

Table of Contents

Introduction	5
Editorial and Associated News [by Dr. Lee D. Miller, President]	8
Dichotomies in Geospatial Analysis.	8
“Spatial Logic?”	9
PDAs versus Tablet PCs.	11
Error Management.	13
New Error Management System	16
Easier Patching.	17
Product Licenses	19
Linux	19
Mac OS X	20
Mac OS X 10.2 (Jaguar).	20
X11 Public Beta 3.	20
Printer Control.	20
Equipment	21
Evolution of Software Distribution Media.	21
Mounting Multiple Monitors.	21
Tablet PCs.	21
Graphics Cards.	21
X Server (alias MI/X 4.0)	23
Rootless (which means, Windows Desktop) Mode.	23
OpenGL.	23
Render Extension.	23
Font Server.	23
X11R6.6.	24
TNTlite® RV6.8	24
Required Activation Code.	24
TNTsim3D™ for Windows	25
Introduction.	25
Distributing a Geosimulation.	27
Smaller Landscape Files.	27
Combine Different Kinds of Terrains.	28
Billboard and Stalk Overlays.	29
Volumes-of-Interest.	30
3D Polygons.	31
Layer Controls.	32
Map View Locator Gadgets.	33
Miscellaneous.	33
Patching.	34
Landscape Builder.	34
Available Now in TNT Development Version.	34
Sample Landscape Files.	34
TNTatlas® RV6.8	36
TNTatlas.	36
No Logo.	36

Miscellaneous.	36
TNTserver 3.0	37
Using JP2 Compression.	37
Serving JP2 Views.	37
JPEG Versus JPEG2000 Views.	37
TNTclients	38
TNTview® RV6.8	38
New Feature Summary.	38
Upgrading TNTview.	39
Installed Sizes.	39
TNTedit™ RV6.8	39
Oracle Spatial Import and Export.	39
Periodic Automatic Backups.	40
Interoperation of Tools.	42
Miscellaneous.	42
Inherited New Features.	43
Upgrading TNTedit.	43
Installed Sizes.	44
Tutorial and Reference Booklets	44
New Booklets Available.	44
Expanded Booklets.	44
Translated Booklets.	44
Tutorial Revision Plans.	45
New TNTmips Features	45
System Level Changes.	45
2D Geospatial Display.	47
3D Geospatial Display.	49
Management of Vector Styles.	55
Virtual (Computed) Database Fields.	59
* Open DataBase Connectivity (ODBC).	60
Landscape Builder.	61
Map Projections and Coordinate Systems.	66
Raster Extract/Copy.	68
Raster Import.	68
Raster Export.	68
Vector Import/Export.	68
Automatic Import/Export Testing.	69
* Oracle Spatial Layer Import and Export.	70
Vector to Raster Conversion.	79
Vector Warping.	80
Convert Regions to Vector Polygons.	80
Directional Analysis.	80
Georeferencing.	80
Fourier Frequency Filtering.	81
Mosaicking.	81
Map Calculator.	81
Transfer Attributes.	81
* Spatial Data Editor.	81

Vector Filters.	84
* Text Layer Controls.	84
Map Layouts.	85
Spatial Manipulation Language (SML).	89
Upgrading TNTmips.	96
Installed Sizes.	97
Internationalization and Localization	97
Operating Languages.	97
Microlmages Authorized Resellers	97
Pakistan.	98
Spain.	98
USA.	98
Zimbabwe.	98
Discontinued Resellers	99
Australia.	99
Canada.	99
Ecuador.	99
Germany.	99
Germany.	99
Latvia.	99
Nigeria.	99
Panama.	99
Peru.	99
USA, CA.	99
USA, CA.	99
USA, CO.	99
USA, SC.	99
USA, WA.	99
Appendix: Abbreviations	100
Attached Color Plates	
Geospatial Analysis now with X11 for Mac OS X	
Auto-Launch and Orbit in TNTsim3D	2-sided
JPEG2000 in TNTsim3D	
Multiple Terrain Surfaces in TNTsim3D	
Billboard Overlays in TNTsim3D	
Volume-of-Interest Overlays in TNTsim3D	
Extrude Polygons as Solid Shapes in TNTsim3D	
More Translated Documentation	
Add Vector Color Palettes via XML	
GeoLock with Relative Zoom	
Multi-View Locator Tool	
Controlling the Multi-View Locator	
Texture Filters for 3D Rendering	
Faster/Better 3D Visualization	
New Vector Style Assignment and Editing	
Improved Style Editor Interface	
Convert CAD Elements to Point Symbols (1/2)	
Controlling Curvature When Warping Vectors (1/2)	

Redesigned Line Pattern Editor	
Establishing Dynamic Relations Between Nodes, Points, Lines, and Polygons	
Improved Linking to Databases via ODBC	
Additions to Landscape Builder	
Import/Export Shapefile Line/Polygon Styles	
Vector Topology Types	2-sided
Behavior of Topology Types	
Oracle® Spatial Layer vs TNT Vector Object	
Importing Vector Objects from Oracle Spatial Layers	
Oracle Spatial Import Options (1/2)	
Editable Object Backup On Demand (1/2)	
Exporting Vector Objects to Oracle Spatial Layers	2-sided
Improved Vector to Raster Conversion	
Controlling Color of Map Grid Tick Marks (1/2)	
Convert Regions to Vector Polygons (1/2)	
Destriping ASTER Images	
Automatic Backup Options When Editing	
Interoperate Tools When Editing (1/2)	
Spline Lines in 3D (1/2)	
Filter Vectors Using Scripts	
WYSIWYG Text Editing	
Inserting Special Characters	
Diagrammetric Map Layout Tools	
Map Grid Labeling Options	
Publish Maps Containing Hatch Patterns	
Creating SML Dialogs	
Build SML Dialogs Using XML	
Sample Dialog Descriptions in XML	
Menus in SML Dialogs using XML (1/2)	
Strict Syntax Checking in SML (1/2)	
SML Dialog with Tabbed Pages using XML	
Nested SML Dialogs using XML	2-sided
Communicate with Visual Basic Programs using SML	2-sided
Suppressing Vegetation in Multispectral Images	
Making Color Separates for Printing	2-sided

Introduction

MicrolImages in its 17th year in business is pleased to distribute **RV6.8** of the **TNT** products. This is the 53rd release of **TNTmips** and adds 190 new features submitted by clients and MicrolImages. What follows is a brief introduction of the most significant of these new capabilities.

The interface components used throughout the **TNT** products to create and assign styles for vector elements have been rewritten to streamline them and add requested features. The design for a new, faster and higher quality 3D rendering model is being introduced. Cartographic features continue to be added to create professionally designed maps and image maps. Import and export of Oracle Spatial tables now permit **TNTmips** and **TNTedit** to be used to prepare or alter these tabular graphics. ODBC support has been improved and permits easier and faster access and exchange of tabular data with Oracle, SQL Server 2000, Access, and other external database systems. The Spatial Data Editor has received much attention but represents a very complex piece of software. It now provides auto-backup features to protect your work. The free **TNTsim3D** geopublishing tool has many new features to improve the real time presentation/publishing of your geospatial results. You asked for a means to connect actions in **TNTview** to Visual Basic programs, and this is available via **SML**. And, to keep all this running smoothly, the error management and patching scheme suggested by several has been instigated.

In this release several subtle refinements are being released, such as line densification for scale and projection changes with corresponding changes in vector sizes. Virtual fields can now be defined between element types for routing and monitoring dynamic systems and bringing GIS topology to bear where CAD data structures do not work.

- **2D Displays:** Multiple views can be geolocked at different view scales so that zooming in one auto zooms the others by the same amount (for example, use synoptic and detailed views). During panning the smaller scale synoptic view will auto pan to recenter on the larger detailed view if any part of it is panned outside the synoptic view.
- **3D Displays:** A third and high quality static perspective model is being released for rendering poster sized 3D rasters or in layouts. It renders a composite multi-layer 3D as fast as the other available models (ray tracing and dense triangulation, which also have minor improvements). It uses a game oriented triangulation server to compute the terrain from the DEM and render it using either the DirectX or OpenGL via the display board's graphics chip. The surface layers or textures are also draped over this model using DirectX or OpenGL.

A new texture server is used to resample all the texture layers selected with a choice of 6 different optional antialiasing and smoothing methods. Choose nearest neighbor or bilinear convolution for maximum rendering speed. Choose full anisotropic mipmapping for maximum quality. It computes the distance and perspective angle to the surface for each texel (which means, screen pixel). It then locates and computes the color of each screen pixel using its distance to interpolate between the average of multiple cells in two small linear arrays each oriented in the direction of view in the two bracketing pyramid layers.

- **Style Editor:** The Style Editor and its interface have been rewritten to improve their ease of use and add a variety of new style features. Point symbols can now

contain embedded characters (glyphs) from any language or symbol font. CAD blocks can be converted to point symbols. A design scale can be specified for scaling all elements and styles in a group.

- **Style Assignment:** Style assignment has a new dialog design. Styles can be assigned to elements with a single mouse selection click. Style assignments can be undone. Samples are shown for each available style. Styles that have been changed are highlighted.
- **Interactive Styling Text Layer Controls:** The **TNT** Text Layer Controls used in all processes including the creation and editing of text blocks for maps now integrates text editing into a single dialog and shows the text elements in the assigned styles. Using markup codes is no longer necessary but still available. All languages are supported including 2-byte Unicode languages, such as Japanese, Chinese, and Korean. Even cursive-like languages, such as Arabic, can be styled and viewed in this fashion and mixed in the text. All characters, especially special characters and glyphs, in the 2-byte font can be viewed in a scrolling window and selected by the mouse for insertion into the text stream in the editor.
- **Personal Color Palettes:** As many personal color palettes as required can now be defined in simple XML documents. Several samples of personal color palettes are provided as XML documents such as the 1024 color USGS palette. Choose personal color palettes anywhere that palettes are used.
- **Map Marginalia:** Many new refinements for designing map grid and marginalia features have been added to improve the appearance of a map. Considerable flexibility is needed in this area to meet national or other map standards, which are quite specific, but highly variable between organizations. For example, new controls are provided for grids and margin tick marks and priorities in cases of conflict. UTM and Lat/Lon label appearance can be selected and multi-label/grid conflicts resolved especially at map corners.
- **Converting to PDF, SVG, and Illustrator:** Conversion of map and other layouts can be controlled by selecting these file types as your “printer.” Hatch patterns can be converted. An Additional Options button on the Page Setup window lets you set certain parameters for PDF and SVG files. Fonts can be embedded, linked, or rendered into rasters. Raster objects in a layout can be converted to PNG, compressed or not, and embedded in the SVG or linked to it.
- **Importing Oracle Spatial:** All Oracle officially supported graphical elements (called geometry types) and their attribute tables can be imported from an Oracle Spatial structure (called a layer) into elements in a **TNT** vector object. An Oracle Spatial layer is a collection of geometries all having the same attributes set in associated tables. The attributes in these tables are converted into **TNT** attribute tables. If the Oracle Spatial layer is geocoded, then the **TNT** object will be georeferenced. The topology of this vector object will be created as your choice of polygonal, planar, or network during this import even though the Oracle Spatial layer has none.
- **Exporting Oracle Spatial:** A vector object can be exported to an Oracle Spatial layer. The proper tabular structure will be created in Oracle to contain this Oracle Spatial layer. Its vector elements will become Oracle Spatial geometry types. Attributes will be placed in corresponding tables. The vector object’s georeference and associated information are added to the Oracle Spatial metadata tables that

index geocoded layers. Note, **TNT** topology will be retained in the export but may be quickly lost by subsequent Oracle operations.

- **Editing Spatial Data:** Editing actions during the use of a tool are now only suspended when a different tool is selected to add a different element type. Incomplete editing activity suspended in this way can be automatically resumed from the last complete action when a tool is again selected. This provides for the more integrated use of the tools that act on different element types. XYZ vector lines can now be splined.

A new backup object can be automatically created and is faster than saving an object when editing is complete. A sequence of these multiple backup objects can be saved on demand, at defined intervals, or when editing has been inactive for a designated number of seconds. The last backup object can be quickly reloaded to restore the edit session or any backup object can be reloaded by selection.

- **Shapefile Styles:** The styles of lines and polygon fills associated with a shapefile can be imported and exported.
- **Real-Time 3D Simulations:** **TNTsim3D** for Windows has many new features and improvements. Polygons can be extruded from a surface as solid shapes. Different kinds of terrain layers can be stacked or the same kind of terrains automatically mosaicked. Multiple surface textures can be associated with each separate terrain layer. Texture layers can be linked, highly compressed JP2 files. Billboard symbol layers can be displayed and controlled as layers. Transparent spherical surfaces can now be rendered from tables to denote Volumes-of-Interest.
- **Landscape Builder:** Many technical features have been added to increase the flexibility of this process and to support creating Landscape Files using the new **TNTsim3D** features noted above. It merges textures and terrains including those already processed into other Landscape Files. It trims textures to the area of the terrains involved. Rasters can be compressed into linked JP2 files as they are converted to textures. Select and set up layers of points and styles for billboard symbols from a vector object. Select and convert polygons from a vector object for extruded shapes. Define position, color, transparency, and portions of spherical volumes.
- **Vector to Raster Conversion:** Uses more accurate polygon fill and line conversion procedures. Select the elements to be converted as all, by attribute, by query, by attachment, or as selected in the view. The user interface has been improved and uses tabbed panels.
- **Warping Vectors:** Set the accuracy for line densification (vertex insertion) to improve the curvature represented when zooming. Automatically densify lines to maintain curvature when changing projections.
- **Interacting with Visual Basic:** **SML** now provides a means of communicating with Visual Basic (or other programs) via ActiveX. For example, an **SML** Tool Script can be used to directly select an element(s) from the current view since it has access to any of the current contents or objects used in a complex **TNT** view. This Tool Script runs from an icon and starts or connects to a running VB program. The **SML** script then allows the interactive selection of a vector element from the active layer in the **TNT** view and sends it to the VB program. The VB program finds a corresponding record in an external database. It then displays the information about the element, supports editing this information in a form or by other means, and then adds or changes it in the external table.

- **XML Interfaces in SML:** XML can now be used to layout all common user interface components for the user controls presented by **SML** scripts for Windows and X. XML support and example XML documents are provided for dialog, nested dialogs, tabbed panels, cascading menu bar with icons, menu button, label, push button, toggle button, color button, edit text, edit number, radio group, combo box, list box, item, and others.
- **Advanced Virtual Fields:** Virtual fields (formerly called computed fields) function just like real fields in **TNT** tables. Previously these virtual fields were restricted to combining real fields for nodes, points, lines, or polygons in a vector object. Now virtual fields can combine real fields between all these element types for a vector object. This is a powerful feature for analyzing and displaying information about these spatially different element types whose real data field might change at any time.
- **Image Destripping:** The portion of the Fourier Analysis process used for destripping images has been rewritten, updated, and equipped with a new user interface.
- **TNTatlas:** Open more than one view window and geolock these multiple views. Geolock multiple views at different view scales. Set an atlas to start up with these multiple view windows using these predefined positions and new interrelationships.
- **TNTserver:** JP2 files (using JPEG2000 compression) can be used as linked raster objects to drastically reduce the size of the **TNTatlas**. **TNTserver** also now can send its results out as the smaller JP2 file in addition to the JPG (JPEG) file.
- **HTML-based TNTbrowser:** The image viewed in this **TNTclient** can be in JP2 format (JPEG2000 compressed) and, thus, considerably smaller than the previous JPG format (JPEG compressed). This reduces the “fetch” time for those using phone modems.
- **Tutorial Booklets:** A selection of the most important tutorial booklets for getting started with the **TNT** products is now available in 15 languages.
- **Mac OS X:** Apple’s X11 X server is now used in place of XDarwin and Orobor-OSX for **TNTmips**, **TNTEdit**, and **TNTview**. Now the **TNT** products use only Apple software and are easier to install and maintain and are somewhat faster in startup and interface actions.
- **Improved Error Management:** Errors in each new biannual official release will be corrected for the Release Version (designated **RV6.8**) kept for just for that purpose and provided for use in the Patch Version (designated **PV6.8**). A single patch will be available for download weekly containing all previous corrections to the official release. All post release development will be done in a separate copy of the official release called the Development Version (designated **DV6.9**).

Editorial and Associated News [by Dr. Lee D. Miller, President]

Dichotomies in Geospatial Analysis.

MicroImages has been focused for the last 10 years upon the development of a geospatial analysis product that contains a complex geographical information system with a fully integrated capability for analyzing remote sensing imagery. Recently the renaissance man or woman really into the idea of the integration in **TNTmips** has convinced us to integrate access to internal and external relational databases, direct use of other products’ geodata formats, surface modeling, GPS collection, and other data sources. At the other end of the analysis is the need for improved geopublishing tools, such as ever more complex visualization tools, advanced cartographic layouts and their conver-

sion to other common publishing formats (PDF, SVG, and AI), distribution of atlases and real time simulations, and so on. Gradually these new capabilities have also been incorporated into the **TNT** products.

Alas, only a fraction of the newcomers involved in geospatial analysis are academically trained to use this wide range of features in a single personal system. This is still the fault of our universities, principally those in and modeled upon the US, whose academicians know only one of these subjects and will teach it that way until they retire. MicroImages has answered with a wide range of tutorials, which will give students and professionals a wide grasp of this integrated approach and an introduction into how to go about applying these tools. Students worldwide download **TNTlite**, comment they want to use an alternative to what is used in class, and figure out what they want to do with it. But, of course, we all know that students have unlimited time but no money. Unfortunately most professional staff hired to work with **TNTmips** are not required to spend the 2 to 4 weeks necessary to go through these tutorials. They are immediately given some specific task and only use these tutorials as reference materials. Under these circumstances, it may take them 2 years to get an overall grasp of **TNTmips** and the analysis tools it offers.

Those using the **TNT** products who were trained in remote sensing, where the **TNT** products began 17 years ago, are experimental in nature. They are tolerant in piecing and patching a solution together to get their often unique analysis completed. Those engaged in GIS are often involved in production work for some enterprise activity and much less tolerant of adjustments and errors in the path to the completion of their work. Cartographers require very precise and meticulous results often dictated by some large set of national level specifications. Those interested in simulation require realism and speed, speed, speed. Those in production work still want a command line approach or something like **SML**. Each specialized use of the **TNT** products imposes its own different set of priorities and goals. Choreographing them all is a challenge, especially in the area of the inevitable error management.

“Spatial Logic?”

During a recent national election, a CNN analyst used the term “spatial logic,” which was new to me. It was used in the context of how they predicted the outcome of the various election contests shortly after the polls close based upon the earliest returns from all the voting districts. Voting districts in this context are polygons that have attributes. One would assume that these attributes are such things as the stratification of the voters in that district into parties, pre-election polling results, even the past distribution curve for each parties’ voters by time of day (which means, counting order). One would assume that dynamically changing attributes are also attached such as weather information and the party breakdown of the earliest votes as the ballot boxes are opened, counted, and tabulated throughout the voting period.

From this, those of you from other nations can understand why our U.S. regulations prevent the prediction of the outcome of elections before the polls actually close. If this were not done, these premature predictions could be used to influence those who have not yet voted. However, election prediction is one of the few areas in which the predictive use of this spatial logic is well controlled. There are very few other regulations that prevent anyone from applying spatial logic to a myriad of other predictions. For example, this author and his students 25 years ago applied the then crude tools of geospatial analysis (using a mainframe) and “spatial logic” to study, forecast, and display the future

land use evolution in the Denver metropolitan area and in shifting cultivation patterns in Northern Thailand.

Spatial Land-Use Inventory, Modeling, and Projection / Denver Metropolitan Area, with Inputs from Existing Maps, Airphotos, and Satellite Imagery. NASA [Goddard Space Flight Center] Technical Memorandum 79710. by Dr. Craig Tom, Dr. Lee D. Miller, and Jerold W. Christenson. August 1978. 210 pages.

Analysis of the Dynamics of Shifting Cultivation in the Tropical Forest of Northern Thailand Using Landscape Modeling and Classification of Landsat Imagery. NASA [Goddard Space Flight Center] Technical Memorandum 79545. by Dr. Lee D. Miller, Dr. Kaew Nualchawee, and Dr. Craig Tom. May 1978. 256 pages.

However, if you reflect on it, outside of mass marketing, most GIS and remote sensing applications focus on how things are today and, perhaps, how they came to be. They are still geared toward preparing static results from geodata and then presenting them in electronic or paper form.

The application of geospatial analysis in our spatial logic is changing for the better but not always for the best of reasons. The Internet has at least gotten us focused on the spatial nature of how things are today. A recently updated map is more valuable than an old one (this year in our car for locating the nearest open gas station or this moment for rerouting around construction sites or accidents). Recent satellite images in 3D on CNN tell us where things were yesterday in the invasion of Iraq and we used these to make our own predictions. Unfortunately, data mining with spatial logic has gotten associated with extracting information from the ever larger body of map, image, and attribute data already in our possession so its human user can tune, or adjust, the related system. We are still not very good at extracting patterns of activity, such as worldwide terrorism threats, from tabular data with possible point coordinates attached. Predicting the spread of disease is one of the few areas in which we are trying to apply spatial logic to spatial geodata for spatial predictive results.

Applying spatial logic requires spatial geodata, which is why we have been using CAD and GIS systems to move from simple tabular databases to new feature oriented data structures that combine tabular data, coordinates, and graphics. However, using spatially oriented feature structures for most predictive purposes will require that these structures provide historic and dynamically updated tabular and graphical data, eventually leading to 4D geodata structures (XYZ and time). For many applications it will also require that this data, at least temporally, possesses some well defined level of topology, which may be 2D, 2.5D, 3D, or 4D depending upon the application.

Where are we collectively in the evolution of geospatial analysis into using our spatial logic? Microsoft is just getting going in convincing us to use its .NET and XML concepts as a possible mechanism for sharing our information. Wireless networking, or at least cell phones, permit mandatory input of spatial information (immediate filing of a police accident report from the car) or tracking vehicles or people by cell phone or wireless GPS.

The best benchmark I know of for the "state of the art" in all of this from a geo viewpoint is where Oracle, a dominant relational database system, is in this evolution. In the Oracle Spatial User Guide and Reference for Release 9.0.1 published in June of 2001 (see complete reference below), it is stated on page 5-3:

"With Oracle 9i, Spatial provides a rational and complete treatment of geodetic coordinates. Before Oracle 9i, Spatial computations were based solely on flat (Cartesian) co-

ordinates, regardless of the coordinate system specified for the layer of geometries. Consequently, computations for data in geodetic coordinate systems were inaccurate, because they always treated the coordinates as if they were on a flat surface, and they did not consider the curvature of the surface."

From this, one assumes that as late as 2 year ago the use of Oracle Spatial was primarily focused upon the storage and use of engineering drawings and similar CAD derived materials. In fact, Oracle is only now about to release a new product called MapViewer that will permit its end-user clients to view and directly use the spatial layers in Oracle Spatial Release 9i. For more information and complete references to these Oracle product manuals see the detailed **TNTmips** section below entitled Oracle Spatial Layer Import and Export.

PDA's versus Tablet PCs.

Personal Assistants Wanted.

Personal Data Assists (PDAs) are mobile data collection and communication devices. However, Tablet PCs will become the housing for our personal assistants, or agents, (PA)—ebooks, maps and image uses, data collection, email, personal mobile video entertainment center (in the car kids will still argue over which video to watch on the rear screen). How long before we snap an image with our camera equipped PA and pull up a map and send the location of a pothole in the street or accident to the proper location. Or, we run the long edge of it over a document page and it's scanned. Only this morning while sitting here and writing this, a truck driver called who was lost 5 miles away trying to make a delivery to me. He had a cell phone and I gave him verbal instructions, but I would have preferred to have sent him a map for his PA in his truck if he had one. This alone is a huge PA application in many nations that have not developed the kind of street address, grid streets, and local maps we use in the US where it easy to locate where the party is or to deliver milk. As delivery services starting with DHL and FedEx and expanding to the cement trucks and mail services will expand in these nations using wireless PA map applications to save huge amounts of wasted gas, time, and money.

PDAs are Tabular Collection Devices.

PDAs and, soon, cell phones are very useful for creating or editing point data in the field or factory, and this is even better if it is overlaid upon a map or image. The downside of this is that most of the software to do this ultimately, like all other software for these devices, is either free or very cheap. This is not a particularly effective component of a business plan unless you are in a position to be the dominant player. I will acknowledge that there is another level of software for these same kinds of applications that involves a company, government agency, or consultant to develop a specific approach to using these devices to solve a specific data collection requirement of that organization. These are still areas of activity where cheap and widely available tools will be developed and most of the cost will be in the labor to develop that specific application. As a result, MicroImages has not created this kind of product even though some of you have requested it. There are already many good, useful, and progressively cheaper products available for this data collection application.

Tablet PCs are Geodata Collection Devices.

Early Opinions.

The emerging Tablet PCs, which run XP, offer an entirely different scenario. First of all, they already run the **TNT** products, such as **TNTatlas** and **TNTview**, used for direct field viewing and for sketching of meaningful graphic elements and their attributing (try drawing a larger polygon on a PDA screen). You could even move up to **TNTedit** on a Tablet, but how much topology reconciliation and advanced geospatial analyses do you want to do in the hot sun? Better to maximize field data collection (drawing, image interpretation, form completion) in the field and do quality control and work (snapping lines, boundary refinement, ...) in the motel and analysis in the office. The very first article I came across relating Tablet PCs to GIS stated these ideas quite clearly and I quote.

“Finally -- the Tablet PC has arrived. No more trying to read a map and the miniscule fonts on a 3 X 2-inch PDA screen. No more awkwardly fumbling with a clamshell laptop trying to enter attribute data in the field via the keyboard. No more hot-syncing PDAs once back in the office. No more paper maps (well . . . not quite).

“Despite all the hype and doubting counter-claims, the Tablet PC will revolutionize mobile GIS. Think about it. We wanted mobility so bad that when PDAs hit the market, GIS users were one of the fastest growing markets for handhelds. As Jim Skog, manager of Hewlett-Packard’s GIS Division noted, ‘I was surprised at the speed of the adoption of the iPAQ and Jornada handhelds. I didn’t really expect tens of thousands of those to get snapped up in GIS usage so quickly.’

“GIS users made extraneous efforts to shrink data and maps to fit on PDAs and create pick-list interfaces for stylus-based data entry. We modified our software to work with PDA operating systems. We were even willing to endure the 3–4-hour battery life.”

[Clearly many were desperate to get images and maps coupled to GPS units to support collection and correction of geodata.]

“Thankfully, with the Tablet PC, mobile GIS users will no longer have to work around the limitations of PDAs (and laptops for that matter). Basically a hybrid between a laptop and a PDA, the Tablet PC incorporates the best features of both -- the mobility and handwriting recognition of PDAs with the larger screen size, full-featured operating system, and high-end computing power provided by laptops. And, thanks to new processor technology and low power consuming components, most models offer battery lives of 4–5 hours -- equivalent to, if not more than, a PDA.

“Regardless of the model type, all Tablet PCs run on Microsoft Windows XP Tablet PC Edition. The operating system is identical to Windows XP except for handwriting recognition components and the ability to use the stylus as a mouse. The operating system also permits users to transition seamlessly from a docked (plugged-in) state to battery operation without shutting down or restarting.”

Quoted from Tablet PCs for Mobile GIS. Geospatial Solutions. 1 February 2003. Vol. 13 no. 2 Jim Englehardt. pp. 40–43 The complete article containing the specifications of most first generation Tablet PCs can be read at <http://www.geospatial-online.com/geospatialolutions/article/articleDetail.jsp?id=45442>.

Another early article on this topic was in a PDA magazine: How to Select a Tablet PC. by Geoff Walker. December 2002. Pen Computing. Vol. 9, No. 47. pp. 14–18. www.pencomputing.com/frames/tpc_how_to_select.html. Clearly they will reposition

their magazine content relative to the fact that we can now carry around a full computer, albeit form improvements are needed.

Short Term Reservations.

I do have some short term problems with the first generation of XP supported Tablet PCs (recall the first primitive, hulky, limited power PDAs). In my opinion, the first generation of Tablet PC devices is made-over portable computers. They are hurried releases that are simply reactions to Microsoft's first release of Windows XP for Tablets. As their designs evolve and other new manufacturers enter the market, they will become truly mobile computing and display devices. Devices that, if low enough in cost, will be kept in many convenient locations (1 in the car, in the office, in the kitchen, and so on). At a low enough price, convenience rules and not price. Even before PDAs, portable computers could be used in multiple locations and some field applications, but their price, form factor, and the ease with which they were lost, stolen, or damaged prohibited their wide adoption.

At the moment these established portable computer manufacturers are in the position of trying to protect their expensive portables by merely retooling their form factor and adding lots of unnecessary "stay at home" features to drive up the price of these first devices to match portables. Few of these companies have much past experience with a truly portable, walking around, wireless equipped form factor. We need a screen we can see in the sunlight, and we do not need a hardware keyboard (except as a cheap USB plug in) or many add-on features all of which hog power. Fujitsu is one of the few companies who have been building and improving portable field Tablet PC-like computers for many years.

If you need to use first generation Tablet PCs now, it is important that you review: Tablet PCs: Ready for Prime Time. by Cade Metz. PC Magazine. April 8, 2003. pp. 100-112. at www.pcmag.com/tabletpctips.

Error Management.

I take errors personally!

Those of you who know me personally know that as a professional I take errors in something that I am responsible for very personally. Others like to tell me that I should not take such things so personally, as its just business. However, going out and "beating up on the software engineers" also doesn't seem to change the reality that, by its very nature, all software has errors. Realizing this, I have managed MicroImages since its founding on the principle that "all software has errors, it is how we work together to fix these errors that really counts." Alas, more and more complex software just seems to have more complex errors. To keep up with this complexity in the **TNT** products, MicroImages is altering its error correction and patching procedures. Corrections of errors will now be made in a separate code base kept isolated from the code base used to develop the next version of the **TNT** products. The technical details of this new approach are described in detail in the section below entitled New Error Management System.

What took you so long?

The Big Picture.

Some of you have previously suggested that MicroImages adopt a dual code base approach. Others may have thought of it or experienced it with other software developers. So, now you are asking why it took MicroImages so long to come to this conclusion. Of-

ten the perception of how MicroImages should proceed is evaluated only in terms of your own objectives. MicroImages has always, since incorporation, taken a very international outlook in our products even though almost all our staff speak only English. This has greatly influenced our software development, marketing, and product support. Serving the entire world with a software product can be a very vague goal and difficult objective and requires a conservative approach in some activities and radical procedures in others. In making each decision, I must place myself in the position of all our clients in all nations, including those of you using older versions of Windows or other operating systems on older computers with low web access.

To summarize, dual development systems were difficult due to MicroImages commitment to international users, cross platform availability, uniform geodata structure, and our short release cycle.

The Strategies of Others.

Other software developers operate with different guidelines with regard to their software releases and subsequent error management. For example, many do not hesitate to quickly and prematurely drop their product's support for an older, aging operating system. However, due to economic constraints, that operating system might still be in common use in some nations (for example, W95 is still widely used in some locations). MicroImages tries to deal fairly with clients, language, and the technology that are in place throughout the world in widely varying economic situations. Ethiopia, where MicroImages has professional clients, certainly does not approach the upgrading of computer technology in the same way as the United States.

You probably operate a single version of Windows. However, the many historical versions of Windows in use impose a wide variety of constraints on software developers, especially in the assembly, installation and subsequent patching of their software products. To be responsive to an international client base with vastly different economies, MicroImages must contend with W95, W98, ME, NT, W2000, XP, XP Home, XP Tablet and all the various patches for each version. Some of these older Windows products are running on slower, older PCs with real memory of 128 Mb. Add to this all the same kind of gyrations on the Mac (9.x, 10.x), UNIXs, and many different vendors differing versions of Linux and their various kernels. Mac OS X Jaguar (10.2) was released 2 weeks after **V6.70** was shipped. Since that time, we have had to deal with the changes from 10.1.5 to 10.2, 10.2.1, 10.2.2, 10.2.3, 10.2.4, 10.2.5 and the switch from XDarwin and OroborOSX to Apple's X11 v.1, v.2, and Public Beta 3. It is a wonder that anyone's application software runs at all!

How do other software companies deal with these issues?

- They delay the release for older operating systems or totally drop support for a new or old operating system.
- They abruptly cease development and stop releasing for an operating system that has low market penetration.
- They do not respond directly to client input on errors at all or do so by indicating that you should wait for the next version (for example, Microsoft has set up its current interim patch system to respond to this criticism).
- They deal with a stable application that is not continuing to rapidly change (for example, constant introduction of new format types, alterations, and issues).
- Or the company is not prudent and conservative in this matter and simply disappears.

The long run objective of operating system developers like Microsoft and Apple is to achieve steady revenue by charging you rent to keep your operating system current or even to run it on their computers where it is kept patched. Application developers also are headed in this same direction as you can clearly see in Microsoft agreements with larger businesses soon to be imposed on smaller businesses by a literal army of salesmen. In the short run, I see a negative trend in all of this. Because of the ease of patching, it is not as necessary for any developers to test a release as thoroughly as if it were going to have to last without patches for a year. However, I believe that moving to isolate the current release and apply cumulative patches to the **TNT** products will be an improvement over our previous approach. On the other hand, it may also have a negative impact as fewer of you will be involved in the design of new features and error testing for the next release of our products. In either case, you will need to be patient, as it will take a few months to get all aspects of this new approach running smoothly.

Why Can We Change Now?

A number of improvements are now in place and more are planned that have permitted MicroImages to convert to a dual code base while continuing to serve our divergent international objectives and clients. I also feel this new approach will permit us to continue to rapidly innovate while periodically releasing new product versions that are reliable or at least reliably repaired. New versions become progressively more reliable and easier to access as we work together to correct the inevitable errors in this very complex software.

These are some of the more important technical reasons MicroImages can now change to a dual development approach.

- InstallShield and similar products for other operating systems are available for packaging Windows applications for installation and use in any version of Windows—with all the components needed for the various historical Windows versions. These kinds of software provide for easy installation but are much more important for product version management. They insure that all the correct libraries and product elements are put in the right places and provide for revised software component installation from a subsequent comprehensive patch file.
- You can now download a larger comprehensive patch file via your faster Internet access bandwidth that you have had to acquire to keep up with your operating system's ever larger and more frequent patches.
- MicroImages has gradually improved its intranet (LAN) system and associated internal version build and error management tools to allow management and coordination of our developments in several versions.
- Automated overnight testing procedures have been initiated and are expanding for some batch aspects of the operation of the **TNT** products (for example, daily map layout and vector topology validation testing prior to **V6.70** and for import/export for **RV6.8**). Implementation of automated testing of aspects of more interactive processes, such as spatial data editing, is currently being studied.
- The **TNT** product release cycle has been lengthened over the past several years from 4 per annum to 3 and now to 2 per annum. This has been possible as the Internet and its increasing bandwidth to your desktop has enabled you to get direct access to error corrections. You no longer have to depend upon MicroImages to fix your errors and get them in the next release sent to you via air express on CD.

Impact on Official Release.

Even with a dual development system, it is impossible to eliminate error propagation—this results from correcting an error in a specific application that causes an error in some totally unanticipated and heretofore reliable application. It is impossible for MicroImages each time we correct an error in the official release patch to check the literally millions of ways you might string together your interactive solution through our millions of lines of code. However, with a dual development system, each correction or patch you apply to the official release will improve that version's overall reliability. Thus, the number of errors you encounter in your patched version of the official release will decay asymptotically with time.

My selfish goal is to reduce the time MicroImages expends dealing with the errors you report by reducing the number of errors and the number of times each must be reported. As you know, your free MicroImages support is provided by professional computer scientists as it often involves complex technical issues. They are not specialists in any discipline to which you plan to apply your **TNT** product. However, more and more of our independent resellers are able to offer you assistance in how to design a geospatial application in your discipline, provide local training in your language, and immediate application support by phone or email. It is my hope that under this new error management system, our software support engineers will now have more time to devote to putting computer systems into place to catch errors in our nightly builds of all the **TNT** software and manage your error identification and correction more efficiently. They can then put more of their time into working with those of you working with us on the Development Version for the next release.

New Error Management System

Two Code Bases.

A new **TNT** error management and associated patching system has been established for use with the Release Version of 6.8 (designated as **RV6.8**) of the **TNT** products.

Patches obtained and applied to correct the errors in **RV6.8** do not include any features added for the next version of the **TNT** products.

Prior to **RV6.8**, a single source code base was maintained by MicroImages and all error corrections were made in that code base. Immediately after a new release was shipped on CD, changes to that code base began for the next version. Typically, the most complex changes to core features (for example, to the geospatial rendering engine (GRE), RVC file structure, topology management, ...) are begun immediately after release. This is necessary since these core items impact many other aspects of the operation of all the **TNT** products. Their early alteration provides the maximum amount of time to mitigate the impact of these basic changes on the next release. Changes are also initiated in specific applications. Using a single code base resulted in errors being introduced into the new version you were just beginning to work with if you applied any patches.

These early changes to the code base of the newly released version of our **TNT** product conflicted with your need to obtain reliable fixes for errors in that release. The patches you obtained usually fixed your specific error but might add another new error some-

where else. It is not possible for MicroImages to determine how the correction of a specific error might impact on a huge interrelated system such as **TNTmips**. Obviously, this is even harder to monitor when spurious errors are coming and going because that same system is in effect “torn apart.”

Two versions for the **TNT** code base are now supported at MicroImages. These code bases were identical on the day of the release of the master CD for **RV6.8** for reproduction. One is being used only to correct errors that you or MicroImages locate in **RV6.8**. This is the version you are installing from your official release CD and that you will patch as needed for your production work. The second code base immediately became the Development Version (hereafter called **DV6.9**) in which new features are being implemented. Anyone authorized to run **RV6.9** can still download and install the latest **DV6.9** weekly, and simply install it as a second **TNT** system to test and use all its new features as they enter this version. Errors you report will be corrected in both versions if they occur in **RV6.8** or only in **DV6.9** if they are caused only by new activities in that code base.

DV6.9 is constantly evolving toward the final official release of **RV6.9**. If you have not purchased **RV6.9** via annual maintenance or some other means, you can still download the **DV6.9** and use any product in it in **TNTlite** mode.

Note: Using the Development Version (**DV6.9**) in professional mode requires that your key be authorized for the use of the next Release Version (**RV6.9**) previously referred to as **V6.90**.

Easier Patching.

Patching the Release Version (RV6.8).

You will no longer be required to download your error corrections in pieces as you did with previous **TNT** versions. Once each week a single Patched Version (**PV6.8**) will be provided for you to download from microimages.com. This **PV6.8** will contain all the corrections to **RV6.8** since its official release on CD. Installing **PV6.8** does not change any of your preferences and does not contain the tutorials, manual, or sample datasets. It will not alter any of your Project Files.

The most recent weekly Patched Version (**PV6.8**) of the **TNT** products contains all the previous patches to the Release Version (**RV6.8**) found on your CD.

You will no longer be able to download the smaller patches needed to correct a single application in **RV6.8**. To obtain a patch for a specific feature you will always need to download the larger, latest **PV6.8**.

When you have patched **RV6.8** (replaced it with a **PV6.8**) the date of this total patch, and, thus, the identity of your current **PV6.8** will be clearly displayed. This “Date of Patch” of your **PV6.8** should always be provided to MicroImages’ software support to identify your **PV6.8** with every communication from you for any technical assistance request or error report. You will find this date on the Help/About TNTmips window and it will be changed by each new **PV6.8** you install.

What is the Development Version (DV6.9)?

MicrolImages has many innovative and imaginative (and patient) clients who work closely with us to guide and test the development of the **TNT** products. If you are in this group, you already know that you have a direct role in the evolution of the **TNT** products. For meeting our short run goals you patiently work with the Development Version and provide both error identification and design feedback, especially when we are working on improvements in your particular area of interest in a **TNT** product. Sometimes your suggestions are easily and quickly incorporated in the Development Version. Sometimes implementing your request is complex even though it may not seem so to you. Or sometimes your need is very specialized and is judged to be of low interest to any other **TNT** client. In these cases, it goes on our big "new feature list" for consideration in the future or as part of the periodic redesign of the entire processes or its low-level core operations.

Windows is very resistant to the installation of multiple copies of any software product. This is very important to keep you from mixing old and new components. Everyone has had the experience of concluding something was not operating properly in an earlier Windows application only to find, after much frustration, that a path led to the wrong version of a component. However, this Windows one-and-only-one-version policy has made it difficult to set up 2 versions of a previous **TNT** product. This is now simple as the **DV6.9** of the **TNT** products has a new and totally different identity in so far as Windows is concerned. Thus, **DV6.9** can be automatically installed and managed separately from **RV6.8** or its subsequent conversion to **PV6.8**, and Windows will treat each as a separate product.

Note: Windows will let you install and patch **DV6.9** whenever you choose as it is a separate product from your **RV6.8** or **PV6.8**

Using the Development Version (DV6.9).

Working with **DV6.9** of the **TNT** products is now more or less equivalent to your past experience with working with a patched version of **V6.70**. However, to work with **DV6.9** you will have to download it from microimages.com and install it. **DV6.9** with alterations and corrections will be replaced weekly rather than the previous Tuesday/Thursday patch procedure.

There is not and will not be a **DV6.8**! The version of the **TNT** products providing access to the new features being added for the next official release (**RV6.9**) is designated **DV6.9**.

Some of you may choose to obtain **DV6.9** to use its newest features on your latest time-critical project. Please do not count on these features working correctly if they are critical to the completion of your project. MicrolImages will, as in the past, be releasing these features as early as possible in their development as part of **DV6.9**. This is so that our early innovators and adopters can get at them via **DV6.9** and provide their suggestions and feedback as early as possible during their development. We will also be fixing errors and adding features in **DV6.9** just as fast as before. Errors encountered in **PV6.8** will usually be fixed concurrently in **DV6.9**.

Please do not request any documentation for any new feature added to **DV6.9**. It is likely that MicroImages' first written reference to a new, usable feature in the DV will be announced by the addition of a color plate to illustrate it at microimages.com. This color plate can only be created and posted by MicroImages when the new feature is at least usable by our writing staff. These new plates introduce new, visually-oriented features to you and often provide an overview of its operation. However, there are also new features being added, such as in the area of database management or at the systems level, which are not readily illustrated. We encourage and appreciate your continuing to work with us in the Development Version as in the past for a better **TNT** product and a better next release.

Patching the Development Version (DV6.9)

The patch for **DV6.9** will be the entire set of **TNT** products to replace the **DV6.9** you already have. New patched versions of **DV6.9** will be provided for you at weekly intervals. You should remove your current **DV6.9** before installing a new **DV6.9**. This is the most reliable way of ensuring that previous DV installations don't affect the latest DV installation. In some cases, the installer may remove the previous DV. Just as with **PV6.8**, **DV6.9** will not provide any tutorials, manual, or sample datasets. Since this will be an entirely separate version of your **TNT** products, a newly acquired **DV6.9** will be easily installed and kept isolated from the **RV6.8** or **PV6.8** on your hard drive. However, it will automatically share your sample data and Project Files.

Patching TNTlite.

Prior to **RV6.8**, **TNTlite** was not updated between versions. As you know, **TNTlite** uses exactly the same code base (now 2 code bases) and executables as the **TNT** professional products. Used in either mode, the **RV6.8** on the official release CDs, any downloaded **PV6.8** or **DV6.9**, and the associated installation and management procedures are identical. As a result, those downloading **TNTlite** from microimages.com will now always get only the latest **PV6.8**. Those who initially obtain **TNTlite RV6.8** by CD can also update it to the latest **PV6.8** by downloading it from microimages.com. Regardless of the route you choose to obtain **PV6.8** from microimages.com, you will be downloading the same **PV6.8** files. All **TNT** products simply use the presence or absence of a Software Authorization Key to determine how to start up, lite or pro.

Product Licenses

A PCMCIA Software License Key is available on special order. This type of key can be used for convenient portable computer operations. Unlike a USB or parallel key, it slides into the PCMCIA slot and does not project out where it can be lost or knocked off. A ruggedized portable computer has internal PCMCIA slots where this key can even be sealed inside the unit. You can see a picture of this and other **TNT** Software License Keys at <http://www.microimages.com/product/keys.htm>.

Linux

If you are using a variant of Linux that supports 64-bit file addressing (for example, RedHat 7.3), you can now create Project Files and other **TNT** files greater than 2 Gb.

Mac OS X

Mac OS X 10.2 (Jaguar).

The **TNT** products now operate under Mac OS X 10.2.5 or later. There is no logical reason for any user of Mac OS X to continue using a version earlier than 10.2, such as 10.1.5. If you are continuing to use these older versions, your **TNT** product may or may not run correctly as MicroImages is no longer maintaining test machines that use anything but the Jaguar version of Mac OS X. If you are using an earlier version of Jaguar (which means, 10.2, 10.2.1, 10.2.2, 10.2.3, or 10.2.4) please install your free upgrade to v10.2.5 before using your **TNT** product.

X11 Public Beta 3.

At Macworld in January 2003, Apple released their free X11 Window System for Mac OS X 10.2. Apple's X11 is a complete X Window System, which requires that you use Mac OS X 10.2. **RV6.8** of the **TNT** products uses this Apple-supplied X11 system. Originally X11 was released as a beta v.1, then beta v.2, and is now at Public Beta 3. Micro-Images has found Public Beta 3 to be stable and suitable for the operation of the **TNT** products. Please do not use beta v.1 or v.2. A color plate for possible promotional use by Mac aficionados is attached and is entitled Geospatial Analysis now with X11 for Mac OS X.

X11 eliminates the use of the XDarwin X Server and the OroborOSX window manager for the **TNT** products. These are both excellent products and **V6.70** worked well with them. However, their use required that you and MicroImages install, manage, and maintain 4 different software environments (Mac OS X, **TNT**, XDarwin, and OroborOSX). At a minimum, this meant that each time Apple upgraded Mac OS X delays were encountered waiting for upgrades to XDarwin and OroborOSX before the **TNT** products could be operated.

Apple's X11 provides the following advantages for **RV6.8** of the **TNT** products relative to **V6.70**.

- Fast startup and operation as X11 is optimized by Apple for use with Mac OS X.
- Single environment for use with the **TNT** products and your other X applications.
- Apple supplied integration between X and non-X products operating concurrently under Mac OS X (for example, cut, paste, ...).
- Easier installation and upgrading.

X11, together with the necessary window manager and the Motif libraries, can be installed in one operation directly from your **RV6.8** CD or from the **TNTlite** product folder you download from microimages.com. If you have already been using **V6.70** with XDarwin and OroborOSX, please remove these products before installing X11. Information on how to purge these products from your Mac OS X system can be found in the same folder as the installation file for X11.

Printer Control.

V6.70 did not present the printer manufacturers' set up window and changes had to be manually set in the printer. **RV6.8** now presents the specific printer's Page Setup window for your setting each time you print.

Equipment

Evolution of Software Distribution Media.

MicrolImages earliest releases of our commercial product predated DOS and Windows (COM anyone?) and were on 8" floppy disks. Next came DOS and Windows using 5.25" floppies. Then we got the then marvelous little 3.5" floppy. Eventually, because of our varied international client base, MicrolImages had to keep releasing on these floppies until everyone in the world had a CD reader. The number of 3.5" floppies for each **TNT** release was well over 50 before the official release was switched to CDs. MicrolImages only recently discarded the 3.5" floppy duplication equipment.

TNTmips RV6.8 fits nicely on a CD for one platform (OS) only, and with sample data and tutorials would have required more than 400 of those 3.5" floppies. But now, MicroImages is hoping you will all have a DVD reader real soon. In fact, MicroImages will soon be shipping two world data sets (GTOPO30 and DCW) in RVC Project File format with each new **TNT** product order on DVD only. These worldwide reference datasets are simply too large to provide on CD as they would span 7 CDs and require dividing this world data into 7 weird subdivisions.

The only way those buying our separate **MI/X** product for the past 2 years has been via a download. In fact, a month earlier some of you downloaded and installed your **TNTmips RV6.8** and reviewed the associated online color plates and tutorials. Some of you have even moved on to the most recent patched version **TNTmips PV6.8**. Obviously, software like **TNTmips** continues to expand in scope requiring new distribution media, methods of installation, and error detection and management.

Mounting Multiple Monitors.

MicrolImages has promoted the concept for years that geospatial analysis (image analysis, GIS, map layout, ...) benefits greatly from a system using multiple monitors. The **TNT** products have been carefully checked for efficient operation on multiple monitors with special gadgets added for this propose. Many good quality display boards now provide direct support of dual monitors. High quality, flat panel displays are available at ever lower and lower prices. By the end of 2003, it is likely that most computer monitors will be flat panel. There are now appearing mechanical mounting systems for convenient assembly or multiple monitors for convenient desktop and other uses. You can review all the interesting possibilities for assembling multiple monitors at www.9xmedia.com. This site also provides information about various graphics boards suitable for use with 2, 3, or many monitors.

Tablet PCs.

The use of these new Windows XP supported systems is discussed in detail above. For a thorough and up to date review of what is available in the first generation Tablet PCs see www.pcmag.com/article2/0,4149,925249,00.aspx for:

Tablet PCs: Ready for Prime Time. by Cade Metz. PC Magazine. April 8, 2003. pp. 100-112.

Graphics Cards.

Graphics chip and associated board developments continue at a rapid pace compared to all other workstation components, and these rapid improvements have a direct impact in geospatial analysis. **TNTsim3D** makes direct use of these advances by using their

hardware implementation of DirectX and OpenGL. Microlmages X server for Windows now directly supplies built-in support for OpenGL [see section MI/X Server below], however, use of this and/or DirectX will only gradually be implemented in existing **TNT** processes, for example, the rendering of 3D displays.

Current Leader.

At the moment, the lead in the graphics cards goes to the ATI's Radeon 9800 Pro over nVIDIA's BFG Asylum GeForce FX 5800 Ultra. As seems to be a very standard continuing practice, both of these boards are being introduced at the US\$400 price point. The nVIDIA board is not shipping as of yet. For comparison tests please see www.pcmag.com/article2/0,4149,980836,00.asp for

ATI Maintains a Rad Lead. PC Magazine. by Konstantinos Karagiannis, 22 April 2003. p. 38

Some interesting points made in this comparison are:

*"Both cards feature support for DirectX 9 [9 is already tested with **TNTsim3D**] and OpenGL, as well as 8X AGP."*

"These differences give the 9800 Pro a memory bandwidth of 21.8 GBps compared with only 16 GBps in the 5800 Ultra".

*"Indeed, our tests showed—at worst—only a tolerable performance drop when 4X anti-aliasing was activated [for the ATI 9800]." [Anti-aliasing is important in improving the appearance of **TNTsim3D** and 3D game operations. Activating it caused a frame rate decrease before this chip.]*

"The new GeForce FX architecture comes in a new form factor. We are not thrilled with the card's thickness—it eats up two card slots because its fan assembly – nor with its huge power draw (you'll need at least a 300-watt power supply). And when the fan kicks in, it's loud." [Another reviewer pointed out that the FX consumes 180 watts and if it were positioned just inside the top of the PC, it could be used as a coffee warmer as well. Certainly this is an impractical power draw.]

Future Orientation.

Can graphics chips continue their advance? Some insight can be gained from an excerpt from an article reviewing ATI's lead and where graphics chip design is headed.

The startup that saved ATI. Armed with management and engineering prowess for ArtX, No. 2 player in 3-D graphics is positioned for the next round of pixel wars. Electronic Engineering Times. Issue 1266. Rick Merritt. 21 April 2003. pp. 18-20.

"The DirectX Spec was driving a new architectural direction in PC graphics. Rather than delivering fixed functions based on approximations using integer math and a graphics pipeline pioneered by SGI, DirectX 8.1 had taken a new course: toward more general-purpose programmable vector processors based on more-exacting floating point calculations.

"Ultimately, it is thought that the DirectX evolution will lead chip makers to create devices based on dozens of computing elements that can calculate polygon vertices and run pixel-shading programs for a variety of graphics and video applications. Sony, IBM and Toshiba apparently share that vision. Their Cell architecture – announced in March 2001, though not yet released – could someday use hundreds of cores in a parallel array to power future PlayStation consoles and a wide variety of other broadband products.

“It’s all about programming now. That’s the new battleground.” said Peter Glaskowsky, editor of the Microprocessor Report. ‘These chips [nVIDIA and ATI] are not distinguished by the number of parallel pipelines or clock rates anymore. The key issue is how much can you do to each pixel you draw, how many programmable instructions you can run per pixel.’ ”

From these kinds of statements it is apparent that graphics chip advances will continue at a rapid rate.

<u>Transistors</u>	<u>Internal Speed</u>	<u>Memory Speed</u>	<u>Average</u>	<u>Price</u>
	(in millions)	(in MHz)	(in MHz)	(in US\$)
ATI Radeon 9800 Pro	110	380	340 DDR	\$18
nVIDIA GeForce FX 5800	125	500	500 DDR	\$18
Intel Pentium 4	55	3070	133*	\$170

* external speed but has four data cycles per clock cycle

This table from the same article is particularly revealing when you compare the rapid advance in graphics design driven by the game industry with Intel’s progress. What is particularly revealing is the almost 10 to 1 price difference in these two kinds of chips. This reflects in some part what happens when you have aggressive competition versus a virtual monopoly.

X Server (alias MI/X 4.0)

A new version **MI/X 4.0** of MicroImages stand-alone X server has been released for use with X programs other than the **TNT** products. Considerably more detail on **MI/X** can be found at www.microimages.com/mix/.

The most significant new features added in **MI/X 4.0** are as follows.

Rootless (which means, Windows Desktop) Mode.

MI/X users can now also optionally choose the Windows Desktop mode. In this mode, the Microsoft Windows desktop is exposed. Each X window will appear and behave similarly to the other Microsoft Windows you have open on your desktop. This feature has been available to **TNT** product users since the release of **V6.70** in August of 2002.

OpenGL.

OpenGL is now built into the X Server for direct use in your X client programs that can make use of it. **TNT** products do not yet make direct use of this built in OpenGL for rendering.

Render Extension.

The Render extension is now supported. This extension is used mainly to support anti-aliased text in X clients. The Render extension was invented by Keith Packard and has been used in the **TNT** products for some time.

Font Server.

X font servers are now supported. This allows a remote server to provide fonts not installed with **MI/X**. This is useful for remote systems like Solaris that use proprietary fonts. It has no direct use in the **TNT** products.

X11R6.6.

The core of **MIX** 4.0 is based on XFree86 4.2.1. This latest upgrade to the X server is part of **RV6.8** and solves several minor problems in X windows.

***TNTlite*[®] RV6.8**

Many university students and interested professionals are downloading **TNTlite**. High school and grade school students and teachers occasionally download **TNTlite**. The number of **TNTlites** downloaded from microimages.com in 2002 was 33% greater than in 2001. Another measure of this increase in interest is the amount of bytes moved out from microimages.com each month. At this time about 300 gigabytes are transferred out (downloaded) monthly from microimages.com. The portion of this bandwidth used for patches is a small portion of that used for **TNTlite**. Furthermore, MicroImages general web site uses a separate web connection from those used to support the download of **TNTlite**. As a result, most of this outgoing traffic is from downloads of **TNTlite**.

Required Activation Code.

TNTmips, **TNTedit**, and **TNTview** used free as part of **TNTlite RV6.8** (which means, without a Software Authorization Key) now require an activation code to start the first time. **TNTatlas** and **TNTsim3D** are freely distributed publication tools that have no lite limits and do not require a Software Authorization Key or this activation code to run.

Who Needs a Code?

An activation code is required no matter how **TNTlite** is installed: from an official **RV6.8** CD, a duplicate of this CD, a download from microimages.com or a mirror site, or by other means. When **TNTmips**, **TNTedit**, or **TNTview** are started the first time without a Software Authorization Key, they present a dialog box for you to enter both your email address and the activation code. The product will then start and restart each time without the code. When any of these 3 **TNT** analysis products are activated in this fashion, they are all activated.

When Will I Need A New Code?

You only need to successfully activate **TNTlite** the first time you start one of the 3 products. You do not need to repeat this process for any subsequent use of **TNTlite**. Your activated version of the **TNTlite** products stays activated! You will only need to obtain a new code if you reinstall **TNTlite** or MicroImages releases a new version of **TNTlite** that you want to install.

How Do I Get One?

A user of the **TNT** analysis products provided as part of **TNTlite** can acquire an activation code by completing the **TNTlite** registration form located at <http://www.microimages.com/tntlite/register.htm>. Upon satisfactory completion of this form, the activation code is emailed to the email address entered into the form. This activation code can only be obtained from MicroImages by this automated return email. If you are going to install from a CD, you may want to acquire your activation code before proceeding with an installation.

If you download **TNTlite** from microimages.com or some other site, after your download starts you will be immediately presented with the registration form. When it is satisfacto-

rily completed, you will be sent an email containing your activation code. Usually you will receive this email by the time your download is complete.

How Soon Must I Use a Code?

As noted, once activated **TNTlite** can be used forever. However, the activation code is only valid for 3 days (72 hours) from the time the code is issued. If you do not use it in that time period, it is voided and you must acquire a new code, but you do not need to acquire or reinstall **TNTlite**. You can acquire an activation code as many times as you like by completing the registration form at www.microimages.com/tntlite/register.htm. You can also use a single activation code to make as many separate installations of **TNTlite** as you wish (for example, on every classroom computer). However, all these installations will need to be completed within the 72 hours that the single code remains active.

What Happens to the Registration Information I Provide?

The email address you supply to acquire an activation code will be used to email you a notice when the next official version of **TNTlite** with important new features is available for downloading. All the information on the registration form is also compiled and supplied to the MicroImages Dealer who speaks your language or is nearby. They may or may not attempt to contact you. This information and your email address are not sold or provided to anyone else.

TNTlite Patches

With **V6.70** and earlier, MicroImages provided only the official release version via **TNTlite**. It was not updated or changed. **TNTlite** users obtained and installed that release version and generally stayed with it. They had less motivation than you, as a professional client, to learn the previous **TNT** patching system. Periodically they would reappear to download **TNTlite** again but were still getting the same version. Now, with the new patching system, a downloaded **TNTlite** package will be **PV6.8** providing all the accumulated patches to **RV6.8**. **TNTlite DV6.9** will also be available for downloading by anyone for possible experimentation with its new features as they are added and tested.

TNTsim3D™ for Windows

Introduction.

TNTsim3D continues to expand to provide you with an even better FREE product to publish, distribute, and permit free use of your geospatial products. Many significant new features have been added. Some of these features are particularly useful and are probably unique, such as support of JPEG2000, multiple textures, virtual mosaicking, stacking multiple terrains, operation in your language, and others.

Are you Ambidextrous?

Previous MEMOs have stressed that **TNTsim3D** is not a flight simulator. It is designed to provide realistic interaction with your geodata and the results produced from it via your geospatial analysis. This can be emphasized by referring to it as a geosimulation, or geosim for short, as it is inherently geographical in nature. Movements within **TNTsim3D** in a geosim may often be described as flying even though no aircraft type performance envelope is enforced. For example, you can jerk around your views to a new orientation at any time with your control devices (for example, using the View-

Center Locator gadget in the Map View). This is appropriate in the operation of a geosim since you do not want to slowly fly to each new viewpoint of interest or have to learn to bank to turn to your viewpoint.

TNTsim3D makes all your input devices (keyboard, mouse, and joystick) active at once. Perhaps you have already found that it operates best if you use both hands (or switch between devices). Your joystick is a convenient device to simulate realistic movement within a geosim. However, **TNTsim3D** also provides you with the mouse-controlled tools needed to occupy specific positions (for example, View-Center Locator gadget) and feature-specific actions, such as the readout of map coordinates of any point in any view.

Sometimes an action can be best controlled if assigned to two input devices. The joystick can be programmed to use a control to move the altitude up and down in a realistic fashion related to the setting used for its velocity and angular change controls. However, the mouse scroll wheel also changes your altitude for fast repositioning at any time or when used with the View-Center Locator gadget. For example, you can fly around over a realistic scene using the joystick. Once you, or your client, become familiarized with the realistic surface, you can use the mouse wheel to scroll up and down through a set of layers of other kinds of data stacked below that surface. These might be various processed image or maps layers overlaid on the same terrain but offset below the realistic reference surface. Or, using the new stacked terrain features in **RV6.8**, these could represent other kinds of non-geographic 3D surfaces of the same X-Y area but totally different Z layers and associated texture overlays.

Maximum utility of a simulation results if you carefully set up to operate it using more than 1 input device.

Integration with FREE TNTatlas.

If the simple feature analysis tools being added to **TNTsim3D** are not enough, you can start a **TNTatlas** for any position in the geosim that you select with the mouse. Then you can immediately use the interactive geospatial tools provided by **TNTatlas** for the corresponding 2D view of that same area for measurements, region actions, and so on. A Landscape File can also be an atlas, or HyperIndex stack, (they are all Project Files) and both can share linked files such as JP2 rasters. Thus, you do not even have to duplicate your geodata when you use these FREE tools to publish and distribute the results of your advanced geospatial analyses.

Conversely, you can also use **TNTatlas** to launch **TNTsim3D**. The MicroImages MEMO for **V6.70** (19 August 02, pages 46–50) described how to use a **TNTatlas** to pan and/or zoom to any desired location and then auto-launch **TNTsim3D** to view that position in 3D. This can be done using a sample Macro Script (distributed with **V6.70**) that is activated by an icon button on the **TNTatlas/X** View window. The button created by this sample script provides a drop-down menu allowing the atlas user to launch **TNTsim3D** in one of three modes: Orbit, Pan, or Stationary. The first two modes start **TNTsim3D** with automatic motion and are especially useful for a new user of **TNTsim3D** who may not be very good at moving around in a geosim. The automatic motion continues until the user activates some input control device and initiates interactive control of the geosim.

Distributing a Geosimulation.

Auto Orbit and Pan.

A new user of **TNTsim3D** for whom you have published your geodata may not be very good at moving around in a geosim. Now for this user you can publish and distribute a geosim that automatically starts from a CD or DVD with automatic motion relative to a point you specify. You can set up the geosim to initially orbit (circle around the specified point while looking inward toward it) or pan (viewer at the specified point and rotating while looking outward). For example, your geosim launches by auto panning around looking out from above a feature, such as a house. Once an input control device is activated (to move toward a distant feature of interest, such as a lake, for example), the automatic motion stops and you or your user now control the motion directly.

Launching **TNTsim3D** into an auto orbit or auto pan mode is defined using a Windows autorun.inf file. In addition to specifying the type of motion (orbit or pan), you can set the location of the point to orbit/pan around (map coordinates and height above surface), the ground width of the main view, the initial direction of view (heading and pitch), and the rotation speed. Auto launch and the format of the autorun.inf file are described and illustrated on the attached color plate entitled Auto-Launch and Orbit in TNTsim3D.

Performance Considerations.

Since orbit and pan motions can be specified as part of any auto launch from a CD or DVD, preprogrammed action and movement can be viewed immediately regardless of what input control devices are available and set up. (**TNTsim3D** provides immediate and automatic access through DirectX 8.x to any control devices you have set up in your system. If only OpenGL is installed, only the keyboard motion controls are available for use with **TNTsim3D**. If neither DirectX 8.x or 9.x or Open GL is installed, **TNTsim3D** gives an error message and fails to open).

TNTsim3D is designed to provide smooth motion regardless of the speed of the medium from which it is reading the data. So even if the user runs the geosim from a slow CD reader, a smooth orbit or pan movement is produced. However, a slow CD reader may prevent **TNTsim3D** from rendering the maximum texture detail in each frame. DVD readers inherently read data much faster than a slow CD and, thus, are not likely to produce this limitation.

Smaller Landscape Files.

Using **TNTsim3D 6.7**, your Landscape Files could be quite large if you assembled geodata for a reasonably sized project area. **TNTsim3D RV6.8** supports the use of JPEG2000 compression and the new limited area extents can reduce your Landscape File size by a factor of 10 to 100. This size reduction is illustrated in the attached color plate entitled JPEG2000 in TNTsim3D. Alternatively, these same new features permit your geosim delivered on CD or DVD to cover a much larger geographic extent by these same factors. Furthermore, now you do not have to combine (mosaic) small texture layers such as orthophotos in advance.

JPEG Compressed Textures.

Linked lossy or lossless compressed JP2 (JPEG2000) texture layers can now be used in **TNTsim3D 6.8**. Using JP2 linked files does not effect your frame rate but may slow down the rate at which detail is filled into the view(s) from front to back (which means, from near to far). Hard drive read rates are not easy for you to improve, but JP2 files

are much smaller, which greatly reduces the time to read these layers. However, this JP2 data must be decompressed in your central processor (CPU) before it is used and, thus, this controls how fast detail can be filled in each frame. The result is acceptable if you are running on a fast (>1 GHz) CPU and will improve if you can provide an even faster CPU.

Virtual Mosaicking of Textures.

Multiple texture layers could be associated with each terrain in **TNTsim3D 6.7**. Each was expanded with null cells to match the full extents of the terrain layer. Thus, using several smaller texture layers would unnecessarily inflate the size of the Landscape File. Texture layers used in **TNTsim3D RV6.8** no longer need to match the extents of the associated terrain layer. Now it is possible and convenient to use multiple smaller textures, such as orthophotos, with a terrain layer of larger extent and let **TNTsim3D** mosaic them during the simulation. This is also illustrated in the attached color plate entitled JPEG2000 in TNTsim3D.

Combine Different Kinds of Terrains.

Virtual Mosaicking of Terrains.

Just as for textures, you can now use a geosim that has multiple terrain layers. This feature is illustrated in the attached color plate entitled Multiple Terrain Surfaces in TNTsim3D. The terrain layers can be adjoining or disjoint patches of terrain, representing the same or a different type of surface. If all of the terrain patches represent the same surface type (such as quad map DEMs), these layers then make up a virtual terrain mosaic that will be assembled by **TNTsim3D** as it loads all the terrains into memory.

Vertical Stacking.

Multiple terrains can also be a 3D vertical stack of layers that represent different kinds of 3D information. Stacking different kinds of terrains permits you to move around in **TNTsim3D** in what appears to be a volume. The attached color plate entitled Multiple Terrain Surfaces in TNTsim3D illustrates such a volume. As demonstrated by the two different terrains in this plate, you might stack a bedrock surface, a groundwater surface, and a land surface. The texture for the land surface would be an image while the other 2 terrains can be presented as color coded values. You can also combine mosaicking and stacking. Thus, you can set up and move around in a simulation that uses separate terrains representing the ocean surface and bathymetry that make a virtual mosaic with the land surface and groundwater layer.

Vertical Offsets.

TNTsim3D positions stacked terrains using their elevation values, so the vertical separation between terrains depends on their relative elevation values. For example, a land surface terrain layer and a groundwater layer may be represented in real values referenced to mean sea level. The elevation difference between these surfaces may be small in comparison to their horizontal extents in the geosim, so they appear close together vertically when viewed from a distance in **TNTsim3D**. Although you cannot set vertical offsets for terrains in **TNTsim3D**, you can set vertical offsets for the visible textures associated with a terrain. Thus, in the above example, you could set a positive offset for any land surface textures to float them above their terrain and a negative offset for the groundwater surface texture to submerge it below its terrain. Either strategy

would increase the apparent separation between surface and groundwater textures and make the groundwater surface more readily visible. Unless you choose, you do not need to manipulate or change the real values in your original terrain layers to create a vertical separation between terrains with similar ranges of data values.

Vertical Exaggeration.

Vertical exaggeration can also be applied in the simulation to exaggerate the relief of the terrains. You can set this on the Options/Terrain tabbed panel. The default value is 1.0 (no vertical exaggeration). Increasing this value will apply increasing degrees of vertical exaggeration. The same exaggeration is applied to all terrain layers.

Billboard and Stalk Overlays.

Billboard Symbols.

Vector point symbols created in **TNTmips** can now be converted to a relational table structure that is used to define a Billboard and Stalk (B&S) symbol layer in your simulation. You can create multiple symbol layers and turn them on/off during a simulation using the same menu used to turn on/off texture layers. Each symbol layer is drawn into each frame in the geosim using either the DirectX or OpenGL functions of the graphics card. Each layer is a set of records in a set of relational tables that specify its location, height, base point, and other scale parameters. The drawing description defining its components, shapes, colors, and other parameters is read from the style assignment tables you created in **TNTmips**.

B&S symbols are drawn in the vertical plane into each frame. They are drawn in all Views you open. They act like billboards that always face you by rotating around the vertical or Z axis of the view (except in the Map View where they lie flat on the terrain). They do not rotate around the X axis and, thus, if you fly over a symbol, it becomes increasingly foreshortened as you approach until you see it edge-on when you are exactly overhead. These effects for a variety of different symbols and symbol layers are illustrated in the attached color plate entitled Billboard Overlays in TNTsim3D.

Floating Symbols and Stalks.

Some kinds of symbol designs are not easy to observe if they are positioned directly on the texture surface. Thus, each billboard symbol can be offset to float above or below the associated terrain layer by a height field in the symbol tables. Offset billboard symbols are automatically connected to their map position on the surface by a vertical line called a stalk. Since symbols can be offset either above or below the surface, the stalk can lead downward to the surface (like a flag pole) or upward to the surface (like a shaft). Stalks are managed in the symbol tables like lines in **TNTmips** and can have line styles such as color, width, and so on.

Vertical Exaggeration.

If you apply vertical exaggeration to the terrain relief in your geosim, the Options/Terrain panel provides a toggle button that lets you choose whether or not to apply the same exaggeration factor to Billboard and Stalk layers. If you turn this toggle on, the lengths of the stalks are exaggerated, but the billboard symbols themselves retain their original proportions.

Volumes-of-Interest.

Spherical Volumes.

You can display layers in your geosim that show spherical Volumes-of-Interest that have been created from 2D or 3D vector points. Solid or transparent spherical shapes are rendered around each point location in each frame. These shapes might represent the range of a cell phone or wireless network transponder, the flight danger zone for a hospital helicopter pad, the safe glide path for an airport approach, a radar's sweep area, and so on. Examples of the kinds of spherical VOIs that can make up a layer in **TNTsim3D RV6.8** are illustrated in the attached color plate entitled Volume-of-Interest Overlays in TNTsim3D.

Tabular Structure.

Each spherical VOI is defined in a relational table structure that is used to draw its VOIs in each geosim frame. VOIs are stored in the same database as the Billboard and Stalk symbol layers. Each individual VOI is defined in this structure as a point in space together with all the other parameters needed to define how it will be rendered by DirectX or OpenGL. These include many optional parameters such as its color and transparency, a surface grid including its spacing and color, inner/outer radius, max/min azimuth angles for sectors, and others. You can view (but not edit) the parameters of each VOI using the View Table option on the Layer menu in the **TNTsim3D** main view.

Transparency.

The color and default transparency for each VOI is determined by the color and transparency set for its individual parent point symbol in **TNTmips**. The color for VOI surface grids is determined by the default (All Same) color for all point symbols in the parent vector object. The preset VOI transparency can be overridden and changed during the geosim using a control on the Options/Extras tabbed panel. The texture and other layers can be seen inside of a transparent VOI, though colored by the intervening VOI surface. If you move your observation position inside a VOI, you can see the texture within the VOI in its true colors and the texture outside the VOI filtered through the VOI's color.

Vertical Exaggeration.

If you choose to exaggerate the terrain relief, the Options/Terrain tabbed panel allows you to toggle the VOI vertical exaggeration on/off. If you toggle VOI exaggeration to on then your spherical shapes will become egg shapes, while toggling it off keeps their shape and measurements unchanged. However, in either case, the center point of each VOI is moved up or down proportionally to the terrain exaggeration defined. If this were not done automatically, all of the VOIs could disappear below the surface.

Other Volumes?

Spheres were the first VOI you requested. Other mathematical volumes (which means, prisms, ovals, cylinders, ...) could be inserted into a geosim using this same procedure. If you can describe a general need for new VOI shapes in **TNTsim3D**, then MicroImages will consider adding them.

3D Polygons.

Introduction.

A Landscape File can now use as a layer a 3D vector object containing polygons that are extruded into solid shapes during the geosimulation. A common application would be to use polygons of building footprints accompanied by a height attribute. However, your solid shapes can represent any kind of polygons you wish to create in **TNTmips**. They could be selected forest stands of a single species whose height is the average tree diameter. They might be lakes whose depth is used to project them downward from the surface.

Use your imagination with regard to how your geodata and the results of your analysis can be portrayed as extruded polygons. You can obscure the entire texture layer, such as by using a soil polygon map with infiltration rate or another numeric attribute as the extrusion height of each polygon. However, often you will want to put your viewer into the context of what is being viewed in a geosim. In this case, you will want to limit the total area of the polygons in geosim. The attached color plate entitled Extrude Polygons as Solid Shapes in TNTsim3D illustrates this kind of application. In these examples the relative area of the solid polygons is limited. As a result, the texture and terrain information of the rest of the area is used to establish the context and location of the extruded shapes.

A polygon can also be extruded through a texture/terrain surface. The pre-extrusion "base" of each polygon is placed in the scene according to its map and elevation coordinate values, independent of any terrain surfaces in the view. (However, the base polygon is rendered as a flat object at either its minimum or maximum Z-value according to its settings for 3D display.) The base polygon can be on, above, or below a particular texture/terrain surface. From this base, the solid polygon is extruded upward or downward according to its associated height (positive or negative) or elevation attribute. If the base polygon is above or below a texture/terrain surface, its solid polygon can be extruded through the surface so the top of the solid is above and the bottom is below the surface. As shown in the bottom illustrations in this same color plate, the undulating terrain surface intersects solid polygons that pass through the surface in the expected irregular shapes.

Sun Position and Shading.

During the operation of your geosim you can choose the shading of the faces of solid shapes. This is controlled by setting the elevation and azimuth of the sun in the Options/Effects tabbed panel. Your extruded solid shapes will not cast shadows onto the surface. Thus, if your texture layer is an image with conspicuous shading effects and shadows, you may want to position the sun so the shading of your solid polygons (for example, buildings) matches the illumination direction in the texture layer. This is illustrated in the attached color plate entitled Extrude Polygons as Solid Shapes in TNTsim3D.

Top and Side Styles.

Just as in static 3D Views in **TNTmips**, styles can be defined separately for the polygon fill (top and bottom of the solid polygon) and for the extruded sides of the solid. The color styles used for the fill and extruded sides of the polygon are those established for polygons in the normal fashion in **TNTmips**. Unlike a 3D View, you cannot use bitmap pattern or hatch pattern fills. These fills do not scale up or down as you interactively

move closer or farther from that solid in your geosim, so **TNTsim3D** does not render them. Although the same fill style is used for the top and bottom of your extruded polygons, their rendered colors will usually differ because of the illumination and shading effects in **TNTsim3D**.

Multilayered Solids.

The side styles for solid polygons can be set up in **TNTmips** to display a stack of differently-colored vertical segments with each segment having a different color. The attribute used to extrude the polygons must have multiple records defining the height or elevation of the top and bottom of the stack and of the boundaries between the segments. This is also illustrated in the attached color plate entitled Extrude Polygons as Solid Shapes in TNTsim3D. In this illustration the boundaries between rock units of different ages are used to define the segment boundaries within each solid polygon.

You could also extrude vertically stacked polygons to represent geochemical, soil, water pollution, or other multiple property assays taken at GPS points. The color of the sides of the slice could represent each individual constituent's identity while virtual fields (computed fields) could be used to determine its assayed amount and, thus, its thickness in the stack. You can set up virtual fields to compute each layer's beginning and ending position in the stack and reuse this approach each time you collect new samples.

Vertical Exaggeration.

When you choose to exaggerate the vertical relief of terrains during the operation of your geosim, the Options/Terrain tabbed panel also provides a toggle to let you exaggerate the vertical dimension of the solid polygons by the same amount. If your solid polygons represent buildings, you may want to exaggerate the terrain but not the building heights. On the other hand, if your solid polygons simply represent map areas with differing numeric attributes, you may wish to let their relative heights be exaggerated along with the terrain.

Layer Controls.

During the operation of a geosim you can now toggle on/off any texture, symbol, volume, and extruded polygon layer. This is done using a cascading menu of terrains and the layers assigned to them. This menu is accessed from Layer on the menu bar in the main view and illustrated on the attached color plate entitled Multiple Terrain Surfaces in TNTsim3D. If you toggle off all layers associated with a terrain, then nothing will show in that terrain's position in any view.

Texture layers are created explicitly for a particular terrain layer, and the Layer submenu for each terrain layer in the geosim shows its available texture layers. Only one texture layer per terrain layer is turned on by default when you start a geosim, but you can turn any of them on or off at any time. Overlay layers (symbol, volume, and extruded polygon) are not explicitly associated with a particular terrain layer when they are created. Overlay layers created from vectors with 3D coordinates (which can be any of the three overlay types) retain their vertical coordinates and, therefore, do not need to reference a particular terrain layer when shown in your geosim. These 3D overlays are listed on the main Layer menu along with the terrain layers, and can be toggled on/off independent of any terrain layer. Symbol and volume overlay layers created from 2D vectors need to be stalked/extruded from a particular terrain layer but are available for use with any terrain layer in the geosim. These 2D overlay layers are listed on each terrain layer's submenu in the Layer menu cascade, and so can be turned on/off for any terrain layer.

Map View Locator Gadgets.

You can now choose from two different Locator gadgets for the Map View: the Arrow Locator gadget and the View-Center Locator gadget. Each gadget has its own configurable features. The Arrow Locator has color selection and the Viewer-Center Locator has viewer, center, and line color selection.

The View-Center Locator gadget is now the default in the Map View and has been improved. The cursor shape changes to a cross when you are near the nadir or “+” end showing that you can drag it if you hold down the left mouse button. Both ends of the gadget (“+” and “circle”) now move more reliably.

When you move all your views with this gadget, any roll you have created in your main view by some other control (for example, via the joystick) will be fixed and maintained as you move the nadir point of the main view around with this gadget. In other words, if your horizon is level (or at some angle) it will be maintained at that orientation as you move your main view with the gadget—you will no longer roll over.

You may at first experience what you think is spurious behavior in the operation of this gadget. Remember that you are manipulating a 3D control gadget. For example, a change in the altitude or nadir position of your main view by some other direct control may automatically and correctly change the position of the circle and the dashed line in this gadget on your Map View.

The circle end of this gadget represents the intersection of the view line of your main view with a fixed point on the surface represented by the circle (which can also be moved separately with the mouse). Thus, if you lower your altitude in the main view with the joystick or mouse scroll wheel, the circle end of the gadget may jump toward your nadir point as a hill in the foreground suddenly obstructs the view to the previously distant intersection of your view line with the terrain. In a similar fashion, for a fixed nadir and altitude, you can drag the circle to a position behind a mountain obstructing your view of that position. This will cause the circle to snap to a new position on the near face of the mountain with a shorter dashed line connecting it to your nadir (which means, the “+” end), which has not moved.

Miscellaneous.

You can select the format of latitude and longitude coordinates as well the units for other parameters displayed in the Readout window. The Projection button on the TNTsim3D Readouts window opens the Coordinate System/Projection Parameters window, where you can choose from the many different Latitude/Longitude formats supported in all the **TNT** products. Menus are also provided for you to choose the units in which to show distance, elevation, angle, and velocity values in the Readouts panels and on the Options/Input panel. If you have changed projection settings for the readouts, you can automatically reset all of these parameters back to their default state using the Reset button in that same window.

You can quickly review the actions assigned to the controls of your input devices (keyboard, mouse, or joystick) in the TNTsim3D Controls window. (To open the Controls window, select Controls from the View menu in the main view). In this window, assigned controls and their actions are shown in black. Control names are now shown in red for any device controls (switch, slider, ...) that have not yet been assigned actions, highlighting the controls that are not yet available for your use. To assign actions to these

controls, use the Configure DirectInput button on the View/Options/Input tabbed panel to open the DirectInput configuration dialog.

A text field in the lower right corner of the main view now indicates in which direction you are looking in this view as well as whether or not the altitude lock is turned on.

Patching.

Correction of errors in the Release Version (**RV6.8**) will be made as part of the single, composite, cumulative, and general **TNT** products patch posted weekly for **RV6.8**. If you download any Development Version (which means, **DV6.90**) of the **TNT** products you will also get the current DV of **TNTsim3D** as part of that single download. Also, as previous, periodically DVs of **TNTsim3D** that introduce significant new features may be packaged for direct download and easy installation as a stand alone Windows product for your FREE use or distribution with your geosimulations.

Landscape Builder.

Please see the section below entitled TNTmips/Landscape Builder for information about the changes to the Landscape Builder process used to build Landscape Files that incorporate the new features added to **TNTsim3D RV6.8**.

Available Now in TNT Development Version.

Interactive Movement of Billboards.

There have been requests from “early adopters” of **TNTsim3D** to permit the selection and movement of Billboard and Stalk points in **TNTsim3D**. For example, using a “move” mode, select a flag symbol in a view with the mouse and move it around on the associated terrain layer to a new position. After you move the symbols, a save option would allow you to save a copy of the overlay with the updated coordinate positions.

The addition of this feature to **TNTsim3D** is underway and should be complete by the time you read this. This is a new feature for eventual release in **RV6.9** and will not be part of any patch you apply to **RV6.8** of the **TNT** products. However, **TNTsim3D** is FREE and does not check the **TNT** software authorization key. Thus, Development Versions (**DV6.9**) of **TNTsim3D** with this and other new features will be posted for download and installation, along with associated sample Landscape Files. You can then try out and demonstrate these new features using the samples. However, be aware that creating your own Landscape Files to use any new **TNTsim3D DV6.9** features like this may require access to a DV of the Landscape Builder created after **RV6.8**. As explained in detail in the section above entitled Using the Development Version (DV6.9), this will require that you have access to a **TNT** key authorized for **RV6.9**, and that you download and install the **DV6.9** version of **TNTmips**.

Sample Landscape Files.

The following 5 sample Landscape Files are provided on the **RV6.8** CD (in the simdata folder) to illustrate some of the new **TNTsim3D** features. Other Landscape Files showing previously available features are on your **V6.70** CD, and you can download many others from microimages.com/products/tntsimLandscapeFiles.htm. Additional new Landscape Files will be added at this download site as they are created to test and demonstrate new **DV6.9** features.

LincNW2.sim (45 Mb).

All three available types of overlay layer are illustrated in this geosimulation of the northwest portion of Lincoln, Nebraska. The single texture layer is a mosaic of color orthophotos with 2-meter spatial resolution. Two Billboards and Stalk symbol overlay layers show locations of schools (color-coded by type) and libraries. Extruded solid polygons are shown for some buildings at the airport, and a Volume-Of-Interest (VOI) layer shows three sets of spherical volumes placed at random locations in the scene. Since the topographic relief is low, you may want to increase the vertical exaggeration in the scene (on the Options/Terrain tabbed panel) and experiment with the separate toggles that control whether exaggeration is also applied to the B&S symbols, VOI, and extruded polygons.

paradox6.sim (12 Mb).

This geosimulation shows geologic information for the Paradox Basin region in southeastern Utah and southwestern Colorado, USA. This is a petroleum-producing area with a relatively simple, layer-cake, geologic structure. The single texture layer fuses a geologic map with relief shading, and also shows the basin outline and state boundaries. The overlay layer shows the use of the color stack side style with extruded solid polygons. Each of the six circular polygons represents a reconstructed section of rock units for that particular location, using data from the United States Geological Survey. The solid polygons extend both above and below the surface. The portions above the surface show reconstructed sections of rock units that once covered that location, but have been eroded away in the recent geologic past. The portions below the surface show the sequence of rock units still preserved in the subsurface. Each polygon has a database record for the top of the section and for each rock unit boundary, with extrusion based on the elevation attribute values recorded in these records. The styles for the color stack on the sides of the polygons are assigned using another table containing the rock unit names.

sthelens3.sim (8 Mb).

This Landscape File shows the volcano Mount St. Helens in southwestern Washington State, USA. A texture layer covering the entire terrain layer shows color shaded relief, blending color-coded elevation with hill shading. Two partial texture layers show color band combinations from Landsat Thematic Mapper satellite images. One shows a color-infrared band combination and the other a natural-color band combination. Each partial texture layer or raster object covers exactly the area shown (they are not padded with null cells to cover the full extent of the terrain). Three B&S symbol layers are also included, showing the positions of ranger stations and visitor centers, survey locations for monitoring the volcano, and radio towers. Most of these B&S symbols were designed to sit on the terrain without stalks (they have height = 0) while the survey points are extruded above the surface on stalks.

FFbasin4.sim (13 Mb).

Two stacked terrain surface layers are used in this geosimulation covering a small desert basin in southern Nevada, USA. One terrain layer represents the ground surface and has two texture layers—a color shaded relief image and a natural-color Landsat satellite image. The other terrain layer, which covers a slightly larger area, represents the upper surface of pre-Cenozoic basement rocks beneath the basin fill. Its texture layer is a color-shaded view of the basement surface. Parts of the edge and interior of this tex-

ture layer where there was no data are transparent. The elevation values in the basement terrain layer are 5000 meters below their true elevations relative to sea level. This downward offset of the basement surface makes it easier to view both surfaces at the same time from the edge of the scene. You can turn off the ground surface texture layer to view the entire basement surface layer or try flying between the two terrain layers.

Roanoke7.sim (34 Mb).

This geosimulation of the Roanoke, Virginia (USA) area has two adjacent terrain layers, each representing the ground surface over half of the geosim (north and south). Each terrain layer has a complete color shaded relief texture layer and two partial texture layers. The partial texture layers are extracts of color digital orthophoto quadrangle images with 2-meter spatial resolution, providing a virtual image mosaic covering the central part of the scene. The source images for the orthophotos were acquired as 1-meter MrSID compressed files, which were converted directly in the **TNTmips** Landscape Builder to partial texture layers linked as lossy-compressed JP2 files. Without the JPEG2000 compression this Landscape File would be much larger.

TNTatlas[®] RV6.8

TNTatlas.

TNTatlas continues to be a free and relatively unique approach to distributing geodata along with a suite of interactive, but local GIS analysis tools. It continues to be disturbing to see **TNTatlas** listed along side simple viewers. Yes, **TNTatlas** can be used as a simple viewer since it will let you view any single geodata file format that can be linked for direct viewing. In fact, Microlimages is giving in to this idea to some extent by letting you “dumb down” **TNTatlas** by hiding features.

However, the most important aspect of having this geopublishing tool available is the availability of **TNTmips** and **TNTedit**. With these tools you can create and prepare your site specific, project specific materials for use in **TNTatlas** or **TNTserver** or shared with a **TNTsim3D**.

No Logo.

Starting a **TNTatlas** from a **TNTsim3D** is interrupted by the presentation of the **TNT** splash screen. As a concession to this application, **TNTatlas** can now be started up to skip the flashing of this splash screen. To do this add `-nologo` to the command line.

Miscellaneous.

The **TNTatlas** Creation Wizard will now copy all files of any type when they are referenced as text layers.

The preferences set for the background color during the construction of a HyperIndex are now used by **TNTatlas**.

TNTatlas for Windows.

This version now supports the zoom and Latitude/Longitude settings in the atlas file as available in **TNTatlas/X**.

TNTatlas for X.

Multiple views can now be opened for the same atlas (for example, side by side) and manipulated separately. These windows may be geolocked, scale-locked, show track-

ing cursors, and so on just as in the Display process. For example, an atlas can be designed using scale-locking so that in 1 view you are zoomed out in the atlas structure so as to provide a synoptic overview. The second view could be in the same structure and zoomed in for more details. Zooming in or out of 1 view would similarly zoom the other at its relative scale.

If you plan to distribute **TNTatlas/X** using multiple views, you probably want to automatically present them in the appropriate size, scale, position, and so on. This can be done in the layout used to start the atlas. Without this kind of auto start setup, the casual user will probably not think to use your atlas in this comparative fashion.

Advance Notice for DV6.9.

Modifications planned for **TNTatlas** for X (but not **TNTatlas** for Windows) will enable you to control which major functions are available for use. This will be accomplished by simply removing their icons from the toolbar. For example, you might omit the Navigation icon if your HyperIndex stack is not designed to use it. You will simply specify in the Startup file which icons are to be presented and which are to be "shut off." Obviously, hiding the icon and, thus, the access to it from the user of the atlas will not reduce the size of the program. However, it will permit the atlas to appear simpler and more specialized for the specific use you have in mind.

TNTserver 3.0

Using JP2 Compression.

TNTserver now supports the use of a HyperIndex (an atlas) that uses images and other rasters in linked JP2 files (JPEG2000 lossless or lossy). The obvious advantage of this is the much higher compression that can be achieved relative to other methods, especially JPEG compression. However, another important advantage is that, at equal quality, more highly compressed images are smaller and, thus, more rapidly retrieved by the **TNTserver** from the file server.

Serving JP2 Views.

You now have the option of choosing that the view (raster) returned by **TNTserver** to the **TNTclient** can be in a JPEG or JP2 format. Obviously the time needed to fetch a smaller view is less as it is directly proportional to the time needed to move it through a network. Fetching a smaller file for the same quality is very important for those who access a **TNTserver** via a phone modem or other low bandwidth service.

JPEG Versus JPEG2000 Views.

The following simple test compares the size of the view fetched from **TNTserver** for rasters of varying content and compression. In all these tests, the size of its view in the HTML version of **TNTclient** was kept the same at 827 by 791 = 654,157 pixels. However, since the view fetched is 24-bit color its size $654,157 \times 3 = 1,962,471$ bytes. The contents of the cache used by the **TNTserver** were purged for each test run. A sample atlas consisting of Ann Arundle County, Maryland (part of the MERLIN online atlas of the Maryland Department of Natural Resources) available via a **TNTserver** at microimages.com was used for each test. The ground area and scale of the view are the same for all tests and only the layer type was changed.

<u>Type of Raster</u>	<u>JPEG</u>	<u>JP2</u>	<u>JP2</u>
-----------------------	-------------	------------	------------

<i>Amount of Compression</i>	75%*	<i>lossless</i>	20:1
Color IR Orthophoto	283,967	1,007,488	98,123
Compression Ratio	7:1 (6.91)	2:1 (1.95)	20:1
Scan of a 7.5' Topo Map	275,873	1,026,079	98,123
	7:1 (7.11)	2:1 (1.91)	20:1
Polygons from a Vector	171,570	608,670	98,123
	11:1 (11.44)	3:1 (3.22)	20:1

*JPEG 75% is the default in the TNTserver

The lossy 20:1 JP2 images produced results of similar quality to the 75% lossy JPEG images. However, they could be fetched 2 to 3 times faster. These tests show that if the users of your **TNTserver** are likely to connect via a phone modem in your nation or user base, serving up JP2 views can have a significant impact on their activity in the **TNTclient**. The disadvantage of this is that the client must still download a plug-in into their browser to support the decompression of the JP2 file.

TNTclients

The HTML client now supports the use of JP2 compression for the views returned from **TNTserver**. However, the decompression of these JP2 views requires that you obtain a plug-in to support it in your Netscape browser or Internet Explorer. For this purpose your **TNTserver** site can provide access to the free plug-in from Adobe as a first and one time step. Adobe Reader 6.0 (formerly Adobe Acrobat Reader) now supports the use of JPEG2000 compression and JP2 files. Thus, it can not be long before browsers catch up and add JP2 decompression and perhaps SVG interpretation as a standard features.

TNTview[®] RV6.8

New Feature Summary.

A variety of new 2D and 3D display enhancements have been added. These include more modes to lock together multiple views and then use them in tandem. The 3D views are being improved in quality with new image smoothing algorithms, such as anisotropic mipmapping, and a new model for rendering relief. You can now easily design and use your own custom color palettes. Images in linked JP2 files now display much faster. New ellipsoids and high precision datums are supported for import and direct conversion during display.

TNTview continues to provide you more customization tools by improvements in **SML**, such as the new XML dialog design procedures. **SML** tools added to a **TNTview** can now also start and communicate spatial and tabular information to Visual Basic and other ActiveX compliant programs. For example, an **SML** Tool Script mode can be invoked by an icon on the view's toolbar. This Tool Script can then support the interactive selection of an element in the view and communicate its location, properties, and attributes to a co-running Visual Basic program for some other analysis or for additional data capture. All of these and many more new features usable in **TNTview** are described in detail in the technical section below on **TNTmips RV6.8** and illustrated in the attached color plates.

Upgrading TNTview.

If you did not purchase **RV6.8** of **TNTview** in advance and wish to do so now, please contact Microlmages by FAX, phone, or email to arrange to purchase this version. When you have completed your purchase, you will be provided an authorization code by FAX. Entering this authorization code the first time you run your product completes the activation of **TNTview RV6.8**.

The prices for upgrading from earlier versions of **TNTview** are outlined below. Please remember that new features have been added to **TNTview** with each new release. Thus, the older your version of **TNTview** relative to **RV6.8**, the higher your upgrade cost will be.

Within the NAFTA point-of-use area (Canada, U.S., and Mexico) and with shipping by ground delivery. (+50/each means US\$50 for each additional upgrade increment.)

<u>TNTview Product</u>	<u>Price to upgrade from TNTview:</u>					<u>V6.20</u>
	V6.70	V6.60	V6.50	V6.40	V6.30	and earlier
Windows/Mac/LINUX	US\$175	275	400	500	555	+50/each
for 1-user floating	US\$210	330	480	600	667	+60/each
UNIX for 1-fixed license	US\$300	475	600	675	725	+50/each
for 1-user floating	US\$360	570	720	810	870	+60/each

For a point-of-use in all other nations with shipping by air express. (+50/each means US\$50 for each additional upgrade increment.)

<u>TNTview Product</u>	<u>Price to upgrade from TNTview:</u>					<u>V6.20</u>
	V6.70	V6.60	V6.50	V6.40	V6.30	and earlier
Windows/Mac/LINUX	US\$240	365	465	545	605	+50/each
for 1-user floating	US\$288	438	558	654	726	+60/each
UNIX for 1-fixed license	US\$350	550	700	800	850	+50/each
for 1-user floating	US\$420	660	840	960	1020	+60/each

Installed Sizes.

Loading **TNTview RV6.8** processes onto your hard drive (exclusive of any other products, data sets, illustrations, documentation files, ...) requires the following storage space in megabytes.

	<u>for V6.70</u>	<u>for RV6.8</u>
PC using W95, W98, WME, NT, W2000, or XP	54 Mb	154* Mb
PC using LINUX (with Intel) kernel 2.0.36 - 2.4	36 Mb	90 Mb
Mac using Mac OS X 10.x	132* Mb	155* Mb
SGI workstation via IRIX	42 Mb	60 Mb
Sun workstation via Solaris 2.x	40 Mb	82 Mb
IBM workstation via AIX 4.x (with PPC)	44 Mb	100 Mb

* The installed size on these platforms is the same for **TNTview**, **TNTedit**, and **TNTmips**.

TNTedit™ RV6.8**Oracle Spatial Import and Export.**

Spatial layers in Oracle Spatial and Oracle Locator can now be imported, edited, and then exported as new layers. Now **TNTedit** can be used as a tool to update or prepare new data for use in these spatial components of Oracle. Oracle does not provide tools

for this. While Oracle, through other products, can ingest CAD-oriented materials into its spatial layers, it does not directly create or use topology in its analysis functions.

A very logical application of **TNTedit** would be to keep the spatial geodata in Oracle current. First the spatial layer is imported. Next it is overlaid on an orthoimage (aerial or satellite derived). Correction, repositions, and additions are made from interpreting the image including updating the imported attributes. The revised object is then exported into a new spatial layer in Oracle.

The details on these new import and export processes are in the corresponding TNTmips New Features section below and in 6 color plates attached to this MEMO.

Periodic Automatic Backups.

In any computer activity, it is important to keep frequent backups. No matter how many times backups are made in any product either a change of mind, human error, or software error will require the use of an earlier backup. However, the frequency with which you make backups depends to a large part upon how fast and innocuously they can be made. **TNTedit** and **TNTmips** now provide for fast backup for the layer being edited under a variety of automated conditions. Fast being a qualitative term here depending upon the size of the object and speed of your computer's memory, bus, and hard drive.

Setting Preferences.

Automatic backups can now be programmed for your Spatial Data Editor operations. A new Backup tabbed panel has been added to the Setup/Preferences window. It provides the means to set if, how, and when these backups will occur and where they will be recorded. This new panel is illustrated in the attached color plate entitled Automatic Backup Options When Editing.

Fixed Time Intervals. The new panel provides a toggle to enable/disable automatic backups. It displays the path to your current backup folder and provides a means to specify or change it. The time interval between backups in minutes can be set and a toggle determines if you are asked if you want to permit the backup at this time or have it automatically completed.

Idle Time Interval.

As an alternative and in addition to being interrupted at the fixed time interval, you can specify how long the Editor is idle in seconds before it performs an automatic backup. A judicious selection of this time will provide you with automatic backups when you leave your desk or temporarily turn to some other task. If this idle time backup is performed, the fixed interval backup timer is reset. As a result, you can use these 2 timers together to fit your habits and get your backups made with a minimum of interference with your work in the Editor. A backup is not created if there have been no changes since your last backup.

On Demand.

You can backup any time you like by using the new File/Backup Objects menu option. This immediately creates a new backup object(s). When your objective is to temporarily preserve your work, this backup operation is automatic and faster (see below) than the Save or Save As operation you have been using. This operation is illustrated in the attached partial color plate entitled Editable Object Backup on Demand. This option was added after the Release Version (**RV6.8**) CDs were reproduced. Thus, you will need to

acquire the most recent patch to **RV6.8**, which will be available by the time you read this.

Number of Backups.

You can specify how many backups of the specific object you are editing will be kept in your backup folder. When this count is exceeded, the oldest copy of that object is deleted. You may be working on multiple objects and separate backups of each are made in your backup folder. As a result, you can set the maximum number of files that can be in your backup folder. When this number is exceeded the oldest file in the folder will be purged. You can not specify more than 1 backup folder since it will be automatically used in several operations that will restore your backup. Set the number of backup files you allow to a large number and provide lots of drive space. This is even a good place to use a second drive or a removable drive.

Drive space is cheap compared with your time. Keep a lot of backups around!

Using Backup Objects.

How are Backup Objects Named?

During your editing session the object you are editing is an object in a temporary workspace. When an automatic backup is made, the contents of this workspace are all saved as a backup object in a new Project File. The name of the Project File in the backup folder containing your backup object is the name of the object you are currently editing followed by the date and time the backup was performed. This is illustrated in the attached color plate entitled Automatic Backup Options When Editing.

Why is it Faster?

The purpose of a backup object is to permit you to restore your editing operation to the workspace to match its condition at the time of the backup so you can resume editing from that point. As a result, this automatic backup is faster than when you save your editing results into a Project File. It uses the operating systems fastest copy to simply mass move the contents of the workspace to the backup object along with undo/redo options and so on to another location on the drive. Topology, search trees, vector optimization, or other final operations are not performed as the purpose of a backup object is to restore it for editing. Your backups will be even faster if the drive used is not the same drive as used for your temporary files and, thus, the workspace. Synchronous fast copy between drives may be faster than asynchronous drive read and writes on the same drive.

Restoring a Backup.

When you start up the Editor, you have the new option File/Open Backup. This choice will automatically open a Select Object window showing all your backup files for immediate selection. You can then select any backup object from the files in your backup folder. Fast copy will move that object to a temporary file and you can resume editing it where you left off. If by some rare chance the Editor should abnormally exit or otherwise crash, you will be asked if you want to load the latest backup files for editing when you restart.

Isolating a Possible Error Condition.

If you are experiencing error in editing a layer, you can use this backup procedure to assist MicrolImages in reproducing this error, which we must do before we can fix it. The Spatial Data Editor is a very complex process often acting on a very complex object. Furthermore, as part of your editing this object, you can perform a complex, difficult, and hard to repeat sequence of interactive operations. Thus, it is often difficult for you to determine what sequence of your operations results in an error. When this occurs, MicroImages often can not reproduce the error and, thus, can not fix it. By backing up very frequently you can iterate back to a prior backup object. By repeating this process, you may be able to isolate the sequence of edit operations that causes the fault to occur. You can then provide MicrolImages with the last useful backup and a description of the steps that cause it to fail. MicrolImages can then load your backup object, repeat the steps, get the error, and fix it for the next patch.

Interoperation of Tools.

You can now switch between drawing tools and element types without losing any of your previous incomplete work in any tool. When you return to that tool, you will find it in exactly the incomplete state you left it in. If you are using a tool, such as drawing a line, you can now start another tool and later switch back and resume drawing the line where you left off. The attached partial color plate entitled Interoperate Tools When Editing illustrates a very simple operation where, during the tracing of a line, the view is zoomed without losing the incomplete line. This new capability is simple to explain, but a big time saver. It permits you to move between tools to complete a complex operation requiring several tools. For example, you can draw a line, switch to drawing a polygon, draw part of a polygon, switch back to resume drawing more of the line, and so on.

Miscellaneous.

Auto-Adjustment of Extents.

If the extent of a vector, CAD, or TIN object is reset when saved because you have made an edit alteration to reduce its extent (for example, you remove a stray outlying point element whose inclusion causes inflated extents in a view), the save action will also cause a redraw using these new extents. Optionally you can shut this off in your preferences so that the saved extent is not reflected on the screen, and there is no automatic redraw.

BSplining Lines In XYZ.

A 3D line in a vector layer can now be selected and splined using a Cubic BSplining or Quadratic BSplining. The line will be splined in the XY plane and new values will be interpolated for the Z coordinates of the new vertices created by the splining. The effect of this kind of splining is illustrated in the attached partial color plate entitled Spline Lines in 3D.

Filtering Vectors.

The Spatial Data Editor can perform the same new filter operations described below for the separate Vector Filter process in **TNTmips**. Zoom and pan operations are now available when using the test feature of vector filter routines.

Object Properties.

All three coordinate types (2D-XY, 3D-XY, and 3D-XYZ) can now be set in the Object Properties dialog on the Coordinate Type option menu. 3D-XY is for vectors that have lines with only a single Z value stored as an attribute, such as contour lines.

Inherited New Features.

The following general improvements in all **TNT** product operations are automatically available in **TNTedit RV6.8**. These improvements are detailed below in the major section on New Features for **TNTmips** and include:

- use of custom color palettes in all display processes,
- much faster display of linked JP2 image files,
- improved use of multiple views and geolocking and a new view locator tool,
- improved quality and faster 3D displays,
- improvements in style assignment and new style and line pattern editors,
- use of new high precision map datums and ellipsoids in rendering layers,
- faster transfer of attributes,
- filter vector objects using a script,
- WYSIWYG entry of text styles,
- new Unicode font display window with interactive selection of glyphs with no or difficult keyboard access in a text string into the Text Layer Controls,
- faster, more interactive group placement procedure for use in layouts,
- convert hatch patterns used in a map layout to SVG, PDF, or AI files,
- **SML** dialogs can be designed with simpler XML documents,
- communicate with other programs via **SML** using ActiveX.

Upgrading TNTedit.

If you did not purchase **RV6.8** of **TNTedit** in advance, and wish to do so now, please contact MicroImages by FAX, phone, or email to arrange to purchase this version. When you have completed your purchase, you will be provided an authorization code by FAX. Entering this authorization code the first time you run your product completes the activation of **TNTedit RV6.8**.

The prices for upgrading from earlier versions of **TNTedit** are outlined below. Please remember that new features have been added to **TNTedit** with each new release. Thus, the older your version of **TNTedit** relative to **RV6.8**, the higher your upgrade cost will be.

Within the NAFTA point-of-use area (Canada, U.S., and Mexico) and with shipping by ground delivery. (+\$50/each means US\$50 for each additional upgrade increment.)

TNTedit Product	Price to upgrade from TNTedit:					V6.20
	V6.70	V6.60	V6.50	V6.40	V6.30	and earlier
Windows/Mac/LINUX	US\$350	550	700	800	875	+50/each
for 1-user floating	US\$420	660	840	960	1050	+60/each
UNIX for 1-fixed license	US\$650	1000	1350	1600	1750	+50/each
for 1-user floating	US\$780	1200	1620	1920	2100	+60/each

For a point-of-use in all other nations with shipping by air express. (+\$50/each means US\$50 for each additional upgrade increment.)

TNTedit Product	Price to upgrade from TNTedit:					V6.20
	V6.70	V6.60	V6.50	V6.40	V6.30	and earlier

Windows/Mac/LINUX	US\$500	750	950	1100	1200	+50/each
for 1-user floating	US\$600	900	1140	1320	1440	+60/each
UNIX for 1-fixed license	US\$750	1200	1550	1850	2000	+50/each
for 1-user floating	US\$900	1440	1860	2220	2400	+60/each

Installed Sizes.

Loading **TNTedit RV6.8** processes onto your hard drive (exclusive of any other products, datasets, illustrations, Word files, and so on) requires the following storage space in megabytes.

	for V6.70	for RV6.8
PC using W95, W98, WME, NT, W2000, or XP	70 Mb	154* Mb
PC using LINUX (with Intel) kernel 2.0.36 to 2.4	67 Mb	121 Mb
Mac OS X 10.x	132* Mb	155* Mb
SGI workstation via IRIX	86 Mb	95 Mb
Sun workstation via Solaris 2.x	72 Mb	118 Mb
IBM workstation via AIX 4.x (with PPC)	92 Mb	137 Mb

* The installed size on these platforms is the same for **TNTview**, **TNTedit**, and **TNTmips**.

Tutorial and Reference Booklets

There are now 71 **TNT** Tutorial and Reference booklets (formerly called Getting Started Booklets). These booklets provide more than 1700 pages and over 3900 color illustrations. Some are up-to-date with the features in **RV6.8** of the **TNT** products, some are not. Each new professional **TNTmips** ships with 3 thick notebooks containing a color printed copy of every booklet. Those of you receiving your **RV6.8** upgrade on CD can view and refer to any booklet using Adobe Acrobat Reader. If you install all these booklets as part of any **TNTmips** product, you can directly access these booklets via Help / Tutorial Overview.

New Booklets Available.

One new tutorial booklet entitled Making Image Maps is being released for the first time with **TNT RV6.8**.

Expanded Booklets.

Three **TNT** booklets have had significant upgrades for use with **RV6.8** of the **TNT** products: Vector Analysis Operations, Sketching and Measuring, and Managing Relational Databases.

Translated Booklets.

Tutorial booklets in most cases are being translated in a priority order. Two booklets, Displaying Geospatial Data and Navigating, are the first two translated. Turkish and Thai have their own books which have been adapted from the tutorials and printed in these languages. Shown below is the number of translated booklets in your language and the total number that have been committed.

Arabic (0 of 6)	Dutch (8 of 8)	Italian (6 of 7)	Finnish (1 of 1)
Korean (18 of 38)	French (6 of 16)	German (10 of 10)	Japanese (14 of 52)
Russian (5 of 13)	Serbian (9 of 24)	Greek (3 of 6)	Spanish (50 of 62)

The covers of some of these booklets in these languages are illustrated in the attached color plate entitled More Translated Documentation.

Tutorial Revision Plans.

The tutorial booklet entitled Writing Scripts with **SML** is at this moment being revised and divided into 2 tutorial booklets. One new booklet will be devoted to how to create user interface components in **SML** and the other will deal with how to implement tools and processes in **SML**. The tutorials pertaining to **TNTsim3D** are scheduled to be upgraded next. Watch for these improvements and others as well as new booklets at microimages.com.

New TNTmips Features

Main or subsections preceded by the asterisk "*" symbol introduce significant new processes or features in existing processes released for the first time in **TNTmips RV6.8**.

System Level Changes.

Custom Color Palettes.

You can now design, save, and reuse your own special color palettes of any number of custom colors to choose anywhere you can select a color palette, such as to style your vector elements, create a new raster, or even for your View window background color. Your color palette can be used anywhere in a **TNT** product that you need to select and assign a color from a palette. The attached color plate entitled Add Vector Color Palettes via XML illustrates their use in the Style Editor. Color palettes can be selected from many different places in **TNTmips** and not all have been updated to allow selection of XML color palettes. Look for color selection windows that have a Palette tabbed panel or a Palette icon to find this new feature.

Color elements for your custom palettes are specified in XML. This same color plate illustrates the form of the XML file that defines your color palette. All you need to create a special color palette is a text editor and the color values for each of your colors. Once you have created an XML color palette, you simply place it in the palettes subdirectory in the directory where your **TNT** product is installed. It will then automatically appear on the list in the Select Palette window presented wherever you can select a palette for use.

Each palette you create can be given its own descriptive name and color array layout. Each color can be named and defined in more than one color system (for example, in RGB and CMYK). The components of the colors in a your palette can be defined in several different data ranges such as 8 bits each for RGB color, 16 bits each for RGB, hexadecimal for RGB, 8 bits each for CMYK, and so on.

Sample Color Palettes.

The following XML color palette files are copied into your palettes subdirectory when you install an **RV6.8 TNT** product. You need to obtain **PV6.8** to have access to these palettes. You can open any of these files in a text editor to inspect its simple XML setup. The MicroImages palette is used by default and provides the same colors as in previous versions of the **TNT** products.

- MicroImages (Default)
- USGS CMY
- U.S. Geological Survey Color Chart

- U.S. Bureau of Land Management (BLM)
- Web (Named Colors)
- Land Based Classification Standards (LBCS) Structure
- Land Based Classification Standards (LBCS) Site
- Land Based Classification Standards (LBCS) Ownership
- Land Based Classification Standards (LBCS) Function
- Land Based Classification Standards (LBCS) Activity
- Canadian Land Inventory Map Series - Ungulates
- Canadian Land Inventory Map Series - Recreation
- Canadian Land Inventory Map Series - General Land Use
- Canadian Land Inventory Map Series - Water Fowl
- Canadian Land Inventory Map Series - Sport Fishing
- Canadian Land Inventory Map Series - Forestry
- Canadian Land Inventory Map Series - Agriculture

Raster Histograms.

When you are viewing a sampled histogram you now have the option to compute and view an unsampled histogram. Choose Recompute from the File menu in the Raster Histogram window.

Histograms computed for floating point raster objects now automatically revise their limiting values to eliminate outlying cell values. Previously, a single outlier high value and/or a single extreme low value would cause Display to produce an abnormally high contrast view. This change reduces the problem of having your floating point data displayed as all light and dark values instead of appearing continuous.

The histogram of 16-bit rasters now uses, as needed, up to 65536 histogram bins.

TINs and Breaklines.

Several improvements have been made to the internal procedures that build and rebuild TIN objects. These changes ensure that Delaunay triangles are preserved for unusual input data structures (for example, many points very close together in a line). In turn, this also ensures that breaklines inserted into the TIN are properly preserved in this and subsequent processing.

Windows and X Desktop.

The Screen tabbed panel of the MicroImages X Server Preferences dialog now provides a button to toggle your user interface back and forth between the X or Windows Desktop mode at your next startup. A button is also available with the **TNT** General System Preferences to present or hide a dialog that prompts you for your desired Desktop mode each time you startup your **TNT** product.

Now if you move the cursor onto a window in Windows Desktop mode, it will automatically activate and come to the foreground for immediate use if you have turned on the *Auto-raise window on mouse focus* option in your X Server Preferences.

KOI-8 text encoding is now supported for additional support of Cyrillic fonts.

Switching Languages.

The default **TNT** product installation will now install all language resource files. Simply select your language in the **TNT** products using the Locale tabbed panel of the General System Preferences window (Support / Setup Preferences).

Miscellaneous.

The symbols for nautical mile and knots have been changed to nmi and kn.

The length permitted for group names has been increased from 16 to 64 characters.

When a group or layout is saved for the first time, the name of the group or layout is used as the default object name.

A Design Scale can now be set for 2D groups for element and style scaling. This is also usable in the Spatial Data Editor so that elements may be styled as they will appear in a subsequent layout composition.

2D Geospatial Display.

Multiple Views.

GeoLocking.

Multiple 2D and 3D views can be opened in all the **TNT** products. If 2D views contain geodata that covers a common ground area (which means, overlapping extents), they can be GeoLocked. **V6.70** kept the scales of all these views the same if scale locking was on. Changing the scale of one view caused the other locked views to change to the same scale and redraw. **RV6.8** provides 3 choices for how GeoLocked windows should relate to each other: Scale Only, Extent Only, and Extent and Scale. Locking by Extent and Scale is the default method of maintaining a common scale and area for all GeoLocked views. Change the scale in 1 view, and all other views will automatically redraw to the same scale. The attached color plate entitled GeoLock with Relative Zoom illustrates uses of these new lock relationships. These new combinations of Geo-Locks can be preset in a screen layout or in the startup layout of **TNTatlas**. For example, an atlas can be automatically opened with GeoLocked side-to-side View windows showing a reference map and an associated image or in some other arrangement using more than 2 windows.

By Extent Only.

Locking a view by Extent Only means that it may not automatically redraw if there are scale and position changes in other GeoLocked views. Since scale locking is off, the relative scale value for this view is ignored, and this view will only change scale using its own zoom in/out controls. Only other views that are also extent locked can cause this Extent Only view to change. This happens only if the smaller extent box of another zoomed in, extent-locked view moves outside the larger extent box of this Extent Only locked view. If this happens, this Extent Only locked view will redraw to center on the smaller extent box for the other zoomed in extent locked view. Another extent locked view can be zoomed out so that it has an extent box that is larger than the Extent Only locked view. In this case, the Extent Only view will not move, change, or redraw when the extent box of that larger view is moved about.

Extent Only locking is useful to create a reference view presenting a macro view of a general area topographic or planimetric map or a zoomed out view of an image. Other extent locked Views can be opened that are zoomed in on the image. They can then be zoomed or panned using the Extent Only locked view for reference and it will only redraw if these detailed views move off the area of the current Extent Only locked reference view.

By Scale Only.

Locking a view by Scale Only means it will automatically redraw to maintain its scale when some other scale locked view is zoomed in or out. Now you can also set a Relative Scale value of greater or less than 1.000 for each scale locked view. This value sets the relative scale to be maintained by this view relative to other scale locked views. Setting scale locking on for two or more views, along with appropriate relative zoom factors, means that they zoom in and out in tandem at these relative scales. It is also important to remember that you can set a display scale range for every object. This scale range determines the display scale at which the object will appear and disappear. Thus, a single GeoLocked view can also add more detailed vector or image layers according to their scale ranges as it is zoomed in as a reference or other scale locked view.

By Extent and Scale.

A view GeoLocked with this setting is both scale and extent locked and will redraw for any scale change in another scale locked view or appropriate location changes in an extent locked view. This is illustrated at the bottom of the attached color plate entitled GeoLock with Relative Zoom.

Multi-View Locator Tool.

This new tool permits you to visualize and adjust the positional relationships of several 2D views opened from the same group or layout. If it is turned on for a view, it shows the rectangular extent boxes on that view of all the other open views. This is illustrated in the attached color plate entitled Multi-View Locator Tool. If some other view has an extent that is larger, it will not show and might be a better choice for opening this tool. If any other view is adjusted in size or scale, its extent box will resize in this tool. You can use the mouse to reposition any of these extent rectangles. After moving the extent box on the view, pushing the right mouse button will cause the associated other view to redraw to conform to these changes. (Remember that on the Mac, the Command key with mouse button is the same as a right button in Windows.) Many complex and useful interactions can take place in the use of this new tool especially if the various views are each GeoLocked with one of the new options above. Several of these are illustrated and discussed in more detail in the attached color plate entitled Controlling the Multi-View Locator.

* Much Faster JPEG2000 Displays.

All **TNT** products can now directly display large JP2 image layers 5 to 10 times faster. This results from an optimization of the buffering scheme used during the decompression of the JP2 files and increasing their minimum tile size to 256 by 256 bytes. With these modifications, the direct display or other use of the linked JP2 file (see TNTsim3D for Windows above) is comparable in speed to the use of a tiled and pyramided raster object. This is particularly significant as now a **TNTatlas** can directly benefit from the reduction in size of images using linked JP2 files. Comparisons of the improvement in displaying a color image are as follows.

Test computer is a Pentium 3 of 650 MHz running W2000.

JP2 test raster is an image of 18648 by 35283 pixels.

10:1 compression (from ~1 GB to 100 MB)		[from hard drive]	[from 10X CD]
Using V6.70	Full View	241 seconds	244 sec.
	1X Zoom	36 sec.	40 sec.
Now RV6.8	Full View	4 sec.	4 sec.

100:1 compression (from ~1 GB to 10 MB)	Using V6.70	1X Zoom	3 sec.	3 sec.
		Full View	[from hard drive]	[from 10X CD]
Now RV6.8		1X Zoom	43 seconds	43 sec.
		Full View	17 sec.	17 sec.
		1X Zoom	4 sec.	4 sec.
		Full View	2 sec.	2 sec.

Note from these numbers that when the compression ratio is high, the slower read rate of a CD, compared to that of a hard drive, is effectively negated. For the 100:1 case, only 1/100 of the time is expended in reading from a CD, and the few seconds needed to display the layer are primarily for the computation of the decompression. Thus, if you have a 2.5 GHz processor to do the decompression (not the 650 MHz processor used in these tests), using JP2 files directly from CD or DVD can be very efficient.

Consider the very significant result this can have on your future publication of **TNTatlases**. Assume you are writing the **TNTAtlas** onto a 4.7 Gb DVD-R drive, which is about 7 times the capacity of the CD-Rs you have been using. Assume that you find 100:1 compression is acceptable for the image layers in the atlas. Assume that you have not been using any compression in the images in the raster objects on your CD-based atlases. Under these most optimistic new conditions, you could place 700 times more image pixels into an atlas, and yet it may be even faster to use from the DVD (depends on your **TNTAtlas** users' hardware).

Now the **TNTAtlas** you could give away on \$2 media represents the equivalent of .5 terabytes of uncompressed images. A couple of years ago none of us even thought in terms of terabytes! This seems like a really large performance jump, but with 1-meter satellite images and 4 to 6 megabyte individual digital camera snapshots, it will not take long to use up this new capability. And those of you who have been making Landscape Files already realize that they produce even bigger storage requirements if you do not want to quickly "fly off the edge of your landscape."

Labels and Leader Lines.

Label styling will now default to "by element" if not previously set for the vector object. This makes styling of automatically generated labels match the way they were created.

Leader lines can now be omitted as an option for on-the-fly labels when their automatic placement positions them outside their associated polygon. This should be used with caution as it will put the labels in some other polygon and no leader will appear to associate the label with its original polygon. This option is the Fit Inside or Outside without Leader choice on the Label Position option menu. The other options for on-the-fly label placement are Always Inside, Fit Inside or None, Fit Inside or Outside with Leader.

3D Geospatial Display.

Introduction.

The 3D display process is being completely overhauled and the underlying code redesigned without significant modifications to the user interface. The first of these improvements is available to you in **RV6.8**. They increase the quality of a direct 3D view by providing much better foreground to background smoothing using computer game concepts for mapping a texture (which means, raster) onto a surface (which means, DEM). In addition to modifications to the Ray Casting and Triangulation methods (now called Dense Triangulation), a third new and faster surface model is also provided called Sparse Triangulation.

Texture Filters.

Background.

Those involved in image processing are quite familiar with the use and limitations of the nearest neighbor sampling and bilinear interpolation methods used in cell sampling operations on a raster. These sampling schemes have been in common use on personal computers for image processing and raster-oriented GIS tasks for years. Nearest neighbor sampling is fast but has many detrimental effects (for example, skipping cells, doubling cells leading to bad aliasing, ...). Bilinear interpolation overcomes these limitations to some extent without dramatically increasing computation times. For example, bilinear interpolation is automatically applied to create pyramid layers in any raster object you link to, import, or create in a **TNT** process. (Some linked raster formats of others, such as ECW and JP2, already provide suitable pyramid layers that are simply used through the link.) Nearest neighbor sampling is still available for pyramiding and other resampling operations on categorical raster objects where any kind of interpolation or averaging would be incorrect.

Improved central processor speeds and advances in display board processors, coupled with the automatic availability of pyramid layers in a raster object, now permit the incorporation of more complex, robust interpolation schemes into the **TNT** processes. Four new interpolation methods are provided in **RV6.8** on your Texture Filter selection list for a 3D display activity: Upper MipMap Nearest Neighbor, Lower MipMap Bilinear, MipMap Trilinear, and MipMap Anisotropic. The objectives of these new methods are to improve foreground smoothing, reduce the dropout effect on linear features in distant areas and corners, and reduce aliasing of linear features, all without requiring special hardware filtering by the display board.

These newer schemes are particularly important in a 3D view where the resolution of screen pixels varies from the foreground to the background. Thus, in a 3D view, a pixel element on the screen covers a small ground area in the foreground but a much larger area in the background. In the foreground, many pixels (screen elements) can fall inside a single raster cell, causing blockiness. In the background many ground cells can map to a single pixel, causing dropouts that are particularly evident in linear features. Applying nearest neighbor sampling and bilinear interpolation to the appropriate pyramid layer can reduce these artifacts. These two methods of controlling sampling in a 3D view are now available through the new Upper MipMap Nearest Neighbor and Lower MipMap Bilinear options. While these are not the most advanced methods, there are special circumstances in which you may wish to apply them.

MipMap Trilinear and MipMap Anisotropic interpolation are also provided for your 3D view and produce the best smooth rendering from the foreground to the background. These methods combine the ground cell values from bracketing pyramid layers for each screen pixel in the view to reduce foreground blockiness, eliminate abrupt shifts in image resolution from foreground to background (zoning), and minimize the impact of dropouts in the background areas. You can review how much better they meet the goals compared to nearest neighbor sampling in the 3D views in the attached color plate entitled Texture Filters for 3D Rendering.

Terminology.

Moving between different technological domains (such as from GIS to image processing to the game industry to solid modeling) can be confusing as different terms are used to convey similar concepts. At the moment smooth, fast, 3D rendering concepts are being

developed by display board manufacturers in response to the computer game industry. Adding their concepts to the 3D view and **TNTsim3D** is introducing some possibly confusing changes into the **TNT** terminology.

TNT pyramid layers, or tiers, have cells of lower resolution (which means, larger ground area) computed from the next higher resolution pyramid layer using bilinear interpolation. These pyramid layers increase in cell size (and, thus, decrease in ground resolution) in a progression by powers of 2. For example, if the original raster's ground resolution is 1 meter, then the complete pyramid structure includes layers of 1, 2, 4, 8 meters, and so on.

The pyramid layers of your raster object in a **TNT** Project File are called tiers in the game board industry. The rendering of the TIN or DEM is called a surface. The raster layers, or tiers, draped on the surface are called textures. Mipmapping is a game board industry term for using the appropriate tier(s), or pyramid layer(s), to derive the RGB texture value (color) of a screen pixel based on its distance to the surface being rendered in the 3D view.

How is Mipmapping Used?

Mipmapping interpolation is computed by your main processor for **TNT** 3D views. Its smooth foregrounds and smooth transitions from foreground to background in 3D views make it well worth its much more complex computations when compared to nearest neighbor or bilinear interpolation schemes. This slowdown can be compensated for by using the new, faster Sparse Triangulation method discussed below for computing the surface, or terrain.

In **TNTsim3D** various forms of mipmapping can also be used and at a much higher rate if your graphics processor makes them available via DirectX or OpenGL. However, at the high frame rates used in games and **TNTsim3D**, one type of scene distortion still remains in the form of aliasing of linear features and edges. The preeminent graphic chip manufacturers (nVIDIA, ATI, and Matrox) are battling it out over whose chip does this the best without a loss of performance. Ask any group of PC gamers and you can get an argument going over whether to turn on these kinds of hardware-implemented mipmapping and antialiasing interpolation/smoothing effects and lose frame rate or accept some aliasing and blockiness and gain speed. However, the very latest ATI 9800 graphics chip appears to have overcome this limitation, as it has been tested and shows no significant differences in its performance with and without the use of its hardware implemented mipmapping and antialiasing options. What is significant about all of this? Now that these procedures are part of the rendering of the 3D view, methods eventually can be added to optionally use these hardware-implemented functions via DirectX and OpenGL to provide much faster 3D views in the **TNT** products.

Upper MipMap Nearest Neighbor.

This method produces views of medium quality but is faster than the best methods. It computes the color of each screen pixel in the view by first determining the projection of that pixel on the surface and its resulting ground resolution. It then uses nearest neighbor resampling to obtain the color for that pixel from the pyramid tier that has the closest, but lower, ground cell resolution. Thus, the resolution varies from front to back in the view. However, you can detect abrupt transitions in resolution from front to back as the increasing distance to the terrain causes the tier used to change. These tier transition effects happen along more or less concave arc boundaries (in a perspective view)

view) or lines (in a parallel view) from near to far. Games played on display boards without advanced graphics processors often show this front-to-back zoning effect.

Lower MipMap Bilinear.

This method produces views that are slightly better than the Upper MipMap method, though it is also slightly slower. It finds the tier of just higher resolution than the projected pixel and then gets the needed pixel color from it by bilinear interpolation. It, thus, deals with 4 times as many cells as the Upper MipMap method and performs more computation in doing the interpolation. In general, the visual transitions in the view are similar to those for the Upper MipMap method except, since the higher resolution tiers are used, the lines of transition are pushed back somewhat toward the background of the view.

MipMap Trilinear.

Trilinear mipmapping produces good quality at a slower rendering rate. The color values for the screen pixel are determined by the two tiers whose resolutions bracket the longest dimension of this pixel projected onto the surface. The determination of the exact cell to use from each of the bracketing tiers is hard to explain in text but is illustrated in the attached color plate entitled Texture Filters for 3D Rendering. The two bracketing cell values are then combined by linear interpolation based upon their ground sizes as compared to the projected screen pixel. Since this interpolation varies continuously as the projected pixel size varies from the foreground to the background in the view, no abrupt resolution transitions are evident. This procedure would be slow in a real-time simulation computer game using any display board that did not perform these calculations in its graphics processor.

MipMap Anisotropic.

Anisotropic mipmapping produces the best quality at the slowest rendering rate. As in the Trilinear method, the color values for the screen pixel are determined by the two tiers whose resolutions bracket the resolution of this pixel projected onto the surface. However, in the Anisotropic method, the shortest rather than the longest dimension of the projected pixel determines its ground resolution, so this method generally selects a higher-resolution pair of tiers for a given screen pixel than the Trilinear method. The determination of the cells used from each of the bracketing tiers is hard to explain in text but is illustrated in the attached color plate entitled Texture Filters for 3D Rendering. The term "anisotropic" is used because multiple cells are taken from the bracketing tiers along the line of view, and their number and treatment can vary for each pixel in every view. The cell values selected from each tier are averaged and the final value for the screen pixel is then computed by linear interpolation as in the Trilinear method. This method provides a parameter called the Anisotropic Limit that controls how many cells can be used from each tier per screen pixel. Increasing this limit can produce cleaner rendering in the background but increases rendering time, as it increases the number of cells that have to be processed from each tier.

Comparing Best Filters.

At first appearance the results of these last 2 methods may appear similar. Both produce good smoothing of the foreground cells for the area where the screen pixels are much smaller than the texture's cells. The texture cells don't show up as blocky, solid-color shapes but instead blend smoothly together. This means that, for a given texture resolution, views that are close to the ground look better using these methods than with

Upper MipMap Nearest Neighbor method. Note that these same improvements will show up in any movies you make, since these same features are available when 3D views are rendered to movie frames, but recording the movie will take longer.

The big difference is in the background. In the Trilinear method, the nice smooth foreground effect is traded off against more distortion in the distant features. Note in the illustration on the color plate in the corner that is farthest into the background, linear features are distorted by being smeared or elongated from side to side. This is because the horizontal dimensions of the cells used from the bracketing tiers are large when compared to the width of the screen pixel projected onto the surface (see plate). The Anisotropic method does a better job of matching the sampling to the decreasing width of the projected screen pixel with distance and does not produce these distortions (see plate). However, you are still left with some dropouts for linear features in the deepest background corner area. This remaining artifact is the subject of current investigation in the game and simulation industry leading to even more exotic proposed methods and titles, such as EWA Splatting and others.

Surface or Terrain Models.

Background.

In a 3D view, a texture is draped over a surface model generated from your DEM using one of three methods. The Ray Casting method that has been available in 3D display directly uses your DEM as the surface. It provides for high detail in the surface topography but can be slow for large, high resolution views, and it cannot be used at high view angles (>75 degrees). The Dense Triangulation and new Sparse Triangulation methods convert the DEM to a regular triangular mesh automatically as part of the process. The Dense Triangulation method, which has been available in previous versions, uses all cells in the DEM to produce a highly detailed surface model, but it is slow, especially for large, high-resolution scenes. The new Sparse Triangulation method added in **RV6.8** achieves its speed by limiting the number of triangles used for the surface to 20,000, regardless of the size and detail of the DEM. The attached color plate entitled Faster/Better 3D Visualization compares the appearance of the same view rendered in each of these 3 methods.

The Need for Speed.

Faster rendering of a 3D view has many obvious potential benefits. Short range benefits include better simultaneous use of 2D and 3D views of the same geodata, less delay in rendering layouts that contain 3D groups, and so on. One immediate need for faster rendering is in the Landscape Builder process. As the Landscape Builder evolves, it can serve as a bridge between what is presented in a static 3D view and the real-time dynamic view of the geodata in **TNTsim3D**. Someday these separate concepts will merge, but this will require major improvements in processor and display board performance. Although the Landscape Builder currently uses only a 2D view, a realistic goal is to provide a 3D view that will show a reasonably accurate snapshot of **TNTsim3D** frames as you change the viewpoint, providing a preview of your geosimulation. Subsequently, in a future release, **TNTsim3D** could provide reverse benefits by providing the 3D display process with the parameters needed to make a high quality, 3D poster-like view or print raster of the current geosim frame.

3D display, like 2D display, works directly with your native geodata objects just as with any other **TNT** process (providing automatic data type conversion, sampling, reprojec-

tion, and so on). As such, it burns a lot of computation cycles. In order for **TNTsim3D** to provide you with a real-time simulation (at a realistic frame rate), it requires that you preprocess your objects into new objects optimized to require less computation (for example, a common projection, certain restrictions on object extents and sampling, compression, and so on). This conversion process in the Landscape Builder takes time if you are preparing a large Landscape File, such as for a DVD. A faster 3D view makes it practical to preview and adjust your design before all this optimization and object conversion takes place. A faster 3D redraw also permits changing your viewpoint with rapid rerendering (2 to 3 seconds) to see how the design looks from various viewpoints.

Sparse Triangulation.

Both Dense and Sparse Triangulation methods create a triangular mesh from the DEM selected for the surface. Since the Dense Triangulation method uses every DEM cell in the triangulation, the highly detailed surface model is too large to be created just once and stored in memory, but must be recomputed for each redraw, change of position, or change in orientation of the 3D view. As noted above, the Sparse Triangulation method limits the number of triangles to 20,000, providing a surface model that is compact enough to be retained in memory so it does not need to be recomputed for each redraw of the 3D view. Sparse Triangulation is therefore much faster at rendering a surface than the Dense Triangulation method, but has some limitation in the level of surface or topographic detail presented in the view. It is planned that a future release of the 3D display process will merge the Sparse and Dense Triangulation methods into what might be called variable-density triangulation to overcome these remaining limitations while still providing fast rendering.

Combining the new, faster Sparse Triangulation method with the new but slower advanced mipmapping interpolation methods provides a good 3D view at all view angles. A scene renders at about the same rate as using the Ray Casting surface method in **V6.70**, which was the fastest approach available to you, but with view angle limitations. If you select one of the simpler methods for draping your texture layer onto the Sparse Triangulation model, you can keep your 3D view in Solid rendering mode (which draws the texture) while moving your viewpoint around and still get an acceptable redraw rate. Previous speed limitations in texture rendering might have limited you to Wireframe mode for previewing movement through the scene.

Improved Polygon Filling.

A more precise polygon fill algorithm has been added for use in rendering extruded polygons in 3D display. In particular, it provides more accurate results when used in the Ray Casting mode.

Patching.

Development has continued on the 3D display process since the Release Version (**RV6.8**) was finalized. New features have been added that will appear in the first patch provided for **RV6.8**. This patch will be posted at microimages.com by the time you receive this MEMO. The new features include the following:

The Sparse Triangulation surface model now provides a choice between perspective and parallel projection, as for the other surface models, rather than just perspective projection. A MipMap Sharpness parameter is now available for use with the MipMapTrilinear, Lower MipMap Bilinear, and Upper MipMap Nearest Neighbor texture filters. The MipMap Sharpness value is a percentage that lets you vary the overall rendering of the

texture from smoother, less detailed (low values) to sharper and more detailed (high values). It does so by adjusting the procedure used to identify which pyramid tier(s) to use for rendering each screen pixel based on its distance from the surface. Higher MipMap Sharpness values adjust the procedure to select lower-level, higher-resolution pyramid tiers for a given screen-pixel distance, so that transitions to less-detailed pyramid tiers occur further back in the scene. The MipMap Sharpness parameter is not available or needed for the MipMap Anisotropic method because this method's design already incorporates the use of more detailed pyramid tiers compared to the other Mip-Map methods.

It is not anticipated that any other new 3D display features will be added to the patches for **RV6.8**. Major changes are anticipated as work continues, but these will appear in the Development Version (**DV6.9**).

Management of Vector Styles.

Microlmages continues to redesign and recode portions of the **TNT** system to make them easier to use. For **RV6.8** this provides a completely reworked interface for the tools you use for the creation and assignment of styles to vector elements. Even more important is the redesign of the way styles are now managed "under the hood." As part of this complete overhaul, new features you have requested have been incorporated. From a technical viewpoint, this new style management code will produce identical appearance and functionality when compiled for X or for Microsoft Windows applications.

Style Assignment.

A new and easier to use Assign Styles by Attribute window replaces the Vector Object Key Attribute Style Assignment window. The most important visual change in this new window is that it displays a graphical sample of every style available in the selected style object. Correspondingly, it also shows that same graphical style sample next to each attribute that is assigned that style. This new window is illustrated in the attached color plate entitled New Vector Style Assignment and Editing.

The new Assign Styles by Attribute window contains 2 scrolling lists. The left list scrolls to present a sample of each style and its name in the style object you have selected for your point, line, or polygon elements. Each style name in the list is sorted alphabetically. The right list scrolls through all the possible attribute values of the field you have selected to control the style of your point, line, or polygon elements. This list is also sorted alphabetically. A sample of the currently assigned style, if any, is shown next to every attribute value. Now you can simply select a style from the source list at the left by name or appearance and assign it, including its sample appearance, to an attribute value with a single mouse click on the "assignment arrow" gadget next to one or more attributes. The sample of this style will immediately appear next to that attribute value. If that attribute value already had a style assigned to it, this new style will replace it, and its sample will change. Any attribute value assigned a new or replacement style in the current session will appear in red text rather than black, which indicates no change.

The Vector Object Key Attribute Style Assignment window in **V6.70** allowed styles to be assigned only to a primary key field. This required that you manipulate your attribute tables to accommodate this limitation. The Assign Styles by Attribute window and new style management procedures in **RV6.8** allow any field, including a virtual (computed) field, to be selected as the style attribute. This is much more convenient. It also creates

new kinds of cartographic functionality where the attributes in a field in a linked external table control the style of those elements.

As before, the New Table icon lists all the attribute values in the current table but with no style assigned to them (style names all become <default style> and the sample is the current default). A new "Save As" icon permits you to save the current style assignment table as a separate style table that can be used as a starting point for a closely related style assignment table for the same vector object. The new style assignment interface is available when styling vector and CAD elements by attribute.

Style Editor.

Selection Operations.

This important procedure has many changes and improvements, which are graphical and illustrated in the attached color plate entitled Improved Style Editor Interface. The top portion of this Style Editor window is similar for all element types and presents a scrolling list of style names and samples for the selected element type (points, lines, or polygons). The bottom portion of the Style Editor varies depending upon which kind of element (points, lines, or polygons) is selected. The scrolling list in the common portion of this window looks just the same in this window as in the Assign Styles by Attribute window discussed above. If you select a style name in this list, it can be directly edited in place. Two icons are also provided in this common area of the Style Editor window. The New Style icon adds a new and empty style name and empty sample to the list. You can then enter the style name into the list and design a new style for it. If you have selected a style in the list, the New Style icon adds a new style with the same style sample as the previously selected style. You can then edit the style's name and use it as a model to alter into a similar style or to create a series of similar styles, for example a series of symbol styles that vary only in color. The Delete Style icon deletes the selected style. The new Style Editor interface is available when creating or editing styles by attribute for vector and CAD objects. It is not yet available for All Same styling.

Point Styles.

The lower portion of the Style Editor window for points provides access to the contents and the edit features for the point styles in the object you have selected. It provides a scrolling list of the point symbols (either predefined or from the style object) by name with a graphical sample if a style that has a symbol assigned is selected at the top of the window. Each point symbol name in the list is preceded by an empty check box. Selecting this box next to a point symbol will assign it to the new point style being assembled in the top half of the Style Editor window.

Often, however, you want to alter or design new point symbols. Three icons are present for this purpose for all element types (points, lines, and polygons) if a symbol (not predefined) or pattern is assigned to the selected style. When you click on the Create or Edit Symbol icon, the Symbol Editor opens with a blank design field. Choose Pattern/Open if you want to edit an existing symbol. The Point Symbol Editor can now convert CAD elements to **TNT** point symbols. This is illustrated in the attached partial color plate entitled Convert CAD Elements to Point Symbols. The Insert Symbols icon lets you select a point symbol pattern from the standard point style objects or from other style objects, from a CAD object, from a CGM file, or from a True Type Font to use or to alter. The Delete icon removes the selected point symbol from the list.

Polygon Styles.

The general operation of the lower portion of the Style Editor appears and functions similarly to the Point Style Editor outlined above. However, polygons can have a separate fill and border style. The polygon portion of the Style Editor provides 2 panels for this purpose. The first panel controls the design of the polygon border styles. The second panel controls the fill pattern for the polygon style.

Border Styles.

Polygon styles include a border style and a fill style. The border style can be set to none, solid, or pattern. If set to none, the style assigned to the lines that make up the polygon boundaries will be used if lines are selected for display. If set to solid, you will get a solid color line of the color and width you designate. You can select the color from the palette provided or click on the Palette icon, select a new palette, and choose from it. When set to pattern, the list of line patterns available in the style object will be shown with a pattern sample and name arranged alphabetically.

Fill Styles.

Fill types include None (show by border only), Solid, Bitmap, and Hatch. If none is selected for the border and the fill type, the polygons will be drawn unfilled with the current default border style. The solid fill mode means you will select a color directly from this panel or open the Select Palette dialog, select a new color palette, which replaces the palette in the Style Editor window, then select from it. The Transparency field lets you set a transparency value for a style with a solid fill.

Bitmap and Hatch fill type choices activate the icons that let you create/edit or select new patterns. You will also get a list of samples with pattern names if the style object contains any. Once a pattern is selected, the Delete icon is also active. If you want to design your own bitmap pattern fill or edit an existing one, select the Create or Edit Pattern icon in bitmap or hatch fill mode.

Line Pattern Editor.

Introduction.

The Line Pattern Editor is completely redone. Its many new features are illustrated in the attached color plate entitled Redesigned Line Pattern Editor. An icon toolbar stretches across the top of this new window. It provides the standard **TNT** icons for New, Open, Save, Save As, and Delete options. An Undo icon is provided to undo your edit operations.

A single **TNT** line pattern is made up of several different primitive or component lines. The color plate illustrates the components of a single line pattern, which is made up of a single red dashed line with transparent segments flanked on both sides by thin black boundary lines. You can now easily create a composite line pattern of this type and design and edit each of its primitive line elements.

Solid and Dashed Components.

Each primitive line element making up a composite line pattern is defined in this Line Pattern Editor window as a separate row in its new tabular form. Each of these primitive line elements can have its own color; start, interval, and size (for dashed and dotted lines); thickness; and offset from the composite line's center position. The measurement units used for all the fields in this design window can be selected from millimeters, inches, or points using a drop down menu on the icon toolbar. In the road example on

the color plate, the thin flanking solid black lines are offset +0.5 millimeters and -0.5 millimeters from the center of the red dashed line, which is 1 millimeter in thickness. The black lines have a thickness of 0.0 millimeters, which means they will be rendered at the minimum width possible at any scale.

Dotted and Crossing Line Components.

The primitive line elements for a composite pattern are not restricted to solid and dashed lines. You will find icons in the toolbar of the window to Add Solid Line, Add Dashed Line, Add Crossing Line, and Add Circle primitive lines. Dotted line and crossing line elements use the same fields present for solid and dashed lines. In the case of dotted lines, thickness refers to the circle boundary when not filled, and size refers to the diameter of the circle. For crossing lines, size refers to how far the element extends from the main line, and thickness is its width.

Component Color.

You can change the color of a line component when it is selected regardless of which field is highlighted. This color can be selected from MicroImages' standard color palette or any of the new custom color palettes you can now create (see the section above introducing Custom Color Palettes).

For convenience you can select several colors from a palette and add samples of them into this window. You can then directly select from these samples and assign these colors without opening the Color Editor window. In other words, you often use a limited number of bright colors in line styles. The Add button adds the selected color in the Color Editor window to this small line color palette within the Line Pattern Editor window. You can then assign the color to any of the line components. When editing an existing line pattern, its colors are automatically added to this small palette.

The one color always available in this small color palette is "variable color." The actual color used for this component is assigned by the style. You select variable color for a pattern component by clicking on the color tile that shows four colors, which is always at the left of the small palette. You can change the color used to represent variable color by clicking on the Colors icon in the Preview panel.

Rendering Order.

Primitive line elements can be rendered over the top of others making up the composite line pattern. Primitive lines are drawn in the order they are shown (from bottom to top). A simple example would be to design a red and white dashed line by defining a solid white line and then rendering a dashed red line of the same width over it. You can change the order of rendering of a primitive line by selecting it and using the Lower or Raise icons provided in the toolbar for this purpose.

Sample View.

As in the previous version of your Line Pattern Editor, at the bottom of the window is a sample area that shows the composite line pattern and automatically redraws for each addition or undo operation. Now you can zoom in and out on this sample pattern using +, -, and 1:1 icons in its separate icon toolbar. Icons on this toolbar will set the sample line to draw straight, zigzag, or sinusoidal. These permit you to check how the line will render in different drawing shape conditions.

Special Effects.

A special Edit panel lets you define cartographic rendering characteristics of your primitive lines. These options are assigned as appropriate for each of your primitive lines when you select them. The End Cap options are Flat and Round and specify how the ends of your lines, dashes, and crossing lines should be rendered. The Join options are Round, Miter, and Bevel and define how the solid and dashed lines will be rendered around corners in the line. If you select a dotted line the Cap and Join options are replaced by a Fill option to select solid or open dots.

Choosing Colors for Printing.

The following important information is paraphrased here from the attached color plate entitled Redesigned Line Patter Editor. You can choose any color for your line pattern elements for their electronic presentation. But, you should be aware of the implications of your color selection for printing if your printer uses dithering and is limited to cyan, magenta, yellow, and black inks. The vector pattern, which uses 2x2 superpixels, is the only color pattern recommended for thin lines, but only supports a limited number of colors (colors that are some combination of 0%, 25%, 50%, 75% or 100% of red, green, and blue). The line will be drawn in the closest of these colors if you have assigned it some more subtle color for electronic viewing. If it is important to you to print your custom color, you can choose another dither pattern (Halftone 2 is recommended), but thin lines may drop out at certain angles. Another option is to use a sublimation or other continuous tone printer or one of the new photo rendering printers, which use higher resolution dither patterns and more ink colors.

Virtual (Computed) Database Fields.

Virtual fields (formerly called computed fields) are attribute fields filled with values that look and function as real fields in every **TNT** application. However, they are actually defined by an associated expression that combines multiple fields (real and virtual) from the various tables that define the attributes of the elements in that raster, vector, CAD, or TIN object. A virtual field can also represent a single field that has been modified by the associated expression (for example, rescaled, logically tested, ...). You can add a virtual field to any internal **TNT** table. You then define its contents by an expression that uses other fields from 1 or more of the other tables as its arguments. If you are not familiar with this powerful **TNT** feature, you can learn how to create a virtual field (computed field) in the **TNT** tutorial booklet entitled Managing Relational Databases.

Those familiar with relational database products, such as Oracle, will be familiar with using a "View." This "View" (a virtual table) can contain fields defined by a query that combines one or more real fields or similarly constructed fields in other Views. This is a somewhat parallel idea to the virtual field in the **TNT** products. However, since **TNT** is a GIS-centric product, these virtual fields are tied more closely with the topology and elements in it.

If the field is virtual, it can be used in the **TNT** products just as if it were real. However, its values are recomputed each time that field is needed in a **TNT** process. For example, if you show a tabular view with any virtual fields, the values of those fields will be computed and shown as if real. Or, you could use a virtual field to define the size of point symbols in a pinmap layer. Whenever you redraw that layer, these virtual field values are computed and resize the pins.

At any time, you can convert a virtual field into a real field in the table in which it has been defined. Furthermore, virtual fields do not have a 1-to-1 equivalence in external database systems. As a result, a virtual field will be evaluated and made real as part of the export of any table(s). For example, virtual fields will be exported as real if a vector is exported to Oracle Spatial as described in the major section entitled Oracle Spatial Layer Import and Export.

One of the significant advantages of a virtual field is that it is dynamically populated each time you use it. Thus, if there are changes in any of the fields used as its arguments, then the virtual field's values automatically reflect these changes. For example, you can create virtual fields in a table of the vector object that combines fields in linked attribute tables in Access. The values in these virtual fields do not exist in the vector object but function as if they were in a real field. But, if the linked or external tables are changed, any use of the associated virtual tables will compute new values whenever they are used or viewed.

Many of you already use virtual fields in a wide variety of applications. However, before **RV6.8**, virtual fields could only be defined by your expressions using the attributes of each separate vector element type (which means, for nodes, points, lines, or polygons). This limited the flexibility of virtual fields especially in connection with their use in dynamic network routing applications using vector topology.

Now you can create expressions to define virtual fields that combine attributes for mixed element types (from nodes, points, lines, and polygons) for internal or linked tables. This creates many new possible uses. You can now use a virtual field to make nodes "line aware" or the lines to be "node aware." Or, a virtual field for a line can be selected or not selected based upon the polygons it crosses. The setup of this expanded capability in virtual fields is illustrated in the attached color plate entitled Establishing Dynamic Relations Between Nodes, Points, Lines, and Polygons.

*** Open DataBase Connectivity (ODBC).**

Connecting to a Data Source.

The procedures for making your ODBC links are explained in more detail in the attached color plate entitled Improved Linking to Databases via ODBC. It illustrates the new Link to Data Source window opened from the Make Table/Form menu choice after showing details for a layer in the Group or Layout Controls window. The Link to Data Source window lists the Data Source Name (DSN) of all your data sources in all the different database systems for which you have installed an ODBC driver. After you select a DSN, this window will list the tables for which you have read permission in that data source. You can then add the table(s) you wish to link to your **TNT** objects. You can have links from different **TNT** objects to external attributes in more than one data source, which can be in more than one database software product if you can keep track of it all.

Pin mapping uses a data source as a primary object, not as attributes. You can use an external data source for pin mapping but you must set up your ODBC link through the import process. The attached color plate also shows a table being viewed simultaneously in Access and **TNTmips** and as a pinmap layer.

Simplified Setup of ODBC.

Improved ODBC connection support is now available. In previous versions of the **TNT** products, the DSN had to be setup on all of the computers that the **TNT** Project File con-

taining the ODBC link was to be used on. Now in **RV6.8**, if the DSN does not exist, the connection code will use the ODBC driver entry itself. However, the ODBC driver **MUST** still be installed on all of the platforms that will access the data source from the **TNT** Project File.

Multi-language Support.

V6.70 used V1.1 of ODBC and **RV6.8** now uses V3.5. This new ODBC driver is also about 2 times faster for its read and write operations. However, the major feature added by this upgrade is the support of other languages by using Unicode in the ODBC driver and data sources that support Unicode. Since connecting to tables using other languages is possible, **TNT** products support of memo fields has been improved, which includes the use of Unicode text.

Binary Fields.

Support has been added to the **TNT** products for a “binary” field type. A “binary” field is simply raw binary data. This modification allows **TNT** to handle any field type specified in ODBC. Binary fields are being added to database products to handle graphical shapes, images and other rasters, and other new materials. For example, a binary field might contain as values variable length compressed images with other fields to tell that system how to interpret it. While this general field type is now recognized by the **TNT** products via ODBC, no use is yet made of its contents.

Landscape Builder.

Introduction.

The Landscape Builder process focuses the flexible geodata preparation and Project File geodata management capabilities of **TNTmips** onto the development of large and complex geosimulations (geosims). Gradually the Landscape Builder is using more and more of these advanced **TNT** capabilities, so that they are also available in **TNTsim3D**. **RV6.8** of the Builder takes this another large step forward. It enables you for the first time to add vector objects in Landscape Files as the source for new layers in your geosimulation. These vector objects can contain point elements that you convert to Billboard and Stalk symbols or spherical Volume-of-Interest (VOI) layers, or extruded polygons that appear as solid shapes in your geosim. The attached color plate entitled Additions to the Landscape Builder illustrates some of the new setup options in the Landscape Builder.

Using Rasters.

Landscape Builder uses many of the flexible **TNT** tools to help you convert raster and vector objects into the specialized forms needed for effective use in **TNTsim3D**. As always the Landscape File is still a Project File that can be used in any **TNT** product. However, its raster and vector objects have been optimized for use in a real-time geosim. As discussed in an earlier MicroImages MEMO, the Landscape Builder converts all raster objects associated with a particular terrain to a common coordinate system. Trying to convert them during the operation of a geosim would slow it down significantly. **RV6.8** adds other Builder optimizations for raster objects, such as exporting them to lossless or highly lossy compressed JP2 files that are automatically linked to your Landscape File.

Using Vectors.

Previously, the only use of vector elements in the Builder was to select and copy them into a texture raster so they appear to lie on the surface. Now vector objects can be selected and used as the basis for adding overlay layers to a Landscape File.

Z Attributes Required.

Often, as you have learned in creating static 3D views, you have vector objects that were only prepared for 2D use and lack some of the attributes needed for use in a 3D view or a geosim. Landscape Builder assists you in adjusting the vector elements to be used in **TNTsim3D**. For example, it will extract the area of interest, convert it to the common coordinate system of the landscape, add needed height attributes for billboards and stalks, define a radius for a spherical VOI, and so on. However, you may also have to use other **TNT** processes to adjust your vector object if it was not originally designed for possible use in 3D View and in **TNTsim3D**. For example, you may need to define a side style for extruded polygons or set a base point, height, and line style for a stalk.

Use Multiple Terrains.

TNTsim3D can now display multiple stacked or side-to-side terrains from a single Landscape File. The Landscape Builder can be used to sequentially add these multiple terrains and their corresponding textures into the file. Simply prepare a new Landscape File with the first terrain raster in it just as in **V6.70**. However, now at any time you can add new, additional terrain or texture rasters into this Landscape File. The extents of the terrains you add do not have to match or even overlap. Terrains representing adjoining pieces of the same surface can be added and will automatically mosaic side-to-side when viewed. Terrains can also stack up vertically if they represent surfaces that overlap in extent but represent different layers with varying Z values. The attached color plate entitled Multiple Terrain Surfaces in TNTsim3D illustrates the application of side-to-side and stacked terrains.

Smaller Extents for Texture Layers.

Landscape Files prepared for **TNTsim3D 6.7** required that the extent of each texture layer match the full extent of the terrain layer. If you used the Landscape Builder to make a texture layer from a smaller image covering only part of the terrain, the resulting texture raster would contain the image surrounded by many null cells (which are transparent in **TNTsim3D**) covering the rest of the terrain extent. You could make a number of such partial textures and view them simultaneously in **TNTsim3D** to form a virtual mosaic. The difficulty was that the extra null cells would significantly expand (which means, inflate) the size of the Landscape File.

The rendering engine in **TNTsim3D RV6.8** no longer requires that each texture layer have extents that match those of the associated terrain layer. Thus, the Landscape Builder has been modified so that it no longer pads a partial texture layer with nulls up to the size of the terrain. This can significantly reduce the size of the Landscape File. You can not create a texture that is larger in extent than the associated terrain. This would cause drastic edge artifacts (which means, cliffs) where the terrain drops to a flat, zero elevation plateau outside the terrain. Now, when a raster object is selected in the Builder for use with a terrain, you can accept its total area if it is encompassed within the terrain, ask for it to be clipped to the terrain edges if larger, or choose any encompassed

subarea. The attached color plate entitled JPEG2000 in TNTsim3D illustrates a virtual mosaic using these new small partial textures.

JP2 Compressed Texture Layers.

Drastically Cut File Size.

Terrain data (for example, DEMs) is sparse, expensive to collect, and limited in resolution. On the other hand, texture overlays of high resolution images of multiple dates can be acquired much more readily (for example, Landsat, Ikonos, QuickBird, orthophotos, ...). As a result the size of your Landscape File built in **V6.70**, and, thus, its geographic extent, was determined by your large texture layer(s). **RV6.8** provides an option that will build terrain layers using JPEG2000-compressed JP2 texture layers from the raster object you select. This can drastically reduce the size of your Landscape File. **TNTsim3D RV6.8** decompresses these linked JP2 textures as it reads and uses them. A geosim using linked JP2 files may even be faster than one that uses internal raster objects since these linked files can be much smaller (see section above entitled Much Faster JPEG2000 Displays). The attached color plate entitled JPEG2000 in TNTsim3D illustrates the application of this new feature.

Define Characteristics.

Once you have selected source objects in Project Files to use to create a texture in a Landscape File, Landscape Builder provides a menu to specify the raster data type in the new file. It provides the option of creating a compressed or uncompressed color-composite raster object (either 8-, 16-, or 24-bit) suitable for use in **TNTsim3D**. For example, if your source raster object had floating-point values, it would be converted to one of these integer composite-color formats for use in **TNTsim3D**. The Landscape Builder performs the data type conversion as it creates the raster object in the Landscape File.

This raster type menu now includes options to store the texture as an external color-composite JP2 file that is automatically linked to the Landscape File. You can choose from various RGB color-composite depths for the JP2 file, from 15-, 18-, 21-, to 24-bit. Often 15-bit color will suffice for a geosim and will further reduce the terrain's size by 15/24. The default compression option for JP2 is lossless compression, but controls activated by your selection of JP2 output allow you to specify lossy compression and a compression ratio.

Nulls Create Artifacts.

Lossless compression schemes such as Run-Length-Encoding (RLE) or lossless JP2 preserve the original cell values but perform limited compression of the complex areas in images. In fact, they can even inflate the sizes of these areas. Lossy compression schemes, including JPEG and JPEG2000, do not preserve large areas of uniform cell values. JPEG2000 lossy compression will change some of the cell values in these areas by 1 data value (for example, from 20 to 21) and a few cells by even more. Problems arise if the uniform area originally contained null values, as is common in many **TNT** raster objects. If some of these cells were altered by the lossy JPEG2000 compression so that they no longer have the correct null value, the intended null areas will no longer be completely uniform or transparent in **TNTsim3D**. They will include spots and streaks of color.

Automatic Null Masks.

When the Landscape Builder creates a lossy JP2 file, it determines the null value for that source raster object. It then automatically creates a compressed binary raster matching the source raster in extent and cell size, with 0 = null and 1 = any other value. It uses a form of RLE compression to store this very small binary mask. Each external linked JP2 raster has an RVC link or stub object in the Landscape File that contains all the descriptive data needed to make the external JP2 file look to **TNT** processes as though it was an internal raster object. The compressed binary mask is stored as a subobject of this RVC link object and describes where the null cells are in the linked JP2 file. When **TNTsim3D** uses the lossy JP2-compressed texture, it also reads the mask to determine if a cell read from the JP2 file should be treated as null and, thus, transparent or should be rendered as a color texture cell.

Upgrading Existing Landscapes.

You may have created valuable but large Landscape Files using **V6.60** or **V6.70** of the Landscape Builder. **RV6.8** provides you with the opportunity to rebuild these Landscape Files using the new compression and masking features to reduce their size. Simply use the new Landscape Builder to select a terrain to transfer from your older Landscape File. You can then successively pick textures from that same older Landscape File and transfer them to the new Landscape File as linked JP2 textures.

Billboard and Stalk (B&S) Layers.

The Landscape Builder can now create a pinmap-like layer from the point elements and their associated styles in a vector object. These pins are displayed to scale as billboards that are always perpendicular to your view in **TNTsim3D**. Vertical lines called stalks can also be defined to support each billboard. These lines can be used to connect a billboard positioned above or below the terrain surface to the corresponding position of the point element on the surface. The attached color plate entitled Billboard Overlays in TNTsim3D illustrates the use of these new layers.

The procedures for creating a Billboard and Stalk layer in Landscape Builder start from a "Convert Vector To" icon button in the Landscape Builder window as illustrated in the attached color plate entitled Additions to Landscape Builder. Choose "Billboard Points..." and you will be prompted to select a vector object whose points and attributes will be used to define the B&S layer. The Builder will then convert these point elements into a set of relational tables in the Landscape File. The tables are used by **TNTsim3D** to display this symbol layer.

Other **TNTmips** processes provide the tools needed to prepare a vector object properly before you select it in the Landscape Builder to be the source for a B&S layer. The points in the vector object can have either 2D or 3D coordinates, but attributes and associated point symbol styles should be set up for the points in the Spatial Data Display process or the Spatial Data Editor. If you want to use different point symbols in the same layer, you will need point attributes to associate the symbols with the relevant points. If you want different symbols to have different offsets from the surface, you will need an attribute specifying the height or elevation. You should give special consideration to the location of the origin (hot spot) for each point symbol in the style object. If the hot spot is at the center of the point symbol (the default location) and you do not specify a long stalk or offset from the surface, the bottom of your billboard may end up buried below the surface. You can use the Symbol Editor (accessed from the Style Editor) if necessary to reposition the origin at the bottom of the symbol.

You can use the 3D display process to set up general 3D parameters and preview the results. The 3D panel in the Vector Layer Controls window lets you select the attribute field to provide the height or elevation of each symbol above the terrain and has controls to set the width and color of the stalks. If the vector object you select includes these 3D settings, Landscape Builder transfers them into the Landscape File for immediate use in **TNTsim3D**. If there are no 3D display settings, Landscape Builder sets a default stalk color and width and prompts you to provide a height to be used for all stalks.

Volume-of-Interest (VOI) Layers.

The Landscape Builder can also use point elements in a vector object to create colored, transparent, spherical volume layers. Choose "Volume Points..." from the "Convert Vector To" icon button to select a vector object whose points and attributes will be used to define the Volume-of-Interest (VOI) layer. As for B&S layers, the vector object should be prepared in **TNTmips** before using the Landscape Builder to create the VOI layer. The vector points can have either 2D or 3D coordinates. The color and transparency for the style assigned to each point determine the color and transparency of its corresponding spherical volume. If you want volumes with different colors, you will need point attributes to associate different-colored point styles with the relevant points. When you convert the vector, Landscape Builder prompts you to enter a radius value to use for the VOI layer. The VOI parameters are stored in a set of relational tables in the Landscape File with fields that can be used to specify an inner and outer radius, sector angles, and other shaping parameters. The attached color plate entitled Volume-of-Interest Overlays in TNTsim3D illustrates these VOIs and the table that contains the point locations.

Extruded Polygon Layers

The Landscape Builder creates a layer of extruded polygons using an approach parallel to that outlined for building a Billboard and Stalk vector layer. The attached color plate entitled Extrude Polygons as Solid Shapes in TNTsim3D illustrates these layers. Choose "Extruded Polygons..." from the same menu and select the input vector object with the polygons to be extruded. This vector must meet stricter requirements than those you use for B&S or VOI layers. First, the vector must have 3D display parameters already set for its polygons in order to be selectable for this conversion. As for points, these parameters are set in the Display process on the 3D panel of the Vector Layer Controls window. This panel allows you to specify an attribute field to set the height or elevation of extrusion. If you want the polygons to have differing heights, then you will need different values assigned for the various polygons. You can also set whether the base of the polygon is drawn at the minimum or maximum level of its intersection with the terrain, and set a style or styles for the sides of the polygons. Fill styles for the top and bottom of the extruded polygon are those used for normal display, and are set on the Polygon tab panel. You should not use Bitmap or Hatch pattern fills, as these do not scale properly for 3D use and are ignored by **TNTsim3D**.

The vector polygons should also have 3D coordinates appropriate to the terrain or terrains they will be displayed with in **TNTsim3D**. (If the polygons have only 2D coordinates, **TNTsim3D** renders them with their bases at 0 elevation, which may be beneath all of the terrains in your geosim.) You can create polygons with 3D coordinates in the Spatial Data Editor by editing over a surface layer containing an appropriate elevation raster. But you may be starting with an existing 2D vector object. In this case you can take advantage of the fact that you must have an elevation raster (DEM) to provide the source for the terrain object in your geosim. The 2D vector object and the DEM can be

processed together elsewhere in **TNTmips** (Process / Convert / 2D Vector to 3D Vector) to transfer Z values from the DEM to the vector element vertices to create a 3D vector object.

When you have selected an appropriate vector object and initiate the conversion, the Landscape Builder transfers the selected vector object directly into the Landscape File, where it is also maintained as a vector object. Since the Landscape File is also a Project File, you can access the transferred vector object's 3D display parameters from **TNTmips** if you need to make any changes in extrusion parameters, styles, and so on.

Map Projections and Coordinate Systems.

HPGN Datums.

Large area satellite images of high resolution, better DEMs, more accurate GPS surveys, and other activities require more and more precise datums. Local datums can now be defined by more empirically derived systems. In the U.S. and in some other nations, this has led to the definition of a locally fitting High Precision Geodetic Network (HPGN) based upon adjustments to points very accurately measured by GPS at many previously surveyed first order benchmarks. Support for conversion to this HPGN datum is now available. These grid files allow datum conversion to within a few centimeters of accuracy. Conversion HPGN grid files are provided for the entire USA.

The conversion of the Tokyo to JGD2000 (Japanese Geodetic Datum 2000) is now supported using HPGN grid conversion files.

More information on the HPGN datum can be found at www.ngs.noaa.gov/TOOLS/Nadcon/Nadcon.html.

EPSG Datums.

The following new geodetic datum definitions from the EPSG (European Petroleum Survey Group) geodesy database have been added.

- Abidjan 1987 (Ivory Coast)
- Accra (Ghana)
- American Samoa 1962
- Ammassalik 1958 (Greenland)
- Aratu (Brazil, 4 variants)
- Arc 1960 (Kenya variants, Tanzania)
- Belge 1972 (Belgium)
- CH-1903+ (Switzerland, Liechtenstein)
- Conakry 1905 (Guinea)
- Datum 73 (Portugal)
- Dealui Piscului 1933 (Romania)
- Deir ez Zur (Syria)
- Dominica 1945
- Estonia 1992
- Estonia 1997
- Fahud (Oman, 2 variants)
- Final Datum 1958 (Iran)
- Grenada 1953
- Hanoi 1972 (Vietnam)
- Indian 1960 (Vietnam, 2 variants)
- Indian 1975 (Thailand)

IRENET95 (Ireland and Northern Ireland)
Korean 1995 (South Korea)
La Canoa (Eastern Venezuela)
Lisbon 1937 (Portugal)
Locodjo 1965 (Ivory Coast)
Luxembourg 1930
Malongo 1987 (Angola offshore)
Manoca 1962 (Cameroon coastal area)
Militar-Geographische Institut (Croatia, Slovenia, Montenegro variants)
Mhast (Angola offshore)
Monte Mario (Italy, 3 variants)
NAD83 (Aleutian Islands, Hawaii variants)
Naparima 1955 (Trinidad)
Naparima 1972 (Tobago)
NGO 1948 (Norway)
Posiciones Geodiscas Argentinas (POSGAR) 1998 (Argentina)
Pulkovo 1942 (Lithuania, Estonia)
Qatar 1974 (2 variants)
Qornoq 1927 (Greenland)
Red Geodesica Venezolana REGVEN (Venezuela)
Saint Kitts 1955
Saint Lucia 1955
Scoresbysund 1952 (Greenland - Scoresbysund area)
Sierra Leone 1968
SIRGAS (South America)
S-JTSK (Czech Republic, Slovakia)
South Yemen
Tete (Mozambique)
TM65 (Ireland)
Trinidad 1903
Xian 1980 (China)
Yemen NGN96

Ellipsoids.

The following ellipsoid definitions from the EPSG (European Petroleum Survey Group) geodesy database have been added.

Average Terrestrial System 1977
Clarke 1866 (Michigan)
Clarke 1880 (RGS)
Everest 1830 (1962 Definition)
GEM 10C
GRS 1980 Authalic Sphere
Xian 1980 datum for China.

Projections.

“Interrupted Goode Homolosine” projection commonly used for AVHRR worldwide datasets is provided.

Oblique Stereographic projection has been added.

Coordinate Systems.

Gauss Boaga zone is now available for the Italy coordinate system.

Stereo 70 zone for the Romania coordinate system is now available.

Raster Extract/Copy.

When a raster object is being extracted or copied using the Raster Extract process, as an option its metadata can be copied to the new raster object.

Raster Import.

Socet Set native DT format output by this BAE soft photogrammetry system can now be imported or used directly via an auto-link.

When importing from ESRI BIL/BIP, the null value will now be obtained from the header file. The data type will also be determined from the header. Note that it may not be possible to obtain these parameters from the header.

ArcGrid import can handle georeference using either the center or the corner of the raster cells as specified in the file.

CMYK images can be imported from JPEG (classic) files.

SRTM importing now provides a "Do not show message again" toggle on the dialog warning you about missing files.

Raster Export.

Exporting to JP2 files now uses incremental writing to reduce RAM requirements during JPEG2000 compression. You can now compress much larger images into JP2 files. Please note that most JP2 plug-ins (for example, for Quicktime Professional, Adobe's for Explorer, ...) do not support decompression of JP2 files larger than 1 to 2 Gb. The Options panel for JP2 export has also been rearranged to put infrequently changed and advanced options at the bottom.

Exporting to TIFF files now permits the selection of compression when exporting 24-bit rasters.

Vector Import/Export.

Importing of large collections of points from text files is now 10 times faster. For example, importing more than 1 million points takes 10 to 15 minutes instead of 90 to 150 minutes.

Database or attribute tables for some external formats may combine numeric and character data in a single field. For example, a field might contain '1John', '17Frank', '34Pete', ... During import, these composite fields can now be separated into a numeric and a string field.

Big Shapefiles.

Import will now handle shapefiles consisting of millions of points (for example, 10 million).

Export to shapefile and E00 will now handle lines with large numbers of vertices (for example, more than 500,000).

Importing Coverages.

The interface for selecting coverages has been improved. You can now select a directory, and the import process will show all, and only, the coverage directories in it, allow you to select one or more, and automatically use the appropriate files within each. The vector objects created will be named the same as the coverage directory. These revisions avoid any confusion you might have with regard to which and how many files to select from a coverage directory and with regard to the name assigned to the vector object that might have corresponded with the specific file selected instead of the coverage's name.

* Shapefile Styles Import and Export.

The graphical and attribute contents of ESRI shapefiles have been imported and exported by the **TNT** products for several years. Now the styles of the lines and polygons stored in the accompanying AVL in the ESRI project file (PRJ) can also be imported and exported to/from a vector object. An imported shapefile immediately displayed as a layer in a view looks just like it was styled within your ESRI product. All the styling components of AVL lines and polygons have an equivalent in your **TNT** styles. This is illustrated in the attached color plate entitled Import/Export Shapefile Line/Polygon Styles where a **TNT** view is compared with that of ArcView.

This color plate also compares views from both products of a vector object that has created and then exported with styles from **TNTmips** or **TNTedit**. You will find that these views compare very closely. However, as you might expect, there are some components of a style that you can define in the **TNT** products that can not be exactly reproduced in the more limited shapefile styles. For example, in **TNT** you have more control over the exact structure of a dashed line. These **TNT** style components are approximated in the export to a shapefile. There are also some **TNT** styling features, such as bit map fill patterns of variable size, that have no equivalent in a shapefile fill pattern, which can only be 8 by 8 cells. If you have used these features in a **TNT** style, when you export that vector object a warning message will pop up. It will indicate the first such condition encountered and continue on to export the valid shapefile styles. You will then have to adjust these styles in your **TNT** product to conform to the more limited style structure supported by the AVL file. This warning system will be modified in **DV6.9** so that a log file is kept recording all **TNT** styles not converted and why.

Point styles in a shapefile are handled as TrueType glyphs and are not supported in **RV6.8**. This capability is being added to **DV6.9** and is nearly complete.

Automatic Import/Export Testing.

External geodata formats are frequently modified, extended, or undocumented features are used requiring adjustment to their **TNT** import and export process. MicroImages also transparently adjusts the Project File as needed by the **TNT** products. Since there are hundred of geodata formats supported by the **TNT** products, keeping all of them operating correctly is a challenge much like trying to hit a moving target. To assist in this, MicroImages has implemented overnight automatic testing of the basic functionality of each import/export. Examples of most of these formats are now automatically imported each night and then exported and compared. If a format is not working, a message is sent to software support and the responsible software engineer.

Import and Export of some external geodata formats provide you with a variety of options that depend on the specific contents of that file. At this time, this procedure only tests a standard example and procedure for each format. Undoubtedly you will continue to encounter file formats that do not work because they do not adhere to their specifications, have been modified, or use undocumented features or features never previously encountered.

*** Oracle Spatial Layer Import and Export.**

Introduction.

Oracle Spatial is an option that can be purchased only for Oracle9i Enterprise Edition. As defined by Oracle "It is a foundation for the deployment of enterprise-wide spatial information systems, and Web-based and wireless location-based applications requiring complex spatial data management." A subset of Oracle Spatial called Oracle Locator is available as a standard feature in Oracle9i Standard Edition and Enterprise Edition. Hereafter, these two Oracle products will be referred to simply as Oracle Spatial.

As an extension of a large and complex Oracle system, Oracle Spatial (and Oracle Locator) stores and manipulates graphical components, georeference information, and attributes as a set of relational tables. **TNT** vector objects are structured directly around the efficient storage of topological elements whose attributes are maintained in relational tables attached to these elements. **TNTmips**, **TNTedit**, and **TNTview** now provide a process to import an Oracle Spatial layer into a vector object. Vector objects created within the **TNT** products, or altered after import from Oracle Spatial, can then be exported from **TNTmips** or **TNTedit** to a new layer in Oracle Spatial.

For those who would like a PowerPoint overview of Oracle Spatial, the following presentation is useful. While it is in Portuguese, it is profusely illustrated in English and, thus, is easily reviewed.

Tecnologia Oracle Spatial. Busca e armazenamento inteligente de dados espaciais. by Ferreira do Nascimento, Chaves da Rocha, and da Nobrega Medeiros. no date. Centro de Ciências e Tecnologia of Universidade Federal de Campina Grande. This is a recent document but unfortunately it is not dated and can be found at www.dsc.ufpb.br/~fabiana/sig/OracleSpatialMapviewer.ppt.

Background.

This new **TNT** import/export process supports the movement of geographical project materials between a topologically oriented approach (**TNT**'s use of vector objects) and a database system (Oracle Spatial's layers). Before you undertake such an operation, it is useful to review the fundamental differences in these approaches to the storage of graphical materials. Oracle Spatial is database-centric with regard to the storage of graphical data. Graphical elements are stored in fields in the database structure that, like all other elements, are located through database searches and then used. Topologically based approaches, such as **TNTmips** use of vector objects, are graphic-centric or GIS-centric, where maintenance of the graphical element and its topological relationships to other elements are paramount, and the management of the attributes of these elements is an adjunct operation.

ArcInfo coverages are topologically based structures even though topology is not rigorously maintained throughout the processes that use coverages. Shapefile and MapInfo

data structures fall between these extremes. ArcGIS is much more database-centric than ArcInfo.

It is also important to remember that, as part of their topologically oriented approach, the **TNT** vector objects will optionally maintain 3 different levels of topology: polygonal (by default), planar, and network. You can create, convert between, and use the topology that is appropriate for a particular application. For example, it may be easier to create a planar topology vector object into which you can quickly sketch elements and then later upgrade it to polygonal topology and add attributes. **TNT's** CAD, TIN, and raster objects are also optimized for area-oriented direct use. For example, raster objects are always pyramided for optimum area access while TIN objects maintain rigorous topology. Color plates entitled Vector Topology Types and Behavior of Topology Types illustrate and further describe these concepts. These color plates were issued with a previous release and have been updated and included here for your review because of their relevance to the current topic.

Advantages of Direct Topology.

Rigorous polygonal topology can be most simply defined as “a place for everything on a surface and everything in its place.” At any time in any application, using a vector object with polygonal topology guarantees that any location has only 1 possible geometric element of a given type. A single vector object of any of the **TNT** topologies is structured to separately maintain points, lines, and nodes and their associated attribute structures. Polygonal topology also maintains polygons in the same structure, whereas the other two types do not store polygons as distinct elements. A vector object defaults to use polygonal topology and, thus, only 1 element of each of these types can exist at a given position, for example, all polygon areas are mutually exclusive regardless of their shape and can not overlap. Lines do not cross each other but are represented as multiple different lines that meet at nodes, and so on.

For simplicity the word “topology” should be understood to mean “polygonal topology.” The lesser topologies of planar and network will be specifically identified as such.

Maintaining topology has the advantage that, at any point in time, all the information for a location on that surface is directly available. It also insures that its elements can be efficiently combined leading to new smaller elements or generalized to combine elements with topology. The disadvantage is that a series of operations that combine objects with topology can result in complex relational database structures for their combined attributes. However, this can also result from the successive manipulation of graphical layers in database-centric systems, such as Oracle Spatial. In either case, to prevent this it is necessary to carefully select and filter how the attribute structures are combined during these operations. Otherwise you must periodically stop and simplify your current attribute structures by discarding unneeded fields and combining records and tables. A database-centric system provides superior tools for these kinds of filter/clean-up operations, since they occur throughout any successive combinations of tables in any application of a relational database, not just the use of the use of their spatial tables.

Topology Only On Demand.

Oracle Spatial and other similar products are designed as efficient relational database systems within which each individual graphical element is treated as a field in a table. These elements may or may not be georeferenced, just as in a vector object. Oracle Spatial provides extensive tools for manipulating these graphical elements. However, a non-Oracle GIS application using Oracle Spatial that requires topology for an operation must build it and then use it outside Oracle Spatial. The application can then display or do what it wants with the many, many new elements and combination of the attributes that may result from intersecting overlapping elements to create a temporary topological condition. If the application needs to store the new graphical elements that result from the topological analysis, it must create or recreate the appropriate Oracle Spatial tables for this purpose. This can quickly build a very complex set of tables.

Terminology.

Oracle Spatial and **TNTmips** have a different vocabulary for spatial data components and operations. The color plate entitled Oracle Spatial Layer vs TNT Vector Object also discusses the terminology and illustrates some interface components. It is important to have an understanding of these differences before moving your graphical components between these systems. The following are definitions of the hierarchal components used in Oracle Spatial and are quoted directly from the Oracle Spatial User's Guide and Reference (Release 9.2), 2002, pages 1-3 to 1-5 (complete reference below). These quoted sections are deliberately presented out-of-order so they are easier to relate to your understanding of the use of similar data organization concepts in the **TNT** products.

Oracle Spatial Element (section 1.5.1).

"An element is the basic building block of a geometry. The supported spatial element types are points, line strings, and polygons. For example, elements might model star constellations (point clusters), roads (line strings), and county boundaries (polygons). Each coordinate in an element is stored as an X-Y pair. The exterior ring and the interior ring of a polygon with holes are considered as two distinct elements that together make up a complex polygon.

"Point data consists of one coordinate. Line data consists of two coordinates representing a line segment of the element. Polygon data consists of coordinate pair values, one vertex pair for each line segment of the polygon. Coordinates are defined in order around the polygon (counterclockwise for an exterior polygon ring, clockwise for an interior polygon ring)."

A key phrase to note in the above definition is a "line string" can be used as an element. As part of the explanation of geometries below you will note that Oracle Spatial uses arcs defined by 3 points as an element as well as the opposite corners of a rectangle.

TNT Points and Segments.

The basic building blocks of a **TNT** vector object are independent or connected strings of XY or XYZ vertices assembled into points, lines, polygons, and nodes, each of which are stored separately in the same vector object. Only straight line connections are used, no arcs, no predefined geometric shapes. This has the advantage of faster computation of intersections and has the disadvantage of using more vertices to define a curve or rectangle.

Oracle Spatial Geometry (section 1.5.2).

“A geometry (or geometry object) is the representation of a spatial feature, modeled as an ordered set of primitive elements. A geometry can consist of a single element, which is an instance of one of the supported primitive types, or a homogeneous or heterogeneous collection of elements. A multipolygon, such as one used to represent a set of islands, is a homogeneous collection. A heterogeneous collection is one in which elements are of different types, for example, a point and a polygon.

“An example of a geometry might describe the buildable land in a town. This could be represented as a polygon with holes where water or zoning prevents construction.”

Oracle Spatial GeometryTypes (section 1.4).

“A geometry is an ordered sequence of vertices that are connected by straight line segments or circular arcs. The semantics of the geometry are determined by its type. Spatial supports several primitive types and geometries composed of collections of these types, including two-dimensional:

- Points and point clusters*
- Line Strings [straight line elements only]*
- n-point polygons [straight line elements only]*
- Arc line strings (All arcs are generated as circular arcs.) [arc elements only]*
- Arc polygons [arc elements only]*
- Compound polygons [combines arc and straight line elements]*
- Compound line strings [combines arc and straight line elements]*
- Circles [arc elements only]*
- Optimized rectangles [rectangles represented only by opposite corners]*

“Two-dimensional points are elements composed of two ordinates, X and Y, often corresponding to longitude and latitude. Line strings are composed of one or more pairs of points that define line segments. Polygons are composed of connected line strings that form a closed ring and the area of the polygon is implied.

“Self-crossing polygons are not supported, although self-crossing line strings are supported. If a line string crosses itself, it does not become a polygon. A self-crossing line string does not have any implied area.

“Spatial also supports the storage and indexing of three-dimensional and four-dimensional geometry types, where three or four coordinates are used to define each vertex of the object being defined. However, spatial functions (except for LRS functions and MBR-related functions) can work only with the first two dimensions, and all spatial operators except SDO-FILTER are disabled if the spatial index has been created on more than two dimensions.”

As you can see, Section 1.4 defines geometries in somewhat more detail but with some contradictions (or at least omissions) relative to Section 1.5.1 since it now includes the idea of a circular arc and a rectangle basic building block. From the above list and the illustrations in this section, the arc primitive (page 1-4) consists of a circular arc between 2 points but must have some additional means of defining the concave side of the arc and its curvature, for example, it is actually defined by 3 points. The rectangle corner appears to be a 2 point polygonal element that is interpreted as being the diagonal of a rectangle.

TNT Vector Elements.

Oracle Spatial geometries are most closely associated with the elements in a vector object: points, lines, polygons, and/or nodes. Thus vector elements should not be confused with the Oracle Spatial basic building blocks defined in Section 1.5.1 and called elements.

TNT's isolated point elements are XY or XYZ positions with an associated set of attributes.

XY or XYZ vertices connected by straight line segments make up line elements. Line elements do not cross each other and can only meet at nodes for rigorous topology. Each line element has a set of attributes. In other words, a line element can also be defined as a set of continuously connected segments with identical attributes. A line element can be of any length from 2 vertices to millions of vertices.

Node elements are also isolated points that represent the connections between 2, 3, 4 or more line elements. If the only 2 line elements entering a node do not have at least 1 different attribute value, the node is unnecessary. Node elements can also have their own attributes, in which case they are treated as point elements.

A polygon element is a set of line elements that, when connected end-to-end, form a closed shape. This connection between a set of line elements to define a polygon element is maintained separately in a vector object and can have its own attributes, which apply only to the inscribed area. Polygon elements can be of any size and shape and can be made up of millions of vertices. Islands within islands within islands ... can be created and maintained.

It can be confusing at first to those using a vector object that you can manipulate and view the line elements bounding all polygon elements either separately or as boundaries of polygon elements since each use has a separate set of attributes. For example, lines that separate pairs of adjacent polygons can be rendered in a road symbolism while the interior of every polygon is rendered in a color to represent its area as differing land use or ownership.

The rigorous polygonal topology in a **TNT** vector object is actually 2.5D. All vertices can be specified as XYZ. However, since only one point, line, and polygon can exist at any location, no area can be specified to be under or above any other area. Thus, full 3D topology is not possible. The other optional topologies that can be specified for a vector object relax various of these rigorous conditions.

Polygonal Topology.

Node elements in polygonal topology define the ends of all lines. Two nodes can not exist at the same XY location. Line elements do not intersect. Line elements can only meet other line elements at a node element. All line elements begin and end at a node. A single point element can exist at a specific XYZ location.

Polygonal topology maintains and stores predefined polygons that do not overlap each other or any other polygon in that object in 2D or 3D. Polygons can have Z values attached to their boundaries, but no polygon can have an interior point shared by any other polygon. Thus, even the simplest concave structure is prohibited. This is why, even if the Z values of the vertices of the polygon vary widely, this is only a 2.5D topology. There are higher orders of true 3D and 4D topology defined, and a description of these can be reviewed on the reverse side of the attached color plate entitled Vector Topology Types.

Planar Topology.

Planar topology requires that all lines start and end in nodes and no two lines cross, as with polygonal topology. However, polygon information is not maintained. The color plate entitled Behavior of Topology Types illustrates how vector objects with planar topology are more like objects with network topology for 2D operations but more similar to objects with polygonal topology for 3D operations.

Network Topology.

Network topology lets lines with common or different attributes cross other lines without inserting any nodes. No area properties are maintained. Thus, if the vertices are XYZ, it is possible to define a 3D structure of lines and points in this topology. If a node is present where XY lines cross, it can specify the type of crossing, such as a bridge, or for 3D lines the minimum flight path clearance since nodes can have attributes assigned.

Oracle Spatial Layer (section 1.5.3).

“A layer is a collection of geometries having the same attribute set. For example, one layer in a GIS might include topographical features, while another describes population density, and a third describes the network of roads and bridges in the area (lines and points). Each layer’s geometries and associated spatial index are stored in the database in standard tables.”

TNT Vector Object.

The term “object” is used in Oracle Spatial and has no direct relation to concept of the objects stored in a **TNT** Project File.

The layer in Oracle Spatial is most closely analogous with the vector object in the **TNT** products. An Oracle Spatial layer can have any combination of geometries. However, the fact that they all share the same attribute set indicates that it might be most efficient to keep geometries representing points, linear oriented geometries, and similar area geometries as separate layers. While not required, this would simplify the common structures of their attributes. In this approach, the layer would be most closely equivalent to the separate collection of points, lines, polygons, or nodes in a vector object, each with its own separate, but common, attribute structure.

Oracle Spatial Data Model (section 1.5).

“The Spatial data model is a hierarchical structure consisting of elements, geometries, and layers, which correspond to representations of spatial data. Layers are composed of geometries, which in turn are made up of elements.

“For example, a point might represent a building location, a line string might represent a road or flight path, and a polygon might represent a state, city, zoning district, or city block.”

TNT Project File.

An Oracle Spatial data model is most closely analogous to a Project File made up of multiple vector objects. The hierarchical structure of the layers in a Data Model would support global queries, searches, and other operations. There are no inherent hierarchical relationships between vector objects in a Project File, so these kinds of global operations on a Project File are not automatically possible above the object level. On the other hand, a Project File can also conveniently contain other kinds of objects: raster, TIN, CAD, and relational database objects making up a project. The HyperIndex

structure created for a **TNTatlas** defines a hierarchal structure for all these **TNT** layers or objects.

Other Differences.

Properties that apply to an entire spatial layer are stored in general metadata tables. For example, these would include a geometries metadata record for a layer containing dimensions, lower and upper bounds, tolerance in each dimension, coordinate system, and so on. Oracle Spatial operations on a layer that require these layer-wide attributes obtain them from this metadata, which has records that pertain to all spatial layers.

TNTmips stores this kind of “adjective” information, such as georeference, extent, datum, ellipsoid and so on, in subobjects that are attached to, and considered part of, a vector or other spatial object.

In Oracle Spatial, there are general look-up tables for distance and angle units, which are defined in **TNTmips** for each object and converted as needed.

Prior to Release 9i Oracle Spatial treated the earth as a flat surface. 9i now supports projected coordinates and comes with reference tables for datums, ellipsoids, and projections where the defining parameters can be looked up as needed by operations.

TNTmips uses more subobjects with the vector object to identify these parameters for operations, such as conversion of the object to a common projection during a display operation.

End User Versus Enterprise Orientation.

TNT products are end-user oriented and are designed around the personal visualization of the results of each operation. You can organize complex and large collections of geodata objects of varying types. There are many kinds of analysis tools that can then be used to transform, combine, or analyze these object types. When working in a **TNT** product, you tend to think about what you want as a project result and then set about acquiring the specific required geodata, selecting the portion or features of interest, and then periodically viewing or printing it. Working with **TNT** products is pragmatic, often site specific, and database operations are not necessarily of paramount importance.

The operation and management of Oracle and, thus, Oracle Spatial are enterprise-wide and something of primary concern to professional IT database specialists. They manage the data models, integrity, and security of the system. They create the end-user applications such as forms to view, edit, and enter primitive data. The individual end user usually does not have much input to the structure of the system.

Oracle Spatial does provide the IT professional with some system level tools to inspect the structure and simple content of an Oracle Spatial layer. These are illustrated in the attached color plate entitled Oracle Spatial Layer vs TNT Vector Object. These include the Oracle Enterprise Manager Console. It shows all tables in your database and can be used to select and view the tabular structure of an Oracle Spatial layer. Hidden away in Oracle Spatial is the Oracle Spatial Index Advisor, which is the only way you can make a simple spatial display and inspection of an Oracle Spatial layer (alas, this useful tool has been poorly named and is hard to find in the reference manuals). To go beyond this simple viewer you must acquire other viewing and spatial analysis tools.

Importing An Oracle Spatial Layer.

The principle reason you may want to import an Oracle Spatial layer is to perform some **TNT** convenient operation upon it that is not provided or is complex to perform in Oracle

Spatial. This may be so that those results can be returned back to Oracle Spatial for use by others or to extract and use Oracle Spatial data in your published project results. The attached color plate entitled Importing Vector Objects from Oracle Spatial Layers shows the import interface and sample results.

Import Creates Topology.

You have your choice of topology types on import of Oracle Spatial layers between polygonal, planar, and network topologies. The partial color plate entitled Oracle Spatial Import Options describes your topology and optional table choices. Choose polygonal topology if the layer contains areas with attributes, such as property maps or soil type polygons. Polygonal topology can also be built for layers that included only lines, but lines that formed closed shapes, in Oracle Spatial. Attributes can then be added to these polygons in the **TNT** products. Choose planar topology if the layer lacks polygons but has or should have nodes at every line intersection. Choose network topology if the layer contains lines that should not have nodes added when projected into 2D because the lines do not actually cross in 3D, such as water, sewer, and other infrastructure representations.

The import process for Oracle Spatial will bring over the geometric features and import the other fields of the spatial table into a Project File database table. The import routine will then search the Oracle database for Primary and Referential constraints (primary and foreign keys in **TNTmips** vernacular) and import those tables into Project File database tables.

Exporting to an Oracle Spatial Layer.

An exported polygonal topology vector has rigorous polygonal topology. You have correct, mutually exclusive areas, and only 1 point, line, and area entity at each coordinate until modified by some subsequent Oracle Spatial process. Planar and network topology are also maintained as elements from a **TNT** vector object become geometries in an Oracle Spatial layer.

The export routine will create a new table for the spatial data (the geometries) and any **TNT** tables in the database object under the vector object (for example, statistics). The table names are checked against the Oracle table names and are modified if necessary to be unique. The export routine does not override or alter existing Oracle Spatial or Oracle tables.

The color plate entitled Exporting Vector Objects to Oracle Spatial Layers describes some of the strict database rules required by Oracle Spatial. The back of this color plate discusses some of the warnings you may encounter on export and how to fix your database if you run into these warnings.

Virtual Fields.

Within your **TNT** project, you may have defined and are using virtual fields (formerly called computed fields, including the new cross vector element type of virtual field [see section Virtual (Computed) Database Fields above]. When you export a vector object to Oracle, these virtual fields are not defined in Oracle. As a result, during export all virtual fields are computed and written out as real, or permanent, fields in the corresponding Oracle tables. A cross element type virtual field is using real fields from different **TNT** vector element types (for example, points and polygons) and will be in the Oracle Spatial table associated with a specific element type (which means, in the point or the polygon

table). The real field computed during export will also be in the table associated with the corresponding Oracle geometry type (which means, with the point or polygon geometry).

Information Resources.

It can be assumed that as a **TNTmips** user, you are already familiar with the advantages, disadvantages, and procedures for manipulating and using a vector object with the different levels of topology. If you are considering the use of Oracle Spatial and are unfamiliar with it, then it is important that you acquire the proper Oracle Spatial reference materials from the extensive collection of Oracle manuals required to document their enterprise-oriented database products. The most important of these related to the use of Oracle Spatial are documented here. If you are not already experienced with the use of Oracle and Oracle Spatial, then you should download these manuals from the web references provided and keep them at hand. All these manuals are available for free download if you sign in (which means, register) with Oracle. You do not have to be an Oracle customer to acquire them.

Oracle Spatial 9.

Oracle Spatial User's Guide and References. Release 9.0.1. June 2001. Part No. A88805-01. 472 pages.

<http://particle.homeip.net/oraclebooks/Spatial%20User's%20Guide%20and%20Reference.pdf>

Oracle Spatial 9.2.

Oracle Spatial User's Guide and References. Release 9.2. March 2002. Part No. A96630-01. 486 pages.

<http://otn.oracle.com/docs/products/spatial/content.html>

You will also find links to documentation on this same page for Oracle Spatial 8.1.5, 8.1.6, 8.1.7, and the earlier Relational Model (now deprecated by Oracle). Please do not bother obtaining manuals for these versions of Oracle Spatial as MicroImages **TNT** support of Oracle is only for Oracle versions 9.0.1 and 9.2.

Appendix C of this documentation briefly explains the differences between Oracle Spatial (Oracle 9i Enterprise Edition) and Oracle Locator (Oracle 9i Standard Edition). It also contains tables listing the functions in Oracle Spatial which are omitted from Oracle Locator.

After you have exported your vector to an Oracle Spatial layer, you will want to view and inspect its contents. To do this, you will want to use the Oracle Spatial Index Adviser available as part of the Oracle Enterprise Edition. Unfortunately this 486 page Oracle Spatial manual only mentions the Oracle Spatial Index Advisor in several very minor ways. It does not provide any details about its availability or use. For 2 paragraphs of information about it you can go to the

Oracle Enterprise Manager: Concepts Guide. Release 9.2.0. March 2002. Part No. A96674-01. 132 pages.

“Oracle Spatial Index Advisor helps you analyze and tune spatial indexes on data. Using this application, you can analyze the effectiveness of spatial indexes, defined on spatial data. The Advisor lets you see if indexes are properly defined for optimum query performance. The application also provides an understanding of the distribution of data through visual inspection.”

“A spatial index is a set of database tiles. With Oracle Spatial Index Advisor, the database administrator specifies the size and number of tiles in a database. The geometric coverage of the tiles has a direct impact on query performance. The Advisor allows the user to see the interaction of the tiles with the geometric coverage and to issue queries against the data to see how typical queries will perform.”

The Oracle Spatial Index Advisor is a console or administrator tool, so you find it as part of the operations covered in the following manual.

Oracle Enterprise Manager: Configuration Guide. Release 9.2.0.2. October 2002. Part No. A96673-02. 208 pages.

“The primary responsibility of first-tier Enterprise Manager clients is to present the user interface to administrators for all their management tasks.

“Depending on what has been installed and licensed, first tier clients could consist of the following components:

“Consoles...

“Integrated management applications, which include Oracle Forms Server Manager, Oracle Policy Manager, Oracle Directory Manager, Oracle Net Manager, Oracle Spatial Index Advisor, Oracle Data Guard Manager, Oracle LogMiner Viewer, Oracle Enterprise Security Manager, and Oracle Text Manager.”

MapView

Oracle 9i Application Server: MapViewer User’s Guide. Release 2 (9.0.4). April 2003. Part No. B10559-01. Beta Draft – Work in Progress. 214 pages.

This is a complex viewing and analysis tool that Oracle is in the process of releasing and testing. This document is not referenced in any way in the 2002 Oracle Spatial manual cited above. It also can not be located readily via any Google search as it is beta and not indexed by Oracle. It was last found by signing in at the following URL.

<http://otn.oracle.com/software/products/spatial/content.html>

Vector to Raster Conversion.

More Control.

The dialog box used in this process has been redesigned to use tabbed panels to manage the selection and conversion of point, line, and polygon elements. You now have all the standard **TNT** selection procedures to control what vector elements will be converted. These include selection in a view, by region, by query, none, all, and so on. The default for cell size in the raster object sets the cell height equal to the cell width. This new dialog is illustrated in the attached color plate entitled Improved Vector To Raster Conversion.

Improved Accuracy.

An important part of the renovation of this process is the improvements in the accuracy of conversion. The vector object has vertices whose coordinates are expressed in floating point whereas the target raster usually has integer precision. The revisions use this floating point accuracy to more accurately assign cells to the correct area along the boundaries of polygons and to represent lines.

Cells are now assigned to the interior of a polygon element if the cell’s center is inside a polygon. This also insures that no cells are unassigned to any polygon for unusual edge

conditions. It will assign a cell to represent a line element if it is intersected by any part of the line. This provides better line continuity since it ensures that the cells representing the line have edge-to-edge contact rather than the previous method, which only required corner contact at some positions. The same attached color plate entitled Improved Vector To Raster Conversion illustrates these better representations of polygon edges and lines in the raster object.

Vector Warping.

Line elements in vector objects are made up of strings of coordinates connected by straight lines called segments. Lines can have too many vertices for the projected use, which slows down displays and many other uses. Applying appropriate line thinning can improve this condition.

Lines can also have too few vertices and consist of very long line segments defined by only 2 end vertices. Long line segments are quite acceptable as long as the internal geometry of the object is not changed. Since warping a vector does change its internal geometry, these long straight line segments are broken into much shorter line segments to approximate the computed curvature. Warping a vector object provides an option for this line densification. The effect of this option is illustrated in the attached partial color plate entitled Controlling Curvature When Warping Vectors.

The approximation of the computed curvature added by densification during warping to all line segments is controlled by the Line Densification Accuracy you enter. This value specifies the maximum deviation of every new, shorter line segment from the actual computed curve. Since the vector object is georeferenced, this value also corresponds to the maximum allowable displacement from the curve on the ground in meters or any other **TNT** linear measurement unit you select. It should be apparent that as you make this value smaller you are increasing the size of your vector object and thus increasing the time it takes to use it in other processes.

Convert Regions to Vector Polygons.

A new process is available to select multiple region objects and convert them to separate vector objects. Previously you could only accomplish this by going through several unintuitive operations in the Spatial Data Editor if all you wanted was to make the conversion to establish topology, add attributes, or use vector object operations such as export. This new process is located at Process/Convert/Region to Vector. It operates like many other similar **TNT** conversion processes. You select one or more region objects, define the output vector objects, and Run. This step is illustrated in the partial color plate entitled Convert Regions to Vector Polygons.

Directional Analysis.

The colors of the elements making up the rose diagram presented in this view can now be changed to enhance its subsequent use in publications. The process has also been modified to ensure that it always has a circular shape when the window is resized.

Georeferencing.

An object you have imported may only have an approximate affine georeference. An option is now available to create an initial control point subobject using the corners of this object. This option can be used to establish its initial rough position to make it available for adding accurate georeferencing. If you create approximate control points with this option they should be deleted after you add your accurate control points.

Fourier Frequency Filtering.

The image destriping portion of the frequency filtering process has been recoded. You can access this feature as before at Process/Raster/Filter/Frequency Filter. The dramatic results of applying this procedure are illustrated in the attached color plate entitled Destriping ASTER Images.

Mosaicking.

A new layout can now be saved over the current mosaic layout.

Map Calculator.

The order of the coordinates can be defined for a "Read File" operation (for example, Lat/Lon or Lon/Lat, N/E or E/N, XY or YX, and so on, shown as XY or YX).

Transfer Attributes.**Huge Polygons.**

The transfer of attributes has been very slow if the process encountered individual polygons each with several million vertices (for example, 10 million). This feature has been redesigned so that it is now practical to use in these huge polygons and takes a fraction of the previous time (which means, minutes instead of hours).

Transferring Line Attributes to Polygons.

A Split At Border operation has been added for use when transferring attributes from lines to polygons. The standard line attribute table that is attached to the polygons by the operation is modified so the length reported is only the length of each line that falls within that polygon. For example, you make a grid cell vector object and transfer attributes from lines to polygons using the Split At Border option to have the length of the roads in each grid polygon attached to that polygon.

*** Spatial Data Editor.****Periodic Automatic Backups.****Setting Preferences.**

Automatic backups can now be programmed for your Spatial Data Editor operations. A new Backup tabbed panel has been added to the Setup/Preferences window. It provides the means to set if, how, and when these backups will occur and where they will be recorded. This new panel is illustrated in the attached color plate entitled Automatic Backup Options When Editing.

Fixed Time Intervals.

The new panel provides a toggle to enable/disable automatic backups. It displays the path to your current backup folder and provides a means to specify or change it. The time interval between backups in minutes can be set and a toggle determines if you are asked if you want to permit the backup at this time or have it automatically completed.

Idle Time Interval.

As an alternative to being interrupted at the fixed time interval, you can specify how long the Editor is idle in seconds before it performs an automatic backup. A judicious selection of this time will provide you with automatic backups when you leave your desk or temporarily turn to some other task. If this idle time backup is performed, the fixed interval backup timer is reset. As a result, you can use these 2 timers together to fit your

habits and get your backups made with a minimum of interference with your work in the Editor. A backup is not created if you have made no changes since your last backup.

On Demand.

You can backup any time you like by using the new File/Backup Objects menu option. This immediately creates new backup objects. When your objective is to temporarily preserve your work, this backup operation is automatic and faster (see below) than the Save or Save As operation you have been using. This operation is illustrated in the attached partial color plate entitled Editable Object Backup on Demand. This option was added after the Release Version **RV6.8** CDs were reproduced. Thus, you will need to acquire the 1st or a later patch to **RV6.8**, which will be available by the time you read this.

Number of Backups.

You can specify how many backups of the specific object you are editing will be kept in your backup folder. When this count is exceeded, the oldest copy of that object is deleted. You may be working on multiple objects and separate backups of each are made in your backup folder. As a result, you can set the maximum number of files that can be in your backup folder. When this number is exceeded, the oldest file in the folder will be purged. You can not specify more than 1 backup folder since it will be automatically used in several operations that will restore your backup. Set the number of backup files you allow to a large number and provide lots of drive space. This is even a good place to use a second drive or a removable drive.

Drive space is cheap compared with your time. Keep a lot of backups around!

Using Backup Objects.

How are Backup Objects Named?

During your editing session, the object you are editing is an object in a temporary workspace. When an automatic backup is made, the contents of this workspace are all saved as a backup object in a new Project File. The name of the Project File in the backup folder containing your backup object is the name of the object you are currently editing followed by the date and time the backup was performed. This is illustrated in the attached color plate entitled Automatic Backup Options When Editing.

Why Is It Faster?

The purpose of a backup object is to permit you to restore your editing operation to the workspace to match its condition at the time of the backup so you can resume editing from that point. As a result, this automatic backup is faster than when you save your editing results into a Project File. It uses the operating systems fastest copy to simply mass move the contents of the workspace to the backup object along with undo/redo options, and so on to another location on the drive. Topology, search trees, vector optimization, or other final operations are not performed as the purpose of a backup object is to restore it for editing. Your backups will be even faster if the drive used is not the same drive as used for you temporary files and, thus, the workspace. Synchronous fast copy between drives may be faster than asynchronous drive read and writes on the same drive.

Restoring a Backup.

When you start up the Editor, you have the new option File/Open Backup. This choice will open a Select Object window showing all your backup files for immediate selection. You can then select any backup object from the files in your backup folder. Fast copy will move that object to a temporary file and you can resume editing it where you left off. If by some rare chance the Editor should abnormally exit or otherwise crash, you will be asked if you want to load the latest backup files for editing when you restart.

Isolating a Possible Error Condition.

If you are experiencing error in editing a layer, you can use this backup procedure to assist MicroImages in reproducing this error, which we must do before we can fix it. The Spatial Data Editor is a very complex process often acting on a very complex object. Furthermore, as part of your editing this object, you can perform a complex, difficult, and hard to repeat sequence of interactive operations. Thus, it is often difficult for you to determine what sequence of your operations results in an error. When this occurs, MicroImages often can not reproduce the error and, thus, can not fix it. By backing up very frequently you can iterate back to a prior backup object. By repeating this process, you may be able to isolate the sequence of editing operations that causes the fault to occur. You can then provide MicroImages with the last useful backup and a description of the steps that cause it to fail. MicroImages can then load your backup object, repeat the steps, get the error, and fix it for the next patch.

Interoperation of Tools.

You can now switch between drawing tools and element types without losing any of your previous incomplete work in any tool. When you return to that tool, you will find it in exactly the incomplete state you left it in. If you are using a tool, such as drawing a line, you can now start another tool and later switch back and resume drawing the line where you left off. The attached color plate entitled Interoperate Tools When Editing illustrates a very simple operation where, during the tracing of a line, the view is zoomed without losing the incomplete line. This new capability is simple to explain, but a big time saver. It permits you to move between tools to complete a complex operation requiring several tools. For example, you can draw a line, switch to drawing a polygon, draw part of a polygon, switch back to resume drawing more of the line, and so on.

Miscellaneous.

Auto-Adjustment of Extents.

When the extent of a vector, CAD, or TIN object is reset when saved because you have made an edit alteration to reduce its extent (for example, you remove a stray outlying point element whose inclusion causes inflated extents in a view), the save action will also cause a redraw using these new extents. Optionally, you can shut this off in your preferences so that the saved extent is not reflected on the screen and there is no automatic redraw.

BSplining Lines In XYZ.

A 3D line in a vector layer can now be selected and splined using Cubic BSplining or Quadratic BSplining. The line will be splined in the XY plane and new values will be interpolated for the Z coordinates of the new vertices created by the splining. The effect of this kind of splining is illustrated in the attached partial color plate entitled Spline Lines in 3D.

Filtering Vectors.

The Spatial Data Editor can perform the same new filter operations described immediately below for the separate Vector Filter process. Zoom and pan operations are now available when using the test feature of the vector filter routines.

Object Properties

All three coordinate types (2D-XY, 3D-XY, and 3D-XYZ) can now be set in the Object Properties dialog on the Coordinate Type option menu. 3D-XY is for vectors that have lines with only a single Z value stored as an attribute, such as contour lines.

Vector Filters.

New options are available in the Vector Filters process available at Process/Vector/Filter.

* Filter By Script.

You can now design your own filters using the new Remove By Script option. This uses the same scripting procedures used elsewhere in other processes. This new procedure is defined in detail in the attached color plate entitled Filter Vectors Using Scripts. When you create a vector object from the import of a CAD object or from geodata imported from some other non-topological system, you frequently encounter a mess that you must clean up. This new scripting tool can be of significant value in cleaning up many of the graphical artifacts in these kinds of vector objects.

When lines are removed by script, 3 options are now provided for managing their attached attribute records. *Combine* will combine the database records if possible for the attachment type of the table. It is not possible for One to One, Implied One to One, and One Record per Element attachment types. *Use polygon with largest area* will use only the record from the largest area among each group of polygons being combined into a new polygon. Apply this option, for example, when sliver polygons are being removed. *Use polygon with smallest area* will use the record of the smallest polygon the group.

Dissolve Polygons.

When polygons are being dissolved, the same three attribute combination options are now provided that were mentioned above for the Remove By Script filter. You might use the *Use polygon with smallest area* option when you are dissolving island boundaries to assign the island attributes to the resulting polygon.

* Text Layer Controls.

What You See is What You Get (WYSIWYG).

The **TNT** multilingual Text Layer Controls in Spatial Data Display now shows text as What You See Is What You Get (WYSIWYG) when you are creating text layer and multi-object legends for your layouts. Since these Text Layer Controls use 2-byte Unicode encoding and fonts, this provides considerable ease in adding styling to layouts in any language supported by the **TNT** products. Now you can easily mix styling within your text block. As you type the text, it will now appear in the text block in the Text Layer Controls window in the style, color, size, and font you have selected. At any time you can highlight a portion of the text in this window and change its style, size, and font. The color plate entitled WYSIWYG Text Editing illustrates these features and how much easier they permit you to directly create styled map legends.

The styles you can now select in the Text Layer Controls window for direct entry or assignment to existing (selected) text strings are Normal, Bold, Italics, Underline, Enhanced, Shadow, Outline, Kerning, and Smooth. The effect of Kerning and Smooth are not WYSIWYG.

Show Formatting Codes.

V6.70 required you to use style control codes to mix text styles in your text block. **RV6.8** still uses style control codes but simply interactively inserts them into your text stream for you and then uses them to render your text. Under some circumstances you may want to view your text in this coded form. For example, there are codes you may have inserted, such as a tab stop, that do not show in the WYSIWYG rendering. The right mouse button brings up a menu that now has a toggle choice of Show Formatting Codes. This choice is also on the Edit menu. This toggle will change your WYSIWYG styled text to code styled text in your standard interface font. You can then toggle back to WYSIWYG using the Show Styling toggle button that now appears on that same menu.

Locating/Entering Special Characters.

Multi-lingual 2-byte fonts can be large and contain many glyphs in many languages. Even in your native language it may be hard to find the location and keyboard entry procedure for a specific character. This can be even more complex if it is a symbol font. Maps often may use these kinds of special fonts and glyphs. In **TNT**, you can now visually review every glyph in a font and interactively enter it in the text string in the Text Layer Controls.

The right mouse button or the Edit menu in the Text Layer Controls now provides a Character Map option. This choice exposes a new Character Map window, which is illustrated in the attached color plate entitled Inserting Special Characters. This window shows you a scrolling table with every glyph in the font you currently have selected in the Text Layer Controls window. It also shows you the glyphs' Unicode value, its descriptive name, and its keyboard entry code. You do not need the keyboard code because a double click of your mouse on any of this information will select that glyph and insert it at your cursor position in the text block in the style, color, and so on, you have selected. You can manually insert the same glyph using <F2>, the glyph's entry code, and <F2> again.

The Character Map window can be accessed wherever you are entering text in your **TNT** product. When you open this window, it stays open for interactive use until you close it or finish that text-oriented activity. You can open it via the right mouse button in the Query Editor, the Text Layer Controls, and other locations, even when you are entering or editing text in a database field. For example, you could insert a special glyph in a database string field and then later insert that same glyph into a search string in a query. Using this window with a database table is illustrated and discussed in the attached color plate entitled Inserting Special Characters.

Map Layouts.

Recent new releases of the **TNT** products have provided you with a variety of new features tailored specifically toward making all types of map and complex display layouts. These enhancements to map appearance have continued in **RV6.8**. For example, map grids and map marginalia can vary greatly according to personal preference and be-

tween national map series specifications. Options are now provided to customize these features to meet a wider variety of requirements.

New features are also provided to assist you in designing your layout components. These include easier to use tools for assigning styles to vector elements (see previous section entitled Management of Vector Styles). Styling now appears as you type in your language into text blocks when preparing multi-object legends (see previous section entitled Text Layer Controls).

Perhaps the most important additions are the new and improved tools to put all these new components together in your layouts. These include procedures for faster and interactive layout design. These are illustrated in the attached color plate entitled Diagrammatic Map Layout Tools.

Layout View.

What You See Is What You Get (WYSIWYG) is very important in designing an attractive map, poster, or engineering report with the **TNT** placement tools available in the Layout View window. However, it takes time to render a complex layout in this form. A WYSIWYG view of your layout is most useful when you are quality checking your design. But its rendering is slower, making movement of groups slow. It is also hard to present dimensions and group relationships within this rendering as they get intermixed and overlaid on the contents of the groups.

Placement Tool.

The Placement tool that lets you reposition groups is now more interactive than before. Changing cursor shapes indicate this tool's action at various locations on your Layout View window. The crosshair cursor indicates that you are over the active group, which can be repositioned if you simply click and drag. The left hand cursor shows that you are over any of the other groups all of which are not the active group. A click with it on any of these groups will make that group the active group (and show the crosshair cursor). The double arrow cursor means you are close to the arrowhead end of an attachment line and a click will let you drag this end of an attachment to another legitimate attachment point. The left arrow cursor indicates that you are not over a group's position box or an attachment arrow.

Faster Redraws.

Clicking the right button in the Layout View window when the Placement tool is active now presents you with a popup menu with toggle buttons for Wireframe View and Solid in addition to Redraw and Lock Scale choices. By default these toggle buttons are not selected and you automatically get the previously available WYSIWYG Layout View for each redraw for your quality control steps. If you select Wireframe View all your groups will be represented by their outline position box only and the view will redraw very fast. If you select the Solid toggle with the Wireframe toggle also on, the position box of each of the groups will be rendered as a solid color polygon, which also renders very fast. These toggle settings persist for your redraw actions as long as you keep the window open or until you change them. Using these 2 new rendering options, you can quickly position groups and see the dimensions of their relative placement as described below.

Lock Scale.

This option appears on the same right mouse button menu as the Wireframe View and Solid. However, it will only appear when this right mouse button is clicked on a position

box representing a specific group. Toggling on this option will lock the scale of the group selected in this manner.

Attachments and Dimensions.

For any of the rendering modes for your Layout View window, you now see attachment lines to show how each group is attached to another or to the layout margin. However, it is often the easiest to see and work with these lines in the Wireframe mode. The arrowhead on 1 end of this line indicates the direction of the attachment. If your cursor is positioned over any attachment line a ToolTip will appear. This ToolTip shows the separation of the 2 groups in the linear measurement units/language you have selected at the design scale you specified for your layout (which means, on the layout if it is printed at the design scale). Attachment and the horizontal and vertical position of the group can also still be viewed and precisely entered in the Group Settings window.

Grids/Tick Marks.

Individual interior tick marks and interior border ticks may not show up well if an image is used in a map, since the image may vary in color and density. You can now interactively select an individual tick mark and toggle its color between the two selected to guarantee that it will contrast with its surroundings. Exterior tick marks in the margin are usually in a white margin background and are often rendered in black. Tick marks inside and outside the margin may line up at the margin and appear as one but are now separate marks and can have their own separate color. The controls for setting up these new color options are illustrated in the attached color plate entitled Controlling Color of Map Grid Tick Marks.

Clipping Limits.

You can now add an overall sketch layer to a group that will ignore group clipping limits. Use this layer to manually add annotations and special labels and other marginalia into the margin or background around a group.

Now when you use a layout template, it will prompt you to enter any needed group clipping limits. This streamlines the process of creating a series of similar maps using a template because clipping is not turned off when the template opens with the new objects in place and the new map grids are generated to fit the clipped extents.

Marginalia.

Grid and tick labels can now be selected from a wide variety of UTM and Latitude/ Longitude formats. Many of these formats and other marginalia options and new features are illustrated in the attached color plate entitled Map Grid Labeling Options.

The spacing between the automatically generated marginalia labels and the ends of the grids, exterior tick marks, or map margin can now be specified.

An N, S, E, or W direction indicator can be optionally added to Latitude/Longitude labels.

UTM labels can be truncated to 1000s of meters by omitting the trailing "000." When this format is used, full UTM coordinates are placed for reference at the bottom left corner.

Map corner coordinates can be expressed with an accuracy of 1/10 or 1/100 of a second or meter. All other grid/tick mark labels can be suppressed so that only the corners are labeled.

When grid/tick labels and corner coordinates overlap only the grid/tick labels will show.

Converting Layouts.

Converting between layouts is a complex task since they are not well defined file formats that can be exported or imported. Layouts, including those prepared in the **TNT** products, are designed as containers for a specific end use in electronic publication or for printing. They can contain or use many different components in many different formats, some of which can be proprietary, such as fonts. Each layout concept has its own principle end-use or design objective, such as PDFs for electronic distribution, Adobe Illustrator for illustration, SVG for web access, and **TNT** for cartography. To maximize their utility in meeting these objectives, they vary in the flexibility, control, and scalability. For example, hatch patterns and their control are more important to cartography than to illustration. The management of fonts and images are other objectives that can vary widely.

Efforts continue to improve the conversion of layouts prepared in **TNTmips** and **TNT-edit** to PDF, SVG, and AI files. Scalability of certain aspects of the components in these formats can be a problem. These limitations can occur when you use cartographic features in **TNT** layouts that are not available in the format you are converting to. If your objective is to prepare a layout in **TNT** and then convert it to another layout format, you simply have to become aware that some features will not scale properly when converted and require that you keep the end use or design scale and fonts in mind when preparing your map or image layout if they are to be used or printed from some other layout format. You will also find that these conversions will present dialogs to control these variable aspects for which there is no 1-to-1 conversion as well as how other variable aspects of the target layouts, such as options for font and image linking or embedding, should be handled.

Converting Hatch Patterns.

You can now convert hatch patterns used in your **TNT** layouts into hatch patterns in PDF, SVG, and AI layouts.

Both relative and absolute scaling can be used for the styling of vector elements to determine how they are rendered in the **TNT** products. However, map cartographic design often uses symbols and hatch patterns that are very specific in the shape and the size of their components and are independent of the scale of the map they are to be used in. These styles are said to be absolute in size and do not vary with changes in the maps overall scale. If this was not the case, and the hatch patterns and symbols were scaled up and down with their use, you would get ugly maps when they were printed to varying scales. But, of much more importance, is that many maps are designed for wide scale use by the general public. Thus, the fill for a map unit is used to identify what is present in the unit, but deliberately does not try to convey anything about its quantity or quality. Letting these patterns, symbols, and line styles change when zooming or printing at different scales might be misinterpreted by the maps' users as a change in the amount or value of the thing it represents.

The problem that arises is that the other common layouts and their associated uses do not support absolute hatch patterns. This means that while a hatch pattern can be designed for use in **TNT** at an absolute scale, this associated information can not be accommodated for these characteristics in these other layouts. Thus, zooming in and out in other products that use these layouts will enlarge or shrink the absolute hatch patterns transferred into them from a **TNT** layout. This is illustrated in the attached color

plate entitled Publish Maps Containing Hatch Patterns. This means, just as in other aspects of converting layouts, that you must be aware of the end destination (for example, PDF, SVG, or AI) and use when you design a **TNT** layout, and design your hatch patterns appropriately.

Improved “Print To” Adobe Illustrator.

The AI file format of Adobe Illustrator has many similarities to PDF file format. Text management is now handled the same as for PDF files in **V6.70**. Circles and arcs are preserved as geometric shapes. Text along a curved baseline in a **TNT** layout follows the curved baseline in the AI file. Variable raster transparency and nulls are converted. Empty layers are no longer created (for example, if you create a text block and then do not put anything in it). Note, PDF files you create in the Adobe Acrobat Distiller can have things like empty text layers created in some other language (for example, Japanese) that will then hang Acrobat Reader in English if that font is not installed, even though everything in populated text blocks is in English. Bit map fill patterns are converted to polylines but are now scaled to 300 dpi. Hatch patterns are converted. Faux bolding for text is not converted as it is not a feature in the AI format.

Improved “Print To” SVG.

Layouts converted to SVG can now include style information. SVG layouts support embedded images and rasters in PNG format or linking to them as external files. You can now specify whether raster objects used in your **TNT** layout should be converted to PNG and embedded or linked as external files. Since these PNG components can be compressed or not you are provided the opportunity to specify that they should use best lossy compression. You can also select to convert to uncompressed rasters if they are categorical (for example, solid polygons fills) or you wish to have the original unaltered rasters and images available, perhaps for further editing of the target layout.

The sample JavaScript, which can be embedded in the SVG file, has an additional option to display map coordinates as you move the cursor over the display.

Improved “Print To” PDF.

Options are now provided to specify how fonts used in the **TNT** layout should be used in the PDF layout. These include using linking to system fonts (no copyright problems), embedding fonts (always guaranteed to be available), and rendering text into PDF shape elements (polygons but with limited scalability and no searchability). If you have used **TNT**'s older non-TrueType fonts, an option is available to auto convert them to a Courier TrueType font.

Spatial Manipulation Language (SML).

The use of **SML** continues to expand, placing new requirements on this spatial scripting language. At one extreme, it is being used by some to build very large repetitive production processes. Another area of application is to build specialized interactive solutions. Some of you simply use it as it was originally conceived—to solve a geospatial analysis problem unique to your needs or application. Requests for new functions are continually received, and these are added whenever possible as noted. Some interesting specialized **SML** scripts were created during this release cycle and are provided as new samples. Tighter checking of **SML** script syntax was requested and is now automatically applied to your scripts. However, development for **RV6.8** was principally focused upon one commonly requested feature, an easier procedure to create custom

interface dialogs for use in both native Windows and X environments and a wider variety of components for use in these dialogs.

XML Dialogs.

SML now supports the creation of user interface dialogs whose components are specified in the widely-used Extensible Markup Language, commonly known as XML. The same XML description of your dialogs will now work in **SML** for Windows (**SML/W**) or **SML** for X (**SML/X**). The older method of creating your dialogs for **SML/X** alone using OSF/Motif classes is still available for those who wish to continue to use it. The attached color plate entitled Creating SML Dialogs compares the code using Motif classes and the XML description needed to present the same simple dialog with toggle buttons and push buttons. By reviewing it, you will find it easier to understand how to create your new dialogs in XML. Now your dialogs can be used in cross-platform applications when you use **SML/X** for Linux, UNIX, Mac OS X, and Windows or **SML/W** for native Windows.

The following color plates introduce in detail the concept of using XML descriptions to build up the visual components of your dialog. Sample XML descriptions are provided for all the common elements present in a Windows dialog. The material these plates contain is fairly complete and will be used to build a **TNT** Tutorial booklet on this subject. While using XML in this fashion provides a quicker and easier way to create **SML** dialogs, you still have the complex task of “hooking-up” each interface component to the portions of the **SML** script that carry out the actual operations on your geospatial data.

Build SML Dialogs Using XML.

This color plate introduces the available XML dialog elements. It also illustrates their use in sample dialogs whose XML descriptions can be obtained for modification and use from www.microimages.com/freestuff/smlscripts.htm.

Sample Dialog Descriptions in XML.

This color plate shows a series of simple dialogs using a variety of elements and provides their XML descriptions with explanatory annotations.

Menus in SML Dialogs using XML.

This color plate shows a sample dialog for a drop-down menu and its XML description with explanatory annotations.

SML Dialog with Tabbed Pages.

This color plate shows a dialog whose general contents are divided into different sets of controls by using two tabbed pages, or panels. The XML description is provided along with explanatory annotations.

Nested SML Dialogs using XML.

This color plate and its reverse side illustrate a real application of a complex interface described in XML that uses a variety of different elements. This is the interface created to control the user input into the sample **SML** script described below in the section Suppressing Vegetation in Multispectral Images. Important segments of this **SML** script and its use of XML are listed and discussed on the reverse side of this plate. The complete **SML** script for use in **SML/X** or **SML/W** along with its embedded XML interface description is available from www.microimages.com/freestuff/smlscripts.htm.

Communicating with ActiveX-Compliant Programs.

What Is It For?

The XML dialogs introduced above now provide an easier method for adding the interface to directly collect and control the inputs needed by your **SML/X** and **SML/W** scripts. However, it is often necessary to communicate from a **TNT** process via **SML** and combine this with input from another non-**TNT** program or series of programs. This could be accomplished in **V6.70** in a rudimentary fashion using command-line and other procedures. Now in **RV6.8** this interprocess communication can be accomplished using a non-**TNT** program that is registered as a Windows ActiveX component. **TNTmips**, **TNTedit**, and **TNTview** are not Windows products and are not ActiveX-compliant and so cannot be used as components by other non-**TNT** programs. But **SML** now provides the capability for an **SML** script to launch an ActiveX component program, communicate information to it, and retrieve information back from it. The ActiveX component can be coded to interact directly with other ActiveX-compliant Windows programs, but its interaction with your **TNT** data is completely under the control of your **SML** script. Using an ActiveX component to communicate with non-**TNT** programs provides considerable programming flexibility by providing more complex kinds of control, exchanges of information, and safe, controlled integration of the **TNT** products into other systems.

A Simple But Widely Adaptable Application.

Introduction.

The simplest example of this new capability is to use an ActiveX component program to present an attractive form to collect information from its user about a point, line, or polygon vector element that they select interactively using an **SML** Tool Script in a **TNT** View window. The identity of the selected element can then be used by the ActiveX component to locate the record corresponding to that element in an external table. The form can then be used to edit or collect additional information about that element and rewrite the altered record into the original database table.

Suppose the polygon selected is a farmer's field. The farm field has an identification code attribute that is passed to the ActiveX component program when it is selected using the **SML** Tool Script. The ActiveX component uses this code to find the record in an external database containing the dollar value of the previous crop raised in that farm field. This previous crop value is displayed in the form. The image displayed with the polygons of farm fields shows that the selected field is fallow (without crop) for the current year. The form is used to change the value of the crop for this farm field in the original database, which is part of some other system.

Using a Visual Basic (VB) Form.

This simple process can be implemented using **SML** and Visual Basic as follows. In a **TNT** View window, add an **SML** Tool Script, which places an icon button on the toolbar of all 2D Views in all processes. When this icon button is pressed, the associated **SML** Tool Script opens an XML dialog that requests that you select a polygon with the mouse. Your selection of any polygon automatically starts a Visual Basic (hereafter VB) program (ActiveX component) and passes it the ID code of the selected polygon. The VB component locates that ID code in an external database and retrieves the appropriate record and populates the form with the current values from that record. If the record is empty or does not yet exist, the cells remain blank and a new record is created. The VB program then permits you to fill in or edit the values in the form's fields.

The VB program defining this form can be as complex as you like and use all sorts of pick lists, constraints, and other filters on the allowed values. When the form is fully completed and you click the OK button, the VB program rewrites the altered record or writes the new record back into the external database. This sequence of operations is illustrated in the attached color plate entitled Communicate with Visual Basic Programs using SML. The key parts of this **SML** Tool Script and the associated VB form program are shown and annotated on the reverse side of this color plate. Both the sample **SML** and this very simple VB program can be downloaded for your trial and modification from www.microimages.com/freestuff/smlscripts.htm.

Return Data to TNT.

Your VB program may only need to edit or populate a table in an external database using information from a polygon displayed in a **TNT** view. However, the database used could be linked to the elements in the **TNT** vector object so that, in effect, you can use a VB form program to edit or populate the attributes of the polygon. This same VB form approach can also be used to display, create, and alter the attributes in a record stored in an internal **TNT** attribute table. After the VB program action is complete the Tool Script retrieves the information entered in the VB form and updates the appropriate record in the original attribute table. In either case, using an internal or linked table, the next step in your **SML** Tool Script could be to redraw the **TNT** view, which could then show the changes in this polygon's attributes by a change in its drawing style (color or fill pattern).

Why Use Visual Basic?

This entire activity could be coded in **SML** but would require more knowledge of **SML** than simply using a Tool Script shell, the precoded element selection procedures, and an optional simple XML dialog. However, the use of VB to create an ActiveX component also provides several advantages. Visual Basic provides a graphical form designer as well as an Application Wizard, either of which make it easy to create attractive forms and other interface components, even for a beginning programmer. VB is also often the only programming language that is taught as part of many professional undergraduate degree programs since it is the easiest programming language to use to complete a project. There are extensive precoded libraries and components that can be purchased or obtained without cost for use in any VB program. There are extensive books, sample code, and nearby friends who can help.

Using ActiveX in Visual Basic.

You can find information on building a VB program with ActiveX controls in the Microsoft Visual Basic 6.0 Component Tools Guide between pages 229 and 243. These pages were consulted for the information needed to create the ActiveX communication used in the sample VB program illustrated in the attached color plate entitled Communicate with Visual Basic Programs Using SML. This is one of the books in the 3-volume Microsoft Visual Basic 6.0 Reference Library, which is widely available and should be consulted by anyone actively using Microsoft's VB.

Using C++, Java, ...

The use of VB in connection with **SML** does not have to be limited to this simple form and database editing project. Spatial or attribute data can be passed to a VB component program by an interactive **SML** Tool Script or by a more batch-oriented Macro Script, and the VB component program can then communicate with other ActiveX-

compliant programs to process and analyze the information. Furthermore, this strategy is not limited to component programs coded in VB. Other languages, such as C++, Java, and others, support creating ActiveX components that can act on data and actions passed to them from the **TNT** products using **SML/X** or **SML/W**.

Understand the Limitations.

Throughout these examples you will note that the **TNT** product remains in control and starts the actions of the external component program via one of the various kinds of **SML** scripts. Any information generated by or through the component program is only communicated back when explicitly requested by code in the **SML** script. Any subsequent action in a **TNT** process, such as a redraw, takes place only in response to commands in the **SML** script. **TNT** processes were designed to be highly interactive, are not native Windows programs, and are not available as ActiveX components. Thus, it is not possible to develop a non-**TNT** program that communicates directly with or exerts control over a running **TNT** process.

Important: **TNTmips**, **TNTedit**, and **TNTview** are still cross-platform using X, are not native Windows programs, and are not ActiveX-compliant or available as ActiveX components.

Production Applications.

SML Tool Scripts can provide the basis for new and interactive approaches as outlined above. **SML** is also used for production applications that use large scripts to prepare specialized products. For example, DigitalGlobe uses a set of **SML** scripts to produce its commercial AgroWatch products. In this application a sequence of **SML** scripts are used to transform the original SPOT and QuickBird multispectral images into vegetation index, soil, and other map products for individual fields. These scripts are organized to produce a smooth production workflow that can produce the final products on CD within hours after the image becomes available. During this flow the operator interacts briefly with these scripts where needed to ingest control points, calibration values, field outlines, and other input parameters. The scripts apply proprietary image analysis algorithms during this smooth production flow to automatically produce final field maps on a CD in the form of a **TNTatlas** as well as in other exported formats for convenient use in other products.

Sample Scripts.

Suppressing Vegetation in Multispectral Images.

A mining exploration company brought this image analysis procedure to the attention of MicroImages by sending in an **SML** script with a request for some assistance in debugging it. With their permission, a user interface was added to the script and it is available now as part of the script library at www.microimages.com/freestuff/smlscripts.htm. Their **SML** script is based upon the paper:

Unveiling the Lithology of Vegetated Terrains in Remotely Sensed Imagery by Robert E. Crippen and Ronald G. Blom. August 2001. Photogrammetric Engineering and Remote Sensing. Vol. 67, no. 8. Pages 935-943.

The concept is based upon the idea that a cell of a multispectral image of a land area has multispectral values each made up of the combined effects of vegetation and the ground surface. If the vegetation cover is light, its contribution to the value of the cell in

each spectral band can be measured by computing the vegetation index for that cell. The value of the cell in the different multispectral bands can then be adjusted using this vegetation index to reduce the contribution of the vegetation to that cell. The adjusted value for the cell in each spectral band now represents the spectral return of the ground surface only.

This technique can be applied to Landsat, ASTER, or other multispectral imagery, and is effective at mitigating the contribution of vegetation cover and exposing the ground surface provided the vegetation cover is light, such as in arid regions (for example, Nevada, South Africa). Thus, from a geologist's or pedologist's point-of-view, the technique can effectively expose the surface lithology over a wide area when these adjusted image bands are displayed in various color combinations. The attached color plate entitled Suppressing Vegetation in Multispectral Images illustrates the results produced by this script applied to Landsat Thematic Mapper images in California and Nevada.

Making Color Separates for Printing.

Significant development efforts in a number of releases prior to **RV6.8** have been focused on adding the features needed to support preparing layouts suitable for professional cartographic map production. This has been prompted in part by the use of **TNTmips** in the national map-making programs of several nations. You have also requested a wide range of new features and improvements for your wide variety of unusual project layouts and printed products.

National mapping programs and other organizations that print maps in large quantities do not print them on large-format inkjet printers with dithered colors. They print maps on large, expensive printing presses with each color printed separately in its own ink. The map data is often input into the printer as a set of TIFF images, one for each color feature layer and ink. The different color ink layers are not overlaid or mixed at any point on the print. Thus, each feature layer has large null areas where none of its features exist and where features in overlying layers cross. If an image background is present it may be printed as a dithered grayscale or color background with holes left for the many solid-color feature overlays.

When you print a map layout to an inkjet printer, **TNTmips** creates a color-composite raster image of the layout for transmission to the printer. However, **SML** can be used to break down this composite to provide the TIFF color separates suitable for use in these large map production runs. A sample **SML** Macro Script that performs this task is illustrated in the attached color plate entitled Making Color Separates for Printing and its major script components are described on the reverse side. The script can be downloaded from www.microimages.com/freestuf/smlscripts.htm. It illustrates the basic approach by processing a map layout with a grayscale orthophoto background and color overlays into the separate TIFF files required for printing: a grayscale TIFF file for the background and a binary TIFF file for each overlay color. While this **SML** was written to produce a specific type of map in quantity and, thus, contains color and ink specifications unique to that map type, it provides a good example of the overall approach and illustrates why you will occasionally need to use **SML** in this kind of application. Each production map and associated printing contractor will have different colors, order of printing, input format, and many other potential variables that can only be accommodated using a scripting language such as **SML**.

Strict Syntax Checking.

Every new script you create in the **SML** or Query Editor window will now automatically include the preprocessor command “\$warnings 3.” This keyword invokes strict syntax checking when the script is run. This feature is illustrated in the partial color plate attached entitled Strict Syntax Checking in SML. Adherence to strict syntax rules involves the following additions to conventional **SML** syntax:

- all variables must be declared before they are used in a statement,
- assigned variable values must match the declared variable type, and
- all statements must end in a semicolon (“;”), but note that preprocessor commands, including “\$warnings 3,” are not normal statements and should not end in a semicolon.

Strict syntax errors will not prevent your script from running. Any violations are simply shown as list entries in the Script Warnings window. You can delete the “\$warnings 3” command from the script, and it will no longer be checked for these strict conditions when run. However, when you use the Syntax menu on the Editor window to manually check syntax as you write or edit a script, strict syntax checking is automatically applied in addition to the conventional syntax check, even if the “\$warnings 3” command is not in the script. Thus, if you edit one of your **SML** scripts written prior to **RV6.8**, strict syntax warnings may appear when you manually check syntax. You can turn off strict syntax warnings during manual syntax checking, as well as when the script is run, by using the preprocessor command “\$warnings 0.”

New Functions.

RasterCompositeToHIS

Converts a composite color raster to 3 separate rasters (Hue, Intensity, and Saturation).

ResourceLookupLabel

Look up localized label from tntxres.txt.

ResourceLookupMessage

Look up localized message from messages.txt.

ResourceLookupTitle

Look up localized title from tntxres.txt.

New Classes.

class POLYLINE

Holds a single polygon without islands.

class MdispPOLYLINETOOL

A tool on a view for drawing a polygon.

class CADELEMOPT

Optional style information for CAD elements.

New XML-Related Classes.

The following classes are part of the XML forms implementation:

class GUI_CTRL_COLORBUTTON

A colored button that pops up a color selection dialog when pressed.

class GUI_CTRL_COMBOBOX

A combobox control.

class GUI_CTRL_LISTBOX

A listbox control.

(There are other control type classes that already existed before **RV6.8**.)

class GUI_DLG

A dialog window. Can be created from an XML description or “the old-fashioned” way.

class GUI_FORM_RADIOGROUP

A group of mutually-exclusive radio buttons.

class GUI_FORMDATA

Container for all of the values in an XML Form.

class GUI_PANE_XML

A layout pane whose contents are described by XML.

class HTMLDOC

An HTML document.

class XMLDOC

An XML document.

class XMLNODE

A node in an XMLDOC tree.

New Keywords.

\$warnings 3

Preprocessor command to set strict syntax checking when script is run.

\$import

Preprocessor command to import an ActiveX/OLE component.

Upgrading TNTmips.

If you did not purchase **RV6.8** of **TNTmips** in advance and wish to do so now, please contact MicroImages by FAX, phone, or email to arrange to purchase this version. When you have completed your purchase you will be provided with an authorization code by FAX. Entering this authorization code while running the installation process lets you to complete the installation of **TNTmips RV6.8**.

The prices for upgrading from earlier versions of **TNTmips** are outlined below. Please remember that new features have been added to **TNTmips** with each new release. Thus, the older your version of **TNTmips** relative to **RV6.8**, the higher your upgrade cost will be.

Within the NAFTA point-of-use area (Canada, U.S., and Mexico) and with shipping by UPS ground. (+150/each means US\$150 for each additional upgrade increment.)

<u>TNTmips Product</u>	<u>Price to upgrade from TNTmips:</u>					<u>V6.20</u>
	V6.70	V6.60	V6.50	V6.40	V6.30	and earlier
Windows/Mac/LINUX	US\$500	750	950	1100	1250	+150/each
for 1-user floating	US\$600	900	1140	1320	1500	+180/each
UNIX for 1-fixed license	US\$800	1250	1650	2000	2250	+200/each
for 1-user floating	US\$960	1500	1980	2220	2640	+240/each

For a point-of-use in all other nations with shipping by air express. (+150/each means US\$150 for each additional upgrade increment.)

<u>TNTmips Product</u>	<u>Price to upgrade from TNTmips:</u>					<u>V6.20</u>
	V6.70	V6.60	V6.50	V6.40	V6.30	and earlier

Windows/Mac/LINUX	US\$600	900	1150	1400	1600	+150/each
for 1-user floating	US\$720	1080	1380	1680	1920	+180/each
UNIX for 1-fixed license	US\$900	1400	1850	2200	2500	+200/each
for 1-user floating	US\$1080	1680	2220	2640	3000	+240/each

Installed Sizes.

Loading **TNTmips RV6.8** processes onto your hard drive (exclusive of any other products, data sets, illustrations, and so on) requires the following storage space in megabytes.

	<u>for V6.70</u>	<u>for RV6.8</u>
PC using W95, W98, WME, NT, W2000, or XP	101 Mb	154 Mb
PC using LINUX (with Intel) kernel 2.0.36 to 2.4	150 Mb	195 Mb
Mac using Mac OS X 10.x	132 Mb	155 Mb
SGI workstation via IRIX	193 Mb	170 Mb
Sun workstation via Solaris 2.x	171 Mb	206 Mb
IBM workstation via AIX 4.x (with PPC)	223 Mb	275 Mb

RV6.8 of the Online Reference Manual in PDF, including illustrations, requires 52 Mb. Installing all the sample geodata sets for **TNTlite** and **TNTmips** requires 247 Mb. The 71 Tutorial Booklets require a total of 149 Mb. The sample TNTsim3D landscape files require a total of 127 Mb.

Internationalization and Localization

Operating Languages.

Note! If your language is missing, please contact MicrolImages for information on plans to add it or to discuss becoming its official translator.

Improved.

Arabic and Chinese operation of the **TNT** products have been significantly improved with the greatly appreciated help of our Resellers.

New.

The **TNT** products can now be operated in Bosnian, Croatian, and Serbian.

Pending New.

Translations of the **TNT** products user interface are being prepared for Swedish, Malaysian, Tamil, and Telugu.

Not Current.

The translation of the interface files for Indonesian, Bulgarian, and Hungarian are no longer being maintained and as a result will not be issued for **RV6.8**. New official translators are needed for these languages.

MicrolImages Authorized Resellers

The following 7 new Resellers in 5 nations were authorized to sell MicrolImages' products since **V6.70** shipped.

Pakistan.

Lahore

Digitek

Hassan Mian
36-J Gulberg-III
Lahore
Pakistan

voice: (9242)586-1266
eFAX: (425)699-7640
email: digitek@post.com
email: digitek@pakistans.com

Spain.

Madrid

URBITEC NETWORKS S.L.

Marina Emelianova
Isaac Newton 1
of-37 Centro E
Parque Tecnologico Madrid
28.760 Tres Cantos, Madrid
Spain

voice: (3491)803-7244
FAX: (3491)803-8668
email: TNTproducts@urbitec.net

USA.

Tucson, AZ

Statistical Research, Inc.

Christopher D. Dore
P.O. Box 31865
Tucson, AZ 85751-1865
USA

voice: (520)721-4309
FAX: (520)298-7044
email: cdore@sricrm.com
www: www.sricrm.com

Coconut Creek, FL

Satellite Imaging Group

Sean Morrissey
5033 NW 50th Court
Coconut Creek, FL 33073
USA

voice: (954)418-9314
FAX: (306)653-0223
email: flyingrhino@earthlink.net

Longmont, CO

ALPI Services Inc.

Pieter Viljoen
1728 Whitehall Drive
Longmont, CO 80501
USA

voice: (720)217-8543
FAX: (303)776-9639
email: pviljoen@direcway.com
www: www.aggrowmarket.com

Zimbabwe.

Harare

Geospatial Solutions Zimbabwe

Kuda Muhwandagara
133 Blakeway Drive
P.O. Box BE 683
Belvedere, Harare

voice: (263)477-8476
FAX: (263)477-8476
email: kuda@taurai.co.zw

Zimbabwe

Discontinued Resellers

The following resellers are no longer authorized to sell MicroImages' products. Please do not contact them regarding support, service, or information. Please contact MicroImages directly or one of the other MicroImages Authorized Resellers.

Australia.

Geo Mapping Technologies. [David Moore] located in South Brisbane is discontinued.

Canada.

ENS Mapping, [Laurie Matheson] located in Calgary is discontinued.

Ecuador.

Aeromapa Cia, Ltda., [Gonzalo Lopez Jacome] located in Quito is discontinued.

Germany.

AVENA, GmbH, [Britta Planer-Friedrich] located in Markt Schwaben is discontinued.

Germany.

IGwU, GmbH, [Barbara Sperling] located in Dresden is discontinued.

Latvia.

Envirotech, LLC, [Janis Dzelzitis] located in Riga is discontinued.

Nigeria.

Business Systems Solutions, [James O. Emadoye] located in Lagos is discontinued.

Panama.

MapIntec Geotechnologies Inc., [Zorel Morales] located in Panama City is discontinued.

Peru.

G.D. Systemas, [Gabino Alva Infante] located in Lima is discontinued.

USA, CA.

Archaeological Mapping Specialists. [Christopher D. Dore] located in Berkeley is discontinued.

USA, CA.

Anthony Williams. [Anthony Williams] located in Monterey is discontinued.

USA, CO.

Common Sense Ag Consulting. [John Radowca] located in Fort Collins is discontinued.

USA, SC.

EPIC Creative Service. [Michael Maloney] located in Rook Hill is discontinued.

USA, WA.

Catafina Geographic Systems. [Rob Kolosvary] located in Vancouver is discontinued.

Appendix: Abbreviations

For simplicity, the following abbreviations were used in this MEMO

W95 = Microsoft Windows 95.

W98 = Microsoft Windows 98.

WME = Windows Millennium Edition.

NT or NT4 = Microsoft NT 4.0 (the **TNT** products require the use of NT4.0 and its subsequent Service Packs). NT4 now has a Service Pack 6a available. Windows 2000 now has Service Pack 2 is recommended if you are working with large files.

W2000 = Microsoft Windows 2000.

XP = Microsoft Windows XP Professional.

XP Home = Microsoft Windows XP Home Edition.

XP Tablet = Microsoft Windows XP Tablet PC Edition.

Mac 10.5 = Apple Macintosh using Mac X version 10.5.

MI/X = MicroImages' X Server for PC microcomputer platforms and operating systems.

GRE = MicroImages' Geospatial Rendering Engine, that is at the heart of most MicroImages products. The current **GRE** will respond and render requests from either X/Motif or Windows.

VB = Visual Basic

Gb = gigabyte (1000 megabytes) or 10^9 bytes

Tb = terabyte (1000 gigabytes) or 10^{12} bytes