and potentially useful comparative approach to the analysis of global grids.

CONCLUSION

It is foolish to believe that a single grid system would ever serve the needs of all users. Nevertheless, for particular applications and disciplines, placement and contrasting of systems within the proposed framework would allow objective decisions to be made about which grid to select. An advantage of the analytical approach to grid selection is that the comparative metrics point out both strengths and weaknesses of any system for a particular application.

Analytical cartography can serve in comparing global grids to conduct meta-analysis of entire systems. The metrics and methods proposed can serve to illustrate possible enhancements, improvements and modifications to existing grid systems that may be of considerable benefit to map producers and users alike. Whatever the outcome, the current era of Internet and World Wide Web based cartography will ensure that the user, rather than the cartographer, surveyor, or geodesist will increasingly influence the future of the mapping sciences. Occam's razor has been proposed as applicable to global grid systems, that is, given two equally useful and powerful grid systems, the better one is the simpler of the two. User demand assessment and user testing are only now becoming regular tools in the cartographer's toolbox. Web mapping both demands immediate solutions to the inadequacies of particular grid systems and provides a somewhat objective means by which user testing can be conducted rapidly and in sufficient numbers to move beyond the currently favored "30 geography students" that human subjects tests of mapping applications tend to use.

The future, quite clearly, will reward those systems that meet their Web searching demands with de facto acceptance and therefore authority. For over a century, cartography has allowed the competitive coexistence of global grid systems devised for different applications. As surveying and mapping yield to mobile mapping applications such as navigation and high-precision positioning, it is hoped that the methods and metrics proposed here can lead to some effective choices for the future based on analysis and quantitative methods rather than subjectivity and bias.

REFERENCES

Clarke, K. C. (1995) *Analytical and Computer Cartography*. Englewood Cliffs, NJ: Prentice Hall.

Clarke, K. C. and P. D. Teague (1998) "Cartographic Symbolization of Uncertainty" Proceedings,
ACSM Annual Conference, March 2-4, Baltimore, MD. (CD-ROM)

Dretske, F. I. (1999) *Knowledge and the Flow of Information.* Stanford University: CSLI Publications.

Dutton, G. H. (1999) *A hierarchical coordinate system for geoprocessing and cartography*. Berlin ; New York: Springer.

Goodchild, M. F. (1994) "Criteria for evaluation of global grid models for environmental monitoring and analysis", Handout from NCGIA Initiative 15, see Spatial Analysis on the Sphere: A Review, by Rob Raskin, NCGIA Technical Report 94-7. Copy courtesy of Waldo Tobler.

Goodchild, M. F. and Gopal, S. eds. (1989) *Accuracy of Spatial Databases*. London: Taylor and Francis.

Howse, D. (1997) *Greenwich Time and the Longitude*, London: Philip Wilson.

Kimerling, A. J., K. Sah, D. White and L. Song (1999) "Comparing Geometrical Properties of Global Grids", *Cartography and Geographic Information Systems*, vol. 26, no. 4, pp. 271-88.

Monmonier, M. S. (1995) *Drawing the line : tales of maps and cartocontroversy*, New York :H. Holt.

Mulcahy, K. A. (1999) *Spatial Data Sets and Map Projects: An Analysis of Distortion*. Ph.D. Dissertation, City University of New York: University Microfilms: Ann Arbor, MI.

Mulcahy, K. A. and K. C. Clarke. "Cartographic Visu alizations of Map Projection Distortion: A Review", Car tography and Geographic Information Science, vol 28, no. 3, pp. 167-181.

Robinson, A. H. et al. (1995) *Elements of Cartography*. New York: J Wiley. 6th. ed.

Shannon, C. (1948) "The Mathematical Theory of Communication", *Bell System Technical Journal*, July/October.

Tobler, W. 1970. "A Computer Movie Simulating Urban Growth in the Detroit Region." *Economic Geography*
46(2):234-240.

Tobler, W. R. and Z. Chen (1986) "A Quadtree for Global Information Storage", *Geographical Analysis,* vol. 18, no. 4, pp. 360-371.

Tobler, W. R. (2000) "The development of Analytical Cartography: A Personal Note," *Cartography and Geographic Information Systems*, Vol. 27, No. 3, pp. 189-194..

Tukey, J. W. (1977) *Exploratory Data Analysis.* Reading, MA: Addison-Wesley.

Universal Geospatial Data Exchange via Global Hierarchical Coordinates

Geoffrey Dutton
Spatial Effects
20 Payson Road
Belmont MA 02478 USA
t/f: 1-617-489-4524
e-m: dutton@spatial-effects.com
web: http://www.spatial-effects.com

Abstract

Hierarchical coordinates can be considered as leaf node identifiers in area quadtrees or nonary trees, numbered canonically. To achieve global coverage, the most straightforward approach is to develop triangular tessellations of a platonic solid fitted to the earth (or other spheroidal body), creating forests of quadtrees. In practice, octahedra and icosahedra seem to be the best choices for a base solid, and triangular quaternary breakdown of their faces also tends to be preferred. The hierarchical coordinate system described is based on the octahedral quaternary triangular mesh (O-QTM). Its form, topology, geometry, numbering system, nesting and areal variations are summarized. Other hierarchical global grids can be designed for geocoding and data exhange purposes, but will tend to have similar properties, advantages and disadvantages as QTM. Geospatial data interchange is then addressed. To overcome limitations and incompatibilities of local coordinate systems and spatial identifiers, QTM's use is proposed for describing positional data, using a binary notation called Loc8. Widespread adoption of this protocol will yield a more efficient, elegant and reliable geospatial data infrastructure, avoid common errors and uncertainties, and may save billions in geodata (re)processing costs. The approach can be made completely transparent to many users, promote distributed spatial data libraries and applications, and conserve network bandwidth during data transmission.

Introduction

This paper describes how a novel, scale-specific notation for geographic location can benefit processing, management, communication and display of geographic information (“geodata”). The approach uses a global multi-scale mesh that surmounts many basic difficulties in handling geodata, including:

- Geometric distortions due to map projections
- Inadequate documentation of projections and datums
- Unspecified or variable accuracy of digitized coordinate data
- Limitations to dealing with geodata at different scales
- Indexing geodata for rapid query and retrieval
- Lack of documentation of roles and significance of features and points

Many of these problems can be traced to unthinking reliance on *coordinate notation* — both planar ("x, y") and spherical ("latitude, longitude"). We model the globe as a *polyhedron*, taking shape as a *regular hierarchical triangulation* of an *octahedron* embedded in a *planet* (fig. 1). As it densifies, four child triangles develop within each existing one, and each has a unique numeric address. Thirty levels of subdivision (which can be expressed in one 64-bit word) yield child facets having areas of about 1 cm, but 20 levels (10m resolution) or fewer usually suffice to fully encode medium-scale topographic map data. The paper discusses various computational properties of the model, briefly describing how data encoding is performed and applied to generalizing maps; resolution can be coarsened by supressing less significant digits and via related techniques. We call this locational data encoding convention **Loc8** ™ (a pun, not an acronym). Figure 1 illustrates the system, showing geographic grid (yellow lines) spaced at 11.25°, which is the spacing of triangular grid points at the third level of global subdivision (black lines).

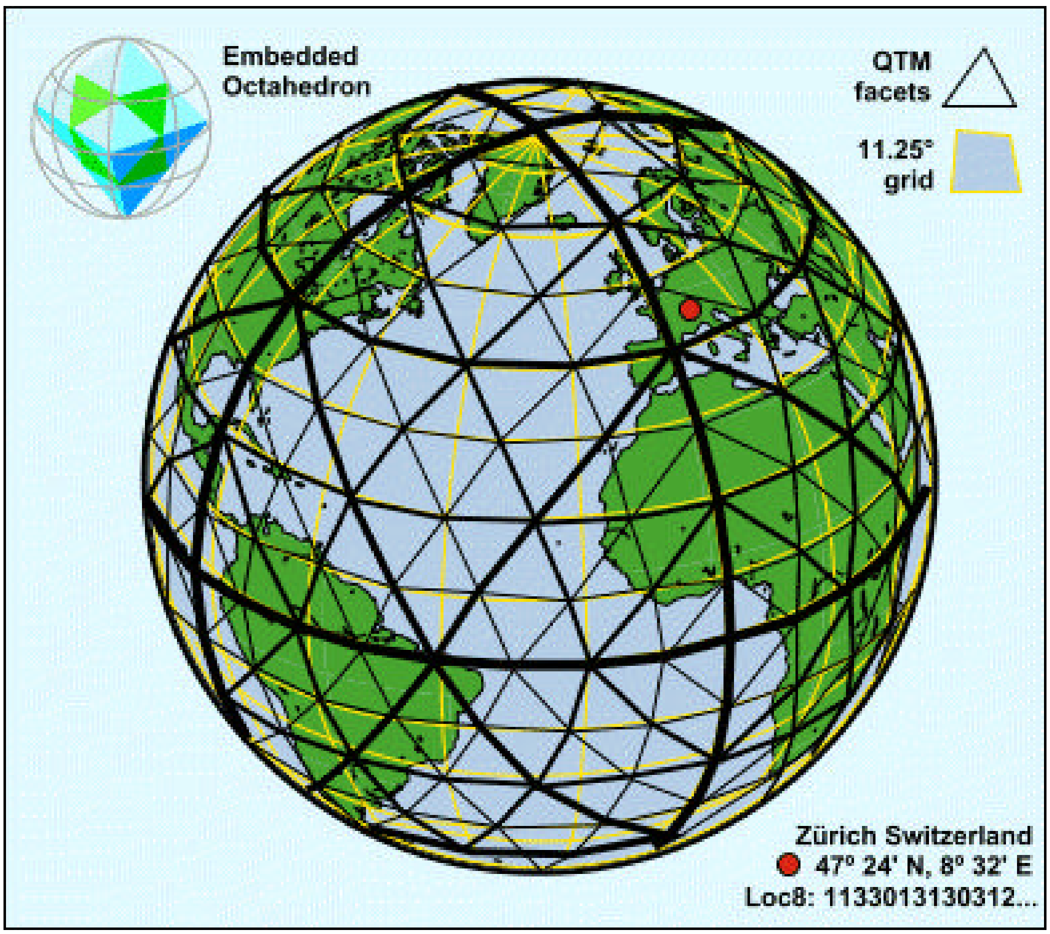


Figure 1: Quaternary Triangular Mesh level 3 with 11.25 geographic grid, showing encoded identifier containing Zürich, Switzerland

The U.S. public sector and geoprocessing industry lead the world in creating a *national geospatial data infrastructure* — related archives of digital maps, aerial imagery and spatial statistical data, much of it available via the internet and modestly priced if not free of charge. These resources are increasingly relied upon for many civil, military and commercial projects. Despite many deliberate, costly efforts to document geospatial data for consumers, geodata are often difficult to use, and their fitness for specific uses is difficult to evaluate in advance.

Other nations are also publishing geospatial data, leading to growing demands to integrate datasets across national borders to enable broader, more powerful perspectives on our planet. Despite ongoing attempts to create international geodata standards, many differences in encoding, scale and accuracy persist, and consequently geodata consumers must spend precious time and resources to understand the structure and limitations of other organizations' datasets.

Many interoperability issues stem from using a "map sheet metaphor" to structure geodata; naively mimicking paper maps in a digital environment fences our planet into countless isolated flatlands that can be spatially related to one another only indirectly, using arcane geodetic equations that may have undocumented components. We believe it possible to completely circumvent these difficulties by providing software for encoding and decoding coordinate data, enabling them to be readily shared, even between local spatial reference systems. Furthermore, by explicitly documenting accuracy of every data point, Loc8 encoding enables geodata to be relied upon at any appropriate scale, dramatically expanding their applicability and discouraging their misuse. This is accomplished through adopting *hierarchical coordinates* (HC) as an intermediary notation for positional data at a deep level.

Hierarchical Coordinates

Geodata encodes positional information in both raster and vector format. Throughout this paper we will focus entirely on vector-based encodings. In GIS databases these are usually implemented as themes, layers or feature classes, elements of which are described by geometric point, line and area primitive shapes ("primitives"), with or without descriptions of their topological relationships. Regardless of the many structural differences such datasets may exhibit, they univerally describe feature locations and geometries using lists of *coordinate tuples*; these are normally pairs of ordinates which either describe planar (x,y) positions on map projections or geographic positions (longitude, latitide) on the globe using a specific datum (sometimes an elevation is appended to each tuple). It is asserted here that many difficulties in using and sharing geodata arise from using coordinate tuple notation, which is insensitive to both scale and error, and that the source of these difficulties is rarely examined. When attempts are made to document positional error, solutions tend to be ineffective, mainly because it is not practical to describe variations in accuracy or scale in sufficient detail without vastly enlarging and complicating geo-datasets. This is because external metadata (of which positional data quality is an important subset) does not scale well as complexity of geo-datasets increases.

To deal with this problem directly, we feel it is necessary to abandon coordinate tuple notation and in its place substitute hierarchical coordinates. These are location identifiers that zoom in on geographic locations by subdividing their spatial domain in a hierarchical, basically recursive manner. One can visualize this as a process of cutting a map sheet into four equal rectangles, numbering each one 0, 1, 2 or 3 in a specific order; a point location of interest will be found on one of the four sections, and this section is cut into four similar ones which are numbered like the four created in the prior step. At each stage uncut pieces are discarded, and the procedure is repeated until the snippet of map remaining is small enough to identify the location without ambiguity. Then the numbers written on the retained pieces are written down, ordered left to right from largest to smallest, as a string of digits. This *quaternary number* identifies the location of interest to a certain (presumably adequate) level of precision, and constitutes an HC.

Notice that, as opposed to a coordinate pair (tuple), a hierarchical coordinate is a single quantity, and thus can be stored in one memory location rather than having to occupy two. This provides a more straightforward way of querying by location, as there is only one key hence one sorting order for coordinates. Other important aspects of planar HCs include:

- The size and coordinate origin of the spatial domain to be subdivided
- The partitioning geometry (2x2, 3x3, ... MxN)
- The spatial ordering of numeric partition labels

The above example describes what is known as a quadtree, of which many variants exist and which is not at all new to geoprocessing. Samet's texts (Samet 1990a,b) provide the classic overview of quadtrees and their applications. As the second bullet above implies, dissections of space other than quaternary ones are possible (although rarely used). Less well appreciated is the fact that quadtrees and their relatives need not tessellate space orthogonally into rectangles, but may also use *triangular* quadrants. This alternative geometry offers useful possibilities, some of which the remainder of the paper will discuss.

The specific hierarchical coordinate system described here is called *quaternary triangular mesh* (QTM).[1] This model has evolved from one designed for digital elevation modeling (Dutton 1984), subsequently recast into its current form to facilitate global spatial indexing (Dutton 1989; Goodchild, Yang and Dutton 1991; Goodchild and Yang 1992; Otoo and Zhu 1993), positional data quality management (Dutton 1990, 1992, 1996), terrain modeling (Lugo and Clarke 1995) and map generalization (Dutton and Buttenfield 1993; Dutton 1999). A Ph.D. dissertation (Dutton (1998) provides an overview of QTM's development, properties and its application to map generalization. To create a QTM, an octahedron is aligned to the earth with its six vertices at cardinal points, fitted to an ellipsoid describing a global datum such as WGS 84. The process of hierarchically identifying locations begins by dividing any of its eight faces at edge midpoints to define four child triangular facets, as figures 1 and 2 illustrate.

---

[1] For further information about QTM, please see http://www.spatial-effects.com/SE-research1.html

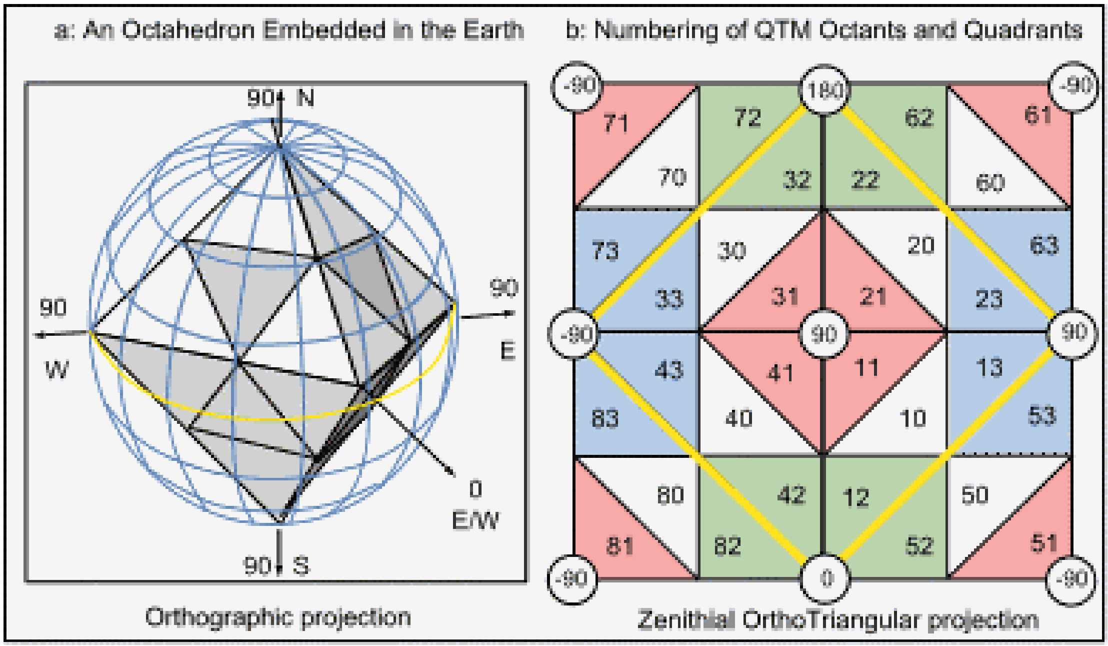


Figure 2: Quaternary Triangular mesh form, orientation and level 1 quadrant numbering

Generating hierarchical subdivisions simply requires computing arithmentic averages of the latitudes and longitudes of vertices. Note that all such first-level (and higher level) facets -- although identical on the octahedron -- vary slightly in size and shape when the new vertices are projected to the spheroidal surface (local relief, were it to be encoded on top of this, would further distort facets). Subdivision may be recursively repeated as many times as are needed to locate positions on the ground: 10 levels of subdivision produce facets about 10 km wide, 20 levels have 10 m resolution, 25 levels have 1 m resolution and 30 levels yield 1 cm facets. By consistently numbering the original eight faces and their children, every facet – no matter how small – will have a unique integer identifier (a base8 digit followed by some number of base4 digits). Longer QTM identifiers (QTMIDs) are more precise than shorter ones, as resolution improves by powers of two. A set of QTMIDs thus is equivalent to a linear quadtree (Gargantini 1982), a set of leaf node identifiers that specify the tree's branching structure by implication only. Interior nodes are not needed, as the only information needed to decode any given QTMID into coordinates that define that facet (or define a representative point for it) are:

- The orientation of the octahedron within the planet;
- The ordering of the octahedron's eight faces;
- The algorithm used to assign child facet numbers (quadrant IDs);
- The number of significant digits the decoding should use, and
- A reference ellipsoid for the region, if required.

Given these parameters and an algorithm, precise geographic coordinates of its defining triangle can be recovered from any QTMID. If the geocode represents a very small feature, it may also alias other small ones nearby if the depth of encoding is not sufficient to distinguish their locations. This could be problematic if data density varies considerably, especially when a fixed level of QTM resolution is used to geocode a large population of features. Of the above five elements, the third one is potentially most confusing. Several schemes have been proposed for numbering QTM facets, each of which produces different geocodes for a given location (Fekete 1990; Otoo and Zhu 1993; Lugo and Clarke 1995; Lee and Samet 1998). However, all but one of these encoding implementations traverse identical sets of triangular facets, because all align the octahedron in the same fashion (note that while Fekete's numbering system is congruent with the author's, his trees are rooted in an icosahedron, as are Lee and Samet's). Therefore, even if different algorithms are used to encode a given feature, a direct 1:1 mapping between the resulting geocodes exists (Fekete's and Lee and Samet's models excepted, again because of the different base solid).

Structure and Encoding of Loc8 Metadata

Enriching existing geodata to use Loc8 hierarchical coordinates involves several steps. Given a geo-dataset (e.g., feature class or layer) described using vector coordinate tuples...

1. Terrestrial positional error (variable or constant) is estimated for every spatial entity
2. Coordinates are deprojected to latitude and longitude on a geocentric ellipsoidal datum (when necessary)
3. Metadata describing types and roles of features and/or specific points are compiled
4. For each entity, geo-coordinates are converted to QTM addresses of specified precision
5. A core Loc8 identifer is constructed for each point containing its QTM address
6. Qualitative metadata applicable to the point is encoded and attached to the Loc8 ID
7. The Loc8 identifier is binary-encoded as a 64-bit word (or successive 32-bit words)
8. The dataset is updated one feature at a time, replacing coordinate tuples with Loc8 IDs

This procedure results in modifying the database to replace all coordinates with binary Loc8 identifiers that capture positional accuracy and selected qualitative attributes. The result is to consolidate positional accuracy and related qualitative metadata parameters into one place, rather than having them specified at each layer, feature class or feature (current GIS data models make feature-level specification quite cumbersome, hence it is rarely attempted). The differece between current practice (in which metadata occupies specially-designated database fields or external files) and how Loc8 can encode positional accuracy or certainty is diagrammed in figure 3.

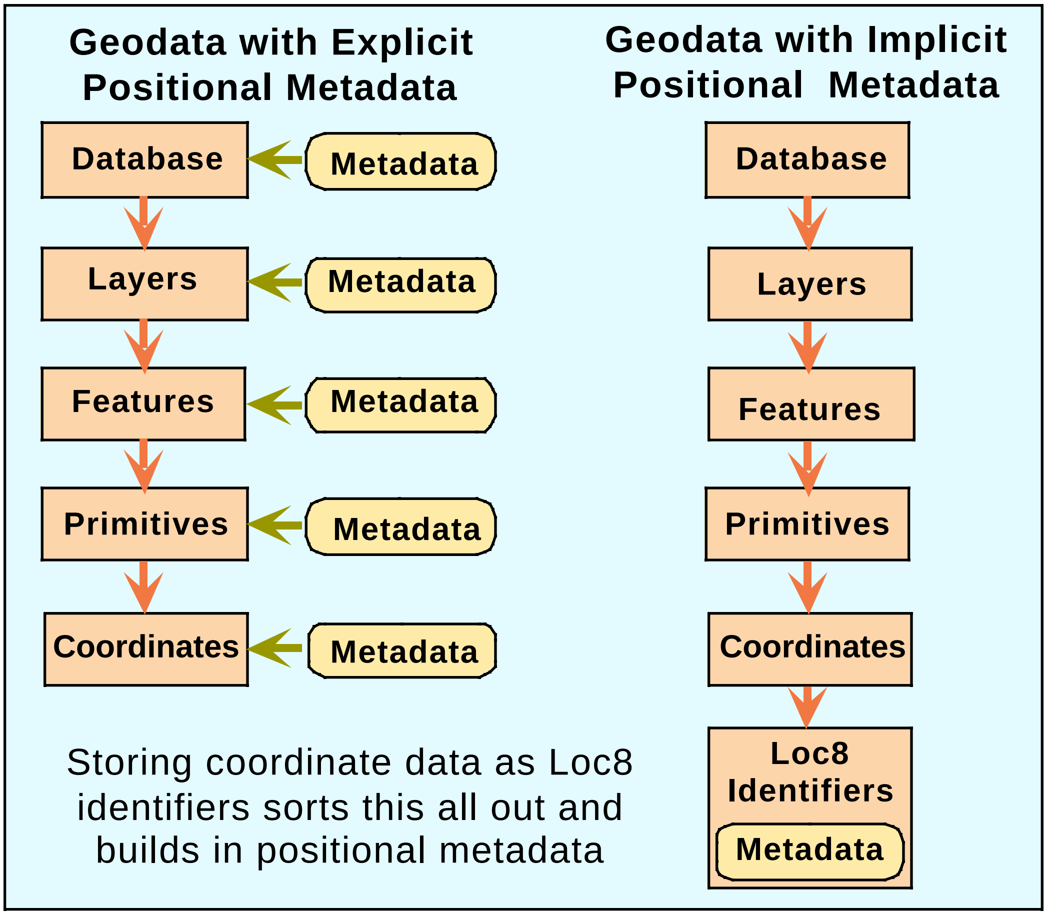


Figure 3: Use of Loc8 Identifiers to encapsulate positional accuracy information

As each Loc8 ID occupies 64 bits, the amount of qualitative metadata that can be included is inversely proportional to the precision of QTM encoding. The exact amount is given by the relation:

$$MB = 58 - 2L,$$

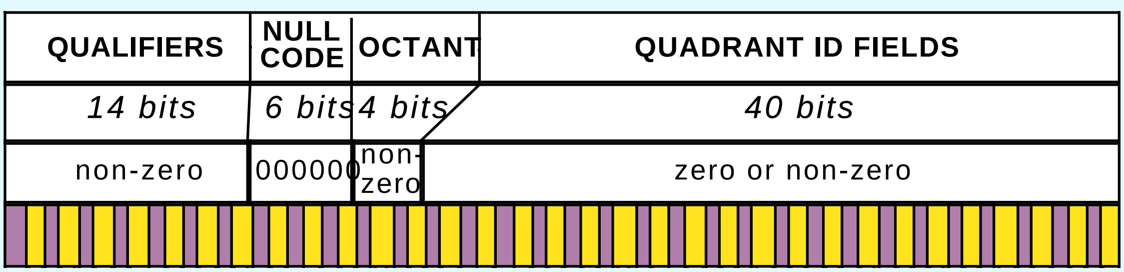
where MB is the number of metadata bits available and L is the number of QTM hierarchical levels of encoding a given point requires. Thus a 20-level encoding (c. 10 m resolution) leaves 18 bits available for each points's metadata, and a 24-level encoding (c. 0.5 m) allows for 10 bits of metadata. Note that the level of QTM resolution is implicit, requiring no bits of storage. As QTM resolution doubles at each level, it is simple to substitute resolution for L in the above equation, given the rule of thumb

$$L = \mathrm{ceil}(\log_2(5.0e6 / R_m)),$$

where 5.0e6 is one-eighth of the earth's circumference in meters (i.e., octant edge length), and R_m is the resolution desired, also in meters. Thus given a spatial resolution in meters, the number of metadata bits available to further qualify such points is by substitution:

$MB = 58 - 2\ \mathrm{ceil}(\log_2(5.0e6 / R_m))$

Ten to 20 bits may not seem like very much space in which to encode attributes or meta-data, but quite a lot of information can be built into such qualifiers. Dutton (1996, 1999) provides examples of information that can be organized into such qualifier fields, aimed at augmenting QTM IDs to aid map generalization operations. The qualifiers described are each coded as 2-bit quantities which are restricted to have only three values (01, 10, 11); the pattern 00 is excluded because it is used as a sentinel to terminate qualifier sets. This means that any number of ternary qualifiers, up to MB/2 or less, can be encoded in a Loc8 ID, or no qualifiers may be stored. Parsing the qualifiers is simple if they are known to use 2-bit ternary encoding; starting at the highest-order bit, successive pairs of bits are examined. Nonzero pairs are qualifiers, and their values and positions are noted. The first zero-value bit pair encountered indicates that all qualifiers have been parsed, and the next nonzero bit-pair encountered is interpreted as the first half on a QTM octant ID (the octant ID occupies 4 bits, and when it is parsed 4 is subtracted from the value stored; the left shift is introduced in order to avoid placing 00 in the two high-order octant ID bits). The remaining low-order bits in the Loc8 ID are all significant 2-bit QTM quadrant IDs. A parsing algorithm is described in (Dutton 1996). An example of the bit-level encoding that results is illustrated in figure 4.



Some possible locational Qualifiers and encoding conventions

QUALI-FIER	SUBJECT MATTER	Code_10	Code_01	Code_11	PROPERTY CONVEYED
1	Vertex Type	Node	Point	Mixed	Topological role
2	Vertex Origin	Defined	Imputed	Mixed	Locational importance
3	Vertex Angle	Sharp	Smooth	Mixed	Local angularity
4	Feature Genus	Point	Line	Area	Topological dimension
5	Feature Origin	Cultural	Natural	Mixed	Source of entity
6	Feature Width	Fixed	Variable	Mixed	Uniformity of extent
7	Feature Priority	High	Low	Medium	General malleability

Figure 4: 64-bit word layout of a Loc8 Identifier having seven ternary qualifiers

While the above describes qualifiers as two-bit ternary fields, they can have any even number of bits; using two bits seems to offer the most flexibility, especially when QTM resolutions are allowed to vary. The Loc8 format enables the lengths of qualifier fields, the number of fields and the number of quaternary digits to vary point by point. As the number of qualifiers in a given Loc8 ID may not be known in advance, and may change from one feature or even one point to the next, their field ordering is significant and rules are needed to interpret them correctly. The rules can be rigid or flexible, as the following illustrate:

- **Rigid:** Each qualifier is two bits, and contains 01, 10 or 11; interpretation depends only on absolute position (i.e., bits 1-2 are qualifier A, bits 3-4 are qualifier B ...); where space restricts their number, “least significant” qualifiers are omitted.
- **Flexible:** Some qualifiers imply that others follow directly, and certain ones might be larger than two bits (i.e., ordering/size is partly contextual).

Whatever scheme is adopted, it needs to be documented in some fashion so that recipients of Loc8 data can parse and interpret qualifier fields (in the absence of such directives the QTMID portion of Loc8 words can still be retrieved, however). Just as QTMID lengths can vary between and even within features or not, the meanings of qualifiers may change, or may remain fixed across an entire dataset. In all cases, some sort of higher-level metadata (akin to a feature attribute table or feature coding guide) needs to be provided at the appropriate level of detail within datasets. This paper will not pursue such concerns beyond pointing out that such higher-level metadata must indicate:

- Names and descriptions of each qualifier
- Sizes of specific qualifiers (number of bit-pairs)
- Descriptions of what specific qualifier values mean
- Ordering or priorities assumed for qualifiers
- Dependencies between qualifiers, if any

At least two types of dependencies may occur. Type I dependency enables qualifier i, by its value, to indicate what the identity of qualifier i+1 is; type II dependency enables qualifier i, by its value, to modify the interpretation of the values of qualifier i+1 (the identity of which would be given by its position in the set). Such metadata must be provided in both machine- and human-interpretable form.

Including qualifiers in Loc8 words is not mandatory of course, and their presence can always be ignored. While handling qualifiers adds to the complexity of creating and using Loc8 data, making such internal metadata available provides a distinct opportunity to geoprocessing communities to add real value and remove sources of potential error from geodata. Furthermore, the content and application of Loc8 qualifiers can vary across datasets, applications and knowledge domains. Qualifiers can be encoded as simple attributes, or indicate specific operations or behaviors. One can think of a Loc8 word as an object that describes its behavior or how it should be used, i.e., as containing an operator and an operand. Methods invoked on a Loc8 object’s data can be specified by the operator portion, or may be defined for the class to which a Loc8 object belongs. This provides a high degree of flexibility in applying Loc8 to encode and use semantic knowledge at very detailed degrees of specificity.

General Benefits of Loc8

Were Loc8 encoding to be adopted for data transmission and database storage, it could simplify a host of applications and enable them to become more transparent, capable and reliable. This will occur beneath users' notice, although many will appreciate enhancements it will bring to data access and productivity. How this can happen is described below. What it would specifically mean for users is difficult to enumerate, but involves at least these three "U"s:

- Usability: encoding a geographic position into Loc8 requires its accuracy to be explicitly stated; this critical metadata is built into all Loc8 data and is available to inform applications of limitations to analysis or dislay.
- Universality: Loc8 handles all locations equally well, regardless of where they lie or what national, state or local coordinate reference systems may exist there; to the extent they are mapped, other planetary bodies can be treated exactly the same way.
- Uniqueness: Although a Loc8 identifier is shared by all locations within a specific triangular patch of ground or water, if that triangle excludes all competing entities, such an address can fully identify a specific feature, and can be enhanced to specify characteristics of it besides location. Thus Loc8 enables construction of globally unique geocodes (similar to postcodes).

Loc8 by itself will not provide complete interoperability, as it constitutes only part of a protocol or standard for handling geodata; other necessary elements include geometries, images, topologies, attributes, relationships, labels, semantics and metadata. While many geodata exchange standards have been proposed and adopted, few standards exist for internal database architectures or interfaces between applications and databases. Much of the progress in this area is being made and reported by the Open GIS Consortium (OGC, Wayland MA, http://www.ogc.org/), which coordinates its activities with industry and standards bodies such as NIST and ISO. By injecting Loc8 into OGC's process — aiming at having it supported as an alternative encoding method — geoprocessing standards can both anticipate and promote Loc8's diffusion into practice as OGC's work starts to impact geoprocessing during the next five years or so.

Unresolved Issues

Several prototypes of Loc8-based software have been built. The author's testbed encodes coordinates of map features, analyzes the encodings and regenerates the features at specified scales using special techniques. This software (written in C) is not fit for general use, but has proven the validity of the concept. An earlier C-language prototype developed by academic colleagues (Goodchild and Yang 1992) successfully demonstrated applicability of a Loc8 variant to common operations on spatial data, but was abandoned when funding expired. Some issues these projects did not address that still need to be explored are:

Bit-level Representation of Loc8 geocodes. To date, software prototypes have represented Loc8 identifiers as strings of characters, as this made handling and debugging them simpler. But *binary encoding* such as is described above is needed to make Loc8-based software more efficient and capable. As Loc8 geocodes need to carry 64 bits of precision, their binary representation is problematic; until recently computer hardware, programming languages and operating systems only supported such quantities in restricted and clumsy ways. These limitations are receding, and settling on an internal format for binary Loc8 codes is thus a high priority. Doing this is not technically difficult, but the protocol must be robust, extensible to include additional identifying information and acceptable to the potential user community. Hence careful multi-sector requirements analyses are needed.

Extended Object Spatial Indexing. Fulfilling Loc8's promise as a spatial indexing mechanism entails developing methods and structures to efficiently represent the locations, sizes and shapes of geographic entities, which range from fenceposts to continents, and may be neither compact nor regular. Several approaches to this involving collections of Loc8 facets have been conceptualized, and need to be implemented and tested. Whether Loc8 is widely accepted may depend strongly on how efficiently software can insert and retrieve many varieties of spatial entities in a "Loc8-native" database.

Native Spatial Analysis Methods. Existing methods to determine spatial properties of Loc8-encoded data need to be augmented, as few such algorithms exist for data in a triangulated polyhedral domain. Examples of spatial property measures include: computing facet areas, finding distances between facets, determining what facets lies within a region and what facets lie between two points. We can perform such operations in Cartesian and spherical domains, but doing so requires converting Loc8 geocodes to coordinates, which is inefficient. Fast, appropriate native methods need to be developed. The most general technical challenge faced is to integrate two very different computational domains, vector (geometry-oriented) and raster (pixel-oriented). Existing GIS packages combine them in certain ways, but in databases the two types of representation are always maintained separately (hence the wisecrack, "raster is faster but vector is correcter"). Our experience encourages us that Loc8 can bring these domains closer together at rather low levels, and this will greatly facilitate fusion of disparate types of geodata (such as digital maps and satellite images), a common yet complex geoprocessing requirement.

Implementing Loc8 in GIS Environments

The era of proprietary database formats for geographical and other information is coming to a close. With few exceptions, major database vendors now provide API protocols for remote access by foreign applications. Besides being voluminous, geographical data can be highly complex and structured idiosyncratically; thus interoperability has improved in rather small steps, but progress is accelerating thanks to continuing industry and governmental efforts to codify geoprocessing standards.

It is this context that Loc8-enabled software can address, providing solutions to nagging problems that are well-known but have yet to be appropriately addressed. One design for implementing Loc8 as middleware is shown in figure 5. The diagram depicts how Loc8 data is encoded, transmitted between applications and decoded for use in a transparent fashion (users may need to indicate or supply relevant metadata, and should be sophisticated enough to exploit the stored metadata in analysis and interpretation). The data interface link at the bottom of figure 5 shows how Loc8 databases or extracts of them can be shared, but a given GIS would normally possess both encoding and decoding capabilities.

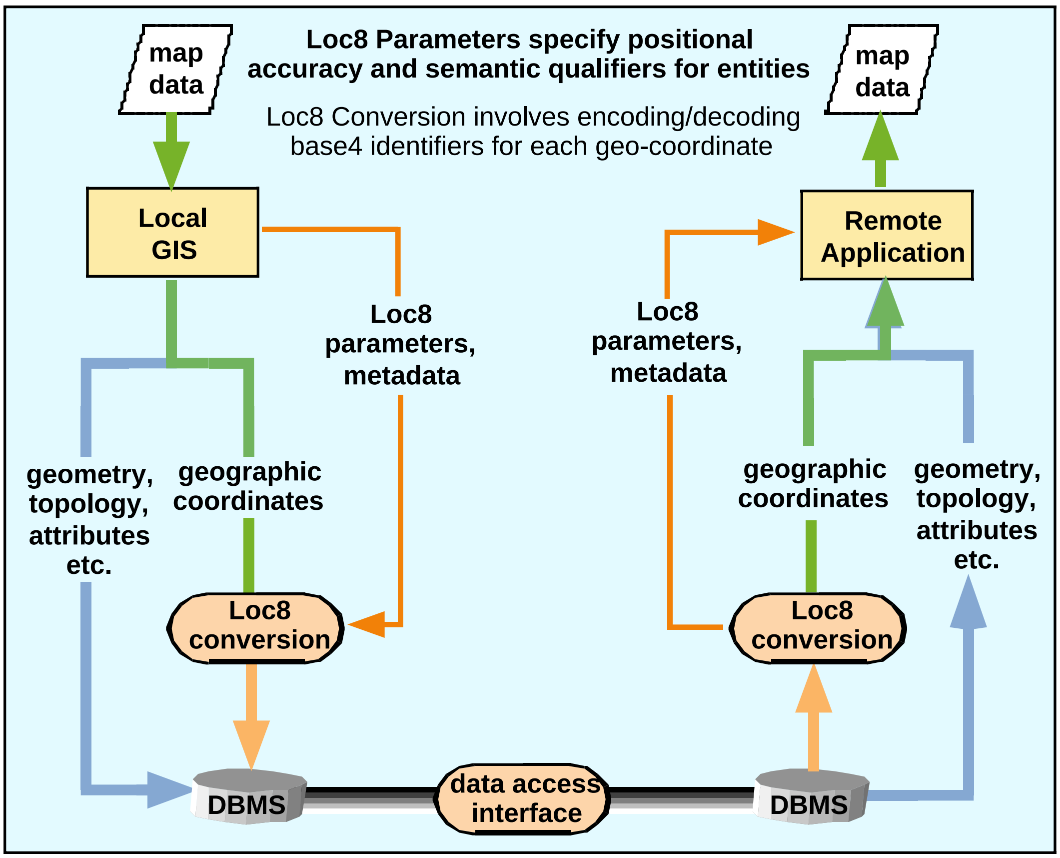


Figure 5: A possible Loc8 geodata conversion architecture

Such a strategy makes sense because (1) translating coordinates to and from Loc8 identifiers should not affect users' views of geodata, (2) GIS software and other applications need not know how to interpret and manipulate Loc8 encodings, and (3) database mechanisms and schemata should be modified as little as possible to handle Loc8 data. The middleware layer will be imposed in between the database and the application; it will encode coordinates as Loc8 IDs based on metadata suppled by the user via a manual interface, file and/or external database. The encoded coordinates will occupy the fields in the database previously allocated to feature coordinates; these fields may need to be redefined in the schema if they were initially described as containing floating-point tuples. The sizes of existing coordinate data fields may change when Loc8 encoding is implemented, but should never grow larger (e.g, two 32-bit fields become one 64-bit field, or two 64-bit fields become one 64-bit field, in which case the unused space can be reclaimed). The Loc8 software layer will receive and emit coordinate and metadata data via an application programming interface (API). In turn, it will communicate to geospatial databases via APIs appropriate to the specific store involved. Flags can be set for features, classes or layers to indicate that Loc8 encoding is present, and schemata inserted to identify whether and how metadata are described within Loc8 fields.

Concluding Observations

The overarching impediment to the adoption of Loc8 is a general complacency and lack of awareness about the inherent limitations of coordinate notation. For years, GIS communities have avoided confronting those problems, apparently preferring to expend millions of hours generating metadata, doctoring deficiencies in one geographic dataset after another, and then waste more time trying to cope with consequences of uncertain data quality to spatial analysis and display. Is this smart? Is it necessary? We think not.

Achieving acceptance for Loc8 is an uphill struggle, because nothing like it has ever been in general use and no rival schemes exist that might supply similar benefits. Some technical objections to this approach may be argued by individuals with expertise in geodesy and surveying. Such feedback is welcomed, and may be useful as we refine our approach. The principal challenge faced, however, is not simply technical; we must convince geoprocessing constituencies of the need for and value of Loc8. This is presently difficult because very few developers and users in geodata communities recognize the impediments erected by reliance on coordiinate tuple notation.

One further limitation to the general use of Loc8 is the prevalence of geodata with projected (planar) rather than geographic (spherical) coordinates. As Loc8 encoding can only be performed from spherical notation, only planar geodata for which inverse projection equations to latitude and longitude are available can be handled. Many spatial datasets — particularly local working data files for applications — do not describe their projection types and parameters. If applications are not sophisticated in their handling of coordinate data, they are not likely to be able to use Loc8 directly. We will initially deal with this by concentrating on inserting Loc8 earlier in the “geodata food chain,” and by encouraging development of data translation utilities to nourish less capable applications.

No map or geographic database can be truly faithful the richness of the world, to its majesty and diversity in space, time and human understanding. GIS professionals confront these limits constantly in designing databases and maps, manipulating and analyzing geodata, and attempting to adapt it to changing needs. Using digital lenses to view the world distorts it in subtle ways that practitioners don't always understand. The technology is thus far from neutral in its consequences, and our growing dependence upon it creates new kinds of spatial effects -- good, bad and often unknown -- not just in models and on monitors, but in the real world itself. We worry about this, have dedicated years to improving the handling and communication of geodata, and are committed to bringing higher standards to geographic information and its application. Creating a global multi-scale geodata infrastructure will be a huge step toward this goal.

References

Dutton, G., 1984, Geodesic modelling of planetary relief. *Cartographica.* 21(2&3), 188-207.

Dutton, G., 1989, Modelling locational uncertainty via hierarchical tessellation. In M. Goodchild & S. Gopal (eds.), *Accuracy of Spatial Databases,*. (London: Taylor & Francis), 125-140.

Dutton, G., 1990, Locational properties of quaternary triangular meshes. *Proc. Spatial Data Handling Symp. 4.* Dept. of Geography, U. of Zurich (July 1990), 901-10.

Dutton, G., 1992, Handling positional uncertainty in spatial databases. *Proc. Spatial Data Handling Symp. 5.* Charleston, SC, August 1992, 2, 460-469.

Dutton, G. and Buttenfield, B.P., 1993, Scale change via hierarchical coarsening: Cartographic properties of Quaternary Triangular Meshes. *Proc. 16th Int. Cartographic Conference.* Köln, Germany, May 1993, 847-862.

Dutton, G., 1996, Improving locational specificity of map data: A multi-resolution, metadata-driven approach and notation. *Int. J. of GIS.* London: Taylor & Francis, 10(3), 253-268.

Dutton, G., 1998, *A hierarchical coordinate system for geoprocessing and cartography.* Lecture Notes in Earth Science 79. Berlin: Springer-Verlag. XIX + 231 pp. 97 figs., 12 plates, 16 tabs. ISSN 0930-0317; ISBN 3-540-64980-8.

Dutton, G., 1999, Scale, sinuosity and point selection in digital line generalization. *Cartography and Geographic Information Science* 26(1), 33-53.

Fekete, G., 1990, Rendering and managing spherical data with sphere quadtrees. *Proc.Visualization '90* (First IEEE Conference on Visualization, San Francisco, CA, October 23-26, 1990). Los Alamitos CA: IEEE Computer Society Press.

Gargantini, I., 1982, An effective way to represent quadtrees. *Comm. of the ACM,* 25(12), 905-910.

Goodchild, M.F. and Yang Shiren, 1992, A hierarchical data structure for global geographic information systems. *Computer Graphics, Vision and Image Processing,* 54(1), 31-44.

Goodchild, M., Yang Shiren and Dutton, G., 1991, Spatial Data Representation and Basic Operations for a Triangular Hierarchical Data Structure. Santa Barbara, CA: NCGIA Technical Paper 91-8.

Lee, M. and Samet, H., 1998, Traversing the triangle elements of an icosohedral spherical representation in constant-time. *Proc. Spatial Data Handling Symposium '98.* Vancouver Canada, July 1998, 22-33.

Lugo, J.A. and Clarke, K.C., 1995, Implementation of triangulated quadtree sequencing for a global relief data structure. *Proc. Auto Carto 12*. ACSM/ASPRS, 147-156.

Otoo, E.J. and Zhu, H., 1993, Indexing on spherical Surfaces using semi-quadcodes. *Advances in Spatial Databases. Proc. 3rd Int. Symp. SSD'93*, Singapore, 510-529.

Samet, H., 1990a, *The Design and Analysis of Spatial Data Structures.* Reading MA: Addison-Wesley.

Samet, H., 1990b, *Applications of Spatial Data Structures: Computer Graphics, Image Processing and GIS.* Reading MA: Addison-Wesley.

Real-Time Global Data Model for the Digital Earth

Nickolas Faust[1,2], William Ribarsky[2], T.Y Jiang[2], and Tony Wasilewski[1]
[1]GIS Center and [2]GVU Center
Georgia Institute of Technology

ABSTRACT

A global data model is presented for the digital earth that provides scalability and fast access. It is shown how the same global hierarchical structure can be used for a wide variety, and perhaps all types, of geospatial data. To maximize efficiency, particular types of data can, at prescribed levels of detail, transition from the global model to type-specific data structures and detail management schemes. This framework has been applied successfully to a variety of data including terrain elevations, phototextures and imagery, maps, buildings, moving or flying vehicles, weather, and other data. Further, the framework provides a geospatial visual data mining approach where one can navigate continuously from global overviews to high resolution local views. This paper presents results for several applications.

Keywords: visualization, global terrain, data model, digital earth, decision-support, large scale data, GIS

1 INTRODUCTION

To attack the problem of dealing with increasingly vast stores of global information for the digital earth, we describe in this paper a general approach to data organization and real-time exploration based on a novel global hierarchical data model. Our recent work has revealed that this framework can be quite generally applied to the earth and anything on it, above it, or even below it. This includes terrain elevations, phototextures and imagery, maps, buildings, moving or flying vehicles, weather, and other data. Further, the framework provides a geospatial visual data mining approach where one can navigate continuously from global overviews to high resolution local views. Because it has an efficient data organization and paging capability closely coupled to a view-dependent data requesting mechanism (that is, it is adjustable based on the requirements of the display platform), the framework is also quite flexible and has been applied to a range of single and networked systems ranging from a single-processor PC to immersive systems with multiple projection screens and coupled computers. In this paper we will discuss our geospatial model, how it fits into an overall framework, and how we have applied it to different types of data in different environments. We will also show that the model is applicable to distributed and Internet-based systems and will discuss our work in that direction.

GVU Center, Georgia Tech, Atlanta, GA 30332-0280
Email: {nick.faust, tony.wasilewski}@gtri.gatech.edu {ribarsky, jiangf}@cc.gatech.edu

For the following reasons our approach offers a general framework for digital earth applications:

- It accepts and integrates all types of geospatial data into a global framework.
- It is scalable to very large data stores and to distributed environments.
- It provides an interactive visualization framework.
- It supports discovery of new information via navigation in context with unfolding detail.

We have implemented these capabilities in a system called VGIS [Lin96, Lin97].

Our approach starts with a hierarchical structure for optimal interactivity for data exploration. We use a novel "forest of quadtrees" with nested coordinate systems for handling global scale terrain data and permitting timely paging of collections of objects in support of real-time navigation. We have found that one can effectively employ this hierarchical structure for a wide range of geospatial data as long as methods adapted to the specific type of data (e.g., terrain versus buildings) are applied at the lower (detailed) levels.

An important aspect of our approach is the idea of a “view” and “view-dependence”. We impose a viewpoint and a limited viewing window on the data presentation. In general there are things within the view and outside the view, things that are close or far away, and things that may be obscured because they are blocked by other things. By view-dependence, we mean that we organize the retrieval of data by these view-dependent factors. Thus we might only retrieve the most prominent details for data that are far away and might not retrieve at all data that are out of view. Of course this approach is useful and efficient for visualization, but it is also a powerful organizing principle for all data handling (for example, GIS data queries). Even for these operations there is an implied viewpoint and viewing window. This approach also supports the ideas of navigation and context. Thus we access details in extended datasets by navigating the data, and we do this while seeing surrounding contextual information that isn’t our main focus but can still provide important supporting details.

In this paper, we present these ideas in detail. We also describe how large collections of objects, such as buildings and weather clouds, can efficiently fit, along with terrain, into the geospatial hierarchy. Application results demonstrate that our approach is both scalable and general because it is able to handle both large scale global terrain information and multiple collections of objects (e.g., cities and weather data) placed around the earth with full interactivity and without extensive memory load. Finally, our approach shows that levels of detail can be naturally incorporated to provide improved detail management.

2 RELATED WORK

There has been a large body of previous work that has considered how to organize and visualize geospatial data. We will not attempt to cover all this work here but rather will focus on the most relevant research and applications. Representing data that span the entire globe requires special data structures that efficiently account for the earth's curvature. Most of these are based on quadtrees [Sam84]. Fekete has developed spherical quadtrees [Fek90] as a global representation for the earth. The main difference between Fekete's method and ours is that he tesselates the earth's surface with a a subdivided icosahedron, while VGIS uses "square" geodetic patches, whose geometry conform to the lat/lon representation commonly used for terrain elevation and imagery datasets. Similar hierarchical data structures have been shown to be useful for global GIS [Goo92].

Much work on the tesselation of terrain has concentrated on triangulated irregular networks (TINs). A number of different approaches have been developed to create TINs from height fields using Delaunay and other triangulations [Gar95, Sch94]. In addition hierarchical triangulation representations have been proposed that lend themselves to multiresolution level of detail (LOD) algorithms [DeB95, DeF95]. Regular grid tesselations, which is the form used in our approach, have also been implemented as terrain approximations [Cos90, Duc97]. Although recent work shows that improved data compression can be attained with TINs [Kim99], we feel that there remains a significant advantage in using regular grids in both dataset compactness and efficiency of dataset building. (See Sec. 4 and [Lin97])

VGIS treats the terrain as a single, connected surface for rendering using a "continuous, view-dependent" LOD representation. Similar approaches are described in in [Cos90, Duc97, Hop98, Xia96]. However, VGIS is the first system to support support real-time view-dependent, pixel-accurate multiresolution rendering of high resolution global datasets.

A number of visualization systems have been implemented that integrate 3D visualization techniques with large geographic information and terrain data. Some systems stress accurate rendering of global images, or accurate modeling of environmental processes, often sacrificing interactivity of the system [Nis93]. Other systems emphasize tight integration of the 3D visualization with the powerful spatial analysis capabilities of GIS [Erv93]. VGIS enables real-time, highly interactive 3D visualizations of the spatial data by building these capabilities into the system from the bottom up.

Recently there has been work that addresses interactive visualization of very large, out-of-core datasets in applications from 3D flow visualization to large scale terrain visualization. From what has been done so far, it is clear that application-control and domain-dependent data organization are essential to achieving good performance [Uen97, Cox97]. Relying on system virtual memory, for example, frequently results in thrashing and abysmal performance. Often a spatial and hierarchical partition of the dataset is used so that one can efficiently load only needed segments. All of these out-of-core techniques consider the paging of continuous volumetric or terrain data. VGIS fits with these application-controlled out-of-core techniques but, unlike the others, it considers more diverse data such as terrain, time-dependent volumetric data, and collections of discrete 3D objects.

3 AN EFFICIENT GLOBAL HIERARCHICAL STRUCTURE

We have built a quadtree and shared cache data structure that together constitute the basic components of our global data model. The global structure is divided into 32 zones, each 45^{o} x 45^{o} [Dav98, Lin97]. (See Fig. 1.) Each zone has its own quadtree; all are linked so that objects or terrain crossing quadrant boundaries can be rendered efficiently. We chose the number and extent of zones based on empirical observations of memory requirements, paging overhead, geometric accuracy, etc. A node in a quadtree corresponds to a raster tile of fixed dimensions and lat/lon resolution according to the level on which it appears in the quadtree. Quadnodes are identified by "quadcodes," which are built in a manner similar to the indices of representations of binary trees, that is, the children of a node with quadcode q are identified by $4q + 1$ through $4q + 4$. In addition, the quadcode contains a quadtree identifier that allows each quadcode to uniquely identify an area on the globe. This structure is replicated in the underlying disk management system so that files are aligned with the quadnodes in the set of linked quadtrees

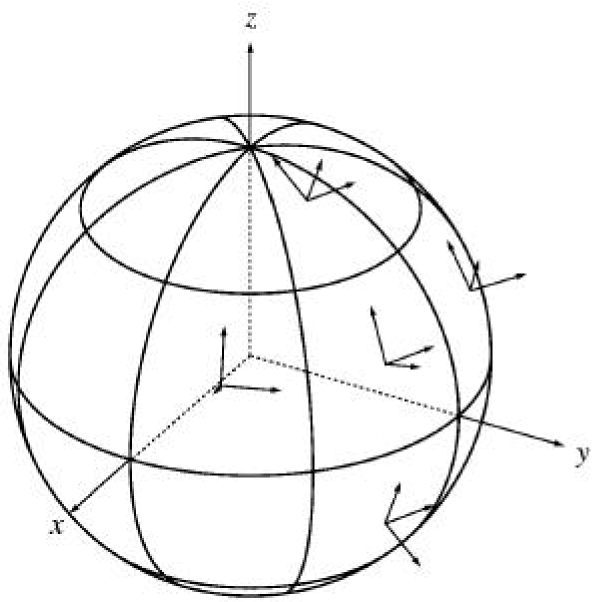


Fig. 1 The Earth divided into 32 zones.. The labeled axes correspond to Earth Centered, Earth Fixed Cartesian XYZ global coordinate systems for each zone.

The quadtrees also define the boundaries of local coordinate systems. If a single, geocentric coordinate system were used, assuming 32-bit single precision floating point is used to describe object geometries, the highest attainable accuracy on the surface of the Earth is half a meter. Clearly, this is not sufficient to distinguish features with details as small as a few centimeters, e.g. the treads on a tank. As a matter of fact, some of the terrain datasets that have been used in VGIS have 10 centimeter resolution. This lack in precision results in "wobbling" as the vertices of the geometry are snapped to discrete positions, which is present in other large scale terrain systems such as T_Vision [Gru95]. We have developed a new approach to overcome this problem [Lin97]; we define a number of local coordinate systems over the globe, which have their origins displaced to the (oblate) spheroid surface that defines the Earth sea-level. The origins of the top-level coordinate systems are placed at the geographic centers (i.e. the mean of the boundary longitudes and latitudes) of the quadtree roots. While the centroid of the terrain surface within a given zone would result in a better choice of origin in terms of average precision, we decided for simplicity to opt for the geographic center, noting that the two are very close in most cases. The *z* axis of each coordinate system is defined as the outward normal of the surface at the origin, while the *y* axis is parallel to the intersection of the tangent plane at the origin and the plane described by the North and South poles and the origin. That is, the *y* axis is orthogonal to the *z* axis and locally points due North. The *x* axis is simply the cross product of the *y* and *z* axes, and the three axes form an orthonormal basis. This choice of orientation is very natural as it allows us to approximate the "up" vector by the local *z* axis, which further lets us treat the height field as a flat-projected surface with little error. Hence, the height field LOD algorithm, which is based on vertical error in the triangulation, does not have to be modified significantly to take the curvature of the Earth into account. However, the delta values (see [Lin96]) must be computed in Cartesian rather than geodetic coordinates to avoid over-simplification of constant-elevation but curved areas such as oceans. Fig. 1 shows the local coordinate systems for a few zones.

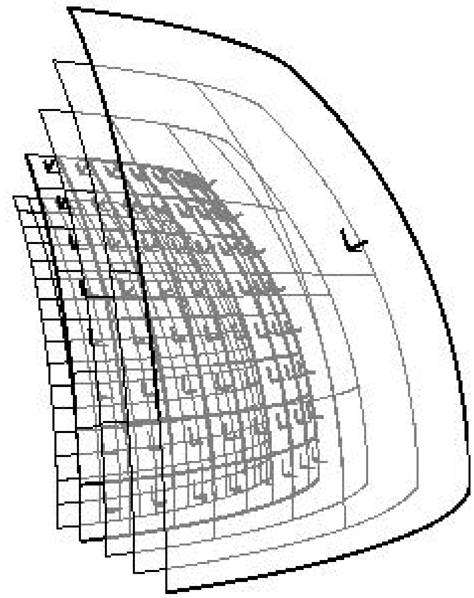


Fig. 2 Nested coordinate systems in a quadtree. 8 x 8 smaller coordinate systems appear 3 levels below the root node.

Using the above scheme, the resulting worst case precision for a 45° x 45° zone is 25 cm---not significantly better than the geocentric case. We could optionally use a finer subdivision with a larger number of zones to obtain the required precision. However, this would result in a larger number of quadtrees, which is undesirable since the lowest resolution data that can be displayed is defined by the areal extent of the quadtree roots. Hence, too much data would be needed to display the lowest resolution version of the globe. Instead, we define additional coordinate systems within each quadtree. In the current implementation, we have added 256 x 256 coordinate systems within each quadtree---one coordinate system per node, eight levels below each root node---resulting in a 1 mm worst case precision. Fig. 2 illustrates a subset of these nested coordinate systems. The terrain and object managers keep track of which coordinate system to use among these thousands of systems and can even transition between coordinate systems for extended objects.

Fig. 3 General global hierarchical framework.

The general approach to using the hierarchical structure is illustrated in Fig. 3. In each zone the quadtree is traversed to a certain level, depending on the type of geospatial data. Below this level a non-quadtree detail management scheme is used that depends on the detailed characteristics of the data. Thus, for example, buildings and terrain have different levels at which separate non-quadtree detail management schemes take over. We describe these schemes further below.

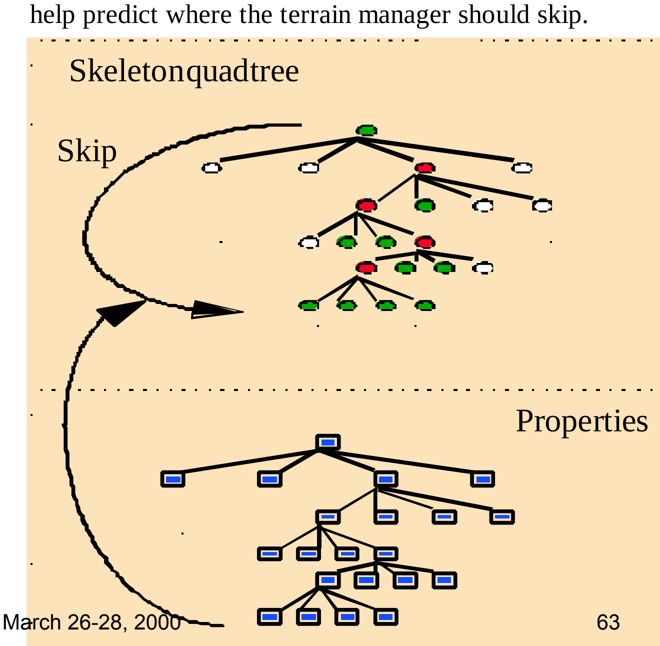
This global hierarchical, nested structure will handle the earth and everything on it at levels of detail from global overviews to fine resolution close-ups. Navigation between these extremes involves changes of up to 10 or more orders of magnitude. We are now extending the structure to everything over the earth including weather and other atmospheric effects [Rib00]. Since the atmosphere is only a thin layer in proportion to the extent of the earth's surface, this should be an efficient approach.

Caching. To conserve memory and promote efficiency, the static, view-dependent data associated with active nodes of the hierarchy are stored in a shared cache. This allows multiple managers for the various data types to access the data without having to replicate it. The shared cache consists of a set of hash tables, one for each data type (e.g., elevations, phototextures, feature data, buildings, moving objects, etc.), which have enough slots to hold all the quadnodes in the dataset. These slots are initially empty and filled with geospatial data whenever a request is processed by a particular server. If a node is no longer needed by any of the managers, the space for it is deallocated. The quadcodes are used as hash keys for accessing nodes in the hash table. Since the hash table slots are initialized at startup, the managers know what nodes exist externally such that no invalid data requests are made to the server. At present all quadtree hash tables are loaded at startup. In the future to maintain scalability, we will impose a structure where only the high level quadtree tables are loaded at startup with an additional paging and caching mechanism to bring in more detailed portions of the quadtrees as they are needed.

Paging. To support parallelism and expandability, there are separate paging threads for the different types of geospatial data. Each thread has a server and a manager. The server loads pages from disk while the manager decides which cells should be loaded (taking into account such things as user viewpoint and navigational speed) and passes it along for display or analysis. This communication path supports a demand-paging approach such as that of Cox and Ellsworth [Cox97]. When data are needed for a node in the quadtree, the manager allocates space in the above shared cache and sends a message to the manager. Message priorities in this queue are changed dynamically according to the importance of the associated request as determined by the manager. Thus, requests that gradually become less important sift towards the end of the queue and get serviced only when no higher priority requests remain in the queue.

We have found that the above page priority procedure sometimes falls short when handling global data. Users of such data frequently fly quickly from a global view where the terrain elevation and imagery data are at 8 Km resolution to views close to the ground where the data are at 1 M resolution or higher, and there may be hundreds or more buildings in view. If the user flies in too fast, the traversal of linked quadtrees by the terrain manager falls well behind the user's navigation. The process can stall in this case, and the pages for the scene currently in view can take quite long to arrive.

Unfortunately the system cannot just jump to the appropriate position in the quadtree. The quadtree has to be traversed to get important properties information, especially quadcell linking data but also geospatial bounding boxes and other data, that are necessary to determine if the object or other data should be displayed or not. To address this problem we created separate sets of skeleton trees, one set for each geospatial data type [Dav98, Dav99]. (See Fig. 4.) This separate structure provides properties information but is lightweight so it can be traversed quickly. Large segments of the indexing trees reside in main memory for fast access. With the flexibility of this scheme we can skip one or more levels before paging in object data. A predictive mechanism is instituted based on user navigational speed and viewing direction to help predict where the terrain manager should skip.



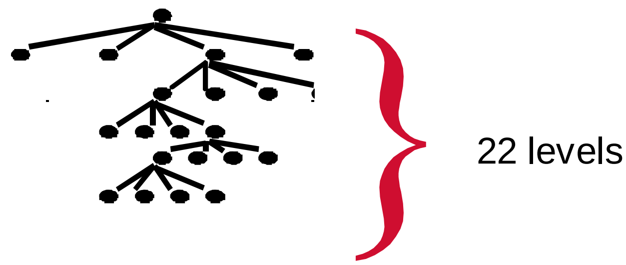


Fig. 4 Skeleton quadtree structure with ability to skip quickly between nodes. If a node does not contain data, properties are retrieved, which tell whether data are needed for the current view and where the data are located.

Since the managers are receiving continuous updates from the user via the user interface, they can use these in their requests to the servers. For example, the object manager can expend more detail on buildings or other objects in the center of the scene.

4 TERRAIN AND STATIC OBJECT DETAIL MANAGEMENT

We give examples of the type-specific detail management schemes illustrated in Fig. 3. We concentrate on terrain and static objects, but the framework can be extended to other types of data.

Terrain. We start with a brief overview of terrain detail management, described more fully in [Lin96] and [Lin97], and present some new capabilities. After the upper level traversal of the quadtree, the algorithm arrives at the leaf nodes. For high resolution data at, say, 1 M spacing this is typically after about 22 levels. The data in these nodes are arranged in a regular grid with a height value at each vertex. (See Fig. 5.) Using the latest viewpoint and view direction, the algorithm determines on-the-fly which vertices to include in the scene display. The decision as to whether a vertex can be removed is based on the screen space distance the vertex travels from its original position to the resulting surface if it were to be removed (i.e., if it's height component were flattened out). The corresponding world space distance is referred to as the vertex's "delta value". If this distance is smaller than a screen space threshold, the vertex is decimated and further simplification of nearby vertices is considered recursively. After the final set of vertices have been selected, the terrain manager produces a single triangle strip for each quadnode, which is simply a list of alternating vertices and texture coordinates enclosed by begin/end triangle strip commands. These display lists are a new capability that emulates the OpenGL counterpart, but are not the same. They are used because the scene managers cannot employ OpenGL display lists directly as only one thread at a time is allowed to issue graphics commands within an OpenGL context. Fig. 6 shows a terrain surface tessellation resulting from application of the continuous LOD algorithm. The threshold in this view is 1 pixel. Note that shading has been applied to the geometry before simplification to retain detail. Even for rough terrain at 1M resolution, this procedure can reduce the number of triangles displayed by a factor of 100 or more [Lin96]. If the terrain is smooth, the reduction is even greater. Further, with a 1 pixel threshold the visual quality of the scene displayed is not perceptually different from a scene rendered with all the triangles.

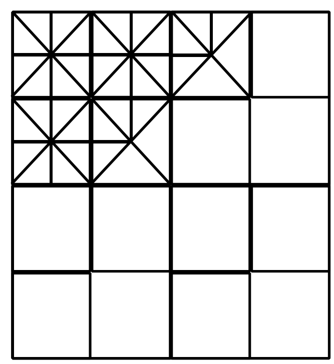


Fig. 5 Quadtree structure specialized for terrain elevations. Each vertex has a height value. For a given view, vertices are removed until a specified error in the height field is reached.

In addition to terrain elevations, we must manage terrain textures. These can be phototexture images, maps, textured shapes describing vegetation patterns, and so on. The texture LOD management bears some similarities to the geometry simplification. Rather than projecting the largest delta value to the screen, the largest (in screen space) *texel* is projected and compared to a threshold. The largest texel is found after taking into account the viewpoint-to-texel distance and the angle the texel normal makes with the viewpoint-to-texel vector. We here assume that all texels lie in a horizontal plane, but may be distributed in elevation within the bounding box of a node. The bounding box is determined by the area of the terrain patch for that node and the maximum and minimum height values within the patch. As a result, low detail is used when the terrain surface is viewed from the side, while relatively higher detail is required for top-down views.

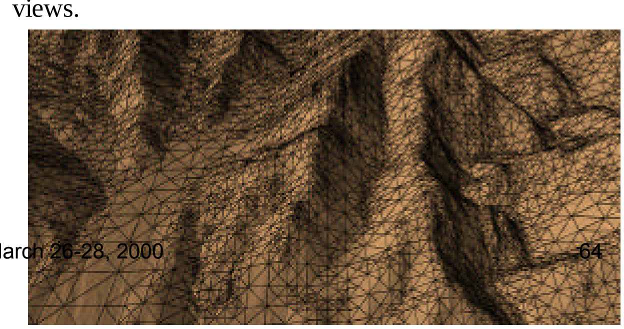


Fig. 6 Terrain surface tesselation showing dynamic detail management based on viewpoint and on terrain roughness.

Static Objects. If we have several cities in our global dataset, we want to quickly determine that the viewer is navigating towards, say, Los Angeles and does not need detailed data for Bombay, Houston, or even San Francisco. Further, as the user navigates to a certain neighborhood within a city, we don't want to page in and then have to cull, one-by-one, buildings for a neighborhood on the other side of town.

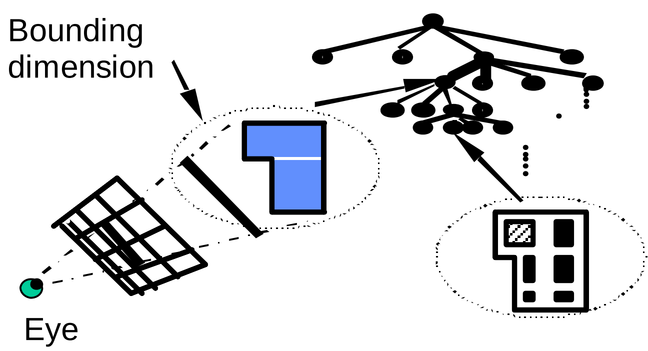


Fig. 7 Viewpoint-dependent detail management for static objects. The bounding dimension of a lower detail representation is checked. If necessary, the higher detail representation (lower right) is used instead.

To achieve these goals we use the global hierarchical structure with a few additions. Object pointers are loaded at the quadnodes determined as described above. Accompanying the pointer is the object location and a bounding dimension obtained from the largest dimension of the object bounding box. Using the viewpoint, object location, and bounding dimension, the system finds a maximum size for the object in screen space in terms of pixels. (See Fig. 7.)

When considering small buildings at urban densities, we use a more automated and compact linked cell mechanism. We must first determine the level to convert from the quadtree to a specific detail management scheme appropriate for buildings or other static objects. Let us assume a quadcell of side L_q. (The bounding box and bounding dimension in Fig. 7 would be of order L_q.) Now let's assume an object with maximum dimension L_o. If $L_o < L_q$, then only the 8 surrounding quadcells might contain objects that would extend into a central quadcell. In fact if we divide the central cell into 4 quadrants and further assume that all objects are placed in the quadcells that contain their centers, the maximum number of quadcells whose objects could overlap this quadrant would be 4, the central cell plus the 3 nearest quadcells to the quadrant. Our approach is to go as deeply as possible using the very efficient quadtree but not to permit more than 4 neighboring (or "linked") cells for considering overlapping objects, since there will be increased overhead from keeping track of the links and from having to consider all the objects in the linked cells. The collection of objects in the linked cells would have to be considered in view frustum culling, collision detection, and many other operations. Thus we choose cells for which $L_o < L_q$. For buildings in an urban setting, L_o could be about 50 meters, the dimension of a typical city block. If a typical building is the size of a house, say 10 meters on a side, we would have to consider on average no more than 100 or so buildings for 4 linked cells. Note that occasionally we will have to consider linked cells that could be in more than one quadtree.

We must also consider carefully extended objects such as a stadium, a large factory, or very large objects created via detail management. These might require considering several blocks as one object. If we use the largest such object in the database to determine the leaf node level of the quadtree, we might also end up with cells containing several hundred smaller objects. To obviate this problem we have discrete representations of such large objects at successive levels of the quadtree, each representation carrying its own list of linked cells. Thus if we flew from outer space down towards an urban area, we might first see a phototextured shape representing the downtown area, which would then be replaced by more detailed shapes representing collections of blocks and tall landmark buildings, and finally these would be replaced by shapes for individual buildings. Although the present system switches between discrete representations, one can imagine a more sophisticated process with more continuous switching of detail.

5 RESULTS

The global data model and the VGIS visualization interface use standard interface and programming elements and run on a variety of platforms. These include UNIX platforms, such as SGI and Sun, and PCs running Windows NT. A variety of display configurations are used including desktop, virtual workbench, and NAVE. (For more details see [Res00].) The virtual workbench and NAVE both use large projected displays that present stereoscopic scenes. The NAVE is a low-cost version--created at Georgia Tech--of the immersive, multiple screen CAVE. It runs entirely on PCs, rather than the SGIs used for the CAVE. Desktop versions of VGIS use the standard mouse interface, but the projected systems, especially the virtual workbench, use head and two-handed tracking. The NAVE also uses a joystick interface.

The results we present are from a single, global database of several gigabytes. Our Army colleagues have built VGIS databases of well over 20 GB without loss of capability. As stated above, we expect the model to scale to much larger data and will be testing this scaling soon. All scenes shown (Figs. 8-10) can be reached by continuous navigation from one place to another. Since typical databases are not likely to have terrain data everywhere at high resolution, our geospatial framework permits accurate placing of high resolution insets on lower resolution backgrounds and also nesting of several resolutions. For example, the state of Georgia (at 60 M resolution) is placed on the U.S. (at 1 Km resolution), and within Georgia are several nested databases around Atlanta (greater Atlanta at 10 M resolution, central downtown at 1 M resolution, and Georgia Tech at 1 foot resolution).

It is important to have the capability to build such large databases incrementally. Unlike most systems with real time data access, the VGIS database can be built incrementally with an efficient terrain dataset builder tool [Lin97] that inserts terrain data in a time that scales with the size of the inserted piece. This time will tend to become much smaller than the time to rebuild the entire database as the latter grows very large.

For navigation of the global database, typical frame update rates are 30 frames per second (fps) or more for an overview of a continent and remain at or above 15 fps even for flying close to high resolution terrain. These frame rates are for an SGI Infinite Reality Engine. Rates on a 300 MHz PC with TNT2 3D graphics card are slightly lower. When a large number of buildings are in view, the frame rates are somewhat lower on either platform. The paging delay for high resolution terrain depends on the speed at which one approaches an area. If one flies from a global overview to a close-up view, high resolution terrain elevations and imagery may be delayed up to a second or two on the Reality Engine and 3-4 times longer on the PC. More gradual fly-ins can be accomplished without paging delays, at least on the Reality Engine.

The several city databases each comprise hundreds of buildings but, due to the hierarchical building structure, frame rates are unaffected until one navigates relatively close to a city. Further, only buildings in the region of the current view are paged in, and only those within the current view are rendered. Significantly higher frame rates can be attained by strategic use of LODs. The efficient fast paging and hierarchical structure ensure that even when one jumps instantaneously to an entirely new location, lower resolution data are displayed quickly and higher resolution data relatively fast.

6 CONCLUSIONS AND FUTURE WORK

We have attacked the problem of increasingly vast stores of global information for the digital earth by presenting a general approach to data organization and real-time exploration. This approach is based on a novel global hierarchical data model. Our recent work has revealed that this framework can be quite generally applied to the earth and anything on it, above it, or even below it. This includes terrain elevations, phototextures and imagery, maps, buildings, moving or flying vehicles, weather, and other data. Further, the framework provides a geospatial visual data mining approach where one can navigate continuously from global overviews to high resolution local views. The framework is also quite flexible and has been applied to a range of single and networked systems ranging from a single-processor PC to immersive systems with multiple projection screens and coupled computers.

Our approach offers a general framework for digital earth applications because:

- It accepts and integrates all types of geospatial data into a global framework.
- It is scalable to very large data stores and to distributed environments.
- It provides an interactive visualization framework.
- It supports discovery of new information via navigation in context with unfolding detail.

We also described how large collections of objects, such as buildings and weather clouds, can efficiently fit, along with terrain, into the geospatial hierarchy. Application results demonstrated that our approach is both scalable and general because it is able to handle both large scale global terrain information and multiple collections of objects (e.g., cities) placed around the earth with full interactivity and without extensive memory load. Finally, our approach shows that levels of detail can be naturally incorporated to provide improved detail management.

The geospatial data model can readily incorporate full GIS capabilities since multiple data layers are already accessed via geocodes. We are now developing a queryable GIS data structure to augment the global model. The GIS layers will be in the hierarchical format presented here to ensure fast access and display.

The approach is extendible to a networked visual data mining system. In fact we have obtained reasonable responsiveness in accessing remote global databases through the VGIS visual interface. Our flexible paging and LOD structures can be adjusted to improve response times and even tuned to the characteristics of a particular client. We are building on these capabilities by inserting an internetwork server into the VGIS architecture. Eventually we plan to develop a completely machine-independent version of VGIS by porting the system to Java3D.

ACKNOWLEDGMENTS

This work was performed in part under grants from the NSF Alexandria Project, the NSF Large Scientific and Software DataSet Visualization program, and ONR. Initial development was made under a contract from the U.S. Army Research Laboratory.

REFERENCES

Cos90 Cosman, M.A., Mathisen, A.E., and Robinson, J.A. A New Visual System to Support Advanced Requirements. *Proc. IMAGE V Conference*, pp. 370-380 (1990).

Cox97 M. Cox and D. Ellsworth. Application-Controlled Demand Paging for Out-of-Core Visualization. Proceedings, *IEEE Visualization '97*, pp. 235-244 (1997).

Dav98 Davis, D., Jiang, T.F., Ribarsky, W., Faust, N. Intent, Perception, and Out-of-Core Visualization Applied to Terrain. *Proc. IEEE Visualization '98*, pp. pp. 455-458 (1998).

Dav99 Davis, D., Jiang, T.F., Ribarsky, W., Faust, N., and Ho, S. Real-Time Visualization of Scalably Large Collections of Heterogeneous Objects. *Proc. IEEE Visualization '99*, pp. 437-440 (1999).

DeF95 De Floriani, L. and Puppo, E. Hierarchical Triangulation for Multiresolution Surface Description. *ACM Transactions on Graphics*, 14(4), pp. 363-411 (1995).

Duc97 Duchaineau, M., Wolinsky, M., Sigeti, D. E., Miller, M.C., Aldrich, C., and Mineev-Weinstein, M. B. ROAMing Terrain: Real-time Optimally Adapting Meshes. *Proc. IEEE Visualization '97*, pp. 81-88 (1997).

Erv93 Ervin, S.M. Landscape Visualization with Emaps. *IEEE Computer Graphics & Applications* 13(2), pp. 28-33 (1993).

Fek90 Fekete, G and Treinish, L. Sphere quadtrees: a new data structure to support the visualization of spherically distributed data. *Proc.SPIE*, 1259, pp. 242-253 (1990).

Gar95 Garland, M. and Heckbert, P.S. Fast Polygonal Approximation of Terrains and Height Fields. Tech. Rep. CMU-CS-95-181, CS Dept., Carnegie Mellon U. (1995).

Goo92 Goodchild, M.F. and Shiren, Y. A Hierarchical Spatial Data Structure for Global Geographic Information Systems. *CVGIP: Graphical Models and Image Processing* 54(1), pp. 31-44 (1992).

Gru95 Grueneis, G. Mayer, P. Sauter, J., and Schmidt, A. T_Vision. *Visual Proc. of SIGGRAPH 95*}, p. 134 (1995).

Hop98 Hoppe, H. Smooth View-Dependent Level-of-Detail Control and its Application to Terrain Rendering. *Proc. IEEE Visualization '98*, pp. 35-42 (1998).

Kim99 Kim, S.-S.; Kim, Y.-S.; Cho, M.-G. A geometric compression algorithm for massive terrain data using Delaunay Triangulation. *Proc. of 7th International Conference in Central Europe on Computer Graphics,* *Visualization and Interactive Digital Media'99*, Vol. 1, pp.124-131 (1999).

Lin96 Lindstrom, P., Koller, D., Ribarsky, W., Hodges, L.F., Faust, N., and Turner, G.A. Real-Time, Continuous Level of Detail Rendering of Height Fields. *Proc. SIGGRAPH '96, Computer Graphics*, pp. 109-118 (1996).

Lin97 Peter Lindstrom, David Koller, William Ribarsky, Larry Hodges, and Nick Faust (1997). An Integrated Global GIS and Visual Simulation System. Report GIT-GVU-97-07.

Nis93 Nishita, T., Sirai, T., Tadamura, K., and Nakamae, E. Display of the Earth Taking into Account Atmospheric Scattering. *Proc. SIGGRAPH '93, Computer Graphics*, pp. 175-182 (1993).

Res00 For more information on various environments and platforms for VGIS, see

Rib00 Ribarsky, W., Wasilewski, T., Jiang, T.Y., and Faust, N. Real Time Weather Data on Terrain. Submitted to *IEEE Visualization '00*, (2000).

Sam84 Samet, H. The Quadtree and Related Hierarchical Data Structures. *ACM Comp. Surveys* 16(2), pp. 187-260 (1984).

Sch94 Schroder, F. and Rossbach, P. Managing the Complexity of Digital Terrain Models. *Computers & Graphics* 18(6), pp. 775-783 (1994).

Uen97 S.K. Ueng, C. Sikorski, and K.L. Ma. Out-of-Core Streamline Visualization on Large Unstructured Meshes. *Transactions on Visualization and Computer Graphics* 3(4), pp. 370-379 (1997).

Xia96 Xia, J.C. and Varshney, A. Dynamic View-Dependent Simplification for Polygonal Models. *Proc. Visualization '96*, pp. 327-334 (1996).

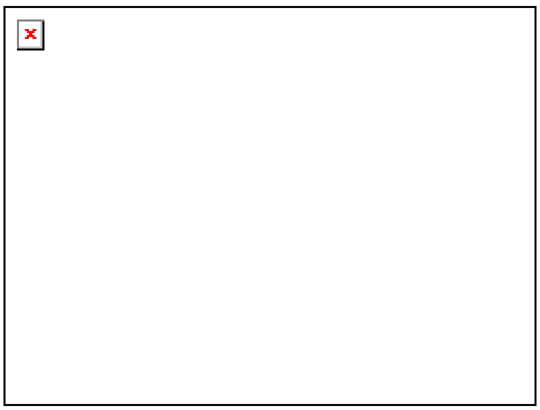
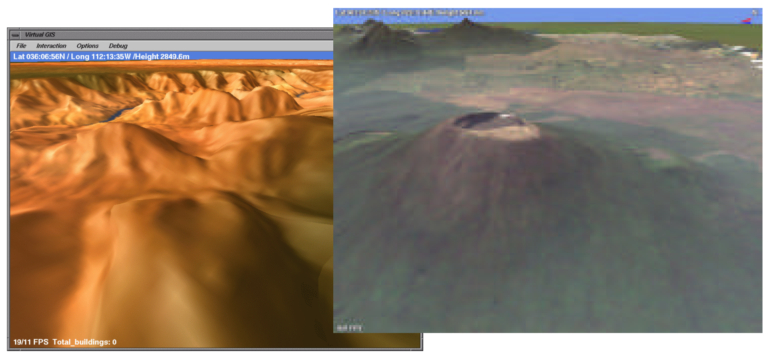
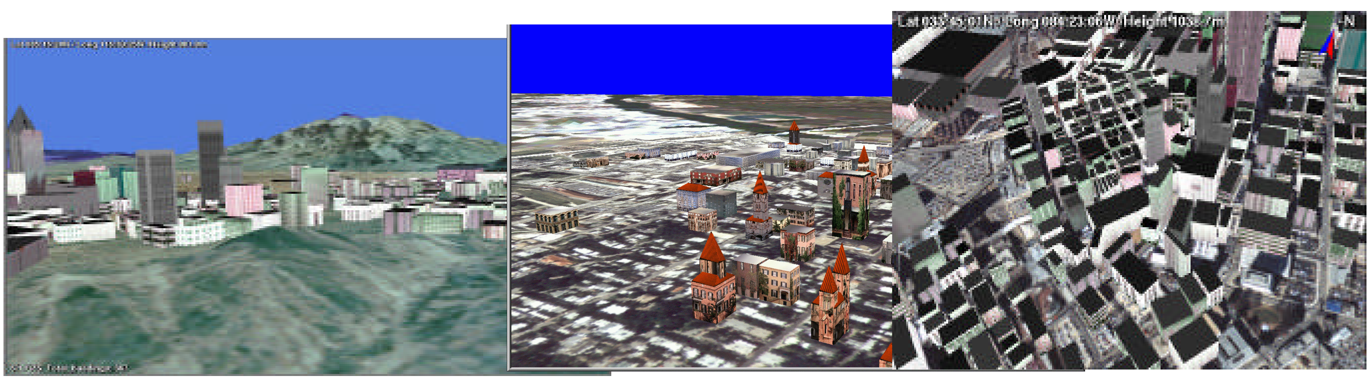


Fig. 8 World overview. This and all following scenes can be reached by continuous navigation.



Figs. 9 Views of the Grand Canyon (left) and a mountainous area in Rwanda.



Figs. 10 Views of cities (from left to right) at NTC in California, Savannah Georgia, and Atlanta. All cities are in the same database. The hierarchical object structure ensures that only those buildings in view are loaded and rendered.

DISCRETE GLOBAL GRIDS FOR DIGITAL EARTH

Michael F. Goodchild[1]

Abstract

Digital Earth is being promoted as a framework into which "we can embed vast quantities of information about the planet" (U.S. Vice President Gore). It provides an organizational metaphor for information about the Earth by allowing users to search over a virtual rendering of the planet; a call for a framework that can accommodate all kinds of information, including data, computational models of processes, text, and images; and a high-end visualization system allowing rapid change of resolution from global to local. These three interpretations all motivate a search for novel hierarchical data structures, but present distinct problems. The paper reviews the various interpretations of Digital Earth, examines progress in various arenas, and develops a series of principles for the design of global gridding schemes.

Concepts of Digital Earth

Information in digital form can often be regarded as a *representation* of some aspect of reality, and this perspective is particularly useful in connection with geographic phenomena, that is, phenomena located on or near the surface of the Earth. Geographic representations are normally confined to spatial resolutions of centimeters or coarser, and extents that range from neighborhoods up to the entire Earth. They include maps, images of Earth from space, textual descriptions of places, spoken records, and digital databases using raster and vector structures.

In recent years there has been increasing interest in holistic approaches to representation. Projects with names like *Virtual Human* or *Virtual Los Angeles* aim to create digital representations of entire systems, and to provide the tools that allow users to explore them interactively, often in immersive environments. A user of Virtual Los Angeles (www.gsaup.ucla.edu/bill/LA.html), for example, will be able to explore various aspects of the city through immersion in an environment that appears to have many of the characteristics of the real city. Of course the underlying representations are limited to certain ranges of resolution, certain easily captured aspects such as built form, and certain intervals of time. But given adequate data and sufficiently powerful computing environments, these projects offer enormous value to educators, prospective tourists and entrepreneurs, city managers, and many others. Virtual anatomy (www.vis.colostate.edu/library/gva/gva.html) could provide the opportunity to work with a representation of a generic human body, and thus could offer medical schools great advantages in training students under carefully controlled and relatively risk-free conditions. Virtual humans and virtual Earth share many characteristics: the need for unambiguous methods for referring to locations within the frame of the whole system; the potential for representation of dynamic processes as well as static form; the need for advanced visualization systems; and similar motivations.

[1] National Center for Geographic Information and Analysis, and Department of Geography, University of California, Santa Barbara, CA 93106-4060, USA.

From this perspective, a virtual or *Digital Earth* would be a unified repository and source for all that is known about the Earth system. It could provide a "one-stop shop" for geospatial data, overcoming many of the problems that are now associated with geospatial data acquisition (data scattered among different servers, in different formats, with different access protocols; redundancy in data production; etc.). It could also contain knowledge about the Earth that is stored in books, journals, reports, and other text-dominated media. And perhaps most importantly, it could contain digital representations of the processes that operate on the Earth's surface and near-surface—the numerical simulation models that represent what we know about how the Earth operates, and how it will change in the future.

It is widely accepted that the term *Digital Earth* (DE) originates with the published version of a speech of the U.S. Vice President, Al Gore. In it, he describes an immersive environment that would allow its users to explore and learn about the Earth and its human and physical environments (the full text is at *www2.nas.edu/besr/238a.html*; a summary was delivered in Los Angeles in January 1998):

> “Imagine, for example, a young child going to a Digital Earth exhibit at a local museum. After donning a head-mounted display, she sees Earth as it appears from space. Using a data glove, she zooms in, using higher and higher levels of resolution, to see continents, then regions, countries, cities, and finally individual houses, trees, and other natural and man-made objects. Having found an area of the planet she is interested in exploring, she takes the equivalent of a ‘magic carpet ride’ through a 3-D visualization of the terrain. Of course, terrain is only one of the numerous kinds of data with which she can interact. Using the system’s voice recognition capabilities, she is able to request information on land cover, distribution of plant and animal species, real-time weather, roads, political boundaries, and population. She can also visualize the environmental information that she and other students all over the world have collected as part of the GLOBE project. This information can be seamlessly fused with the digital map or terrain data. She can get more information on many of the objects she sees by using her data glove to click on a hyperlink. To prepare for her family’s vacation to Yellowstone National Park, for example, she plans the perfect hike to the geysers, bison, and bighorn sheep that she has just read about. In fact, she can follow the trail visually from start to finish before she ever leaves the museum in her hometown.
>
> She is not limited to moving through space, but can also travel through time. After taking a virtual field-trip to Paris to visit the Louvre, she moves backward in time to learn about French history, perusing digitized maps overlaid on the surface of the Digital Earth, newsreel footage, oral history, newspapers and other primary sources. She sends some of this information to her personal e-mail address to study later. The time-line, which stretches off in the distance, can be set for days, years, centuries, or even geological epochs, for those occasions when she wants to learn more about dinosaurs.”

Several principles and challenging ideas underlie this piece of technological fantasy. First, the immersive environment provides a very rich form of communication between the information store and the learner, unimpeded by the constraints of a single medium, and not limited to the visual channel or to the traditional and narrow concept of a paper map as we normally understand it. Many of the constraints of traditional mapping disappear in the DE environment, including the need to project the Earth onto a flat surface, and the limitation of an unmodifiable and therefore static display. Second, the vision mixes types of data that are readily communicated by *rendering* into something resembling their true appearance, such as topography and land cover, with other types that will have to be communicated symbolically. This second type includes information on population, health, or environmental quality. Cartographers are familiar with the problems of mixing these two types through their experience with symbolic enhancement of orthographic images. Other information mentioned in the speech is geographic only in the sense of having a footprint; the contents of newspapers and oral histories will have to be represented iconically, and their contents communicated in some appropriate way, since they are not geospatial and therefore cannot be mapped onto the Earth's surface.

More fundamentally, perhaps, DE embodies a novel metaphor for the organization of digital information and construction of user interfaces. The current generation of computer operating systems, such as Windows 98, makes use of the metaphor of the desktop, with its clipboards, filing cabinets, and briefcases, because this is the environment most familiar to office workers. This tradition goes back to work at the Xerox PARC laboratories in the 1960s, but came to dominate Microsoft operating systems only in the late 1980s with Windows. Yet the office is not a natural environment for thinking and learning about the surface of the Earth, and *office* is not the first thing that comes to mind when we think of Columbus, or von Humboldt. Since all such information relates to some geographic location, it would be far more effective to use the Earth's surface itself as the organizing metaphor. For example, rather than look in a filing cabinet under Z, someone interested in Zimbabwe would find it much easier conceptually to reposition a digital globe to the right part of Africa (or to look up Zimbabwe in a digital rendering of the back-of-the-atlas gazetteer, and see the globe repositioned automatically). DE replaces the office with the Earth as the dominant user interface metaphor. In that sense it offers a significant contribution to the growing interest in digital libraries that support search for information by geographic location, or *geolibraries* (NRC, 1999).

Finally, DE is not static. In an immersive digital environment, and given sufficient general knowledge in the form of models of processes, it is possible to imagine the user of DE being able to simulate future environments, by executing models of urban growth, or species extinction, or tectonic uplift, and observing their consequences for any part of the Earth. It is also possible to imagine modeling past environments, by running simulations backwards in time. In this regard, DE is seen as having immense power for education. A static DE would be a good basis for learning the facts of the Earth's special geography. But a dynamic DE would allow students to explore the processes of general geography, and their implications, in compellingly realistic form and using boundary conditions representing environments that are familiar to them. For example, a student would be able to learn about tectonic processes by modeling the appearance of California 1, 10, or 100 million years from now.

The idea of DE as a digital library of simulation models raises interesting and challenging questions. Although libraries have evolved highly effective systems for abstracting and indexing books and journals, and although numerous organizations now offer access to substantial resources of geospatial data (e.g., the National Geospatial Data Clearinghouse, www.fgdc.gov), the knowledge contained in simulation models, which is arguably among our most sophisticated knowledge of how the Earth system works, remains largely uncataloged and inaccessible. In contrast to knowledge in books and journals, no infrastructure currently provides the tools to support a scientist wishing to search for, evaluate, download, and run simulation models. An infrastructure for organizing, describing, and sharing simulation models would provide a very valuable service to the Earth science community.

The speech has spawned a substantial level of interest in DE, as a search of the WWW will reveal. DE projects are under way to model tectonic and other geophysical processes, to explore Earth imagery, to deliver the services of a map and imagery library, and to teach about the Earth and its human and physical phenomena (www.alexandria.ucsb.edu/adept). An interagency working group has been meeting for the past two years under the auspices of the U.S. National Aeronautic and Space Administration, and a number of prototypes of DE are under development in the private sector. The first International Symposium on Digital Earth was held in Beijing in December 1999, and another is planned in Canada in 2001.

There is every indication that DE's vision of an integrated source of knowledge about the Earth will attract the attention of researchers, developers, agencies, and corporations. This paper is about the importance of discrete global grids to that vision, and is intended to introduce global grid researchers to the potential of DE. The next section discusses the role of grids in DE, in three contexts: georeferencing, indexing, and discretization. This is followed by sections on functionality, and the assessment of grid options. The paper ends with a brief concluding section.

DE applications of global grids

Georeferencing

In DE all information is referenced to positions on the Earth's surface, through point locations, *footprints*, or similar mechanisms. Services must be provided to allow users to locate places, to compare different sources of information about the same place, and to define the locations associated with information being input to the system.

Many methods exist for defining location on the Earth's surface. To be useful they must define location uniquely, at least within a specified domain (e.g., the placename Springfield is unique in the U.S. only within the domain of a state); and have meaning that is widely shared within a defined information community. Latitude and longitude referencing satisfies these requirements, but is accessible only to a comparatively small information community, as is referencing by widely adopted coordinate systems such as UTM. Placenames are the dominant method in the broader community, but are subject to a number of caveats:

- domains less than the Earth are often needed to establish uniqueness; or placenames must be coupled with one or more domain names (e.g. Springfield, Illinois, USA);
- the spatial resolutions associated with placenames are often undefined or unknown;

- the meaning of placenames can be context-specific (Los Angeles may have different meanings to speakers located in Southern California, New York, and Moscow);
- alphabets, diacritical marks, and spelling variations cause problems; and
- the meaning of placenames may change through time.

Discrete global grids offer two major advantages for georeferencing. Besides being unique and domain-independent, appropriately indexed or *linearized* grids express three-dimensional location in a single string, and make resolution explicit in the length of the string (Goodchild and Yang, 1992). Grids with these properties have been adopted in many contexts, including various systems for indexing national topographic map series; regional georeferencing schemes such as Go2; and the quadtree indexes used in spatial databases (Samet, 1989).

A user of DE may not be concerned with the precise details of internal georeferencing, requiring only services that recognize various forms of georeference, deliver predictable responses based on them, and make them interoperable. Internally, however, DE should employ a system with the union of all desirable properties. Discrete global grids appear to have the edge in this respect. Their major disadvantages—lack of a large information community familiar with them, and lack of an intuitive relationship between pairs of codes and proximity—are not important for internal purposes, but would be problematic if they were exposed to most users.

Spatial indexing

An index can be defined as a mapping of information into a linear structure, such that information on a given topic always appears at the same place in the structure. Once an index is built, it can make retrieval of information on a given topic much faster, since no search has to occur because the location of information is computable. A spatial index linearizes information based on location. Besides speeding search, a spatial index provides additional advantages in spatial databases, because of certain characteristics of their applications. The probability of a request involving two locations ***x*** and ***y*** usually decreases with the distance between them, and the probability that a request for data on ***y*** will follow a request for data on ***x*** is similarly inversely dependent on distance. It was this issue that led Morton to implement an early form of quadtree indexing in the Canada Geographic Information System (Foresman, 1997). Quadtrees and R-trees are two popular forms of spatial index (van Oosterom, 1993).

Linearized discrete global grids are clearly applicable in DE, and offer substantial advantages in data retrieval and processing. Further advantages in search over distributed archives are discussed in a subsequent section.

Discretization

If one accepts the proposition that the geographical world is inherently continuous, with an infinite number of possible locations, then digital representation must necessarily abstract, generalize, or approximate. Most forms of representation involve *discretization*, or the reduction of continuous dimensions to discrete form, most commonly with respect to the spatial dimensions. Tesselations thus replace the continuous spatial dimensions with a finite number of discrete elements, and reduce variation within elements to some simple form, often a constant but also possibly a polynomial function, that can be described with a small number of parameters. In the modeling world the two common options are *finite element* methods, which replace continuous spatial variation with variation over a tesselation of triangles or quadrilaterals modeled using polynomials in the continuous spatial dimensions, and *finite difference* methods, which approximate spatial variation using constant values within square cells. In the geospatial data world finite difference representations are commonly known as rasters; finite element representations are known in the case of linear variation over triangular elements as *triangulated irregular networks*, and other options are generally not used. Not all geospatial data are discretized in the spatial domain—much representation of global climate data for modeling purposes relies instead on discretization in the spectral domain.

If DE is to implement simulation models, consideration must be given to the discretization inherent in its representations. Finite difference methods require regular tesselations, and the square tesselation that is normally used can only exist on a flat surface. The basis for estimating surface derivatives in GIS similarly depends on a raster representation and a flat projection of the Earth's surface. For example, the equations shown below give the standard estimation method for the first derivatives using finite differences. Discrete global grids such as QTM (Dutton, 1998), based on near-regular triangular tesselations, are more compatible with finite element methods.

$$f'_x \approx [-z_{i-1,j-1} - 2z_{i-1,j} - z_{i-1,j+1} + z_{i+1,j-1} + 2z_{i+1,j} + z_{i+1,j+1}]/8d$$
$$f'_y \approx [-z_{i-1,j-1} - 2z_{i,j-1} - z_{i+1,j-1} + z_{i-1,j+1} + 2z_{i,j+1} + z_{i+1,j+1}]/8d$$

Finally, there are obvious advantages to adopting a single approach to discretization within DE. Hierarchical methods that allow the same structure to be used over a wide range of spatial resolutions, such as the quadtree or QTM, would provide a single, consistent approach. But if QTM were to be adopted for DE, simulation models that exploit finite differences would have to be rewritten to use the finite element approach. Alternatively, a hybrid approach could be devised that would estimate derivatives for the irregularly shaped elements of QTM, but use the finite difference approach of assuming constant values within elements. For example, the derivatives at the center of a triangular element could be estimated using the approach on which the equations above are based, but adapted to the irregular geometry of QTM, by fitting a plane through the values in the element and its three full neighbors (analogous to estimating derivatives using a cell and its four full neighbors in a raster), or through the values in the element and its full and diagonal neighbors (a variable number in the QTM scheme, but always equal to 8 in a raster).

Functionality

Several aspects of DE functionality bear directly on the topic of this conference. Reference has already been made to questions of indexing and discretization, which underlie functionality. The choice of underlying structure is also likely to be dictated by a number of other aspects of functionality.

First, DE musts support a full range of georeferencing schemes, geodetic datums, and projections. Data for DE will be drawn from a wide range of sources, and merging data at global scales always raises geodetic questions, particularly when non-global datums are involved. As noted earlier, there are obvious advantages in a structure that offers consistency across a wide range of scales; a single address for any global location; and positional accuracy and spatial resolution that are explicit in the coding scheme.

Second, DE must support rapid zoom, and the Gore speech talks about zooming in real time from global to neighborhood resolutions. On a commonly available display unit of approximately 1000 by 1000 picture elements a full display of the Earth requires data with spatial resolution in the 10km range. Neighborhood display is likely to require 1m spatial resolution or better, implying that DE must support zoom over at least 4 orders of magnitude. Moreover, a complete coverage of the Earth's surface at 1m resolution requires approximately 10^{15} data elements. Hierarchical structures such as QTM have inherent advantages here, since they provide a single representational structure for unlimited ranges of resolution, and adapt easily to incomplete coverage, or coverage with variable resolution.

Third, DE must support visualization. Much high-performance visualization hardware uses triangular elements, providing fast capabilities for pan, zoom, and rendering. DE displays will always be user-centered, implying that resolution can be allowed to degrade away from the center of the field of view. In this sense DE is in sharp contrast to practice in traditional cartography.

Search for data in DE

DE was presented earlier as a new metaphor for the organization of geospatial information resources. In this context, DE is envisioned as an environment in which the availability of information is discovered by the user through interaction with a virtual Earth. Information might be rendered directly onto the planet, or its existence might be represented symbolically. Prior to display, however, DE would somehow have to establish the existence of information, and determine its properties.

Currently, two mechanisms exist for discovery of such information within the distributed archive of the Internet. First, solutions such as the National Geospatial Data Clearinghouse (NGDC; www.fgdc.gov) rely on standards for data set description. The FGDC metadata standard (Content Standards for Digital Geospatial Metadata) provides such a standard. Participants in the clearinghouse provide metadata records for all of the data sets that are available on their servers, and users are able to search these records through a common portal. Geographic footprints are commonly expressed as bounding latitude and longitude values.

Second, a well-informed user might be able to narrow the search to a small number of servers. Many users of geospatial data know where to look for standard types of data, such as Landsat imagery, or digital raster graphics. This knowledge might be described as *collection-level metadata* (CLM), providing the general characteristics of the data sets stored on a given server or collection, in contrast to *object-level metadata* (OLM), which is used to describe the characteristics of single data sets. At this time no standards for CLM exist. In practice, many collections are defined by theme, or by data type. The Alexandria Digital Library's CLM is particularly complex, since the collection is based on the imagery, maps, and data sets in the collection of the University of California, Santa Barbara's Map and Imagery Laboratory, one of the world's largest.

One possible solution to the CLM problem is to base server contents on a geographic tesselation. If we could agree, for example, that all data sets whose footprints intersect Santa Barbara County should be available on a server maintained by the county, then search for such data would be vastly easier than it is under the current arrangements, with such data scattered over a large number of servers maintained by different agencies, institutions, and jurisdictions. Thus discrete global grids provide a potential solution to the CLM problem, by forming the basis on which data could be allocated to a network of servers.

Conclusion

At this point DE is little more than a vision. As such, however, it may be extremely useful in providing researchers and developers with a target for activities. Current technology is not sufficiently powerful to deliver all DE services, but we can certainly implement some services using current technology, and imagine a time when more powerful technology would be available. Moreover, the Vice President's vision includes significant references to central facilities—in good geographic tradition, if a service is too expensive for individuals to afford, yet still of substantial value, then it makes sense for the community to provide it as a public service at a central location.

DE can clearly benefit from developments in discrete global grids, which can provide the georeferencing, the indexing, and the discretization needed for geospatial data sets. They have properties, in particular hierarchical structure, uniqueness, explicit representation of spatial resolution, and consistency, that make them superior to any single alternative. Moreover, discrete global grids based on triangles have distinct advantages for high-performance visualization and for finite-element modeling. Of course DE must be capable of working with alternatives, for input and output, or for narrowly defined applications within the general DE structure, such as modeling on projected flat surfaces using finite differences.

Discrete global grids have many overlapping motivations, from uniform sampling of the Earth's surved surface to optimal locations for facilities. This paper has explored one more: the vision of a unified repository of all that is known about the Earth's surface and near-surface. The vision of Digital Earth appears to be capable of attracting significant interest, in communities that range from vendors of high-end visualization systems to the military.

Acknowledgment

The Alexandria Digital Library and the Alexandria Digital Earth Prototype (ADEPT) are supported by the National Science Foundation and by other agencies participating in the Second Digital Library Initiative.

References

Dutton, G. (1998) *A Hierarchical Coordinate System for Geoprocessing and Cartography*. Lecture Notes in Earth Science 79. Berlin: Springer-Verlag.

Foresman, T.W., editor (1997) *The History of Geographic Information Systems: Perspectives from the Pioneers*. Upper Saddle River, NJ: Prentice Hall PTR.

Goodchild, M.F., and S. Yang (1992) A hierarchical spatial data structure for global geographic information systems. *Computer Vision, Graphics and Image Processing: Graphical Models and Image Processing* 54(1): 31–44.

National Research Council (1999) *Distributed Geolibraries: Spatial Information Resources*. Washington, DC: National Academy Press. Also available at www.nap.edu.

Oosterom, P.J.M. van (1993) *Reactive Data Structures for Geographic Information*. New York: Oxford University Press.

Samet, H. (1990) *The Design and Analysis of Spatial Data Structures*. Reading MA: Addison-Wesley.

Comparing Intercell Distance and Cell Wall Midpoint Criteria for Discrete Global Grid Systems
By MJ Gregory, AJ Kimerling, D White and K Sahr

Many diverse applications study processes and patterns at a global scale. To aid in this research, discrete global grid systems (DGGSs) enable environmental modeling, monitoring and sampling across the earth at a variety of spatial scales. A DGGS can be evaluated on a set of topological and geometric criteria, two of which, intercell distance and the "cell wall midpoint or Heikes-Randall criterion", form the basis of this study. These two properties have been noted to be important for dynamic modeling applications.

This study focuses on results obtained from measurements of six different global partitioning methods. Each method was further subdivided into different design choices, which included frequency of cell subdivision (2- or 3-frequency edge partition), predominant tessellating shape (triangle, hexagon, diamond or quadrilateral), and base modeling solid (octahedron, icosahedron, or sphere). Intercell distance and cell wall midpoint measures were statistically normalized to be comparable among the methods studied. The results were further standardized to a common mean intercell distance (89.02 km) to determine performance rankings for the methods. Finally, the distortion of each method was presented graphically to understand the underlying spatial pattern.

For intercell distances, the Fuller-Gray method had the best performance, while two quadrilateral grids (Tobler-Chen and Equal Angle) performed substantially worse. For the cell wall midpoint criterion, the Equal Angle grid had the lowest overall distortion with the Snyder and Fuller-Gray methods also performing relatively well. The Tobler-Chen grid generally had the poorest performance for this property, especially at the higher recursion levels studied. All methods based on the icosahedron outperformed those based on the octahedron for both criteria studied. Aggregation of triangles into hexagons and diamonds seemed to have little impact on intercell distance measurements, although dual hexagon aggregation had markedly different statistics and spatial patterns for the cell wall midpoint property. Frequency of subdivision seemed to account for very little variation. Maps of spatial variation for both criteria show patterns of localized distortion which are unique to each method studied.

EarthView -- a "planet browser"

By Paul Hansen and TonVan Sant
Geosphere

There have been many proposals for a "digital Earth" system. The most prominent being that put forward by Al Gore and being developed by NASA. In such a scheme, the internet has sites which contain data that is geo-referenced; i.e., the data describes some aspect of the Earth at some location. In the case of satellite imagery and terrain elevation data, it is possible for a digital Earth system to include a 3D graphics representation of the Earth or parts of it, and provide a real-time fly-thru and viewing of the data.

EarthView is a software system that does this by pre-processing the image and terrain data into a grid of multi-resolution tiles, and quickly calculating which tiles are required for a particular view. The term "multi-resolution" means that some tiles contain data at higher or lower resolution than other tiles; for example, a view of Earth from a great distance in space would use lower resolution tiles than a closer view. The contents of a single low-resolution tile will require 4 tiles at the next higher resolution level, since it will appear to be twice as wide and twice as high, and more image detail will be seen.

A unique feature of this software is the way that the north and south polar regions are handled. Standard cylindrical (Mercator) projection has the drawback that the polar regions are "stretched" more as latitude increases, while it has the benefit that latitude lines are horizontal and longitudinal lines are vertical. EarthView can take advantage of this benefit from the equator to the 45th parallel (north and south), but from there to the poles converts the data into separate regions with the pole in the center. This is also somewhat distorted, but when these regions are mapped to a globe in computer graphics, the result is a high-quality simulation that avoids mapping problems and saves diskspace by reducing the "polar stretch".

In this mapping, the entire globe is represented in six portions, four around the equator, and two polar. Within each of these 90 degree regions there is then a recursive subdivision into 4,16, 64, etc. subtiles, down to the desired resolution. The EarthView software can manage these tiles at extremely high speed on today's low-cost computers, so that everyday citizens can view a digital Earth with the sophistication that was only recently available to expensive business, government and military installations.

The World Geographic Reference System

By Jordan T. Hastings
Go2 Systems, Inc.

Go2 Systems, a California company, has developed a new coordinate system for digitial geospatial data, particularly suitable to emergent wireless Internet and Web-mapping applications. The patented Go2 system, termed the World Geographic Reference System (WGRS), strikes a balance between precise numerical notations, such as latitude/longitude or Universal Transverse Mercator, and colloquial names, which are inherently approximate. The numbers accommodate machine computations, whereas the names facilitate human references. WGRS is based on a irregular gridwork of oblique stereographic projections, centered on important physical and/or cultural geographic features, especially cities. These grids may overlap, and also may have activated sub-grids nested to arbitrary depth. Within the WGRS grids, named places are catalogued, and can be referenced, at a variety of scales from fully-extended 100Km x 100Km regions, down to 1Km x 1Km neighborhoods, and even smaller features, eg. mouths of springs, public monuments, MacDonalds (tm) restaurants. In effect, WGRS provides a multi-scale, areally-explicit gazetteer, organized in consonnance with significant features on the landscape. Syntactically, WGRS coordinates appear as a compact string of "dotted digit pairs" that interleave Northing/Easting measurements in a single object relative to a named grid, eg. "US.CA.LA.45.36.72.81". This notation, which is conveniently reminiscent of Uniform Resource Locators (URLs), is modestly mnemonic, quickly entered using a variety of input devices (keyboards, phone keypads, voice recognition systems), and also reliably communicated across the Web. WGRS applications currently being developed by Go2 Systems and partners include U.S. 9-1-1 emergency systems, and an international "yellow-pages" directory for both the wired- and wireless-Internet.

Challenges of Georeferencing Places: Development of Digital Gazetteers

By Linda L. Hill
Alexandria Digital Library
University of California

Gazetteers are fundamentally dictionaries of named geographic places/features; each name entry in a gazetteer is defined by at least one spatial representation and by at least one type (category). Gazetteers can be free-standing or they can be services providing gazetteer access to other sources of geospatial data (e.g., GIS data). Place name access to georeferenced data and information of all sorts (indirect geographic referencing) is a powerful unifying strategy for distributed georeferenced systems. Challenges of broad use of gazetteers in georeferenced systems include shareable data formats, query and response service protocols, efficient text and spatial queries, and complete descriptions of the data for appropriate use (e.g., authenticity, accuracy, temporal attributes). Graphic responses to questions such as "Where is the Mississippi River?" should be answerable from a set of gazetteer and GIS services. We should be able to discover images, text, maps, data, etc. about the Mississippi River Valley area through the use of the place name (indirect georeferencing). The gazetteer role is to translate names to spatial representations so that information is discovered by spatial footprint as well as by names. The Alexandria Digital Library has done extensive work on gazetteer development, including leading an NSF-sponsored workshop on Digital Gazetteer Information Exchange (DGIE). In this presentation, it is proposed that geographic names can provide a key to interoperability among various representations of spatial objects.

Discrete Global Grids -- One User's Perspective

By Ralph Kahn
Jet Propulsion Laboratory
California Institute of Technology

I study climate, primarily by analyzing remote sensing data from satellites. So I am a user, though not a developer, of techniques related to discrete global grids. In December 1999 we launched Terra, the first in NASA's series of Earth Observing System (EOS) satellites. The EOS program is aimed at detecting, understanding, and eventually helping predict climate change on global and regional scales. With the polar-orbiting Terra satellite, we begin collecting planet-wide data over long enough time scales, and with sufficiently stable calibration, to address truly global problems.

Retrieval algorithms convert the "Level 1" measured radiances into "Level 2" geophysical quantities, such as aerosol amount, cloud height, and surface reflectivity. The spatial and temporal sampling pattern for Level 1 and 2 data, determined by the satellite orbit and instrument characteristics, is usually highly non-uniform. Discrete global grids are used to create "Level 3" data sets, uniformly sampled in space and/or time, from the Level 2 data. Techniques routinely employed for this purpose are primitive, in part because available data sets were adequate to addressing only the simplest global climate change problems.

Whenever data taken at different resolutions are used together to calculate a new quantity, grids are likely to play a role. The global radiation balance, for example, is assessed by combining measurements of visible and infrared fluxes with those of latent and sensible heat contributed at the surface. More sophisticated global grids with than the ones traditionally used are needed to calculate finite-difference quantities, such as the horizontal gradients of energy, momentum, and material. And there is the need to "validate" data taken from orbit over the long term and at many places, by comparing it with measurements made near-simultaneously by aircraft-borne and ground-based instruments. Since the data taken from different platforms vary widely in their spatial sampling, having appropriate grids and associated data processing techniques would be a great asset. This talk will summarize, from one climate researcher's view, the hopes, needs, and limitations in the application of discrete global grids to the new Earth observations.

Unexpected and Complex Behavior of Hierarchical, Multiresolution Cellular Automata

By Ross Kiester[1] and Kevin Sahr[2]

[1]USDA Forest Service
Corvallis, Oregon

[2]Department of Computer & Information Science
University of Oregon
Eugene, Oregon

We have implemented hierarchical, multiresolution cellular automata on subsets of a variety of global grid topologies. Using a simple rule sets analogous to the well known "Game of Life" due to John Conway, we have found that multiresolution systems appear to manifest infinite memory and produce dynamics similar to fractional brownian motion. These systems may crash or persist for long periods of time. The probability that a given system will persist depends on the topology of the grid, the size of the grid, the number of resolutions, and the the number of cells "alive" at time 0. Whether or not a given configuration persists also depends exquisitly on the exact inital spatial configuration. The value of this system for simulating demographic, ecological, and biogeographic process is discussed.

An Adaptive Grid and Associated Advanced GIS Techniques for Air Quality Models

By Maudood Khan, M. Talat Odman, Hassan A. Karimi, Michael F. Goodchild

Hassan A. Karimi
Dept of Information Science and Telecommunications
University of Pittsburgh

Air quality models (AQMs) simulate the transport, transformation, and fate of pollutants emitted into the atmosphere from various sources. They are increasingly used to design emission control strategies. The accuracy of AQMs depends on the resolution of relevant atmospheric processes. Inaccurate results may lead to huge investments in ineffective strategies and still leave the public health at risk. Current AQMs have fixed grids with predetermined resolution. However, there is a need for increased resolution in places of high chemical activity. The location of such places may vary throughout the simulation as the meteorology and emission patterns change. To reduce the uncertainty in the AQM results, we have developed an adaptive grid model and a set of advanced geographic information systems (GIS) techniques and tools that together can provide the optimal resolution automatically. One main characteristic of the adaptive grid technique is that it preserves the topology of the grid, that is each cell always has the same neighbors.

One of the most important issues in developing the adaptive grid AQM is the processing of emissions. Current GIS, which are used to retrieve emissions data (source location and emission rates) and map them onto fixed grids, cannot be used. Much more efficient GIS techniques are necessary for real-time emission processing over adaptive grids that evolve during the simulation. Every time the adaptor changes the grid point locations, the following three-stage processing is performed: (1) point sources are allocated into the appropriate grid cells, (2) line sources are intersected with grid cells to determine their contribution to each cell, and (3) area sources are mapped onto the grid cells. It is desirable to complete this processing in a fraction of the time required by the other AQM calculations. For this, we have designed very efficient search and intersection algorithms that take advantage of the unchanging topology of the grid.

Another important issue is the identification of criteria to drive the adaptor. A linear combination of solution features such as the gradients and curvatures may be used to generate a weight function. The adaptor will then move the grid points, clustering them around the regions where the weight function has high values. This will increase the resolution where it is needed more. Since the number of grid points is constant, this can only be achieved by decreasing the resolution in other places where it is needed less but the final result is a more favorable positioning of grid points such that the errors are reduced. Deciding which criteria to use is not a trivial matter in AQMs. There are too many species with very different spatial distributions and different contributions to pollution. Including the gradients of all the species in the weight function may lead to a more uniformly spaced grid than it is desired. We are developing criteria that take into account the chemical reactivity of the atmosphere and give more weighting to those species that are more important in the formation of pollution.

The performance of the adaptor and its efficiency to cluster grid nodes where needed, were evaluated using surface elevation data for the island of Hawaii. Initially, surface elevation values were specified on a 4x4 km resolution grid. Then two grids were created with 25% of the number of original data points: a fixed grid with 8x8 km resolution and an adaptive grid with variable resolution. The adaptive grid retained the data values much better than the fixed grid. The adaptor clustered the grid points near sloping terrain including the coastline and mountainous regions. The result was a reduction of the overall absolute and normalized errors as well as the local maximum error.

The development of other pieces of the adaptive grid AQM, such as the meteorological data processor, are underway. Full-blown regional-scale air quality simulations based on the developed adaptive grid and advanced GIS techniques will be performed in the near future.

Comparative Study on Map-projection Based and Direct Spherical Tessellation for Global GIS

Shohaku Ko[1]
Ryosuke Shibasaki[2]

[1]Center for Environmental Remote Sensing
Chiba University
Chiba City, Japan

[2]Center for Spatial Information Science
University of Tokyo
Toyko, Japan

Global environmental problems have attracted increasingly public attention in recent years. The GIS for managing and handling global environmental information (Global GIS) is expected to be a useful tool in global environmental studies. A tessellation scheme of a spherical surface of the globe forms one of the important basis of a Global GIS especially in using raster data. However, conventional latitude-longitude tessellation scheme has several inconveniences, such as low efficiency in data storage and analysis. After reviewing requirements of spherical tessellation schemes for Global GIS, we propose a set of evaluation indices and make a comprehensive comparison of spherical tessellation schemes, including conventional and proposed ones. Through the comparison, North-Up ZOT projection-based tessellation proposed by the authors is found relatively suitable for Global GIS, especially in applying existing GIS software.

Ellipsoidal Area Computations of Large Terrestrial Objects

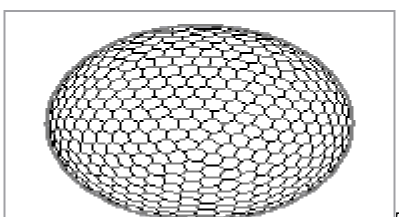
Hrvoje Lukatela, Geodyssey Limited http://www.geodyssey.com/
Paper presented at the International Conference on Discrete Global Grids,
Santa Barbara, 26 - 28 March 2000

Abstract

The mathematics of area computation on the ellipsoidal planetary surface is straightforward; it is however rarely implemented in its rigorous form. Most spatial information systems dealing with two dimensional objects treat the area not as a simple derivative of the object definition geometry, but rather as an artifact of its representation in a particular planar representation. This approach fails when no single, canonical planar representation is either practical or desirable, or when the spatial extent of the object exceeds the useful coverage of a single planar projection.

The Hipparchus geopositioning model represents two dimensional terrestrial objects in the context of an irregular global grid consisting of spheroidal Voronoi polygons. This paper presents the strategy and outlines the implementation of area computation for such objects. It assumes that efficiency is as important as is the precision, and that the objects can be of any size, shape and topological complexity. The speed and accuracy of the computation is examined by applying it to a large, global object of high data volume and considerable topological complexity.

Introduction



This paper presents a particular numerical solution to a rather straightforward and well-defined mathematical problem: given a two-dimensional object of an arbitrary complex topology on a surface of an ellipsoid of rotation, find its area. Several elements of the underlining mathematics are worth noting at the outset:

- The total surface of an ellipsoid of rotation can be expressed as a closed (but transcendental) function of its semi-minor and semi-major axes.
- The area of a pseudo-trapezoidal figure bounded by two meridians and two parallels leads to a single ellipsoidal integral over the latitude domain.

- The area of a triangle - with geodesics as its sides - leads to a double ellipsoidal integral over the geodesic lengths.

Even the mathematically simpler - trapezoidal - decomposition results in an integral that requires a binomial series expansion. In addition, the substitution of the continuously changing width of latitude-bands for a figure bounded by a geodesic leads to an approximation, the magnitude of which depends on both the band height and the local azimuth of the geodesic. The decomposition into triangles results in a similar, but even more complicated numerical solution. In either case, the decomposition of topologically "deep" two dimensional objects into either pseudo-trapezoidal or triangular components can lead to rather complex implementation problems. A good overview of the ellipsoidal geometry and the algebra and calculus used to implement common geodetic computations can be found in either Jordan (1958) or Bomford (1975).

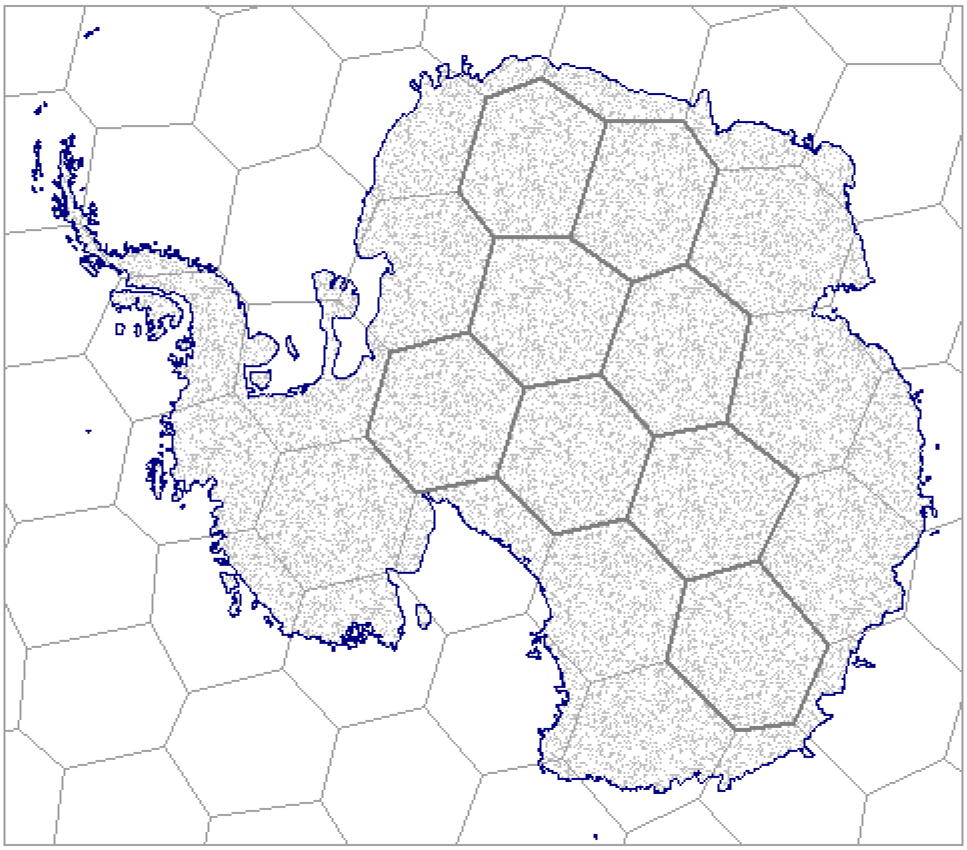
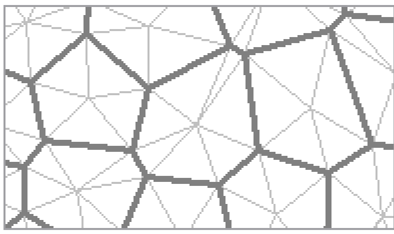


Figure 1: A two-dimensional object in a spheroidal Voronoi grid

If the two-dimensional ellipsoidal surface object is already represented in a form which provides a convenient basis for the geometrical decomposition, it will be advantageous to use such representation as a basis for the area computation. This is the case in the Hipparchus Geopositioning Model, where an object is represented in an irregular global grid of Voronoi polygons. (The full presentation of this model is outlined in a web-resident publication titled: Hipparchus Geopositioning Model: an Overview). It is an example of a "constructive index", where the object geometrical integrity remains intact, and the infrastructure required to access parts of the object that belong to one index block (data cell, tile... etc.) exists "in addition" to the data defining its geometry. Figure 1 depicts an object in a global Voronoi grid. We will - later on in this text - recapitulate those features of this type of object representation which influence directly the area computation. For now, we note only that there are some cells ("boundary cells") through which the object boundary "meanders" and some cells ("interior cells") which are completely encompassed by the object. The area of an object will be determined by summing up the area of all interior cells and adding to this the partial area - that portion covered by the object - of all of its boundary cells.

Area of Ellipsoidal Voronoi Cells



We will first outline the method for determining the area of any Voronoi cell. This is done by subdividing the cell into triangles with neighbor edges as a base and cell center as the opposite vertex. The edges of these triangles are segments of great circles on a sphere related to the ellipsoid (on which both the object boundary and cell-center coordinates are defined by their geodetic coordinates) by conformal ellipsoid to sphere mapping. While these edges are not ellipsoidal geodesics, the maximum line displacement is both relatively small (2.5 meters for a line similar to the one discussed below in the context of UTM mapping) and the "gain" and "loss" tend to be of the same magnitude when all the edges along the cell boundary are considered. The area of these triangles is then computed by spherical productions, but using a constantly changing radius determined by the mean curvature of the ellipsoid surface in the cell center.

The method presented here shares two salient principles with the one devised by Tobler and Kimerling, as described in Kimerling (1984): the first is the use of local ellipsoidal radii in spherical triangle area productions and the second is the tracking of positive and negative area contributions based on the direction of the boundary segment. The latter parallels the procedural makeup of the planar polygon area computation and provides an effective way to keep the non-numerical programming complication within reason, while at the same time avoiding any restriction on the level of the topological complexity of the objects for which the area is calculated.

In geodetic computations, the expression for the mean radius of the ellipsoid surface in a given point is commonly referred to as Gauss' formula: `r = sqrt(m*n)` (where `m` and `n` are radii of the meridian and prime vertical, respectively). The conventional computation of the mean radius uses the ellipsoidal eccentricity term - an unnecessary complication in computation on digital computers. The preferred method uses `tsq`: a latitude (`phi`) dependent square of the free term of the tangential plane, defined directly as a function of the sine and cosine of the point of tangency and the major and minor ellipsoid semi-axes (`a` and `b` respectively), as follows:

```
s = sin(phi)
c = cos(phi)
```

```
tsq = a*a*c*c + b*b*s*s
r = (a*a*b)/tsq
```

This method of mean ellipsoid radius computation is especially convenient for systems such as Hipparchus, which represent the location of a point using the direction cosines of its ellipsoid normal: the evaluations of `s` and `c` require no expensive transcendental computations. Also, the `tsq` term finds a repeated use in many different ellipsoid geometry propositions.

No *a priori* error term has been derived for this "finite element" ellipsoid-to-sphere approximation; the total area of all cells so computed can however be compared (and adjusted) to the total ellipsoid surface. (See below, under "Some Numerical Results", for details). Since we assume that a system will be required to produce the area of many different objects, represented in the context of the same global grid, an array of areas - one element for each cell - will be conveniently stored with the other data in the structures representing the Voronoi polygons.

Computing the area of a spherical triangle for which the lengths of all three sides (**a**, **b** and **c**) are given can be done using two different approaches: by applying Legendre's approximation, or by L'Huilere's theorem. The former states that the area of a spherical triangle will be getting closer to the area of a planar triangle with the same side lengths, as the ratio of the triangle perimeter divided by the spherical radius gets smaller. The latter is, on the other hand, a rigorous evaluation of the spherical triangle area in terms of the three sides. To compare the two:

```
P = sqrt(s*(s-a)*(s-b)*(s-c))
```

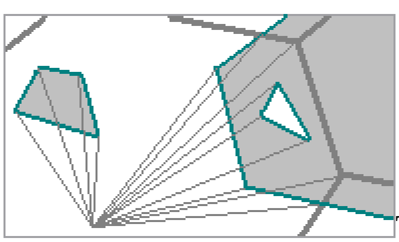
versus:

```
P = 4*atan(sqrt(tan(s/2)*tan((s-a)/2)*tan((s-b)/2)*tan((s-c)/2)))
```

(where `s` is the common shorthand notation for the half-sum of all three sides)

Implementations which impose a low maximum cell radius limit might take advantage of a considerably faster Legendre's approximation. The implementation used to derive the numerical examples given below is, however, based on the second (L'Huilere's) expression, applied to a "local" sphere with the radius equal to the mean ellipsoid radius at the cell center.

Area Computation in Boundary Cells



The determination of the area of an object in a boundary cell requires no additional mathematics. If an object is topologically well-defined, its area inside a boundary cell will consist of a finite number of distinct "faces", each bounded by a closed ring. The ring may consist of either the segments that all belong to the object boundary, or, in the general case, a combination of object boundary and cell edge segments. In a procedure directly paralleling the usual implementation of planar polygon area computation, we can traverse the ring, and accumulate triangular areas subtended from each segment and the coordinate origin - adding or subtracting, depending on the radial direction of the boundary segment respective to the coordinate origin.

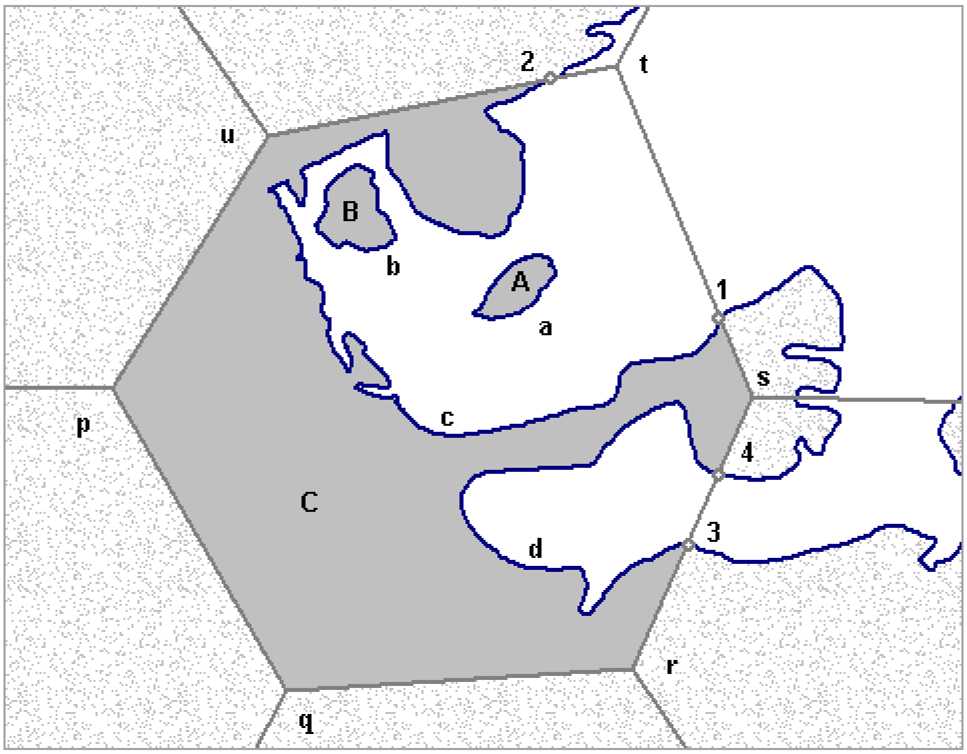


Figure 2: Geometry of a boundary cell

We will next identify some area computation pertinent features of the object geometry definition in an example of a boundary cell, as depicted in Figure 2. (It is an enlarged part of the object shown in Figure 1). The topological decomposition of the object presented in the following is essentially the same as it would be if the object was a planar one - thus additional details, definitions and code segments can be found in many sources describing planar computational geometry - for instance Bowyer (1983).

The fundamental geometrical element is a "fragment": an ordered array of vertices forming the object boundary, in either a closed ring, or starting and ending on the cell boundary. There are four boundary fragments in the example: **a**, **b**, **c** and **d**; two of them are closed rings (**a** and **b**), and two are connecting cell boundary crossing points (fragment **c**, connecting crossing points **1** and **2**; and fragment **d**, connecting points **3** and **4**). For each fragment, we note that the start point is the point where the object boundary "enters" the cell, and the fragment end point is the point where the boundary "exits" the cell.

Fragments are directed in the mathematically positive sense, such that the object interior is always on the left-hand side of the boundary line. This ensures that the "voids" will produce a negative contribution to the total area, regardless of how "deeply nested" the topology of the object is.

One or more fragments form the boundary of a "face" - a single continuous, connected area. In the example, there are three faces: **A**, **B** and **C**. Faces **A** and **B** are formed by single ring fragments **a** and **b**. Face **C** is formed by fragments **c** and **d**. In addition to the two fragments, the boundary of face **C** also requires two cell-edge line segments: first one from the end point of the fragment **d**, to the cell vertex **s**, to the start point of the fragment **c** and the second, from the end point of fragment **c**, through the cell vertices **u**, **p**, **q** and **r**, to the start point of the fragment **d**.

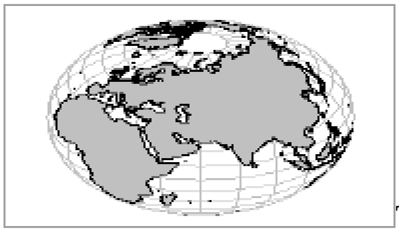
A canonical representation of the Hipparchus system two-dimensional objects identifies only the distinct boundary fragments, and not the faces they form. No spatial algebra proposition (e.g., "polygon overlay") requires this knowledge, and all such propositions would thus be burdened with the additional code if required to keep track of the fragment/face relationship. The area computation algorithm must therefore "construct" the faces as and when required. This information is used implicitly, in the ring traverse order, and is not stored permanently. This "construction" is trivial for faces which are formed by a single closed fragment (**A**, **B**); and somewhat involved for the faces (**C**) that are composed of both object and cell boundary.

The computation of an object area in a boundary cell consists of two phases. The first one is a simple traverse of all fragments. Closed fragments present no special problem: their area contribution is accumulated as they are encountered. For open fragments (i.e., those that start and end on the cell boundary) both boundary crossing points are stored in a table, which lists the point type (start or end), fragment identification (in form of a pointer), cell neighbor index of the crossing point, and the distance from the closest "upstream" cell vertex. If the cell across the **p-q** edge is the first (index **0**) neighbor in the (circular) list of cell neighbor cells, then for fragment **c** two entries are stored in the table: one for entry point **1**, with neighbor index **3** and **s-1** distance; and another for exit point **2**, with neighbor index **4** and **t-2** distance. Similar entries would be made for boundary points **3** and **4**, when fragment **d** is processed. In addition, a list of fragments is doubly-linked with the table elements. This ensures that at the end of the traverse of an open fragment, its end-point can be retrieved from the table.

The boundary crossing point table is then sorted, with the neighbor index as the primary and upstream vertex distance as the secondary ordering element. The table in the example would thus be reordered as **3**, **4**, **1**, **2**. This table is considered to be circular, just like the ordered list of cell neighbors.

In the second phase we traverse the faces by starting at the first previously unvisited cell entry (fragment start) point retrieved from the sorted table. The table element is marked as "visited", and the fragment is followed - accumulating the area at each fragment vertex - until the fragment end (cell exit) point is encountered. The table is then searched for the next (in the circular sorted table order) fragment start point. When one is found, the cell vertices (if any) between the two points are traversed and their area contribution is accumulated. If the found entry point has not been visited before, we mark it and traverse its fragment, if it was visited before we have completely encircled a face. Another "unvisited" entry point from the table (if any are left) starts the same process for the next face; if none are left, the boundary cell area computation is completed.

Some Numerical Results

The numerical testing and verification of the area computation presented here differs markedly from that used in Kimerling (1984). There, the comparison is made between the area of a rhomboid computed from its vertex coordinates in UTM projection and the area computed on the ellipsoid using geodetic coordinates of the four equivalent vertices. However, the straight line in UTM projection with a length of almost 250 km (for the largest of the verification figures) is in the general case different from the projection of the geodesic - in this case, the maximum displacement between the two is considerable, and varies significantly between the easterly (11.3 meters) and westerly (28.9 meters) edges. If the geodesic edges connecting the vertices on the ellipsoid are projected back to the UTM plane (as a sufficiently dense array of vertices), and the area is recomputed, it changes by approximately 1 in 10000 - quite a bit above the precision level of the ellipsoidal area computation claimed by both methods.

Numerical and timing tests performed and presented here use a data object derived from the world coastlines coverage of the Digital Chart of the World (see References, DCW, 1992). It provides multiple "layers" of general-purpose geographical data, commensurate in the precision and density with that of a 1:1 million paper map, using angular geographic coordinates in 5 degree "tiles" on the WGS 84 ellipsoid.

Before the DCW data could be used, however, numerous topological inconsistencies - occurring primarily on the DCW tile edges - had to be detected and removed. The resulting data set consists of 1.3 million vertices, in 27.3 thousand boundary rings - the largest of them encompassing all of the landmass of Europe, Asia and Africa. As a Hipparchus canonical 2-dimensional data set, the object size is slightly over 12 MB. (Raw coordinate data - at 8 bytes per point - takes approximately 11 MB of that).

The average coastline boundary segment is slightly under one kilometer, and less than 1.5% of the segments exceed 3.5 kilometers in length. It is thus safe to assume that the boundary segments - computationally represented by segments of great circles on a conformal sphere - are coincident with the projection of the ellipsoid geodesic connecting the two boundary vertices. (For objects with long boundary segments Hipparchus vector algebra offers a fast yet highly precise approximation of the geodesic line computation as a mid-point between two points: one on the direct and the other on the inverse vertical intersection. (Details of these and other vector-algebra based geodetic techniques can be found in an online Hipparchus Tutorial).

By simply inverting the order of boundary vertices in each ring, we can produce two conjugate objects: one representing the continental landmasses and islands, and the other representing the global Ocean. Both objects are represented in a global Voronoi grid of 2432 cells. The area of both objects has been calculated, with an obvious expectation that their perimeters (a natural by-product of the computation) will be the same, and that the sum of their areas will be equal to the total planetary surface.

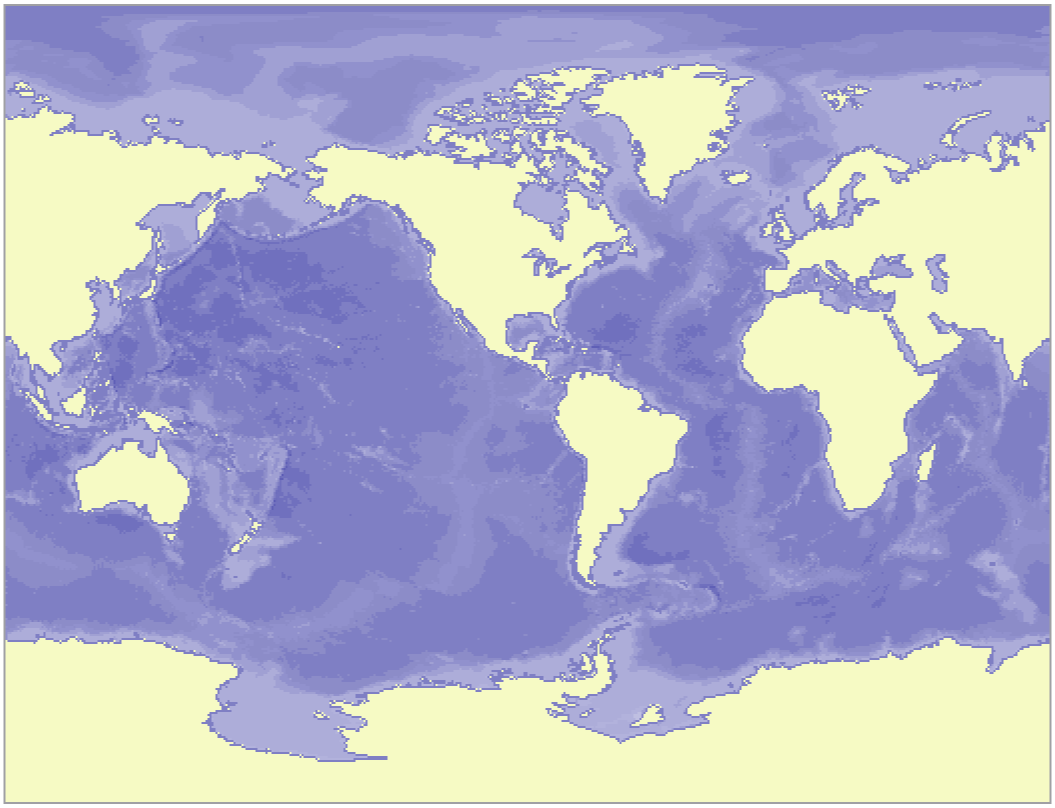


Figure 3: Seven Seas - A Geometry Object

The area computation program initialization consists of the instantiation of the Voronoi grid as a memory-resident object and of the steps necessary to establish the memory-mapping access to the files containing the two objects. The area computation is packaged as a Hipparchus Library function named h7_RsetAreaPerimeter(); its parameters are four pointers to given data: to the Voronoi grid descriptor, some workspace, ellipsoid geometry parameters, and header data of the object for which the area is required; plus two pointers to returned values: the area and the perimeter.

The results presented here have been obtained using the code compiled with the GNU C compiler V.2.95.2, carried out on a 400 MHz Pentium II under Linux kernel 2.2.14. (Performance under NT was only marginally slower).

```
Ellipsoid: WGS 84, a=6378137.0e0, b=6356752.3141e0
```

The 2432 element cell-array area initialization took 0.19 seconds and produced the following values (square meters):

```
Ellipsoid area:           510065621716336.1
Total area of all cells: 510065575723515.5
Difference:                       45992820.6
Relative difference (one in):   11090113.0
```

Area computation for both objects took 3.94 seconds (each), and produced the following values (square meters, meters):

```
Landmass area: 150998900960532.0, perimeter: 1249923047.850
Oceans area:   359070890924373.1, perimeter: 1249923047.850

Ellipsoid area:           510065621716336.1
Landmass + Oceans         510069791884905.1
Difference:                     4170168569.0
Relative difference (one in):     122313.0
```

Conclusions

The following conclusions seem to be justified:

- Computation of the ellipsoid cell areas using local mean curvature at the cell center produced the results which are within one in 11 million of the total ellipsoid surface area. For the grid and objects as examined in the example, this is two orders of magnitude better than the results of the object area computation, and thus adequate.
- Computation of two very large planetary complement objects produces results which are within one in 122 thousand of the total ellipsoid surface area. The obtained level of accuracy is a result of a combination of factors; the major one probably the inevitable rounding error in a very large number of (over 1.3 million) of relatively narrow triangles. The obtained accuracy compares favorably with that in many operational systems which compute the areas in the plane - at the outset a significant departure from the true object geometry.
- The very short time needed to carry out the area computations of large objects makes it practical to compute it only as and when this information is required. This facilitates the system design in which the area of a global object of any size and/or shape is not treated as an independent attribute, but rather as only one in a repertoire of measures that can be derived from its canonical geometry definition.

References

1. ______, *Digital Chart of the World* (CD publication), United States Defense Mapping Agency, Fairfax, VA 22031-2137. July 1992.
2. Bomford, G. *Geodesy*, London, Oxford University Press, 1975 ed.
3. Bowyer, A and Woodwark, J. *A programmer's geometry*, Sevenoaks, Butterworths, 1983.
4. Jordan, Eggert and Kneissl, *Handbuch der Vermessungskunde* (Band IV: Mathematische Geodaesie), 10th ed. Metzlersche Verlag., Stuttgart, 1959
5. Kimerling, A. J.*Area Computation from Geodetic Coordinates on the Spheroid*, Surveying and Mapping, Vol. 44, No. 4, 1984, 343-51.

A Seamless Global Terrain Model in the Hipparchus System

Hrvoje Lukatela, Geodyssey Limited **http://www.geodyssey.com/**
Paper presented at the International Conference on Discrete Global Grids,
Santa Barbara, 26 - 28 March 2000

Abstract

This paper outlines the steps used to construct seamless global triangular networks similar to the planar TIN commonly used to facilitate terrain modeling and volumetrics. It is based on global coordinates and a planetary surface tessellation using spheroidal Voronoi polygons. The techniques used to extend the surface modeling across the Voronoi cell boundaries are presented. The paper also outlines the strategy used for the efficient retrieval of only those parts of the model that are visible in a transient view, as well as the platform- and projection-independent approach to surface rendering.

Introduction

Representation of the physical surface of the Earth in digital systems is a subject of considerable current attention. As the area of the coverage of such systems increases, it becomes necessary to provide methods to model very large, continuous surface conglomerates in a manner which does not violate the surface integrity (i.e., which does not impose hard partitioning as an artifact of the digital model), but, at the same time, provides an efficient spatial index to a small section of the surface of transient interest.

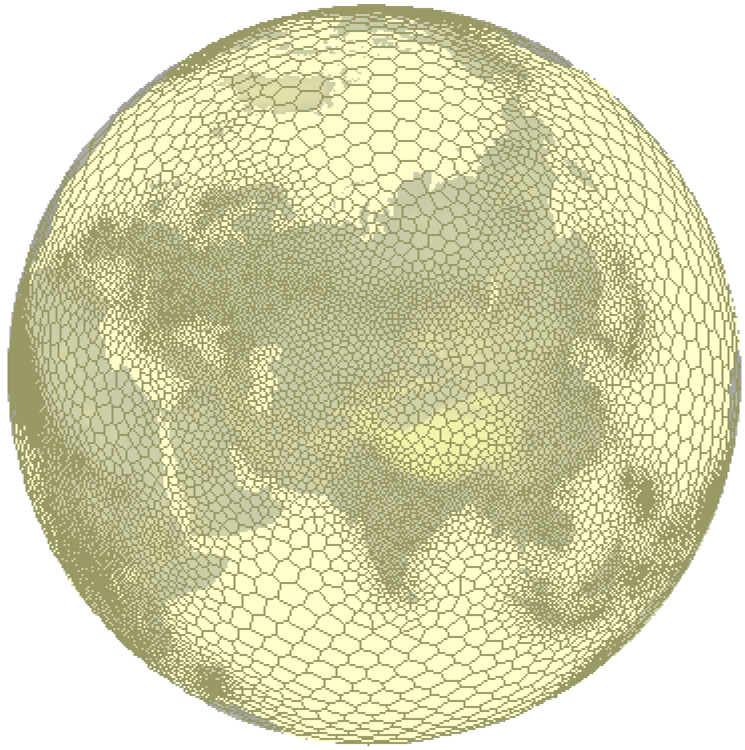


Figure 1: A "global grid" of Voronoi polygons

The Hipparchus Geopositioning Model (outlined in considerable detail in a web-resident publication titled: Hipparchus Geopositioning Model: an Overview) provides a method of construction and manipulation of geometric objects of various levels of complexity (points, lines, areas and surfaces) in a manner which imposes no restrictions on their spatial size or shape. The system is based on two major computational geometry constructs: use of the vector components of the ellipsoid normal instead of the latitude and longitude angles and data-density driven tessellation of the planetary surface using a global grid of spheroidal Voronoi polygons.

The position of a point on the surface of the ellipsoid is best given by a numerical definition of the normal to the surface at that point. The common global coordinates - latitude and longitude - are angles: the first of the two is between the normal and the Equatorial plane and the second between the Equatorial plane projections of the normal and the projection of the prime meridian. Traditional geodetic computations (for instance: given are two known points, determine the length and the azimuth of the shortest line that connects them...) are based on trigonometric functions of those angles and on the expansion of the power series of the eccentricity of the ellipsoid. The angles, however, present two problems when used for computations on digital computers: transcendental functions (sines, cosines) require many more CPU cycles than the algebraic primitives (addition, multiplication) and their areas of singularity must be compensated with complicated and error-prone code. Thus replacing angles with vector components of the ellipsoid normal was noticed as potentially beneficial as soon as the digital computers were incorporated into the geodetic practice (cf. Bomford (1975), remark on formulae "symmetrical and better for computation...", p. 593 - under 'Cartesian Coordinates in Three Dimensions').

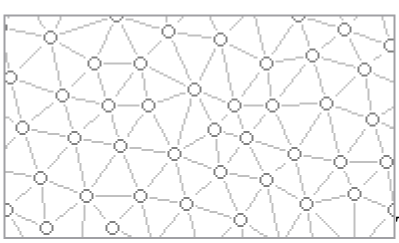
Hipparchus computations are consistently based on the ellipsoid normal given by its vector components instead of the latitude and longitude angles.

A "global grid" is, in the most general sense, a geometrical subdivision of the planetary surface which assists in the organization (partitioning, indexing, etc.) of globally-distributed digital data. "Regularity" of the grid usually translates to simple data structures and straightforward classification algorithms. Since no regular and isometric tessellation of the sphere (beyond five platonic solids) is possible, practical applications have two alternatives: to retain in the design of the "grid" as much "regularity" as possible, to be rewarded with a considerable algorithmic elegance, but at the expense of the ability to perform fast and accurate geodetic computations (see Dutton, (1999) for a well-known example of such approach), or to accept the "irregular" nature of the grid while attempting to make computations based on it as fast and as precise as possible.

The Hipparchus system makes no attempt to produce a "regular" grid. Instead, the grid is designed so that any particular implementation of the grid can match the density of the data that inhabits it and so that the parameters which numerically define the grid "cells" assist (and not hinder) the speed and precision of geodetic computations and spatial algebra productions. (Figure 1 shows a sample of such grid with the density derived from the density of the human population). It is based on the surface subdivision known as "Voronoi polygons": a purposefully selected finite and discrete set of "cell-center" points subdivides the surface so that any surface point is uniquely associated (i.e., it "belongs to") the member of the set that it is closest to. The point classification is accomplished using only distance calculations. Point, line, area and surface object sets are in turn defined in terms of the cells they occupy and the vertex coordinates, represented not by a the full (global) vector representation of the ellipsoid normals, but by the vector difference from their respective cell center points. An extended introduction to and computational geometry treatment of the Voronoi polygons can be found in O'Rourke (1994).

This paper outlines the canonical representation of the surface object, and explores the techniques which are used to operate upon it. While the context and the examples used in this paper center on the representation of the terrain, such objects can be used to represent any continuous function for which a sufficiently dense set of point-location dependent scalar values are known, and about which we know enough to postulate that each planetary surface point will have one (and only one) measured or interpolated function value.

Triangulated Irregular Network - TIN

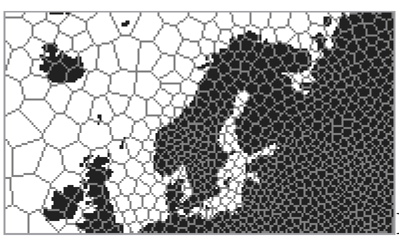


The triangulated irregular network (TIN) is an often used surface representation method in planar computational geometry. Description of TIN data structures and the algorithms (including C language source code) can be found in both Ambroziak (1993) and Lischinski (1994). The TIN approximates a continuous surface with a mesh of triangles which more or less coincide with it. The quality of this approximation depends on the combination of the method by which the elevation measurement points have been selected, and the triangulation strategy. The implementation described herein assumes a more-or-less uniform density of significant points and starts with a simple "least-diagonals" fast iterative planar triangulation algorithm. It is open to accept different triangulation strategies - presumably matching more closely the peculiarities of the input data generation process.

The data structure used to represent a planar TIN is simple and shows only minor variations from one implementation to another. It consists conceptually of an array of points and an array of triangles. The two may be doubly linked, but commonly only the triangles are linked to their vertices. Additionally, each triangle is linked to its three neighbors, with a flag value to signal that a triangle edge is at the same time the edge of the TIN (i.e., it is an "outside" edge). Most triangulation algorithms produce a structure in which all outer edges form a planar convex hull. A point array element consists of planar coordinates and elevation. If the shading or perspective rendering is anticipated, triangle normals might be included in the data; this however would be worthwhile only if the cost of re-computing the triangle orientation far outweighs the cost of additional storage and, usually of even greater significance, the time required to access it.

The main difficulty in the implementation of computational geometry algorithms usually stems from the need to properly predict all (or at least all likely) degeneracy and singularity cases. (See in particular comments under "Robustness" in Lischinski (1994)). While the implementation described in this paper is no exception, it is interesting to note that all such problems were encountered (and hopefully resolved) in the planar ("in-cell") geometry domain.

A TIN in a Global Voronoi Grid



In general, the Voronoi polygon grid used to model the TIN will be used to provide the spatial framework for a number of other object classes. The only requirement that the implementation of TIN data will place on the grid is that the density of the elevation points remains approximately an order of magnitude above the density of the Voronoi polygon centers. In this - and a number of other characteristics - a TIN object in a global Voronoi grid parallels the combined characteristics of Hipparchus point and line sets. This TIN will additionally consist simultaneously of two levels of triangle/elevation data: high-volume triangles with source data points as their vertices and low-volume, large triangles with cell-center and end-points of cell edges as their vertices.

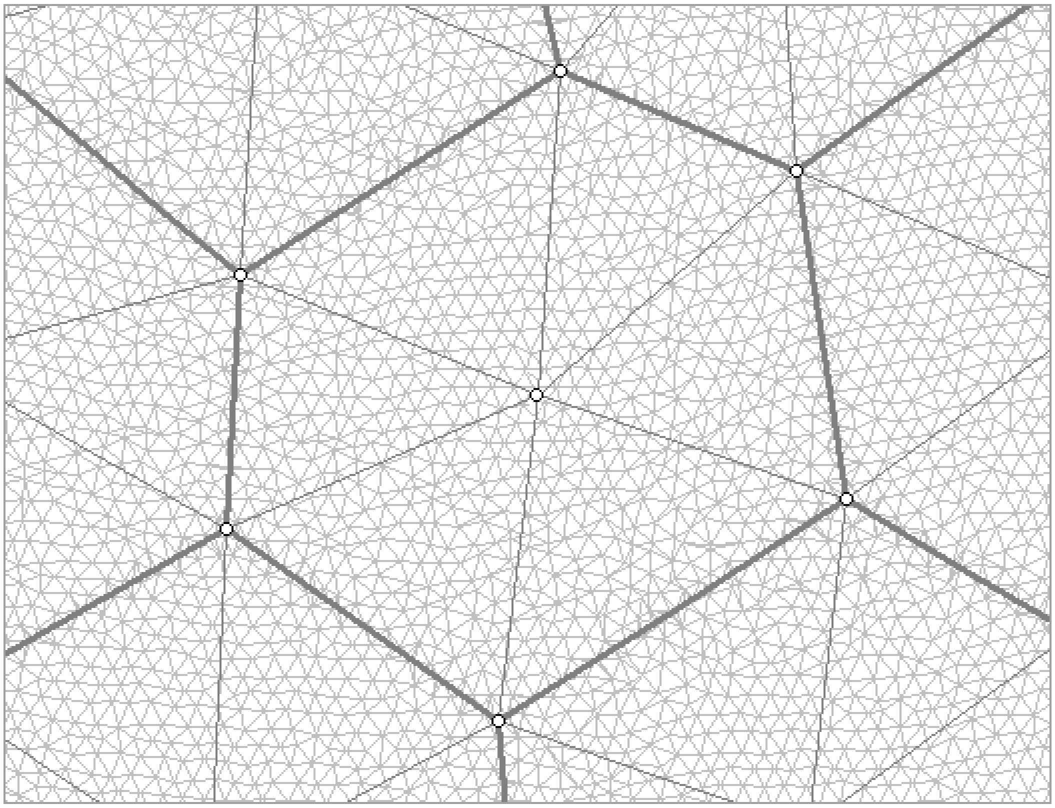


Figure 2: TIN in a Voronoi grid

The source data used in a construction of the Hipparchus TIN object is a Hipparchus point set with elevations. The construction algorithm proceeds cell by cell, transferring the point coordinate and elevation data from the point set into a two-level TIN structure consisting of both individual triangles and the values that describe the elevation of cell-sector triangular vertices. In each cell, the process starts with a selection of all those points that are inside the cell and those points in the neighbor cells that fall within some distance of the shared edge. A triangular grid produced in this step is then intersected with the cell boundary. All triangle edge/cell edge intersection points are assigned an interpolated elevation and marked as "points on the hull". (This can be done, since the Voronoi polygon is at the same time a convex hull of all points in its interior). These points are kept in the final surface representation. Description of planar convex hull and algorithms used to construct it can be found in Sedgwick (1983).

A second triangulation (that includes only cell interior and cell-edge points) follows, this time as a faster, "forced hull" process. The triangles of this second triangulation are stored - together with the point coordinate and elevation data - in a final cell-oriented series of structures representing the surface. Similar to other Hipparchus objects, it is a hierarchical set of tables, at the center of which is a table of "cell headers". It describes that part of the surface which is "over" the cell, and its most important elements are:

- Cell identifier
- Coordinate array pointer and count
- Count of the cell interior points in the above

- Triangle array pointer and count
- Triangle neighbors array pointer and count
- Minimum, maximum and mean cell elevations

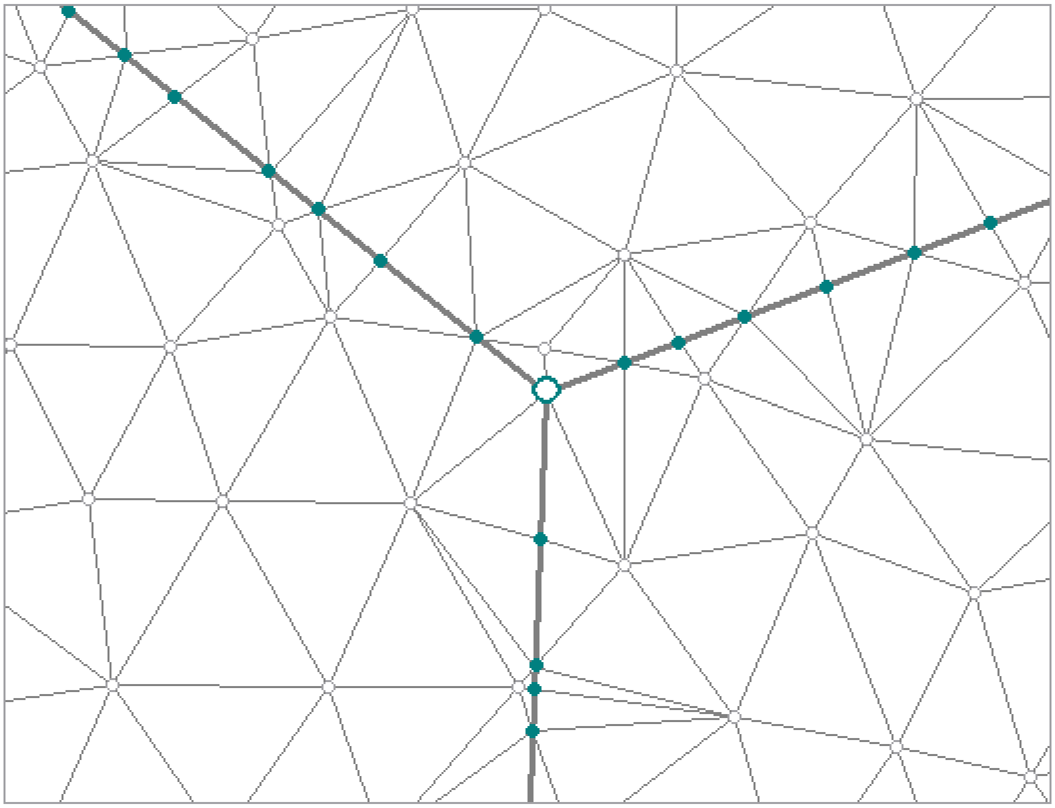


Figure 3: Different types of elevation points

The final TIN point array contains three types of points (clearly distinguished in Figure 3), in order-of-magnitude decreasing numbers:

- Elevation points transferred from the source point set.
- Cell/triangle edge intersections with linearly interpolated elevations.
- Cell vertex points with spatially interpolated elevations.

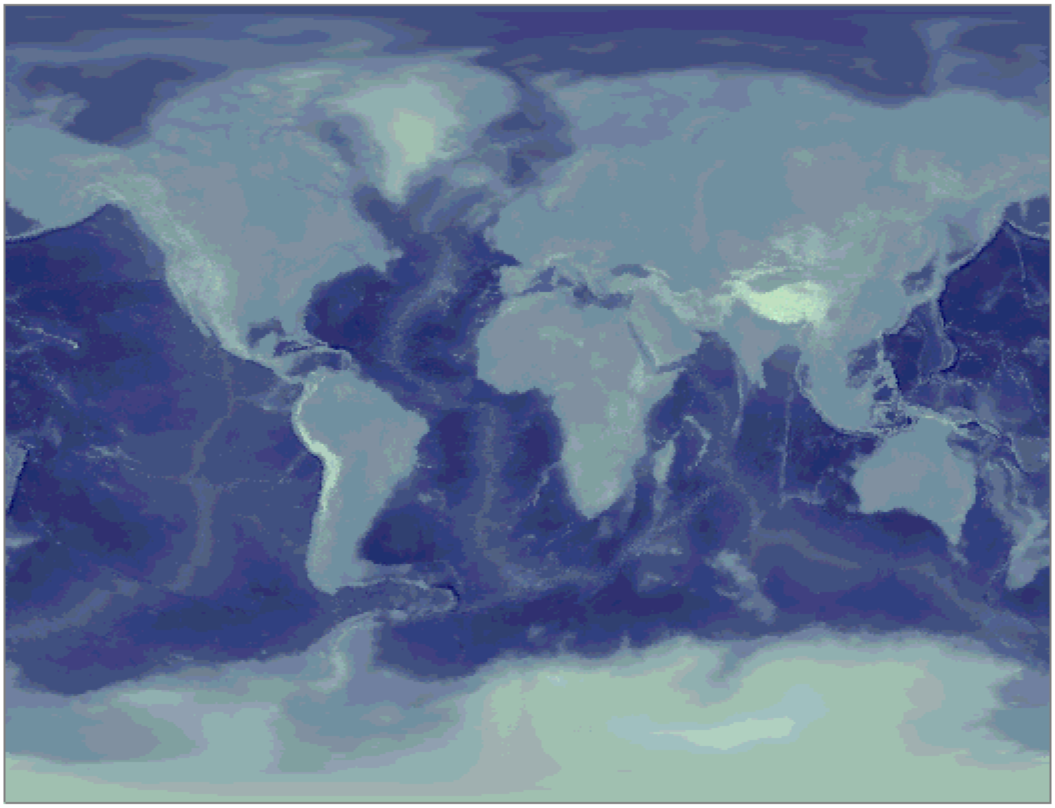
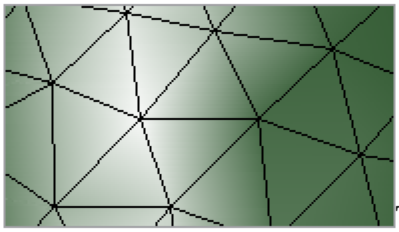


Figure 4: Hypsographic rendering of a global TIN

The test data set (see Figure 4 for its hypsographic representation) used to produce the illustrations of this paper, has been derived from a 5-minute gridded set of elevation points, and used extensively in the development and code testing. A high density Voronoi grid was used as a sampling framework, to avoid an artificially high (and meaningless) data density in high latitudes, and to reduce the point count to a level commensurate with the overall quality of the elevation data. The result of this process was a Hipparchus point set with about 200K points covering the planet in a generally uniform pattern. In addition to its use in the testing, it is anticipated that it will also be used to complete the planetary surface coverage for local (regional, continental, etc.) elevation data sets available in higher density and precision. This strategy is not unlike the frequent use of a Hipparchus Voronoi index center-point set called "isotype", which provides a similar function for regionally-biased grids.

The size of the disk files is approximately 2.5 MB for the source point set, and approximately 8.4 MB as a TIN. In the current implementation, all in-cell triangle indices are 32-bit integers; this removes any practical restriction on the number of triangles in a single cell. Elevations are recorded as 32-bit floating point numbers.

TIN Rendering

TIN rendering is implemented in a series of functions that belong to the general section of the Hipparchus Library dealing with "geographics". A TIN can be rendered in a "reduced dimensionality" mode, to generate only the points and lines (triangle edges) without the full surface representation. This mode was used to generate Figure 3. The Figure and the other illustrations in this paper were created using a scripted geographical workbench program called GALILEO, available for free download from Geodyssey's web page at http://www.geodyssey.com/.

In cartographic applications a TIN is most often rendered as one of the two graphical artifacts: a hypsographic scale color fill or a shaded surface. The first assigns to each pixel a color dependent on the point elevation, the second assigns a color dependent on the spatial orientation (slope and azimuth) of the surface. Figures 5 and 6 illustrate the difference: both show the map background-fill "layer" generated from a global TIN for an area of the European Alps. Hydrographic network and political boundaries are also shown; both are expected to be - to some extent at least - in an easily verifiable relationship to the surface elevations.

Of the two, hypsographic scale color fill is more demanding of the graphical programming, as it requires painting of pixels that belong to a single triangle with a range of colors. (As opposed to shading, where all the pixels that represent one triangle are in general of the same color). We will therefore in the following concentrate on the hypsographic scale TIN rendering.

The central computation geometry process used in this type of rendering is the ubiquitous triangle "gradient fill": given the x and y (image) coordinates of three vertices and their associated "z" value, fill all pixels "inside" the triangle with a color value commensurate to the interpolated value of "z". This interpolated value is a value that a respective point of the plane defined by three points (vertices) would have.

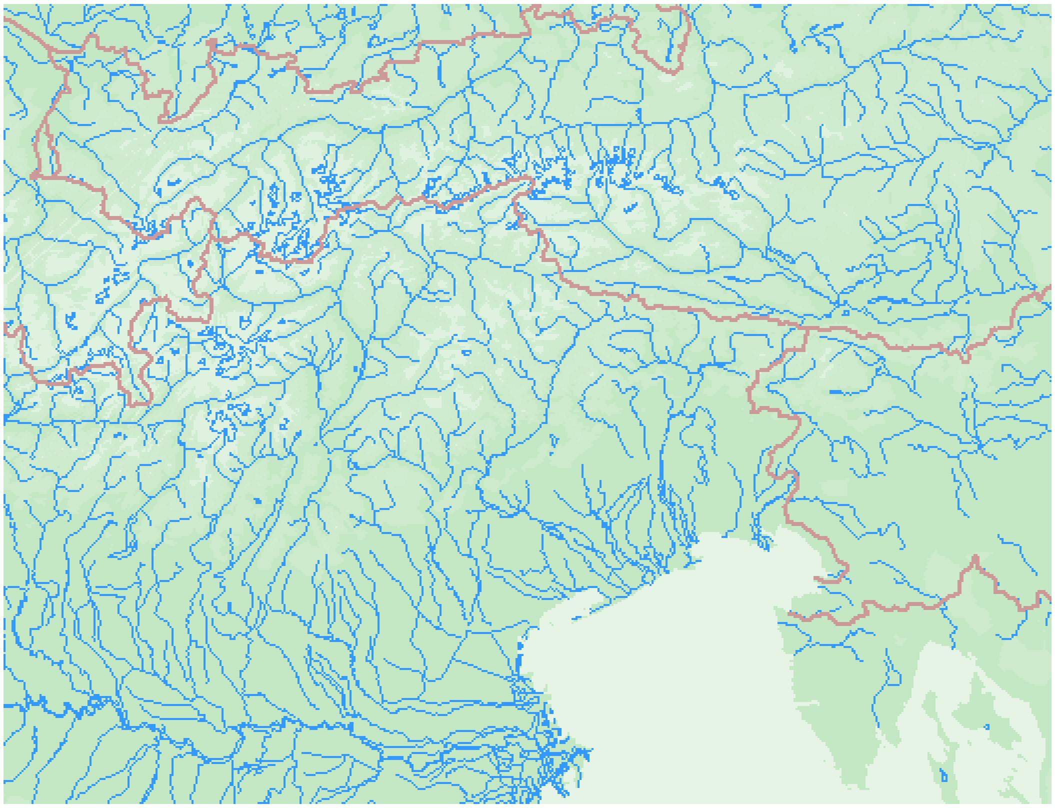


Figure: 5 Hypsographyc rendering

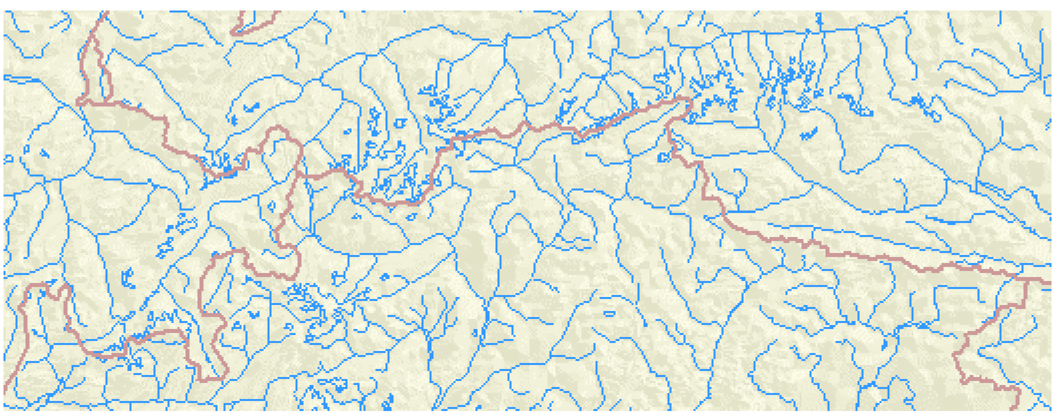


Figure: 6 Shaded rendering

Because of the multiplicity of environments in which this type of rendering is performed, there are different levels at which the division of labor between the application code (including any libraries used), the graphical platform and - possibly - the display hardware can take place. For the Hipparchus Library, we assumed there are at least three such levels, and so provided the means to allocate control to a lower level component (i.e., either the graphical platform or the hardware) at any of those three levels. Their application program interface can be described by the dimension, as follows:

Point

A single pixel is set to a color of the invoker's choice.

Line

Pixels of a scan-line segment with a given y and two end-point x coordinates are set to a linear gradient of colors linearly interpolated between given end-point color values.

Area

All pixels inside a triangle are filled with colors interpolated between the given color values of the vertices.

The first level is available in even the most basic graphical application development environments; the last one is implemented in most "2-D accelerated" graphical device drivers and is supported in graphical platforms targeting such devices. Thus an application rendering a TIN would keep invoking Hipparchus Library functions to the point-level under the Win32 API, and pass a whole triangle to an Open GL API.

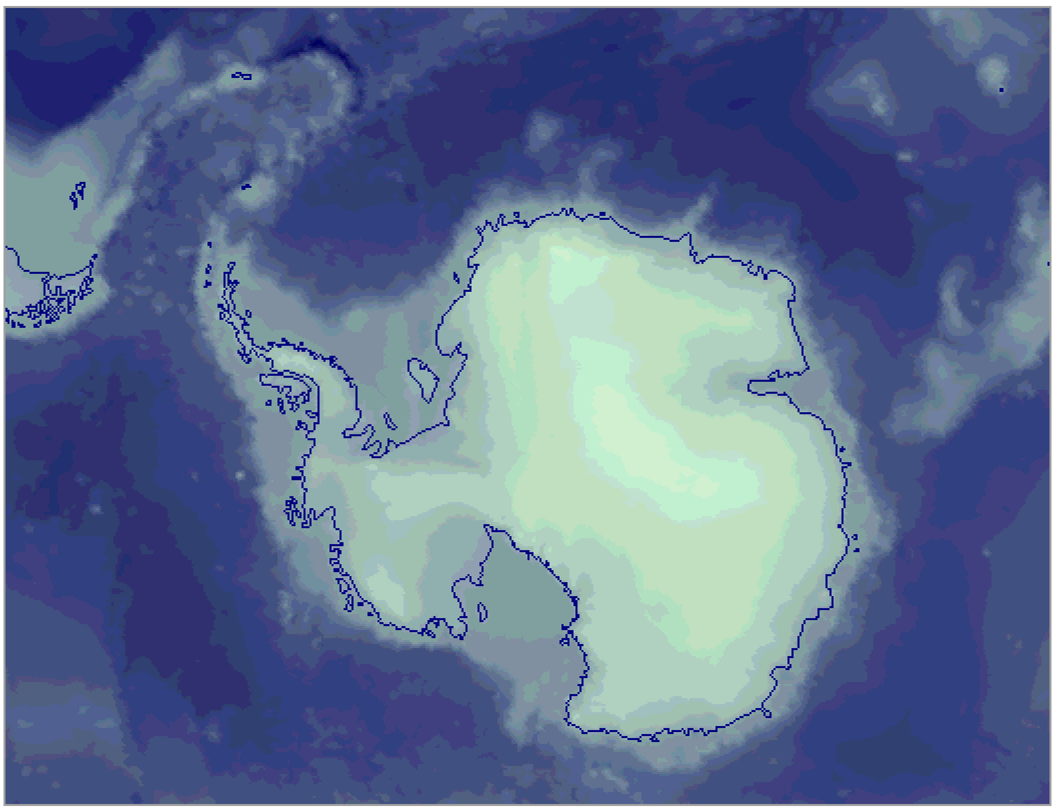


Figure: 7 Sample surface, rendered in (relatively) large scale

A high-latitude section of the global TIN depicted in Figure 7 illustrates another unique advantage of the TIN representation in a Voronoi grid. As mentioned before, all vertices on the cell edge will have an interpolated elevation assigned to them. Likewise, a mean elevation value for the whole cell is a natural by-product of the TIN construction process. This data (cell mean and edges values) can be used as a "generalized" representation of the same TIN: it consists of a set of cell-segment level triangles, probably an order of magnitude fewer per given area than the original TIN. (cf. outlined points and large triangles formed by them in Figure 2). When the scale of rendering becomes sufficiently small, the application can choose to render only the large triangles, thus decreasing the rendering time considerably. Applications will often store Hipparchus Binary Objects (on disk or in a database) so that the low-volume cell-related values are separated from the high-volume point coordinates. If this is the case, rendering based only on the cell-level values (such as depicted on the small-scale map in Figure 8) will require at least an order of magnitude fewer disk transfers than the rendering of Figure 7 - despite the fact that both are generated from the same surface object.

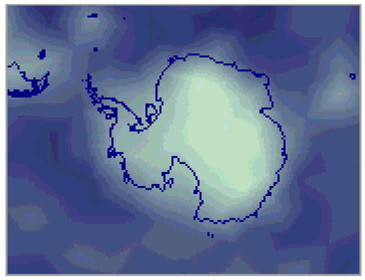


Figure 8: Cell-level generalization

Conclusion

A TIN as a Hipparchus Binary Object shares many architectural similarities with point sets, line sets and regions constructed and recorded in the context of global Voronoi grids. Its construction is based on a planar triangulation within the cell boundaries. Triangle information storage follows the common planar TIN model, but an additional set of elevations is stored to represent the cell edge elevation profile. Simple, straightforward and conjugate linear interpolation of elevations on the edge guarantees that no artifact will be introduced at the cell edge. The cell-based data structure representing the whole TIN provides a simple and fast determination of the elevation at an arbitrary selected surface point, and triangle rendering fits well with the API of common graphical platforms. A novel approach to the problem of generalized rendering is another benefit of the Voronoi grid.

References

1. Ambroziak, R.A., Cook C.A., Woodwell, G.R. and Wicks R.E.: *Data, Software, and Applications for Education and Research in Geology* (CD publication), United States Geological Survey Open File Report 93-231, 1993.
2. Bomford, G. *Geodesy*, London, Oxford University Press, 1975 ed.
3. Dutton, G. H.:*A Hierarchical Coordinate System for Geoprocessing and Cartography*, Springer Lecture Notes in Earth Sciences 1999.
4. Lischinski, D.:*Incremental Delauny Triangulation*, article published in The Graphics Gems Series, Volume IV, pp. 47-59, AP Professional 1994.
5. O'Rourke, R.:*Computational Geometry in C*, Cambridge University Press 1994.
6. Sedgewick, R.:*Algorithms,* Addison-Wesley 1983.

•••

Digital Earth: Building the New World

Yvan G. Leclerc, Martin Reddy, Lee Iverson, and Nat Bletter

SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025, USA

Abstract: Digital Earth is a massive, distributed framework for managing and visualising georeferenced information over the Internet. Imagine being able to walk down the streets of Paris, seeing the sights and hearing the sounds around you, without leaving your home. You are able to converse with a friend who is also online, and the two of you decide to fly to Tahiti together. As you travel around the globe, you stop momentarily at places of interest to view three-dimensional (3D) representations of local structures, learn about representative works of art, browse historical information for the area, or view real-time video and audio feeds of local events. This vision presents virtual reality with many substantial challenges, for example: rendering complex georeferenced data for the entire planet, managing and displaying embedded multimedia data, visualizing massive scenes in real-time, and developing useful navigation interfaces for interacting with these scenes.

Keywords: 3D geography, georeferenced information, multiuser, terabytes, terrain visualisation, virtual heritage, virtual environments, visual simulation, VRML.

1. Introduction

The Digital Earth project at SRI aims to build the infrastructure and visualisation tools for a massive, open, distributed, interactive, multi-resolution, 3D representation of the earth, in which multifarious information can be embedded that refers to specific points on that earth. No system has been previously attempted on the scale that we propose here. Digital Earth is a compelling multimedia experience that presents virtual reality with a number of significant challenges. Before proceeding, we will expand upon our opening description to clarify its meaning and implications.

- By the term massive, we mean quantities of data that could not be stored on a single disk or maintained by a single organisation. Most existing terrain visualization or flight simulator applications work by pre-loading all data into main memory on start up. However we will be dealing with aggregate data sizes in the order of many terabytes and therefore cannot afford this luxury.
- SRI's Digital Earth project is an open venture in that we intend to use open standards to build the Digital Earth, as well as contribute freely-available data and

tools to the general public. We feel that this is an important issue in today's marketplace.

- Considering the massive quantities of information that we envisage, it is necessary to distribute the Digital Earth contents over many servers around the world and to allow multiple organisations to contribute and manage their own data. This work will take advantage of emerging network technologies such as the Next Generation Internet or Internet2.
- The Digital Earth will be interactive in that users can actively control their destination and flight path around the planet. They can also select types of information that they wish to browse and view multimedia content such as movies, images, and text. We also desire collaborative capabilities so that multiple remote users can communicate with each other, follow the same paths, or perform teacher-student learning activities.
- In order to interact with massive quantities of distributed data in real-time, we need to employ hierarchical level of detail (LOD) techniques. A single computer could not possibly handle all of the data in the Digital Earth if it were transmitted as a flat structure. For example, colour imagery for the entire world at 1 m resolution would require over 1 petabyte (10^{15} bytes) of memory. If we had elevation data for the world at 30 m resolution, this would produce a geometric model of over 500 billion (5×10^{11}) polygons.
- Finally, the Digital Earth involves many different types of georeferenced data, not just terrain elevation and imagery. For example, we could insert virtual representations of buildings at their correct geospatial location, visualise weather phenomenon, or annotate locations with place names.

This vision sprung from a speech that the Al Gore, Vice President of the United States, made to the California Science Center in 1998 entitled, "The Digital Earth". In this speech, Gore challenges the scientific community to build a *"multi-resolution, three-dimensional representation of the planet, into which we can embed vast quantities of geo-referenced data"* (Gore, 1998). Gore gives the following example of the utility of such a facility:

> *"Imagine, for example, a young child going to a Digital Earth exhibit at a local museum. After donning a head-mounted display, she sees Earth as it appears from space. Using a data glove, she zooms in, using higher and higher levels of resolution, to see continents, then regions, countries, cities, and finally individual houses, trees, and other natural and mad-made objects. [...] She is not limited to moving through space, but can also travel through time. After taking a virtual field-trip to Paris to visit the Louvre, she moves backward in time to learn about French history, perusing digitized maps overlaid on the surface of the Digital Earth, newsreel footage, oral history, newspapers and other primary sources."*

In this paper, we present a technical infrastructure that will enable the Digital Earth to be built and maintained. We also introduce solutions for interacting with and visualizing these data sources using virtual reality technologies. A prime focus of this work is to enable users with a standard PC configuration to view the Digital Earth over the web using standard solutions such as VRML97 (Virtual Reality Modeling Language); but also to allow for more complex configurations such as a CAVE or Immersadesk interface via custom browsing software.

2. Infrastructure

The Web today already allows access to trillions of bytes of data of a wide variety, including text, maps, images, 3D models, music, and video. However, there is currently no way to find all available data about a given geographic location or area because the only organizing principles that span all data are either keywords/keyimages (used by search engines) or hyperlinks. In other words, data on the Web today is typically not georeferenced in any way.

Also, the sheer volume of geographically indexable data about our world, including satellite and aerial imagery, 3D models of terrain and buildings, and maps, is so large that no single organization can possibly produce, store, or even index all of the data. What is needed, therefore, is an organizing principle, or infrastructure, that can be used to georeference any type of data from around the world without having to modify any of the data itself. This infrastructure must be able to:

1. Store the terabytes of data that represents a baseline set of data covering the real world.
2. Allow very fast indexing of petabytes of data of all kinds by geographic area and various metadata attributes.
3. Make it simple for producers to either add their data, or a pointer to their data, within the infrastructure.

Our solution is based on a hierarchy of Web servers that would be distributed around the world. Each server in this hierarchical organization has a DNS (Domain Name System) name that represents a given geographic area of the earth, called a cell. The server is responsible for all data contained within this area and having detail within a limited range. It is also responsible for providing DNS service to those servers corresponding to smaller cells within its own cell, and these servers are responsible in turn for correspondingly more detailed information about their smaller cells.

For example, instead of a domain name like www.ai.sri.com, we might have an address that defines a geographical location using a hierarchal format such as minutes.degrees.tendegrees.geo. An example of this would be 37e47n.1e5n.10e20n.geo. The first part of this address defines a 1 minute x 1 minute grid of the earth, the second part defines a 1 degree x 1 degree grid of the earth, and the third part defines a 10 degree x 10 degree grid of the earth.

Each of these geographic scales is managed by a (potentially) separate server. By limiting the area and range of details in this way, each server becomes responsible for only a tiny fraction of all of the data available around the globe. Furthermore, by using DNS names that represent the geographic service area of the server, clients can immediately determine which server to query without the need for a search engine or global name server.

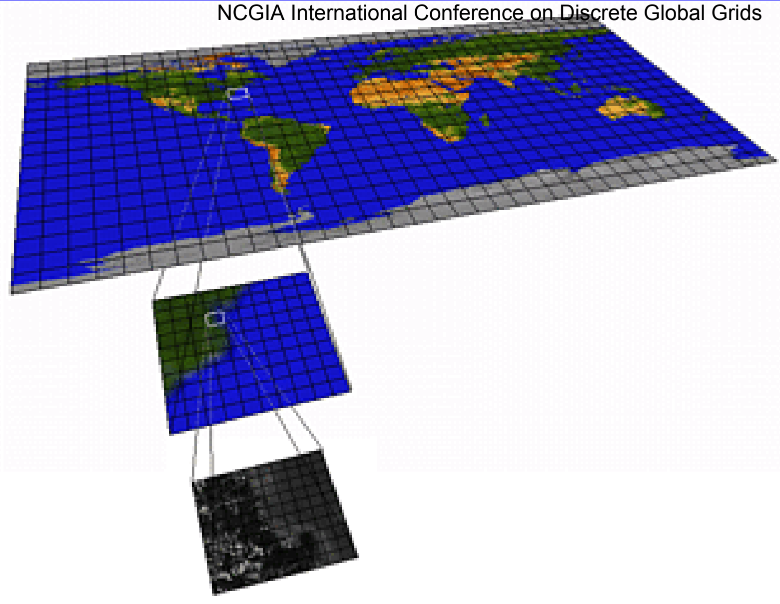


Figure 1. An illustration of the use of DNS to index geographic data hierarchically. Each cell in the top layer is a 10 degree x 10 degree region of the planet with a unique DNS address. The lower layer represents the cell containing Washington, D.C. This in turn contains a grid of uniquely addressable 1 degree x 1 degree cells, which in turn contain a grid of 1 minute x 1 minute cells.

3. 3D Representation

The Digital Earth will consist of huge volumes of distributed data. We require that the user be able to interact with these, experiencing minimal download times and high frame rates. This requires implementing level of detail techniques to manage the streaming and display of terrain data.

Multi-resolution terrain is best represented using a hierarchical structure such as a tiled, pyramid scheme (Falby et al., 1993; Hitchner and McGreevy, 1993; Leclerc and Lau, 1995). This involves progressively downsampling an image or elevation bitmap to produce the multi-resolution pyramid. Each level of this pyramid is then segmented into a grid of equally sized rectangular tiles, for example, 128 x 128 pixels. A tile at one level of the pyramid will therefore map onto four tiles on the immediately higher-resolution level, that is, the tiles at the higher-resolution level cover half the geographical area of the former. Using such a representation, we can progressively transmit higher resolution data around the area of interest while other regions remain in low resolution (see Figure 2, below). The use of tiling also allows us to fetch and display only those sections of the Digital Earth that are visible from a certain vantage point.

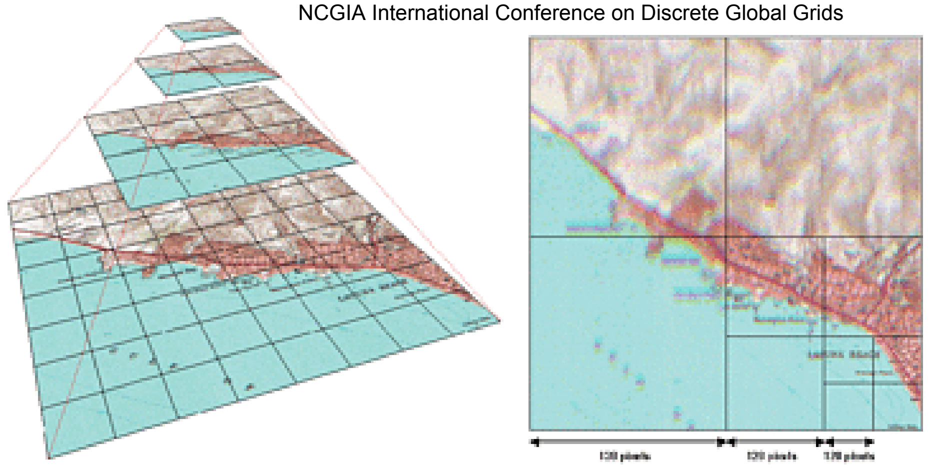


Figure 2. (a) A tiled, pyramid representation for terrain, and (b) Illustrating the use of this structure to produce viewpoint-dependent, multi-resolution imagery.

One of the principal goals of this work is to allow multiple types of georeferenced data to be embedded into the global infrastructure. For example, we could have 100 km resolution data for the entire globe but recursively insert higher-resolution datasets for smaller regions of interest, for example, a 1 km resolution dataset for the conterminous United States and a 1 m resolution dataset for Yosemite Valley, CA. Then we could insert 3-D models for buildings in Yosemite Valley, some of which could contain hyperlinks to various multimedia presentations about the National Park.

4. Coordinate Systems

Most computer graphics system use a Cartesian coordinate system to model all objects in 3-D space (e.g. OpenGL, 1997; ISO/IEC, 1997). In terms of georeferencing, this coordinate system is most similar to a geocentric coordinate system, where all locations are specified in units of meters as an (x, y, z) offset from the center of the planet. However, most georeferenced data are provided in some geodetic or projective coordinate system.

A geodetic (or geographic) coordinate system is related to the ellipsoid used to model the earth, for example the latitude-longitude system. A projective coordinate system employs a projection of the ellipsoid onto some simple surface such as a cone or a cylinder, for example the Lambert Conformal Conic (LCC) or the Universal Transverse Mercator (UTM) projections, among many others (Synder, 1987). Each of these coordinate systems was designed for slightly different applications and offers particular advantages and restrictions. For example, some projections can only represent small-scale regions, others are conformal (same scale in every direction), and others can be equal area (projected area corresponds to the earth's physical area over the entire projection). Figure 3 provides an illustration of a number of contemporary coordinate systems.

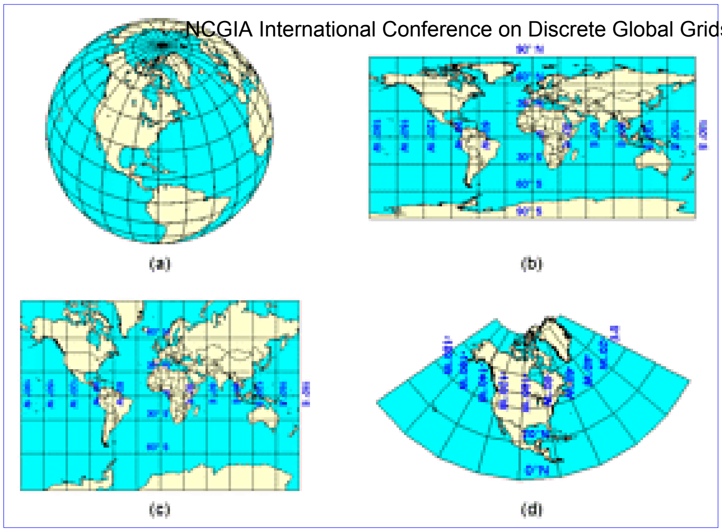


Figure 3: Examples of geodetic and projective coordinate systems. (a) Orthographic projection, used for perspective views of the Earth, Moon, and other planets, (b) Latitude/Longitude graticule, used to locate points on the Earth's surface via a grid of meridians and parallels, (c) Mercator projection, used for navigation or maps of equatorial regions, (d) Lambert Conformal Conic, used by USGS for topographic maps. (Images adapted from Dana, 1998. Reproduced with kind permission.)

We use the SEDRIS (Synthetic Environment Data Repository and Interchange Specification: Coordinate System API in order to perform conversions between various geographic coordinate systems. We have also converted part of this library to Java (the GeoTransform package so that we can perform these conversions on the fly inside a VRML97 scene. Using the GeoTransform package we have developed new VRML nodes to provide support for geographic data in VRML97 and contributed this work to the GeoVRML Working Group, an official working group of the Web3D Consortium. All of the above software are freely available under open source like licensing schemes.

There is a further problem to face when dealing with geographic coordinate systems: that of precision. Most graphics systems only support rendering using single-precision floating point numbers, and VRML97 can only store floating point numbers in single-precision (ISO/IEC, 1997). IEEE single-precision values are represented using 32-bits with a 23-bit mantissa (IEEE, 1985), thus providing around 6 digits of precision (22^3 = 8.39 x 10^6). Given that the equatorial diameter of the earth is 12,756,274 m under the WGS 84 ellipsoid (Synder 1987), then we will only be able to represent geocentric values down to the order of 10s of meters. In order to get round this problem, we define an absolute georeferenced location for each terrain dataset or feature, and an implicit local coordinate frame against which the geometry is referenced. Internally, the absolute location is stored as double-precision geocentric coordinates. Each vertex of a model is transformed into a double-precision geocentric coordinates and the single-precision difference of these two values is used to render the vertex.

5. Visualisation

We intend the Digital Earth to be browsable using commercial off-the-shelf software. In order to do this, we rely on the VRML97 format to describe the terrain data so that the user will be able to interact with the data using a standard VRML97 plug-in for their web browser. We also intend to enhance an existing terrain visualisation system, called TerraVision, in order to efficiently read these VRML representations. A general user can therefore use the VRML browser interface to easily gain access to the Digital Earth, while a more serious user could install TerraVision in order to use an application that has been specifically tuned to the data and which offers various virtual reality interfaces.

5.1. The TerraVision System

TerraVision is a real-time, distributed terrain visualization system that has been developed over several years at SRI International (see Figure 4). It was originally developed as part of the DARPA-funded Multi-dimensional Applications Gigabit Internet Consortium (MAGIC) and Battlefield Awareness and Data Dissemination (BADD) projects and has been specifically designed to browse massive terrain and other data distributed over a fast network (Leclerc and Lau, 1995, Reddy et al., 1999). It incorporates features such as an active map display, 2-D pan and zoom display, 3-D flythroughs, time of day and fog selection, incorporation of georeferenced models such as buildings or roads, and support for virtual reality devices such as head-mounted displays and the CAVE.

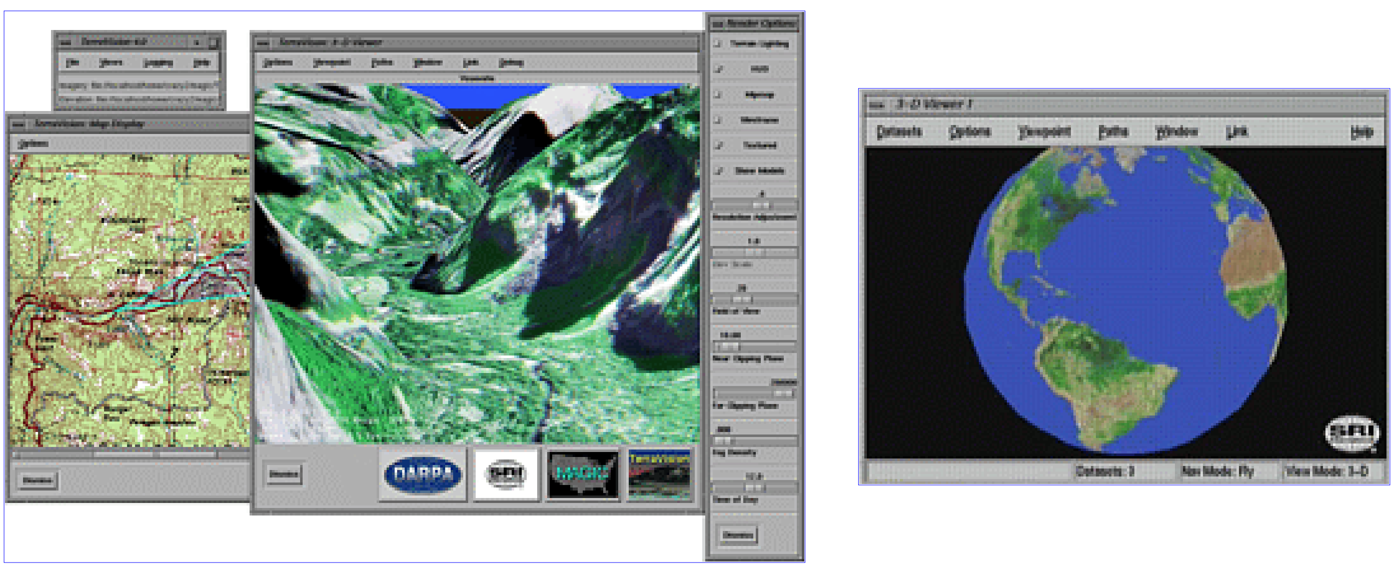


Figure 4. Left: screenshot of the TerraVision terrain visualization system showing the 3-D viewer and the co-registered Map viewer. Right: an MPEG movie showing a TerraVision fly-through from the globe to Menlo Park, CA.

5.2. A VRML Browser

An important goal of our work is to enable open solutions. One facet of this goal is the adoption of various open standards, such as VRML, to give a wide cross-section of users access to the content. By employing VRML as the file format to represent the multi-resolution structure of the Digital Earth, we allow users to interact with it using standard off-the-shelf VRML browser software. VRML browsers are produced by several companies and are available for a range of platforms. These are often provided for free as plug-ins for Internet browsers such as Netscape Communicator (NC) or Microsoft's Internet Explorer (IE). IE5 was shipped with a pre-installed VRML plug-in.

By default, a VRML browser will display a 3-D scene, perform any key-frame animations that are specified, and allow the user to interact with the scene by using a mouse. Certain objects can be defined as hyperlinks so that when the user clicks over them, an action is performed such as loading a new VRML scene or displaying an HTML page. It is possible to extend the base functionality of a scene by embedding Java code directly into objects to define their behavior, or to control the VRML browser from an external Java applet running in the Internet browser. These features enable us to encapsulate much of the Digital Earth functionality into a standard VRML application. For example, we will be able to navigate around a multi-resolution, 3-D representation of the globe; embed multiple terrain datasets as well as other features such as buildings, roads, and textual annotations; and click over features to display other multimedia objects. However, it is likely that certain capabilities will not be available in a standard VRML browser, or that they will be available at a lower performance level. For example, TerraVision will offer more sophisticated and optimised tile management techniques, perform stitching between tiles of different resolutions, implement terrain-specific navigation models, and so on.

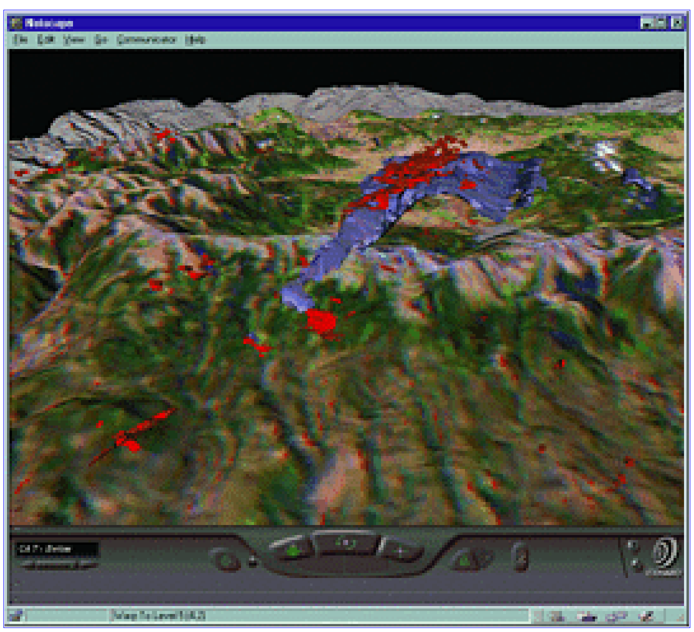


Figure 5. An example showing a VRML browser display a terrain model of the Colorado Rockies and showing the embedding of a georeferenced object representing a clear air turbulence isosurface. (Weather model provided by National Center for Atmospheric Research, Boulder, CO.)

6. Navigation

The issue of efficient navigation of our global structure is a crucial one. Without a means to interact with the data in a timely and appropriate manner, our work will be of limited practical value. There are a number of dimensions to the issue of navigating large, planetary models. We will discuss some of the most important issues here.

1. **Terrain Following:** The earth is round, not flat. As we navigate over the earth's surface we should therefore expect to follow a curved flight path. We therefore desire a navigation method that will maintain a particular height above the surface of the earth. In order to do this we need to know the gravitational up vector for a particular region of terrain so that we can calculate height above the terrain and also orient the user correctly. This up vector could be found by using the 3D normal to the plane which is tangent to the reference ellipsoid at the region in question.

2. **Altitude-based Velocity:** The velocity at which the user can navigate should be dependent upon their height above terrain. For example, when flying through a valley at a height of 100 m above the terrain, a velocity of 100 m/s could be considered relatively fast. However, when viewing the entire globe from space at an altitude of 20,000 km, zooming in at the same speed would be painfully slow. We therefore scale the velocity of the user's navigation in an attempt to achieve a constant pixel flow across the screen. We have found that a simple linear relationship, assuming height above the earth's ellipsoid, gives satisfactory results.

3. **Active Maps:** When flying low over an area of terrain, it is often difficult to maintain a context of your position in the world. We therefore employ overhead and map displays to provide this context. We can project the user's 3-D geocentric location onto the map display so that the user can easily ascertain their location in the world. Additionally, we can allow the user to click over the map and then move the viewpoint directly to that location.

4. **Multi-Modal Navigation:** Mouse gestures are often insufficient to provide good navigation in a complex environment. TerraVision will provide the ability to navigate using a number of inter-related modalities such as pen-based gestures and spoken language. This capability will be built using SRI's Open Agent Architecture and prior experience in building such an interface to a 2D map. For example, the user could say something like "Take me to Dundee". The system would then automatically descend to this city. Or the user

could ask "What's the name of this river?" while sketching it on the terrain, and so on.

7. Conclusions

We have described the Digital Earth concept: the dream of building a massive, distributed model of the earth where users can fly from outer space down to street-level scales and interact with large amounts of georeferenced multimedia data. More than just a pipe dream, we have shown how we may build a scalable framework to support this vision by introducing a geographically based DNS naming convention; we have presented a number of problematic issues and provided solutions to these, for example modeling massive, distributed multi-resolution terrain, supporting geographic coordinate systems, dealing with double-precision data, navigating around a large globe structure; and finally, we are developing solutions to allow users around the world to interact with the Digital Earth in an efficient and seamless manner.

References

Dana, P. (1998), "Geographer's Craft Project Web Page", Department of Geography, University of Texas at Austin. Available on-line at http://www.utexas.edu/depts/grg/gcraft/notes/mapproj/mapproj.html.

Falby, J. S, Zyda, M. J., Pratt, D. R. and Mackey, R. L. (1993). "NPSNET: Hierarchical Data Structures for Real-Time Three-Dimensional Visual Simulation". Computer and Graphics, 17(1):65-69. Pergamon Press, UK.

Gore, A. (1998). "The Digital Earth: Understanding Our Planet in the 21st Century". Speech delivered at the California Science Center (CSC), Los Angeles, CA, 31 January. Available on-line at http://www.ai.sri.com/digital-earth/gore.html

Hitchner, L. E. and McGreevy, M. W. (1993). "Methods for User-Based Reduction of Model Complexity for Virtual Planetary Exploration". In Proceedings of the SPIE: the International Society for Optical Engineering, vol 1913, pp. 622-36.

IEEE 754-1985 (1985). "IEEE Standard for Binary Floating-Point Arithmetic". IEEE Standards Publication. http://standards.ieee.org/.

ISO/IEC 14772-1:1997 (1997). "The Virtual Reality Modeling Language". December 1997. Available on-line at: http://www.web3d.org/Specifications.

Leclerc, Y. G. and Lau, S. Q. (1995). "TerraVision: A Terrain Visualization System". Technical Note 540, Artificial Intelligence Center, SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025. Available on-line at http://www.ai.sri.com/pubs/technotes/aic-tn-1994:540/document.ps.gz

OpenGL Architecture Review Board (1997). "OpenGL Reference Manual: the Official Reference Document to OpenGL, Version 1.1" (Second Edition). Editors: Kempf, R. and Frazier, C. Addison Wesley, MA. http://www.sgi.com/software/opengl/guides.html.

Reddy, M., Leclerc, Y. G., Iverson, L. and Bletter, N. (1999). "TerraVision II: Visualizing Massive Terrain Databases in VRML". IEEE Computer Graphics and Applications, Special Issue on VRML, 19(2): 30-38. Available on-line at http://www.ai.sri.com/~reddy/pubs/CGAVRML/

Snyder, J. P. (1987). "Map Projections - A Working Manual". U.S. Geological Survey

Pixel Loss and Duplication during Projection of Global Grids

Karen A. Mulcahy

Department of Geography
East Carolina University

Little attention has been paid to the consequences of transforming global grids by map projections. The transformation of a discrete, raster structure is very different from the transformation of vector data. Vector data employs the time tested point-to-point method, is very flexible in terms of transformation, and standard cartographic theory regarding map projection transformation applies. The transformation of less flexible, global raster data sets is more complex. Given current implementations of projection transformation, data changes at the pixel level are inevitable. To track the changes at the pixel level, two metrics *PL* and*PD,* are applied to the study of global equal area projections. Projections such as the Craster Parabolic, Eckert IV and VI, Goode Homolosine, Sinusoidal, Mollweide, and Hammer-Aitoff all maintain the property of true relative area. While, one might assume that all pixels would be retained in a transformation from the Earth surface to the plane but this is not the case. In addition, the primary goal when choosing a projection is to minimize distortion. In the choice of global equal area projections, the property of angular distortion is often minimized. The results at the pixel indicate that the opposite choice may be appropriate. Greater angular distortion provides a higher retention of original pixels without duplication.

Design Issues in the Go2 Grid System

Val Noronha

Digital Geographic Research Corporation
Mississauga, Ontario, Canada

The Go2 vision brings a system of geographic coordinates into public use for the first time. This is a big step forward for location science, and while it presents opportunities, it raises several challenges in terms of comprehension, usability, consequences of misuse, and even legal liability. For example, because the system is "overtiled," there is a many to one mapping of Go2 addresses to real world locations. Second, although the map projection provides for addresses to be specified in meaningful distance units, the multiplicity of projection centers results in grid discontinuity. Third, both the alphabetic and numeric components of an address contain potential for error, confusion and misunderstanding. This presentation examines these technical issues in depth, and discusses approaches we are pursuing in system design to minimize problems. It covers the choice of map projection, handling of overtiling, design and assignment of alphabetic codes, and error-proofing for mission critical application.

Discrete Global Grids Applicable to Spatially Balanced Sampling

Anthony Olsen[1], Denis White[1] and Don L. Stevens. Jr.[2]

[1]US EPA Western Ecology Division

[2]Dynamac Corporation

Developing efficient survey designs for natural resources, such as lakes, streams, forests, requires that the spatial distribution of the target population be explicitly considered. During the past 10 years, a joint effort among geographers and statisticians has resulted in an approach to survey design based on the concept of a discrete global grid. These new spatially-balanced survey designs have been successfully applied to over 100 environmental monitoring programs within the United States as well as in a number of other countries. First, we describe the theoretical requirements for spatially-restricted designs. Second, we illustrate how discrete global grids are applied to complete a hierarchical randomization that results in a spatially-balanced sample. The concepts are illustrated with applications to surveys of lakes in Wyoming, streams in Mid-Atlantic states, and terrestrial lands in western states. The survey designs are based on a new discrete global grid system. The system uses an icosahedral model of the sphere, a diamond hierarchical structure, and Snyder’s equal area projection.

Practical Properties of the Spiral Points

Robert Raskin

Jet Propulsion Lab, 300-320

A uniform hexagonal grid over the sphere can be closely approximated by a set known as the spiral points (Rakhmanov et al., 1994; Saff and Kuijlaars, 1997). These points are generated along a spiral from pole to pole, with successive points separated in latitude by a fixed increment of sin(latitude) and in longitude by a fixed arc length along a parallel. Rakhmanov et al. found that this family of grids possesses excellent asymptotic properties with respect to uniformity, although the reasons for its fine performance and close approximation to a hexagonal grid are not known. A significant advantage of this set is that it can be generated for any number of grid points n. Some practical properties of this grid are presented, including possible data structures, methods of finding adjacent points, and uniformity of point distribution, based upon empirical tests performed for a wide range of values of n.

Rakhmanov, E.A., E.B. Saff, and Y. M. Zhou, Minimal discrete energy on the sphere, Mathematical Research Letters, 1, 647, 1994. Saff, E. B. and A.B.J. Kuijlaars, Distributing many points on the sphere, Mathematical Intelligencer, 19, 5, 1997.

The Relationship Between Wavelets and Quadtrees for Global Grids

Robert Raskin

Jet Propulsion Lab, 300-320

Quadtrees and wavelets have in common the ability to represent and store data in hierarchical form. The connection between these two representations in the context of global grids is explored in this presentation. Wavelets are ordinarily thought of as a spectral representation and quadtrees as a data structure. However, wavelets and quadtrees can be made to have identical reconstruction algorithms. For this case, the Haar wavelet is used and the quadtree stores only differences from the parent value at any given level; the reconstruction involves summing the values of each node from root to terminating leaf. For global grid schemes that use triangular decomposition, this would enable applications to control the resolution at which retrieval and analysis operations are performed. This feature may be of advantage in massively large, high resolution, global datasets. The isomorphism between wavelets and quadtrees suggests that a quadtree can be further generalized in the same way that the Haar wavelet is generalized to other "mother" wavelets. These other wavelets may represent a closer fit to the variations inherent in the data, as an appropriate wavelet choice can capture most of the spatial variations at levels closer to the root of the tree. Also discussed will be some extensions to the concept of wavelets that have been proposed for the sphere and how these might be represented.

A Preliminary Comparison of Proposed Topologies for Geodesic Discrete Global Grid Systems

Kevin Sahr

Department of Computer & Information Science
University of Oregon

Multi-resolution regular partitions of the platonic solids have been proposed as an alternative to traditional floating point coordinates for encoding geospatial location on digital computers. Proposed topologies have included grids based on triangles, hexagons, and diamonds. In this paper we will give a mathematical model for such systems and then describe the implementation of that model to create a software testbed for evaluating the performance of these systems. Preliminary performance comparisons of the three major topologies as location-based spatial data structures will be discussed.

Finding a (Buffon's) needle in a hierarchical triangular haystack

Ashton Shortridge
National Center for Geographic Information and Analysis,
Department of Geography, University of California, Santa Barbara
ashton@geog.ucsb.edu

Abstract

This paper develops a method for determining the probability for a line to intersect an equilateral triangular tile such as the ones comprising a global data structures like the quaternary triangular mesh (QTM). Knowledge of intersection probabilities may be valuable for applications involving global QTMs, including estimating complexity and time required to process a distance operation and identifying optimal resolutions in the hierarchy for specific operations. The solution for this problem is presented, its application to the QTM is examined, and its utility is discussed.

1. Introduction

Imperialists, death star commanders, and spatial data modelers alike would agree that partitioning the globe is a useful exercise. For spatial data modelers in particular, geographic tiling of a global database may be desirable if not downright necessary. Storage retrieval, and analysis of information from high resolution, globally extensive datasets are facilitated by partitioning them into more manageably sized tiles (Goodchild, 1989).

The intersection of geographical features of interest with tile boundaries may be a positive or negative experience for the GIScientist. On the positive side, the tesselation may act as a spatial indexing system, and the intersection allows a linear or areal feature to be encoded appropriately. For example, several adjacent square cells in a (hypothetical) high resolution remotely sensed image are classified as water. The arrangement of these cells may reasonably approximate the size, shape, and position of the lake. If the spatial resolution of the cells were coarser, the lake would not be characterized as well. On the negative, the tesselation may subdivide the dataset into different large files, in which case the analyst must download and process several times the volume of data contained in a single tile. For example, the global digital elevation dataset GTOPO30 is divided into 33 tiles. To obtain the data for, say, Nebraska, two 11 megabyte compressed tar tiles must be downloaded, uncompressed, imported into a GIS, clipped, and finally merged, although the amount of data within Nebraska is a small fraction of that contained in a single tile. In both positive and negative cases, knowledge of the intersection probability of tile boundaries with geographic features is useful.

Buffon's Needle-type solutions from the field of geometric probability provide a framework for deriving probabilities for a number of common tile-feature intersections, including line-line, line-square, and line-rectangle. This paper reports on the extension of Buffon's Needle to the equilateral triangle case. More formally, I consider the earth's surface modeled as a quaternary triangular mesh at a particular level of decomposition. This surface consists of equal-sized equilateral triangular tiles completely filling the space, so that all locations fall in one and only one tile. A needle (a straight line segment of length *n*) is dropped randomly onto the surface. What is the probability that the needle intersects a tile boundary? Knowledge of this probability has implications for the analysis of data stored in global data structures like the quaternary triangular mesh (QTM), which partitions the world into successively finer levels of nested equilateral triangles.

In Section 2 the design of the quaternary triangular mesh is outlined, and the utility of the probability of intersection for segments upon this mesh is identified. Section 3 describes the classical Buffon's Needle problem and a general approach for the solution of similar problems in geometrical probability. In Section 4 the Needle problem for an equilateral triangle is framed and the solution is derived. To my knowledge this solution has not appeared in the geographic literature, nor have I found reference to the problem in a (limited) review of the geometric probability literature. Section 5 applies this solution to the quaternary mesh structure. Three general questions are identified which can be answered using the framework. This discussion is illustrated with global dataset examples. The paper concludes in Section 6 with a look at extensions of geometric probability for global tessellations.

2. A hierarchical triangular data model for global GIS

There are a number of ways in which the surface of the earth may be recursively decomposed into a hierarchy of tiles for the purposes of storing, retrieving, and analyzing spatial data. One of these spatial data models, termed the quaternary triangular mesh (QTM), was proposed by Dutton (1984, 1989) and further developed by Goodchild et al. (1991) and Goodchild and Yang (1992). Construction of the quaternary triangular mesh begins by projecting the earth on to an octahedron with the poles at the top and bottom points. Each facet of the octahedron is an equilateral triangle; level 0 in the hierarchy is this set of eight facets. Each triangle can be subdivided into four equilateral triangles by connecting the center points of each side; these triangles comprise level 2. Any triangle within this system can be recursively decomposed to increasingly higher (finer) level levels in the hierarchy.

The quaternary triangular mesh has several advantages as a method of structuring global spatial information. Triangles at higher levels nest congruently within coarser, lower level triangles. Data structure density may vary with data density, since the number of levels can be extended in data-rich locations. The level of a triangle is implicit in its index number, making it straightforward to determine the precision of the data. Efficient spatial operators have been developed for the QTM, making it a useful structure for very large data sets. Dutton (1989) is an excellent source for more information on the QTM and its application for building triangular hierarchical data models.

Intersection of spatial elements with the tiles of the data structure is necessary for construction of geographical databases and for much spatial analysis performed upon them. Usually intersection is calculated deterministically, since feature locations are typically known prior to performing the operation. However, there is utility in knowing the probability of intersection without reference to feature-specific locations. Knowledge of intersection probabilities may be valuable for applications involving global QTMs, including identifying optimal level (tile size) for desired levels of accuracy in encoding objects, and estimating complexity and time required to process a distance operation. Buffon's Needle-type solutions from the field of geometric probability provide the framework for deriving such probabilities. The next section discusses Buffon’s Needle and outlines the basic framework.

3. Geometric probability and Buffon’s Needle

Geometric probability is a branch of mathematics that is concerned with the probabilities associated with geometric configurations of objects. Among the most well known of these applications is the Buffon’s Needle problem. The classic Buffon’s Needle problem and its solution are as follows. Parallel, equidistant lines are spaced *s* units apart in the plane. A straight needle of length *n*, where $n < s$, is dropped onto this plane at random. By random, it is meant that the location of the center of the needle and its orientation are uniformly and independently distributed. The probability *P*(int) that the needle will intersect one of the lines is:

$$P(\text{int}) = \frac{2n}{p\,s} \qquad (1)$$

Many clever proofs for this solution and for extensions to the classic problem have been developed over the years and may be found in several sources, including Klain & Rota (1997), Gani (1980), and Solomon (1978). Here I wish only to sketch out an approach to the solution that will clarify the method used on the triangle problem. The converse probability – that the needle does not intersect a line – is more straightforward to calculate. Once this is known, the probability of intersection is simply one minus the probability of no interception. I begin by noting that center of the needle must fall between two adjacent lines, and so its center position can range from 0-*s*. The orientation of the needle can range from 0-π, relative to the vertical. Therefore the total size of the solution space is the area π*s*. For the needle not to intersect a line, only a subset of the π*s* domain is feasible. The ratio of this subset to the total indicates the probability that the needle falls entirely between the lines. The remaining problem is to identify this subset, which involves a healthy dose of trigonometry and integral calculus.

4. Buffon’s Needle solution for the equilateral triangle

Consider a surface tiled by equilateral triangles with sides of length *s* and heights *h*. This tesselation is equivalent to a region of the earth tiled to a constant QTM level. Toss a needle of length *n*, where $n < h$, randomly onto this surface. What is the probability that the needle will intersect a triangle boundary? As with the classic problem outlined in the previous section, the solution involves first identifying the probability that the needle does not intersect the triangle. The problem is then to identify the ratio of the feasible (no intersection) solution space to the total solution space.

I begin by investigating the converse probability, that the needle falls completely within a single triangle. It is clear that the centerpoint of the needle must fall within a particular triangle, and that boundary intersection is entirely determined by the location of this centerpoint, the angle α of the needle relative to the triangle, and the length of *n* relative to *s*. A few definitions are now required. Orient the equilateral triangle as in Figure 1 and label the sides S1, S2, and S3 as shown. The orientation of the needle, angle α, is relative to S1.

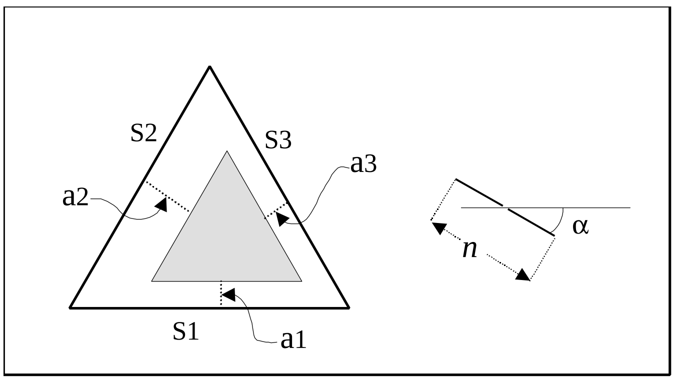


Figure 1. *An equilateral triangle with sides S1, S2, and S3. A needle (shown on the right-hand side) is dropped onto the triangle. Shaded region is that part of the triangle in which the centerpoint must fall for the needle not to intersect a triangle edge.*

Figure 1 illustrates a particular case for a particular *n* and α. For the needle to fall completely within the triangle, its centerpoint must fall in the shaded region of the figure; that is, its center must be farther than:

$$\left.\begin{array}{ll} a1 = (n/2) * \sin\alpha & \text{from S1} \\ a2 = (n/2) * \sin((\pi/3) - \alpha) & \text{from S2} \\ a3 = (n/2) * \sin((2\pi/3) - \alpha) & \text{from S3} \end{array}\right\} \text{for } 0 \le \alpha < \pi/3 \qquad (2)$$

For needle angles ranging from $\pi/3 \le \alpha < 2\pi/3$, $a2$ becomes negative, so over this range $a1$, -1*$a2$, $a3$ hold. For angles ranging from $2\pi/3 \le \alpha < \pi$, both $a2$ and $a3$ are negative, so $a1$, -1*$a2$, -1*$a3$ hold.

The shaded region shown in Figure 1 is an equilateral triangle for any $n < h$, where *h* is of course $s * \sqrt{3}/2 \approx 0.866s$. Figure 2 indicates how the length of the side of this shaded triangle (denoted *t*) may be identified. From the figure, it can be determined that:

$$t = s - (d2 + d3), \text{ where}$$
$$d2 = e + f2,\ d3 = e + f3, \text{ and} \qquad (3)$$
$$e = \frac{a1}{\tan \pi/3},\ f2 = \frac{a2}{\sin \pi/3},\ f3 = \frac{a3}{\sin \pi/3}$$

for any particular α.

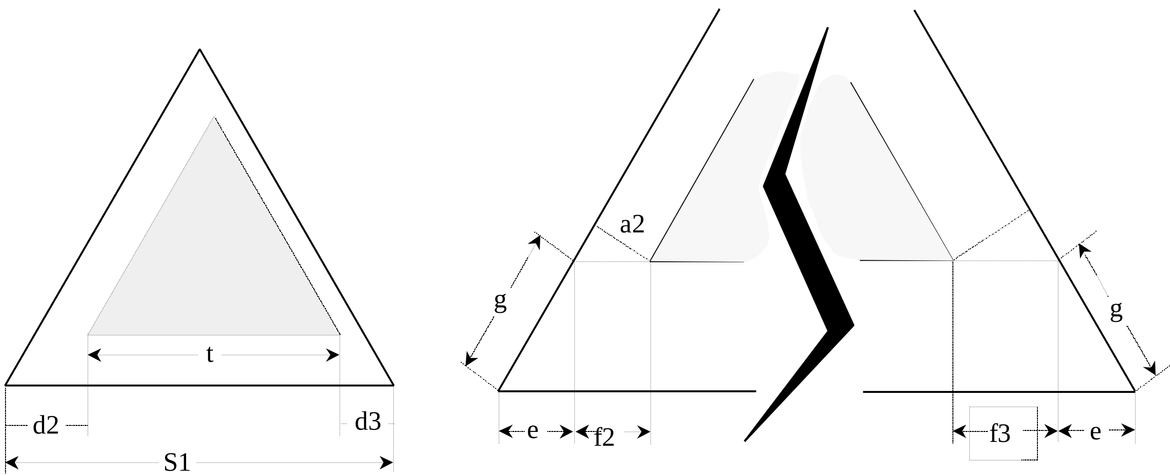


***Figure 2**. Left-hand side shows relationship between the shaded triangle side t and S1. Right-hand side shows important lengths for calculating the length of t. See text.*

The area of the space in which the centerpoint must fall for the entire needle to lie within a single triangle is then $t(\alpha)^2\sqrt{3}/4$, again for any particular α. Identifying the solution volume requires integrating across the possible values of α, which range from 0-π. Because the values of a2 and a3 differ depending on α, however, three separate integrations must be performed:

$$V1 = \int_0^{\pi/3} \left(\sqrt{3}/4\right) * t(\alpha)^2 \, d\alpha$$
$$V2 = \int_{\pi/3}^{2\pi/3} \left(\sqrt{3}/4\right) * t(\alpha)^2 \, d\alpha \qquad (4)$$
$$V3 = \int_{2\pi/3}^{\pi} \left(\sqrt{3}/4\right) * t(\alpha)^2 \, d\alpha$$

The author can attest that the solutions for these integrals are lengthy and tedious; therefore they will not be shown here. As may be intuitive, *V1=V2=V3*. The range of possible needle center locations and orientations is contained in the volume $\left(\sqrt{3}/4\right) * \pi s^2$. Dividing the sum of the integrals by total volume gives the probability that the needle will fall completely within the triangle:

$$P(\text{noint}) = \frac{V1 + V2 + V3}{\left(\sqrt{3}/4\right) * p\, s^2} \qquad (5)$$

Solving for *V1-V3*, this equation expands to:

$$P(\text{noint}) = \frac{3 * \left(\sqrt{3}/4\right)\left(\dfrac{3p\, s^2 + 2p\, n^2 + 3n^2 \tan p/3 - 12sn \tan p/3}{9}\right)}{\left(\sqrt{3}/4\right) * p\, s^2} \qquad (6)$$

from which we obtain:

$$P(\text{noint}) = 1 + \frac{2p\, n^2 + 3n^2 \tan p/3 - 12sn \tan p/3}{3p\, s^2} \qquad (7)$$

or, rewritten:

$$P(\text{noint}) = 1 - \frac{12sn \tan p/3 - 2p\, n^2 - 3n^2 \tan p/3}{3p\, s^2} \qquad (8)$$

and therefore the probability that the needle intersects a triangle boundary is:

$$P(\text{int}) = \frac{12sn \tan p/3 - 2p\, n^2 - 3n^2 \tan p/3}{3p\, s^2} \quad \text{for } 0 \le n < s\sqrt{3}/2 \qquad (9)$$

How does this probability change as *n* varies with respect to *s*? Figure 3 is a plot of the function. When the n is 26.57% the length of s, the probability that it intersects a triangle boundary is 0.5. When $n = s\sqrt{3}/2$, the probability of intersection is 0.996. Although the formula is not valid for the entire range of possible values of *n*, probabilities for larger values are very close to one.

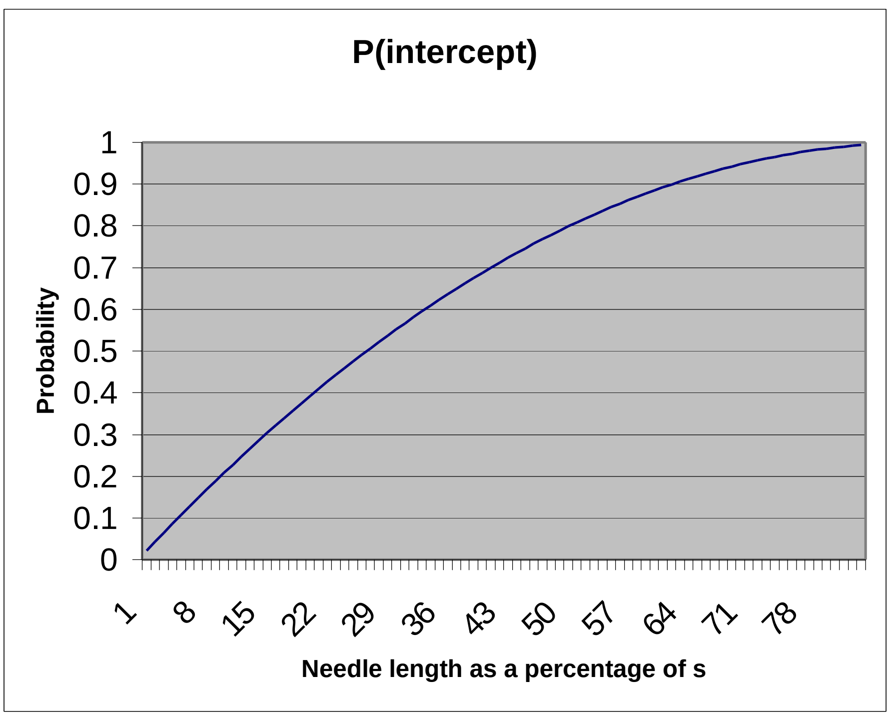


Figure 3. *Probability of interception for a needle on an equilateral triangle.*

5 Applications for the triangular hierarchical data model

The probability model for the “needle in a triangle” appears well suited for application to a quaternary triangular mesh. As discussed in Section 2, the QTM consists of nested levels of triangles. Higher levels correspond to smaller triangle sizes and greater subdivision of the earth’s surface. Table 1, taken from Goodchild & Yang (1992), indicates the lengths of the triangle sides for different levels of decomposition of a spheroidal model of the earth with radius 6378 km along the equator and 6356 km along the meridian. The values in the table indicate that triangle side along the meridian is approximately 99.7% as long as the side along the equator due to the differing lengths of the axes.

Level	Degree	Along Equator (lat)	Along meridian (lon)
0	90º	10018.538 km	9983.8912 km
2	22º 30’	2504.6345 km	2495.9953 km
4	5º 37’ 30”	626.1586 km	623.9988 km
6	1º 24’ 22.5”	156.5397 km	155.9997 km
8	21’ 5.625”	39.1349 km	38.9999 km
10	5’ 16.40625”	9.7837 km	9.7500 km
12	1’ 19.11015”	2.4459 km.	2.4376 km.
14	19.77539”	611 m	609 m
16	4.9438”	153.9 m	152.3 m
18	1.23596”	38.2 m	38.1 m
20	0.3089904”	9.55 m	9.525 m

Table 1. Length of triangle edges at increasing levels of subdivision (from Goodchild & Yang, 1992).

An additional departure from the equilateral model is that triangle side length for a given level is dependent upon latitude. The sides of polar level *n* triangle are approximately 1.2533 times longer than those of their equatorial brethren. For any level of decomposition, then, there is not complete agreement with the equilateral model described at the beginning of Section 4. Differences appear minor enough for this model to produce useful results.

Three related questions with practical implications for global database design can be answered using Equation 9:

- Given a particular precision (level), what is the probability that points separated by *n* will be stored in separate triangles?
- What level of precision is required to ensure that a certain percentage of points *n* distance apart are stored in separate triangles?

- Given a particular precision (level), how close can two points be before the probability of encoding them in separate levels drops below *p*?

In the first question the level of subdivision is known, as is the needle length, and the probability of intersection for the length is of interest. For example, consider a continental-scale scattered point dataset with elevation observations at each point. This dataset is to be used to help develop a global digital elevation model on a hierarchical triangular mesh. The desired precision is a level-12 decomposition with triangular tiles about 2.4 km on a side. When exactly one point falls in a tile, the point elevation is assigned to the tile. Some form of averaging must be used if more than one point falls in a tile. It is known that many points are approximately 600 meters from their nearest neighbors. It is therefore of interest to determine the probability that point observations 600 meters apart will fall in the same tile. The solution is obtained directly from Equation 8. The probability that observations spaced 600 meters apart fall in the same tile is 0.531. One can expect to run the more complicated assignment algorithm over half the time when points are this far from their nearest neighbors.

The second question is of particular importance for database design. If the distance between points of interest is approximately known and an acceptable percentage of points assigned to the same tile can be agreed upon, then it is straightforward to identify the level in the hierarchy to which the database must extend. Consider a hypothetical example: a global database of airport locations. The developer is not especially interested in differentiating between airports in the same city, but in the distribution of airports serving separate populations. It is important that at least 90% of airports more than 50 kilometers apart be stored in different tiles. What level of decomposition is required?

The solution for this problem may be obtained by noting that Equation 9, written in terms of *s*, is the quadratic:

$$3p\,P(\mathrm{int})s^2 - 12n * \tan(p/3)s + \left(2p\,n^2 + 3 * \tan(p/3)n^2\right) = 0 \qquad (10)$$

Plugging in *P* and *n*, which are known, we can solve for *s* using the quadratic formula. The maximum triangle side that satisfies the problem is 80.645 km long. A level 7 subdivision of the earth, with triangle side length of approximately 78.27 km, would be required.

In the third case, the level of the hierarchy is known, as well as the desired probability of intersection, and what is of interest is the length of the needle. Consider the following example: a digitized coastline is being stored with a precision of level 14 (triangles about 610 meters on a side). The coast is represented by straight line segments of varying length, the placement and orientation of which are presumed random. It is also assumed that the vertices are significant locations along the line and it is desirable to store many of them in the QTM database. If two points are close together and fall in the same level 14 tile, possibly significant variation will be lost. It is therefore of interest to determine the segment length beyond which the probability of intersection drops below 80 percent.

As in case two, the solution is obtained by writing Equation 9 as a quadratic in terms of *n*:

$$\left(-2p - 3 * \tan\left(p/3\right)\right)n^2 + 12s * \tan\left(p/3\right)n - 3p\ P(\text{int})s^2 = 0 \qquad (11)$$

Inserting the known values for *P* and *s*, a valid solution for *n* can be obtained using the quadratic formula. Here, the segment length is 306.14 meters. Vertices closer than this distance have a probability of less than 0.8 of falling in separate level 14 triangles.

6. Conclusion

This paper has reported on the utility of geometric probability for global hierarchical data models based upon the quaternary triangular mesh. The first two sections of the paper briefly reviewed the QTM and its applications for global database organization. The third section described the classic Buffon's Needle problem, and the fourth section developed an extension of the Buffon's Needle approach to the equilateral triangle. The fifth section applied this extension to the hierarchical QTM. Three general questions were identified that could be answered using the equations derived in section 4, and illustrations of each of the questions were provided.

I see several directions for future work in this area. One extension (no pun intended) to the classical Buffon's Needle problem is the long needle case. Given a needle longer than a triangle side, the question becomes, what is the expected number (and variance) of tiles to be intersected? Second, this paper has focused solely on probabilities for line segments to intersect tile boundaries. Probability of tile intersections for areas would be a useful extension. Simple areas like rectangles and circles probably have analytical solutions. For more complex polygons, simulation appears to be a useful alternative. Third, I have not addressed the important issue of dataset scalability. Point addresses can be stored at arbitrarily fine levels in a QTM, and this precision can vary from point to point. Applications of geometric probability that go beyond a particular level of precision would be especially interesting and useful. Finally, it would be useful to identify intersection probabilities for alternative tile shapes like hexagons.

In the spectrum of activities that comprise the development and use of global GIS, geometric probability appears to have greatest application for data modeling decisions. If critical application specifications are known (e.g. mean trip distance for a proposed routing system, or the distribution of between-vertex distance of lines in a land information system) then optimal hierarchy levels can be established. Alternatively, the probability that features intersect tile boundaries at particular levels of the hierarchy can also be determined. Perhaps the greatest utility of the approach developed here is that it quantifies the relationship between the level of the hierarchical data structure and the real world features the structure attempts to represent.

References

Dutton, G., 1989, Planetary modeling via hierarchical tesselation. Proceedings Auto Carto 9, E. Anderson (Ed.), (Baltimore: ACSM), 462-471.

Dutton, G., 1984, Geodesic modeling of planetary relief. *Cartographica* **21**(2&3, Monograph 32-33), 188-207.

Gani, J., 1980, The problem of Buffon's needle. *Mathematical Spectrum*, **13**(1), 14-18.

Goodchild, M. F. and Yang, S., 1992, A hierarchical spatial data structure for global geographic information systems. *CVGIP: Graphical Models and Image Processing* **54**(1), 31-44.

Goodchild, M. F., Yang, S., and Dutton, G., 1991, Spatial data representation and basic operations for a triangular hierarchical data structure. *NCGIA Technical Report 91-8* (Santa Barbara: National Center for Geographic Analysis and Synthesis).

Goodchild, M. F., 1989, Tiling large geographical databases. In Design and Implementation of Large Spatial Databases, A. Buchmann, O. Günther, T. R. Smith, Y.-F. Wang (Eds.), (Berlinn: Springer-Verlag), 137-146.

Klain, D. A. and Rota, G.-C., 1997, *Introduction to Geometric Probability* (Cambridge: Cambridge University Press)

Solomon, H., 1978, *Geometric Probability*. CBMS-NSF Regional Conference Series in Applied Mathematics 28 (Philadelphia: Society for Industrial and Applied Mathematics).

Acknowledgements

Michael Goodchild and Kevin Curtin contributed ideas, questioned my assumptions, and pointed out several blunders. Any remaining errors are solely my responsibility. This work was financially supported by the National Center for Geographic Information and Analysis and the National Imagery and Mapping Agency (grant #NMA 202-97-1-1021).

Developing an Equal Area Global Grid by Small Circle Subdivision

Lian Song, A. Jon Kimerling and Kevin Sahr

Department of Geosciences
Oregon State University

Modern environmental monitoring and modeling requires partitioning the earth's surface into a global grid optimized for survey sampling and unbiased, spatially complete data collection of relevant environmental phenomena. Prime requirements are that cells comprising the grid be equal in area, regular in shape, and highly compact on the earth's ellipsoidal surface. No existing global grid fully meets these criteria.

We propose a new equal area global partitioning method based upon small circle edges on the earth's surface, which we call the "small circle subdivision method". A detailed description of this method is presented, including its mathematical derivation and geometrical comparison with alternative methods. The small circle method appears to be the best developed to date to satisfy the essential criteria for a global grid.

Interoperable Coordinate Transformation and Identification of Coordinate Systems

Daniel Specht
Engineer Research and Development Center

March 28, 2000

Interoperable Coordinate Transformation and Identification of Coordinate Systems

ABSTRACT

The OpenGIS Consortium (OGC), an industry group, has developed COM, CORBA and Java interface specifications allowing vendors to develop mutually interoperable geospatial software. OGC established the Coordinate Transformation Working Group (CT-WG) in April 1998. The working group developed a Unified Modeling Language object model of coordinate systems (CS) and coordinate transformations (CT) in coordination with ISO/TC211. In June 1999 OGC released this object model as part of a request for proposal. In February 2000 OGC accepted interface specifications (unpublished as of February 2000) drafted in response to this proposal.

This paper describes the object model and those specifications. The object model describes a CS as a collection of axes and at least one datum. A recent draft of the object model is available at http://www.opengis.org/techno/request.htm under "request 9."

The specification provides a common way of identifying CS and of accessing CT services that support accuracy calculation. When implemented, these specifications will ease data import; users of compliant applications will import data unaware of its coordinate system. If the application cannot import data in a given coordinate system, a compliant server will transform the coordinates to the native coordinate system.

INTRODUCTION

The OpenGIS Consortium (OGC) is a non-profit organization that promotes interoperable geoprocessing. The goal is that geospatial applications access data and services using standard interfaces. Clients using standard interfaces need not know much about the server. OGC defines OpenGIS as "transparent access to heterogeneous geodata and geoprocessing resources in a networked environment. The goal of the OpenGIS Project is to provide a comprehensive suite of open interface specifications that enable developers to write interoperating components that provide these capabilities." (http://www.opengis.org).

Representatives of software vendors, government agencies and academic institutions that belong to OGC write requirements for interface specifications. These requirements are published as a Request for Proposal (RFP). Member organizations then submit proposals for these specifications. After OGC accepts a proposal as an Implementation Specification OGC submits it to the International Standards Organization Technical Committee 211, ISO/TC211 (http://www.statkart.no/isotc211/) for acceptance as an ISO standard.

Interface standards are widely used outside of the geospatial community. For example in the Microsoft environment a user can cut out a piece of a spreadsheet and drop it into any conformant application (e.g., MS Word). The application will have access to spreadsheet services and data stored on the spreadsheet. To the user it appears as if the spreadsheet is running on the application. What we cannot do yet is to tell our spreadsheet that column H is Transverse Mercator coordinates and please change them to Lambert coordinates using a CT service on the network.

However OGC has developed several sets of geospatial interfaces. For example users with two applications compliant with the Simple Features specification for COM could (in theory) cut and paste a geospatial feature from one application to the other running on the same network.

THE ABSTRACT MODEL OF COORDINATE TRANSFORMATION

The US Army struggled with poor CT software for years. Interoperable CT software would help solve this problem. With this in mind I chartered a CT Working Group (CT-WG) at OGC. The group developed the Abstract Model, an object model of CT and CS using Unified Modeling Language (UML), a notational language used for object oriented analysis and design. UML is widely used and has been standardized by the Object Modeling Group (http://www.omg.org). Figure 1 shows what a UML object class looks like.

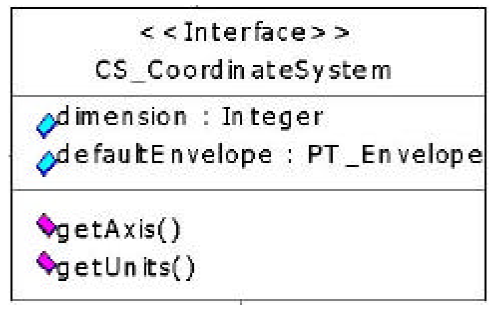


FIGURE 1. The class has three parts. The name of the class is in the top part, the middle part lists attributes (or properties) of the class and the lower part lists methods, i.e. implementable functions. In this case the methods get metadata, the axis and units.

ACHIEVING CONSENSUS

The CT-WG struggled to reach a consensus on what the model should look like while coordinating with our counterparts at ISO/TC211 to maintain conformance between the ISO/TC211 model and our own. This was not always easy.

Consider a point stored as latitude, longitude and elevation above sea level. Latitude and longitude are referenced to a geodetic datum while elevation is referenced to a vertical datum.

ISO/TC211 calls this a compound coordinate reference system, composed of two coordinate reference systems, each with its own datum, shown in Figure 2. (The diamonds on a stick indicate strong aggregation; the datum is part of the Coordinate Reference System which is part of the Compound Coordinate Reference System.)

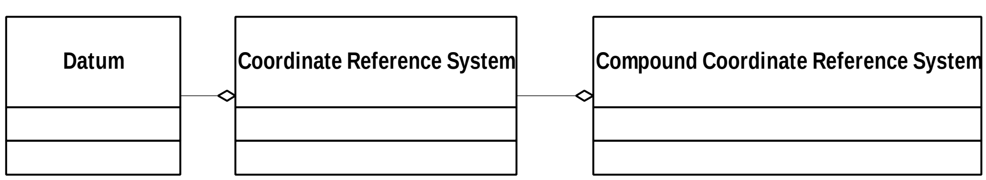


FIGURE 2. An ISO view

Some OGC members felt this was a single coordinate reference system with two datums and that the model should not include a compound coordinate reference system (Figure 3). There were other proposals as well.

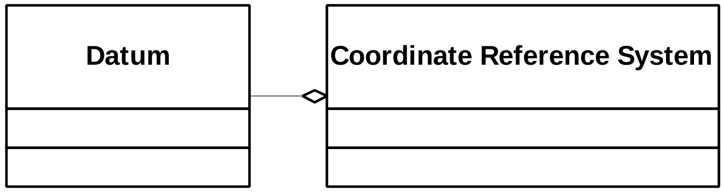


FIGURE 3: An early OGC view

The CT-WG could not come to a consensus and created a UML model that left the issue open. This UML model is part of the OpenGIS Abstract Specification, http://www.opengis.org/public/abstract/99-102r1.pdf which was included in the RFP.

OGC asked members to propose an Implementation Specification that solves this problem. The proposed solution is shown in figures 4 and 5. The revised UML model will be published shortly as part of the implementation specification.

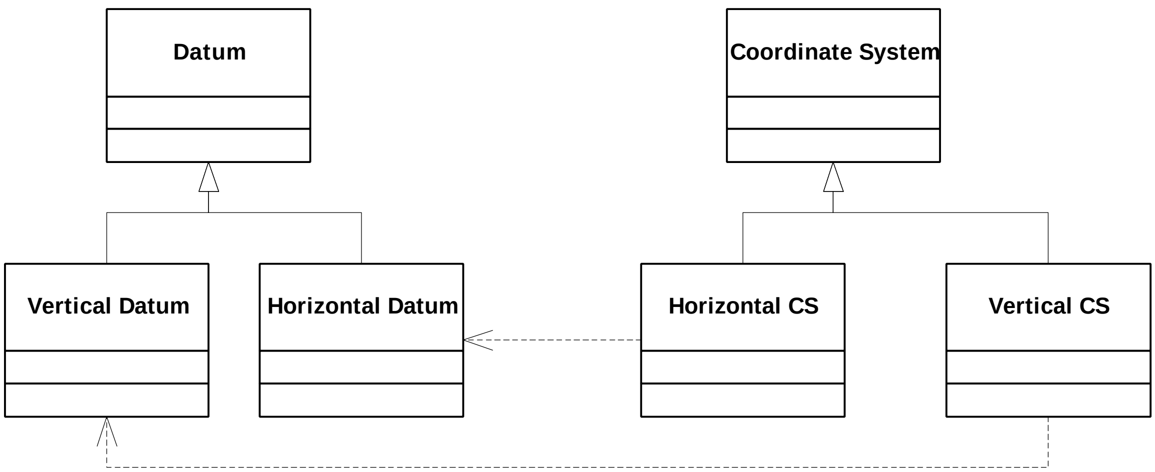


FIGURE 4. A representation of the final model. The object at the pointy end of the triangle is the superclass and the object at the other end is the subclass. The dotted line with the arrows indicates dependency; if the datum goes away so does the CS. These relationships are not explicit in the actual UML model, part of which is shown in Figure 5.

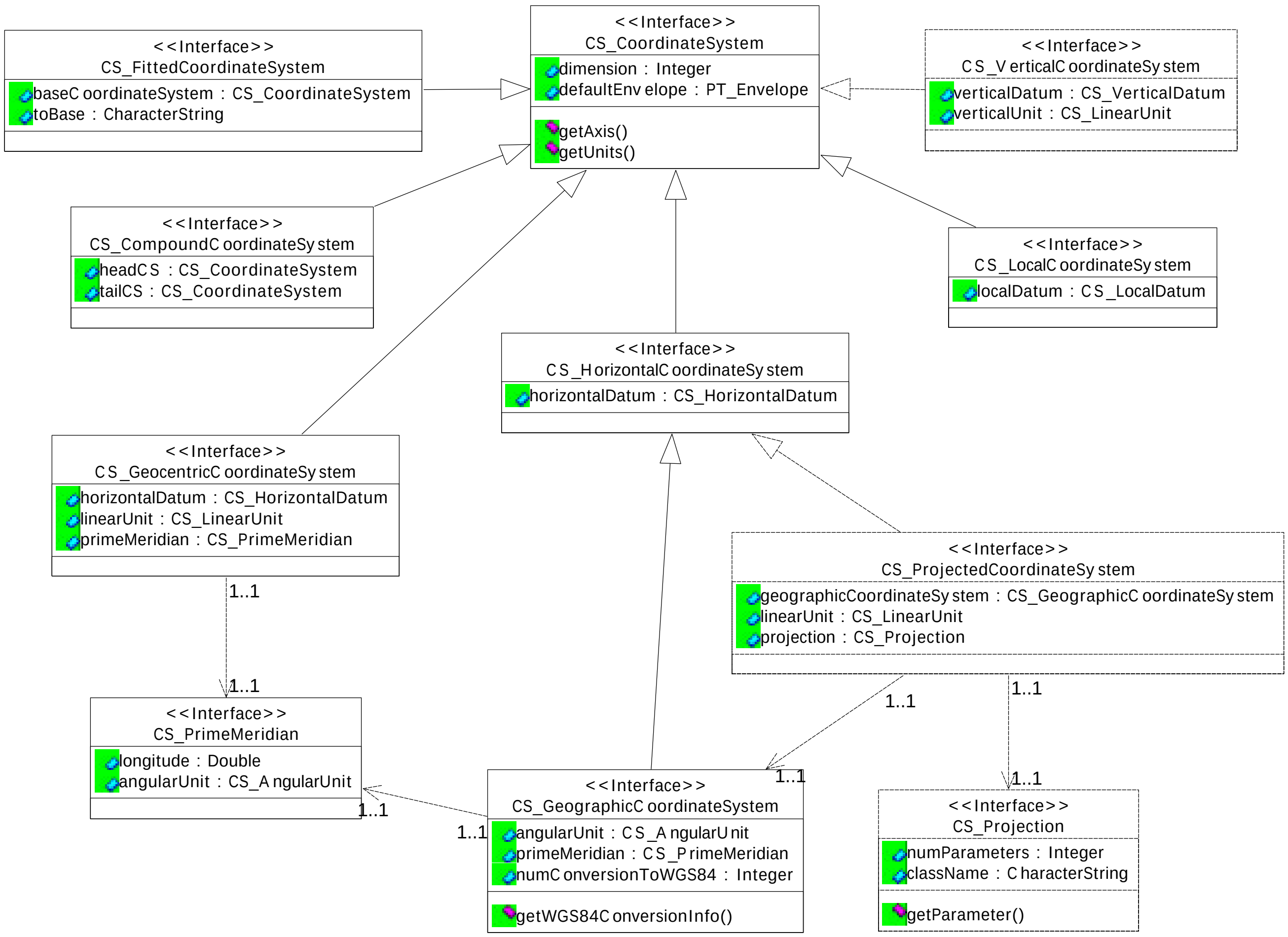


FIGURE 5. Here there is no association between the datum superclass and the CS superclass. Instead there are relationships between subclasses, e.g., a Vertical CS and a Vertical Datum. This model uses the term Coordinate System rather than Coordinate Reference System

IMPLEMENTATION MODEL

The object model for CS includes a number of CS classes and a number of datum classes.

A **Compound CS** includes a horizontal and a vertical CS and references two datums, a horizontal (geodetic) and a vertical datum. However where the Compound CS is 3-D ellipsoidal both the horizontal and vertical components may reference the same datum.

A **Horizontal CS** is a 2-D CS that may be **Projected** (a cartographic projection) or **Geographic** (latitude and longitude). Horizontal CS use horizontal (geodetic) datums.

A **Vertical CS** is a vertical 1-D CS (e.g., vertical with respect to a plumb line or to an ellipsoid). This references a vertical datum such as a sea level datum or a horizontal (geodetic) datum.

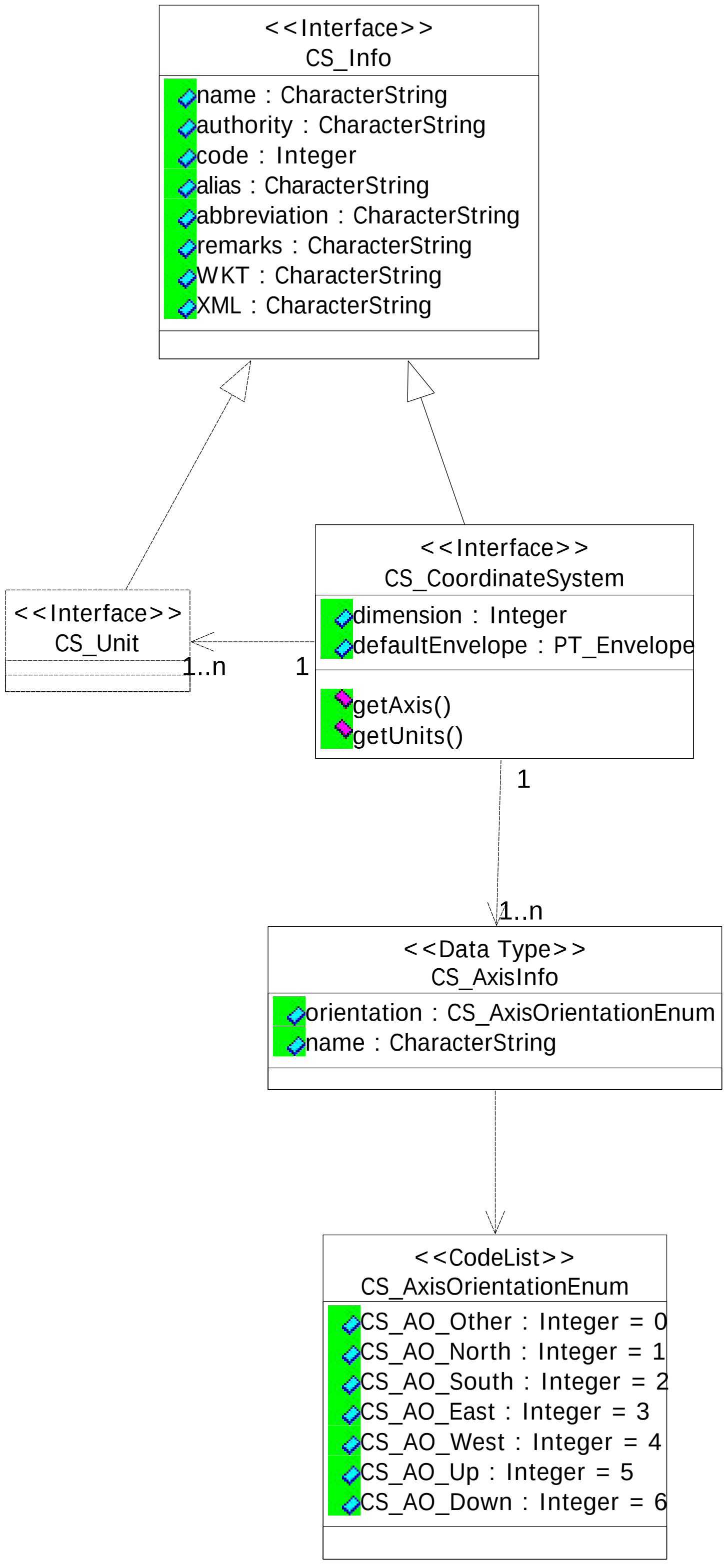


FIGURE 6. This specification supports users who create a coordinate system in any number of dimensions.

A **Local CS** references a local datum. A Local CS cannot be transformed to another CS. Once the Local CS is georeferenced it is no longer a Local CS. Note that a local datum is not necessarily 2-D Cartesian.

Geocentric CS are 3-D Cartesian geocentric CS that reference a horizontal datum.

Fitted CS sit "inside another CS. The fitted CS can be rotated and shifted, or use any other math transform [a mathematical function] to inject itself into the base CS." (unpublished draft specification). A Fitted CS references the datum of the base CS.

This specification will also support creation of coordinate systems of any dimension, because each of these CS and datum have the metadata shown if figure 6, representing ellipsoid, geoid, prime meridian, axis, parameters (both projection and transformation parameters) and unit (i.e. linear or angular units). The specification also includes enumerated lists for things like datum type (e.g., Altitude Barometric and Orthometric) and axis orientation. Attributes in CS_Info are from ISO/TC211 Spatial Referencing by Coordinates.

IDENTIFICATION

Interoperability requires a universal way to identify a CS, even if the CS is unique to a small project. A client cannot import data without knowing the CS. However no one has volunteered to keep the list of all CS for the entire geospatial community. The best anyone can do is to keep a list of lists.

Our CS identifier consists of an authority (one of the lists within the list) and a code (an integer). The authority is whoever assigned the integer to the CS. For example the European Petroleum Survey Group (EPSG, http://www.petroconsultants.com/products/epsg21.html) assigned 26778 to the Kansas South state plane coordinate system, so our CS identifier is EPSG:26778. This system is distributed, extensible and supports legacy name lists (if they are translated into integers).

INTERFACES

There are interfaces for CT (not discussed in this paper), to create a CS and to access CS metadata (i.e., attributes or properties). The interfaces support accuracy of a transformation and of a coordinate, but not precision of a CS.

The specification includes three profiles (i.e., versions), Microsoft Interface Definition Language (MIDL) specifications for COM interfaces, Interface Definition Language (IDL) specifications for CORBA interfaces and Java source specifications for Java interfaces. All three profiles have analogous functionality.

CONCLUSION

The CT specification will provide a common way of identifying CS and of accessing CT services that support accuracy calculation. When implemented, these specifications will ease data import. Users of compliant applications will import data without having to be aware of the CS of the imported data because the client and server will use the same identifier for a given CS. If the application cannot transform imported data to the native CS , a compliant server will transform the coordinates to the native CS. Compliant software will therefore import data more reliably and be able to deal more effectively with distributed data.

Compliant software is already emerging. A GIS vendor (Cadcorp Ltd, UK) has already fully implemented the COM profile of the CT specification. The specification supports the software's ability to transform an image or a feature (i.e. a feature geometry). Figure 7 shows how this prototype Cadcorp product has transformed an image.

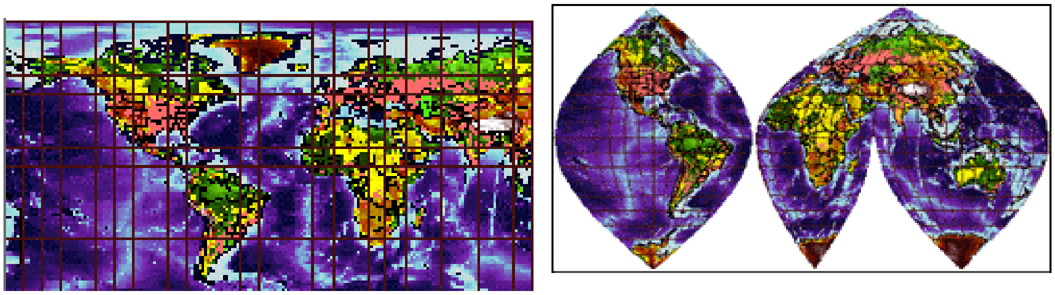


FIGURE 7

The image on the left in figure 7 was downloaded using the Open GIS Web Map Server Interface Specification (http://www.opengis.org/info/press/wm_overlays_pr.htm). Within months web browsers will overlay geospatial data views from various and diverse web servers that use this specification. Overlay will be possible because of interoperable coordinate transformation software.

DOD plans to modify its library of CT functions and its CT application (GeoTrans, available from iatbsw@tec.army.mil) so that it conforms to the OGC CT (Java) interface specifications. GeoTrans could serve CT on the web to any compliant application. Compliant applications would use the CT service invisibly to the user.

OGC maintains the CT specifications, improving them based on feedback from software developers and developing additional CT interface specifications.

REFERENCES

Geographic information - Spatial referencing by coordinates, (Committee Draft) project no. 19111 (15046-11), N814, 1999, ISO/TC211 WG3

OGC Request 9: Core Task Force, Coordinate Transformation Working Group, A Request for Proposals: OpenGIS Coordinate Transformation Services, 1999, OpenGIS Consortium

OpenGIS Implementation Specification: Coordinate Transformation Services, OpenGIS Project Document, 2000, Cadcorp Ltd. (unpublished)

UML Distilled, Applying the Standard Object Modeling Language, Fowler, M. and Scott, K., 1997, Addison Wesley

Experiments In Modeling Global Trade

Waldo Tobler

Professor Emeritus
Geography Department
University of California, Santa Barbara

An experimental spherical quadratic transportation model is applied to world exports and imports. The data consists of a table of 1987 trade (in millions of dollars) between seventy countries distributed thoughout the world. One version of the model uses the 4900 spherical distances between the country centroids. The model then generates supply and demand Lagrangians and attractivity and turnover potentials from the table margins. A variant of this model includes the summed exports and imports as weights. Both of these model estimates can be compared to the actual trade table. An alternate form of the model solves the spherical Poisson equation to get the attractivity potential. Gradient and streakline fields are computed from this net movement potential, and are shown in the form of maps.

Conference Program

International Conference on Discrete Global Grids
Radisson Hotel
Santa Barbara, California, USA
26-28 March 2000

Sunday March 26

Opening keynote presentation - Prof. Waldo R. Tobler

Monday March 27

Session I - Motivations
Chair: Jon Kimerling

- *Discrete Global Grids: One User's Perspective*
  Ralph Kahn, Jet Propulsion Laboratory
- *Discrete Global Grids and Digital Earth*
  Michael Goodchild, University of California, Santa Barbara
- *Discrete Global Grids Applicable to Spatially Balanced Sampling*
  Anthony Olsen, Denis White, US EPA; and Don L. Stevens. Jr., Dynamac Corporation

Session II - Computational Advances
Chair: Ralph Kahn

- *Augmenting SAND with a spherical data model*
  Houman Alborzi and Hanan Samet, University of Maryland, College Park
- *A Preliminary Comparison of Proposed Topologies for Geodesic Discrete Global Grid Systems*
  Kevin Sahr, University of Oregon
- *Practical Properties of the Spiral Points*
  Robert Raskin, Jet Propulsion Laboratory

Session III - Gazetteer Services, Reference Models & Interoperability
Chair: Keith Clarke

- *Universal Geospatial Data Exchange via Global Hierarchical Coordinates*
  Geoffrey Dutton, Spatial Effects
- *Interoperable Coordinate Transformation and Identification of Coordinate Systems*
  Daniel Specht, Topographic Engineering Center
- *Challenges of Georeferencing Places: Development of Digital Gazetteers*
  Linda Hill, University of California, Santa Barbara
- *Digital Earth: Building the New World*
  Yvan G. Leclerc, M. Reddy, L. Iverson and N. Bletter, SRI International.

Session IV – Transformations
Chair: Geoffrey Dutton

- *Pixel Loss and Duplication during Projection of Global Grids*
  Karen A. Mulcahy, East Carolina University

- EarthView -- a "planet browser"
  Paul Hansen and Tom Van Sant
- *Real-Time Global Data Model for the Digital Earth*
  Nick Faust, William Ribarsky, T. Y. Jiang, and Tony Wasilewski, Georgia Institute of Technology
- *The Global Spatial Data Model (GSDM)*
  Earl F. Burkholder, Global COGO, Inc.

Tuesday March 28

Session V - New Grids
Chair: Anthony Olsen

- *The World Geographic Reference System*
  Jordan T. Hastings, Go2 Systems, Inc.
- *A Seamless Global Terrain Model in Hipparchus System*
  Hrvoje Lukatela, Geodyssey Limited
- *EASE-Grid: A Versatile Set of Equal-Area Projections and Grids* Mary J. Brodzik and Ken Knowles, University of Colorado
- *Developing an Equal Area Global Grid by Small Circle Subdivision*
  Lian Song, A. Jon Kimerling and Kevin Sahr, Oregon State University

Session VI - Applications
Chair: Karen Mulcahy

- *Experiments In Modeling Global Trade*
  Waldo Tobler, University of California, Santa Barbara
- *Unexpected and Complex Behavior of Hierarchical, Multiresolution Cellular Automata*
  Ross Kiester, USDA Forest Service; and Kevin Sahr, University of Oregon
- *Finding a (Buffon�s) Needle in a Hierarchical Triangular Haystack*
  Ashton Shortridge, University of California, Santa Barbara
- *An Adaptive Grid and Associated Advanced GIS Techniques for Air Quality Models*
  Maudood Khan, M. Talat Odman; Hassan A. Karimi, University of Pittsburgh; and Michael Goodchild, University of California, Santa Barbara

Session VII - Comparisons
Chair: Hrvoje Lukatela

- Comparative Study on Map-Projection Based & Direct Spherical Tessellation for Global GIS
  Shohaku Ko, Chiba University; and Ryosuke Shibasaki, University of Tokyo
- Criteria and Measures for the Comparison of Global Geocoding Systems Keith C. Clarke, University of California, Santa Barbara
- Comparing Intercell Distance and Cell Wall Midpoint Criteria for Discrete Global Grid Systems
  M. J. Gregory, A. J. Kimerling, D. White, and K. Sahr, Oregon State University

Session VIII - Analysis & Summary
Chair: Michael Goodchild

- *The Relationship Between Wavelets and Quadtrees for Global Grids*
  Robert Raskin, Jet Propulsion Laboratory
- *Ellipsoidal Area Computations of Large Terrestrial Objects*
  Hrvoje Lukatela, Geodyssey Limited
- *Design Issues in the Go2 Grid System*
  Val Noronha, Digital Geographic Research Corporation
- *Closing Remarks*
  Steering Committee