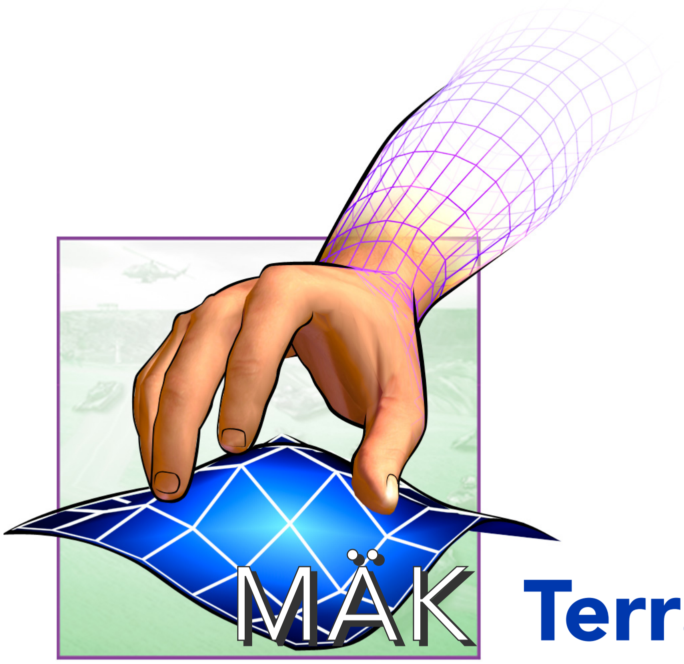
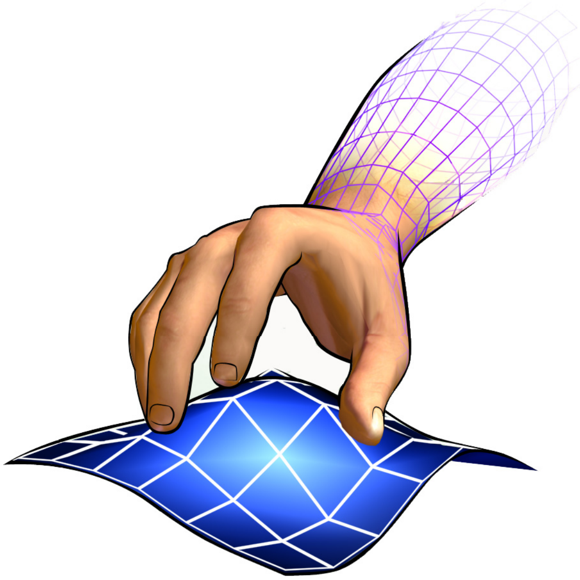


Terrain Database Tool

Users Guide

VT MÄK
A company of VT Systems



MÄK Terrain Database Tool

Users Guide

VT MÄK

A company of VT Systems

For third-party license information, please see "Third Party Licenses," on page xii.

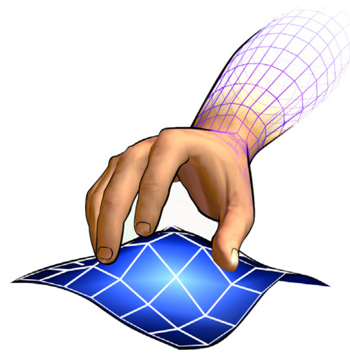
VT MÄK
68 Moulton St.
Cambridge, MA 02138 USA

Voice: 617-876-8085
Fax: 617-876-9208

info@mak.com

www.mak.com

Revision TDB-4.2-1-090707



Contents

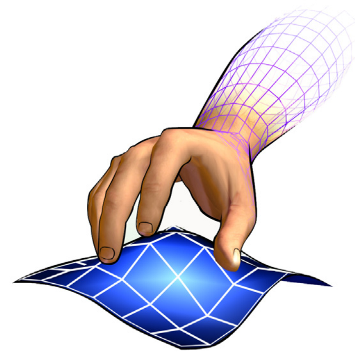
Chapter 3 Editing GDB Terrain Databases

Chapter 4 Editing Images

Chapter 7 Metafile Records

Appendix A The MÄK Specification Model

Index



Preface

This manual is written for persons who will use the MÄK Terrain Database Tool (TDB Tool). The manual assumes that you are familiar with your operating system and that you know how to work in your computer's graphical window environment.

Please see release documentation for information about changes and updates to the TDB Tool since this manual went to press.

How This Manual is Organized

The chapters are organized as follows:

Chapter 1, *Introduction* provides a brief introduction to the TDB Tool and MÄK Terrain Format.

Chapter 2, *Managing Projects* explains how to start the TDB Tool and how to create and manage terrain projects.

Chapter 3, *Editing GDB Terrain Databases* explains how to edit terrain soil types and polygon colors.

Chapter 4, *Editing Images* explains how to edit images for your terrains.

Chapter 5, *The TDB Tool Command Line Interface* explains how to use the command line interface to get access to all features of the TDB Tool.

Chapter 6, *Navigating and Configuring the Map* covers basic use of the graphical user interface (GUI).

Chapter 7, *Metafile Records* explains terrain metafile formats and parameters for the various database formats supported by the TDB Tool.

Chapter A, *The MÄK Specification Model* describes the MÄK specification model used when vector network data is imported into GDB databases.

TDB Tool Documentation

The TDB Tool is distributed as part of VR-Forces, VR-Vantage, and MÄK Plan View Display. The following documentation for the TDB Tool is included with those products:

- *MÄK Terrain Database Tool Users Guide*.
- Online help.

Online Manuals

The online (PDF) version of *MÄK Terrain Database Tool Users Guide* uses hypertext links for the entries in the table of contents, for all index entries, and for cross references. PDF documentation also supports full text searching.

To view and print PDF documents, you need a copy of the free Acrobat Reader software (version 5.0 or later). To obtain a copy, visit www.adobe.com.

MÄK Products

The MÄK Terrain Database Tool is a member of the VT MÄK line of software products designed to streamline the process of developing and using networked simulated environments. The VT MÄK product line includes the following:

- VR-Link® Network Toolkit. VR-Link is an object-oriented library of C++ functions and definitions that implement the High Level Architecture (HLA), the Distributed Interactive Simulation (DIS) protocol, and the Test and Training Enabling Architecture (TENA). VR-Link has built-in support for the RPR FOM and allows you to map to other FOMs. This library minimizes the time and effort required to build and maintain new HLA or DIS-compliant applications, and to integrate such compliance into existing applications.

  VR-Link includes a set of sample debugging applications and their source code. The source code serves as an example of how to use the VR-Link Toolkit to write applications. The executables provide valuable debugging services such as generating a predictable stream of HLA or DIS messages, and displaying the contents of messages transmitted on the network.
- MÄK RTI. An RTI (Run-Time Infrastructure) is required to run applications using the High Level Architecture (HLA). The MÄK RTI is optimized for high performance. It has an API, RTIspy®, that allows you to extend the RTI using plug-in modules. It also has a graphical user interface (GUI) for the MÄK RTI rtiexec.

- VR-Forces®. VR-Forces is a computer generated forces application and toolkit. It provides an application with a GUI, that gives you a two dimensional view of a terrain database.

  You can create and view local entities, aggregate them into hierarchical units, assign tasks, set state parameters, and create plans that have tasks, set statements, and conditional statements. VR-Forces also functions as a plan view display for viewing remote entities taking part in an exercise. Using the toolkit, you can extend the VR-Forces application or create your own application for use with another user interface.

- VR-Vantage™. VR-Vantage is your 3D visualization solution. It includes two end-user applications, MÄK Stealth and Vantage IG, and the VR-Vantage Toolkit. The Stealth displays a realistic, 3D view of your virtual world. You can view this world from the inside of a simulated moving vehicle, or place the eyepoint at another moving or stationary location. The Stealth lets you switch rapidly among several predefined viewpoints while the simulation is underway.

  Vantage IG is a configurable desktop image generator for out the window (OTW) scenes and remote camera views. It has most of the features of the Stealth, but is optimized for its IG function.

  The VR-Vantage Toolkit is a 3D visual application development toolkit. Use it to customize or extend Stealth and Vantage IG or to integrate VR-Vantage capabilities into your custom applications. VR-Vantage is built on top of OpenSceneGraph (OSG). The toolkit includes the OSG version used to build VR-Vantage.

- MÄK Data Logger. The Data Logger, also called the Logger, can record HLA and DIS exercises and play them back for after-action review. You can play a recorded file at speeds above or below normal and can quickly jump to areas of interest. The Logger has a GUI and a text interface. The Logger API allows you to extend the Logger using plug-in modules or embed the Logger into your own application. The Logger editing features let you merge, trim, and offset Logger recordings.
- MÄK Plan View Display. The Plan View Display (PVD) provides a 2D, "big picture" of a virtual environment. It shows simulated entities, such as vehicles, soldiers, and tanks moving over a 2D-map display.
- VR-Exchange™. VR-Exchange allows simulations that use incompatible communications protocols to interoperate. For example, within the HLA world, using VR-Exchange, federations using the HLA RPR FOM 1.0 can interoperate with simulations using RPR FOM 2.0, or federations using different RTIs can interoperate. VR-Exchange supports HLA, TENA, and limited DIS translation.
- MÄK Gateway. The Gateway is a translator application that allows DIS exercises to operate as members of HLA federation executions that use the RPR FOM, thereby preserving the value of simulation code written to support DIS. The MÄK Gateway is a standalone application and does not require you to write additional code for its use.

How to Contact Us

For TDB Tool technical support, information about upgrades, and information about other MÄK products, you can contact us in the following ways:

Telephone

Call or fax us at:	Voice:	617-876-8085 (extension 3 for support)
	Fax:	617-876-9208

E-mail

Sales and upgrade information:	info@mak.com
Technical support:	support@mak.com
VR-Vantage support:	vrv-support@mak.com

Internet

MÄK web site home page:	www.mak.com
License key requests:	www.mak.com/support/licenses.php
Product version and platform information:	www.mak.com/support/productversions.php
For the free, unlicensed MÄK RTI:	www.mak.com/products/rti.php
Support FAQ:	www.mak.com/support/faq.php

Post

Send postal correspondence to:	VT MÄK 68 Moulton St. Cambridge, MA, USA 02138

When requesting support, please tell us the product you are using, the version, and the platform on which you are running.

Document Conventions

This manual uses the following typographic conventions:

`Monospaced`	Indicates commands or values you enter.
`Monospaced Bold`	Indicates a key on the keyboard.
`Monospaced Italic`	Indicates command variables that you replace with appropriate values.
`{ }`	Indicates required arguments.
`[ ]`	Indicates optional arguments.
|	Separates options in a command where only one option may be chosen at a time.
(|)	In command syntax, indicates equivalent alternatives for a command-line option, for example, `(-h | --help)`.
/	Indicates a directory. Since MÄK products run on both UNIX and Windows PC platforms, we use the / (slash) for generic discussions of pathnames. If you are running on a PC, substitute a \ (backslash) when you type pathnames.
Italic	Indicates a file name, pathname, or a class name.
sans Serif	Indicates a parameter or argument.
➤	Indicates a one-step procedure.
Menu → Option	Indicates a menu choice. For example, an instruction to select the Save option from the File menu appears as: Choose **File → Save**.
i	Indicates supplemental or clarifying information.
!	Indicates additional information that you must observe to ensure the success of a procedure or other task.

Directory names are preceded with dot and slash characters that show their position with respect to the TDB Tool home directory. For example, the directory *tdbtool/doc* appears in the text as *./doc.*

Hardware and Operating System Naming Conventions

Unless otherwise specified, references to UNIX include Linux, regardless of the type of computer it is running on. References to PCs refer to Intel processor-compatible computers running a version of Microsoft Windows.

Mouse Button Naming Conventions

An instruction to click the mouse button, refers to clicking the primary mouse button, usually the left button for right-handed mice and the right button for left-handed mice. The popup menu refers to the menu displayed when you click the secondary mouse button, usually the right button on right-handed mice and the left button on left-handed mice.

Third Party Licenses

MÄK software products may use code from third parties. This section contains the license documentation required by these third parties.

Boost License

VR-Link, and all MÄK software that uses VR-Link uses some code which is distributed under the Boost License. All header files that contain Boost code are properly attributed. The boost web site is: www.boost.org.

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

libXML and libICONV

VR-Link and all MÄK software that uses VR-Link, links in libXML and libICONV. On some platforms the compiled libraries and header files are distributed with MÄK Products. MÄK has made no modifications to these libraries. For more information about these libraries please see the following web sites:

- The LGPL license is available at: http://www.gnu.org/licenses/lgpl.html
- Information about IconV is at: http://www.gnu.org/software/libiconv/
- Information about LibXML is at: http://xmlsoft.org/

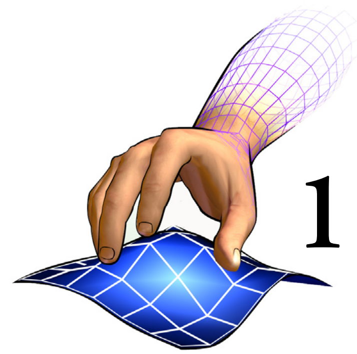
pThreads Library

VR-Exchange links with the pThreads win32 library. The library is distributed under the GNU Lesser General Public License (LGPL). MÄK has made no modification to this library. For information about the pThreads win32 library please see: http://sourceware.org/pthreads-win32/

EHS, PCRE, and PME

The MÄK RTI links with the EHS, PCRE, and PME libraries. These libraries are distributed under the GNU Lesser General Public License (LGPL). MÄK has made modification to these libraries only to allow them to compile on the supported platforms. For more information about these libraries please see the following web sites:

- EHS: http://xaxxon.slackworks.com/ehs/
- PCRE: http://www.pcre.org/
- PME: http://xaxxon.slackworks.com/pme/index.html



1

Introduction

This chapter describes the TDB Tool and how to install it, and introduces the MÄK Terrain Format and MÄK Terrain Project files.

1.1. The TDB Tool

The MÄK Terrain Database Tool (TDB Tool) is a utility program that converts terrain databases into MÄK Terrain Format (GDB) for use with MÄK applications such as VR-Forces, MÄK Plan View Display (PVD), and MÄK Stealth. VR-Forces and the PVD use the TDB Tool implicitly when they import databases. Stealth customers use the TDB Tool as a utility to convert terrain databases to GDB format for display in the Stealth.

The TDB Tool has a command line interface and a graphical user interface (GUI). If you run the TDB Tool without command line options, it runs in GUI mode. If you use any of the command line options, it runs without the GUI.

The TDB Tool GUI lets you:

- Create MÄK Terrain Projects (MTP), from which you can generate GDB terrain databases
- Create MÄK Terrain Descriptor files (MTD), which allow you to associate raster images with GDB files
- Edit the soil types of polygons
- Add vector data to GDB terrain databases
- Apply color to polygons
- Split images and create multiresolution papermap files for inclusion in MTD files.

---

i The MÄK applications that include the TDB Tool do not necessarily support all of the features of the TDB Tool or the files it generates. Please see the documentation for your application for a list of limitations, if any. Throughout this manual we may refer to MÄK products that are particularly relevant to the TDB Tool feature being described.

1.2. Installing the TDB Tool

The TDB Tool usually is installed as part of the primary MÄK application that you have purchased. If you receive an updated TDB Tool independently of another MÄK application, do one of the following:

- On Windows, run the installer
- On Linux, untar the files into the directory of your choice.

1.3. MÄK Terrain Format

MÄK Terrain Format (GDB) stores polygonal data and associated surface information, and vector data and its associated traits. There are no grid or elevation tables in the GDB file.

The polygonal data can consist of triangles, polygons, or both, with the restriction that all polygons must be convex, planar, solid (no holes or other degeneracies), and do not contain multiple vertices that are colinear. Almost all polygons in GDB files share vertices that are within a specified distance of each other to reduce memory usage. The default is 0, to ensure correctness so that only exactly the same vertices are shared by default. However, you can override this default when you are sure it is safe to be more aggressive.

The vector data is a series of points and edges, where edges are composed of segments. The edges can be connected head-to-tail to form areal features or left unconnected to form linear features. Each vector feature has a set of associated characteristics, which provide the necessary detail for usage. For example, a linear feature composed of many segments may be used to represent a road. The associated characteristics may contain the width of the road, the number of lanes, the specific feature type code, a name, or any other useful data the user has associated with it.

The GDB file format supports referencing other GDB files from within a GDB file. An approximate representation of a terrain section can be stored in the parent GDB file, along with a reference to the actual GDB file, should it become necessary to load it. (This provides limited paging of terrain data.) A single parent GDB can have many references to child GDB files.

1.3.1. Database Formats Supported by the TDB Tool

The TDB Tool supports the following database terrain and vector data formats:

- GDB
- OpenFlight
- MetaFlight
- DTED
- ESRI Shape
- CTDB versions 4b, 7b, and 7l
- VMAP
- DFAD and DFD.

The TDB Tool can convert only those terrain databases supported by the platform on which it is running.

1.3.2. Converting Terrain Databases to MÄK Terrain Format

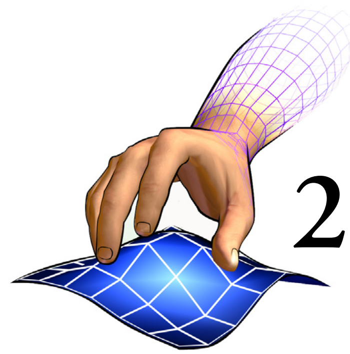
For VR-Forces or the PVD to use terrain database data, the data must be in GDB format. When one of these applications opens a non-GDB database, it accesses the TDB Tool to convert the database into GDB using default values. You can save a converted database to GDB format directly from these applications. However, when you use the default values, you do not optimize the resulting database for memory usage and performance. Therefore, it is strongly recommended that you convert external terrain files to GDB using the TDB Tool, so that you can fully optimize the files.

1.3.3. Geocentric Terrain Databases

VR-Forces, the PVD, and the Stealth usually run and display simulations that take place within one UTM zone. If you want to run a simulation over an area larger than a UTM zone, you must use a more appropriate projection. The GDB format supports Lambert Conformal Conic, flat earth, and geocentric projections. For large area databases, geocentric is recommended. You can load geocentric databases in any supported format and convert them to GDB, and you can convert databases that are already in GDB format to geocentric.

i

Because geocentric databases are only approximations of the curved surface of the earth, there can be odd interactions between the ground and object models if the sizes of the polygons are very large. Only the vertices are actually at the correct altitudes in a geocentric terrain, and as such, the interior of the polygons may report altitudes significantly lower than expected. This is an inherent problem in approximating a curved surface with flat surfaces, similar to trying to approximate a circle with an octagon.



2

Managing Projects

This chapter describes how to start the TDB Tool and how to create and save MÄK Terrain Projects.

2.1. Edit Mode and Project Mode

When you use the TDB Tool, you work either in project mode or edit mode. The mode is displayed on the status bar.

In project mode, you create or edit a MÄK Terrain Project (MTP). A project lists all of the data (terrain, vector, shape, and so on) that you want to combine into a GDB terrain. The output from a terrain project is a GDB file and a MÄK Terrain Descriptor (MTD) file, which lists the GDB file created by the project, the terrain database to use in the VR-Forces 3D Front End, and the image files associated with the project, if any. (This is because image data is not included in GDB files.)

Creating a terrain project allows you to convert terrain data to GDB format with a customized set of inputs and attribute values. Saving a project allows you to reproduce the GDB file as needed.

If you have a GDB file, you can load it in edit mode. In edit mode, you can:

- Edit the soil types of polygons
- Add vector features
- Apply color to polygons
- Associate images with the GDB file
- Manipulate the images that you want to associate with the GDB file
- Update the MTD file associated for the terrain database.

When you associate images with a GDB file, you save these associations in the MTD file. Thereafter, you can load the GDB file and its images into VR-Forces or the Plan View Display by loading the MTD file.

For details about edit mode and MTD files, please see Chapter 3, *Editing GDB Terrain Databases*.

2.2. Starting the TDB Tool

- To start the TDB Tool in Windows, choose **Start** → **All Programs** → **MAK Technologies** → *application x.x* → **Tools** → **TDB Tool**.
- To start the TDB Tool GUI on Linux or from the Windows command line, run *./tools/tdbTool/bin/tdbtool.exe* (for the Stealth), or *./bin/tdbtool.exe* (for VR-Forces or the Plan View Display).

The TDB Tool window opens with a minimal menu and nothing loaded.

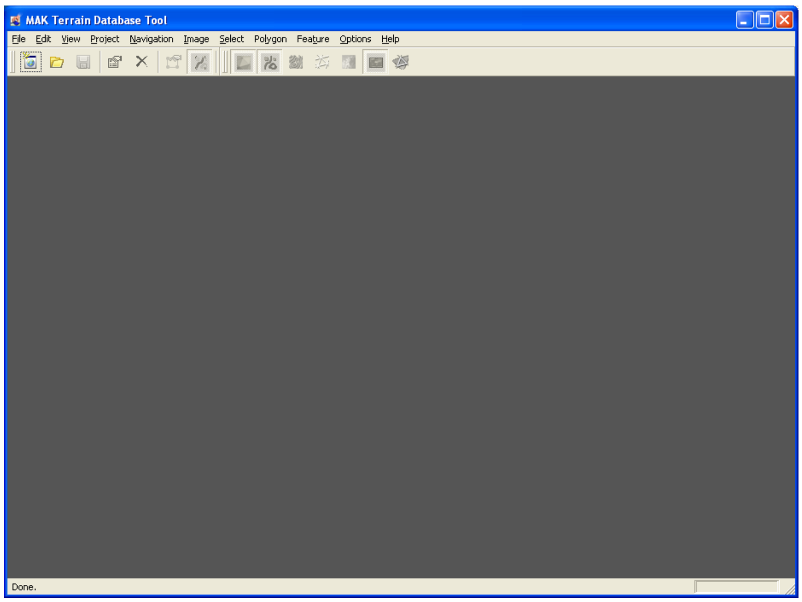


Figure 2-1. TDB Tool window at startup

2.3. Creating a Project

When you create a project, the TDB Tool clears the Project tab and the map of any existing data, becomes ready to import data for the project, and enters Project Mode. If you have not previously loaded a project or database, the Utility toolbar is added to the window (Figure 2-2).

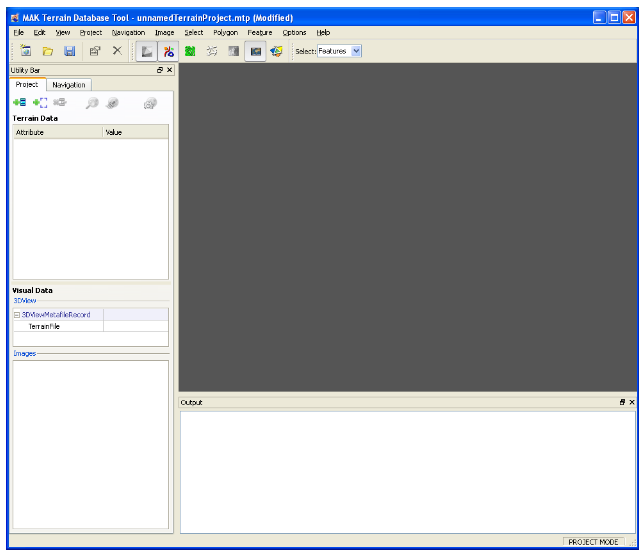


Figure 2-2. TDB Tool with a new project

The Project tab on the Utility toolbar has a set of buttons for frequently used tasks.

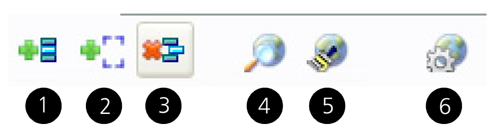


Figure 2-3. Project tab buttons

In the following list of tasks, the numbers reference the buttons in Figure 2-3:

- Add Data From File(1)
- Add Terrain Extent (2)
- Remove Data Record (3)
- View Terrain (4)
- Clear Map Area (5)
- Build Terrain (6).

For more information about the Utility toolbar, please see "The Utility Toolbar," on page 6-2.

➤ To create a project, choose **File** → **New** → **Terrain Project**.

2.3.1. Adding Data to a Project

You can add the following types of data to a project:

- Terrain database (for example, GDB, DTED, or OpenFlight)
- Vector data files (for example, DFD or shapefiles)
- Terrain extent
- Raster files (for example, BMP, JPG, or PNG).

! Although you can specify more than one terrain database or terrain extent in a project, when the terrain is built or viewed only one of them is used to build the GDB terrain, as follows:

- If you select multiple terrain databases, only the first selected terrain database in the list is used.
- If you select multiple terrain extents, only the first selected terrain extent in the list is used.
- If you select a mixture of terrain databases and extents, only the first terrain is used.

For details about importing particular types of data, please see the following sections:

- "Adding Multiple DTED Files," on page 2-8
- "Adding Images to a Project," on page 2-10
- "Importing Vector Data," on page 2-14
- "Importing OpenFlight Files," on page 2-17
- "Importing MetaFlight Files," on page 2-20.

To add data to a project:

1. Choose **Project** → **Add Data From File**, or click the Add Data From File button on the Project tab. The Add Data dialog box opens.
2. Select the data source that you want to import. (You an select multiple data sources at one time.)

---

i Files with the extension *.shp* can be terrain files or vector data files. Since the TDB Tool cannot tell which type of data it is from the extension, if you want to import shapefiles, you must explicitly choose Shape Feature Files or Shape Terrain Data Files in the File Type list before you select a shapefile.

If you need to add DTED files that are located in multiple folders, please see "Adding Multiple DTED Files," on page 2-8.

3. Click Open. A record for each datum is added to the project, for example, a GdbMetafileRecord for a GDB terrain, or a FltMetafileRecord for an OpenFlight terrain (Figure 2-4).

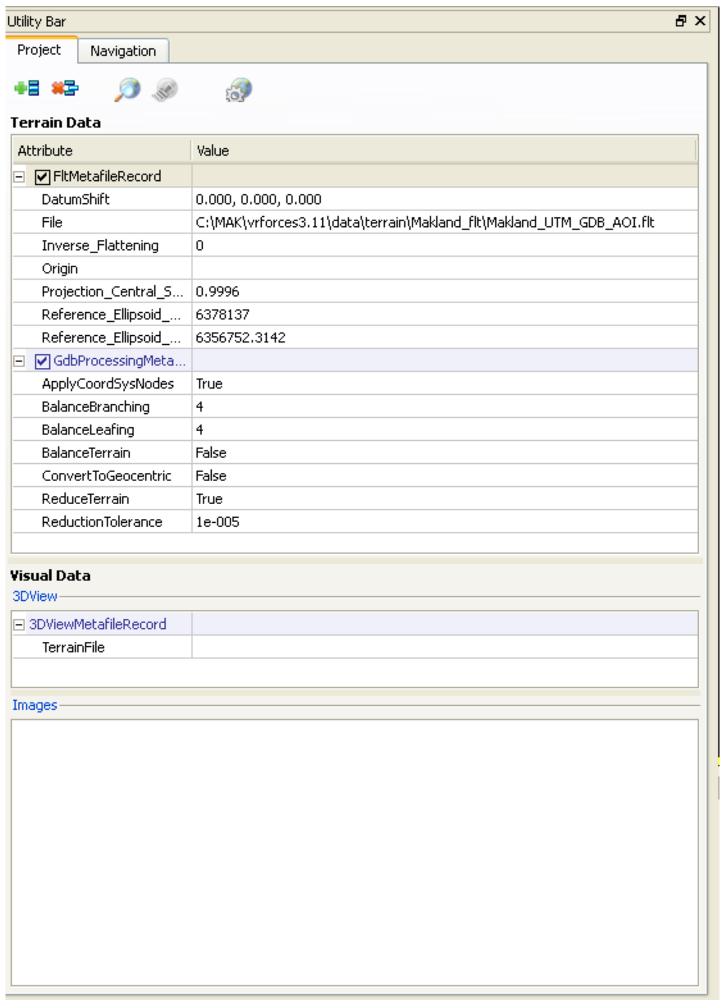


Figure 2-4. An OpenFlight record in a project

Adding Multiple DTED Files

You cannot add multiple level 0 DTED files that are located in multiple locations at the same time. You must add them using the metafile record File attribute.

To add multiple DTED files:

1. On the Project tab, in the Terrain Data group box, select the Open button next to the File attribute. The Edit Files dialog box opens.

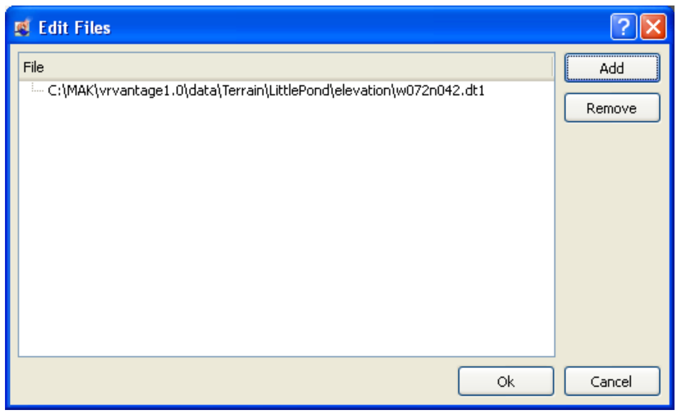


Figure 2-5. Edit Files dialog box

2. Click Add. The Add Files dialog box opens.
3. Select the appropriate DTED files.
4. Click Open. The files are added to the Edit Files dialog box.
5. Repeat steps 2 through 4 until all the DTED files you want are added.
6. Click OK.

Adding Terrain Extents to a Project

A terrain extent specifies an area that consists of a flat square composed of two or more triangles, depending on the tessellation level specified. You would use a terrain extent when you do not need polygonal data for your simulation or do not have any real data. Typically, you would add vector data or overlay a raster image, or both, on the terrain.

To add a terrain extent to a project:

1. Choose **Project** → **Add Terrain Extent**, or click the Add Terrain Extent button on the Project tab. The Extent Information dialog box opens.
2. Select the coordinate system from the list, and enter values for the Northeast and Southwest corners, and the TessellateX and TessellateY. TessellateX and TessellateY specify the number of square cells that compose the area. (A square cell is made out of two triangles.)
3. Click OK. An ExtentMetafileRecord is added to the project.

On the project tab, you can edit the coordinates of the extent.

2.3.2. Editing Data Attributes

When you add data to a project, a record for each data item is added. Each record type has a set of attributes. These attributes are assigned values based on the information in the data you are adding. Some attributes are assigned default values. You can edit these attributes on the project tab. Some attributes describe the characteristics of the terrain. Others provide instructions to the TDB Tool for optimizing the data when the GDB terrain is built. For details about optimizing databases, please see "Reduction," on page 2-22. For descriptions of the attributes, please see Chapter 7, *Metafile Records*.

> Some attributes are read-only.

To edit project data:

1. Expand the list of attributes for the record you want to edit by clicking the plus sign.
2. Click the value for the attribute you want to edit. The field becomes editable (Figure 2-6).

> Some fields are editable in place. Others require you to click a button to open a dialog box.

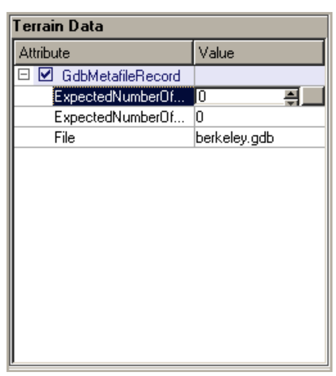


Figure 2-6. Editable attribute

3. Enter the new value.

2.3.3. Adding Images to a Project

You can associate multiple images with a terrain. Images are layered and only the top image displays unless you apply transparency to it. You can rearrange the order of images. For details, please see "Specifying the Display of Raster Map Layers," on page 6-12 and "Specifying the Transparency of Raster Map Layers," on page 6-14.

The TDB Tool can include multiresolution papermaps in a project. To create a multiresolution papermap, you use the TDB Tool's image splitting feature. For details, please see "Multiresolution Papermaps," on page 4-5.

---

i Like terrain databases, when you add an image to a project, it does not display. To see the image, you must view the project or build it.

To add an image to the terrain:

1. Do one of the following:
   – Choose **Project → Add Data From File**, or click the Add Data From File button on the Project tab. The Add Data dialog box opens.
   – Right-click in the Images group box on the Project tab and choose **Add Image**. The Add Image Data dialog box opens.
2. Select the image to add.
3. Click Open. The image is added to the project.

---

i When adding an image, the projection of the image must be the same as the projection of the terrain. For example, a geodetic image with a geodetic image file.

2.3.4. Specifying Visual Data (3DViewMetafileRecord)

When you create a project, the TDB Tool automatically adds a 3DViewMetafileRecord to the project tab. In an MTD file, this record specifies the terrain database to open in the VR-Forces 3D Front End for scenarios that use the MTD file. (The VR-Forces back-end uses the GDB file specified in the MTD file.) If your project is importing an OpenFlight or MetaFlight terrain, when the TDB Tool builds the project and creates the MTD file, it automatically specifies the FLT or MFT file in the 3DViewMetafileRecord. If you created the project using some other type of terrain, the 3DViewMetafileRecord is left blank unless you manually specify a terrain for the TerrainFile attribute. For details, please see "3D View Metafile Record," on page 7-10.

> **i** If the 3DViewMetafileRecord in an MTD file is blank, and you specify that MTD file for a scenario in the VR-Forces 3D Front End, VR-Forces opens the GDB file specified in the MTD file, unless you specify a different terrain when you create the scenario.

2.3.5. Viewing a Terrain

When you add data to a project, the TDB Tool does not automatically import the specified data and display it. The TDB Tool works this way because large terrains can take a long time to display. Not displaying it automatically can save you time. If you want to view the terrain data that you added to a project, you must specifically make the TDB Tool display it.

> **i** If you make changes to a project after you display the terrain, the display does not get updated automatically. You must view it again to see the changes.

To view the terrain:

1. Specify the records that you want to include in or exclude from the view, as follows:
   - To include a record, select the check box to the left of the record name.
   - To exclude a record from the view, clear the check box next to its name.
2. Choose **Project** → **View Terrain**, or click the View Terrain button in the Project tab. The TDB Tool displays the terrain in the map area.

2.3.6. Building a Terrain

When you add data to a project, the TDB Tool does not automatically build a terrain with the data. To create a GDB database, you must explicitly build the terrain. When you build a terrain database, the TDB Tool saves it to the name that you specify. It also creates an MTD file for the terrain, with the same name as the database.

! If you generated the terrain from a GDB file and you save the new terrain to the same name, it overwrites the old file. If you have made any changes from the source GDB file, you will not be able to recreate the original project. If you want to be able to recreate the original project, save the new terrain to a different name or keep a copy of the original GDB file in a different directory.

! A project can contain multiple terrain database and extent records. However, when you build a terrain, only one of the databases or extents can be used. If more than one is selected, the TDB Tool decides which to use as follows:

- If you select multiple terrain records, only the first selected terrain record in the list is used.
- If you select multiple extents records, only the first selected extent record in the list is used.
- If you select a mixture of terrain records and extent records, only the first terrain record is used.

To build a terrain:

1. Select the records that you want to include in or exclude from the terrain, as follows:
   – To select a record, select the check box to the left of the record name.
   – To exclude a record from the view, clear the check box next to its name.
2. Choose **Project** → **Build Terrain**, or click the Build Terrain button in the Project tab. The TDB Tool prompts you to enter a name for the terrain database.
3. Enter a name for the terrain database. The TDB Tool builds the terrain. (If the TDB Tool detects that the terrain conversion process may exceed the amount of available memory, it prompts you to change the conversion algorithm. (For details, please see “Building Large Terrain Databases,” on page 2-13.)) Then it displays an informational message that it has created the terrain and MTD file. It asks you if you want to go into edit mode.

4. Do one of the following:
   – To go into edit mode, click Yes. If you have unsaved changes to the project, you are prompted to save the project. Then the TDB Tool switches to edit mode.
   – To remain in project mode, click No.

Building Large Terrain Databases

Converting large terrain data source files to GDB using the standard algorithm can use up all of the available memory on a computer, which can cause the build process to fail. If the TDB Tool determines that building a database may exceed memory, it offers you the option of building it with an alternative algorithm. This algorithm may be much slower than the standard algorithm, but is more likely to be successful.

2.3.7. Saving a Terrain Database

To save a terrain, build it, as described in "Building a Terrain," on page 2-12. It is saved as part of the build process.

2.3.8. Deleting Records from a Project

You can delete records from a project. If you have already built or viewed a project that included the deleted records, the deletions are not automatically applied. You must view the terrain again or rebuild it for the deletions to be applied.

To delete a record from a project:

1. In the Project tab, select the record to remove by clicking its name.
2. Choose **Project** → **Remove Data Record**, or click the Delete Record button on the Project tab.

---

i The state of a record's selection check box does not affect record deletion. The name of the record must be selected to delete it.

2.3.9. Clearing the Map Area

Clearing the map area removes the terrain and raster images. It does not affect the contents listed on the Project Panel.

➤ To clear the map area, choose **Project** → **Clear Map Area**, or click the Clear Map Area button on the Project tab.

2.3.10. Saving a Project

When you save a project, the TDB Tool saves the records for all input data.

To save a project:

1. Choose **File** → **Save**. If this is the first time you have saved the project, the Save Project dialog box opens.
2. Enter a name for the project.
3. Click OK.

Saving a Project to a New Name

➤ To save a project to a new name, choose **File** → **Save As** and enter a name for the project.

2.3.11. Closing a Project

➤ To close a project, choose **File** → **Close**.

2.4. Importing Vector Data

You can include vector data, such as Shape, VMAP, DFD, and DFAD files, in a terrain project. When you build the terrain, the data is included in the resulting GDB file. The TDB Tool includes the following default mapping files for importing DFAD and DFD files into GDB:

- *dfad_sedris_map.txt* maps each type of DFAD/DFD feature to a default SEDRIS feature. You can edit this file or specify a different map file with the `tdbtool -e` option.
- *importFile.txt* defines the list of DFAD/DFD features to import into the GDB file. Each possible feature specifies IMPORT or NOIMPORT. You can edit this file or specify a different feature file with the `tdbtool -d` option.
- *sedris_colormap.txt* defines the color representation for each subcategory in the SEDRIS code, in the format RGBA. You can edit this file or specify a different color map file with the `tdbtool -c` option.
- *shp_sedris_map.txt* is used for importing shapefiles.
- *FID_SMC.txt* defines the list of FID and SMC codes for DFAD and DFD basic features.

! You cannot import vector data into geocentric databases.

For more information about importing vector data, please see Chapter 7, *Metafile Records*.

2.4.1. Clipping Vector Data to the Terrain Extent

By default, when you import vector feature data, it is clipped to the terrain extent. Clipping reduces the size of the database and removes unwanted data. Areal features that intersect the terrain are not clipped. You specify clipping with the **Clip** attribute.

2.4.2. Planarizing Vector Data

By default, when you import vector data, it is planarized. Planarizing flattens the vector data and adds intersections of edges. It may take an extremely large amount of computation time on large vector datasets. Therefore, be careful about planarizing your data sets. You can explicitly planarize vector data with the **Planarize**, **Tolerance**, and **ExcessiveEdgeCount** attributes in MTP files.

2.4.3. Loading Shape *.prj* Files

The data in shapefiles may be accompanied by a *.prj* file. The shapefile loader uses the data in the *.prj* file to project geodetic data to UTM. The loader only processes Central_Meridian, SPHEROID, False_Easting, False_Northing, and Scale_Factor data. If this information conflicts with the settings in the metafile record, it overrides the metafile record. The *.prj* file must be in the same directory as the shapefiles. If it is not present, the data in the shapefile is interpreted as geodetic information. For more information about shapefiles, please see "Shapefiles," on page 7-25.

2.4.4. Importing VPF Files

The structure of VPF files described in VPF Metafile Records"VPF Metafile Records," on page 7-14. This section describes how to add a VPF database to a project. Because VPF databases are created from a set of directories and files, it is not as simple a matter as just selecting a particular file to add to the project.

For this example, assume the area of interest is contained in a single database. You want to import a new feature type (roads) into your terrain database.

This creates a VPF Attribute layer for the terrain project. From this attribute layer the user must next select the Library they intend to use from the database. A VPF database may contain many libraries. For example the DNC database of the US has about 50 libraries for the different ports/coastal areas. The Lat/Long eventually defines which libraries are loaded. It is currently up to the user to select the library that encompasses the user's area of interest.

To add a VPF database to a project:

1. Open a project or create a new one.
2. Choose **Project** → **Add Data From File**, or click the Add Data From File button on the Project tab. The Add Data dialog box opens.
3. Select the Library Attribute Table (LAT) file for the database that you want to import. Since VPF uses the file system to organize itself, there is not any one file that represents the database. The TDB Tool uses the LAT file, which is in the root of the VPF directory as a starting point for loading the database. This creates a VPF Attribute layer for the terrain project.
4. Select the library you want to use from the database. A VPF database may contain many libraries. For example the DNC database of the US has about 50 libraries for the different ports and coastal areas. The Lat/Long eventually defines which libraries are loaded. It is up to you to select the library that encompasses your area of interest.
5. Specify the layer (trans) that you want to import.
6. Specify the feature class (road) that you want to import. You can select "All" to load all of the features in a layer.
7. Repeat this process for each layer that you want to import.

2.5. Importing OpenFlight Files

You can import OpenFlight database files with mesh nodes that have triangle strip primitives. (Triangle strips are the only primitive available and supported by MultiGen-Paradigm at this time.)

2.5.1. Support for Maps with Localized or Unlocalized Origins

The TDB Tool can load databases with both localized or unlocalized origins.

An OpenFlight terrain database may be composed of multiple OpenFlight files (or tiles.) In unlocalized databases, all OpenFlight files are built with respect to one origin. The OpenFlight reader loads terrains with “Local Origin Offset” set to zero or set to the same value for all the tile files.

In localized databases, each OpenFlight file contains its own origin information. The “Local Origin Offset” values can be different in the tile files. In Figure 2-7, each cell represents a terrain file. The dot represents an origin.

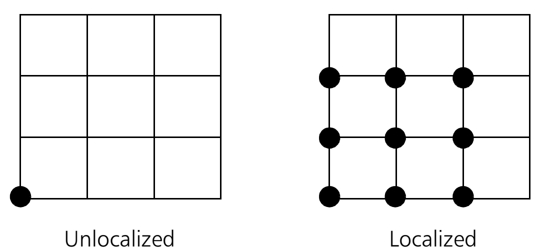


Figure 2-7. Unlocalized and localized terrains

2.5.2. Converting OpenFlight File Texture Data to GDB

OpenFlight files do not contain information about soil characteristics, which VR-Forces models need to calculate entity mobility.

To provide soil characteristic data for VR-Forces, when the TDB Tool loads an OpenFlight terrain database, it tries to deduce soil type information from the names of the texture files that are usually associated with OpenFlight files.

By default, the TDB Tool tries to determine the soil types by looking for keywords, such as road, street, grass, water, and so on, in the name of the texture file. If this method does not adequately capture soil type information in the file you are importing, you can map the texture files to soil types in the *./data/terrain/surfChar.map* file.

Table 2-1 lists the MÄK soil types to which you can map the OpenFlight texture files.

Table 2-1: MÄK Soil type

asphalt	dryGround	orchards	shallowRiver
boulder	forest	pavedRoad	shallowStream
building	grass	rock	softSoil
cultivatedFields	gravelRoad	sand	swamp
deepLake	mud	shallowLake	tree
deepRiver	ocean	shallowPond	USRailroad
dirtRoad			

Map the OpenFlight texture files to the supported soil types, for example, (from the *surfChar.map* file):

```
SurfaceCharacteristicMap
{
   Match      HL_water                       ocean
   Match      Water_Ocean_Deep_Blue          ocean
   Match      Tree_Evergreen                 tree
   Match      Tree_Canopy_Evergreen          forest
   Match      Road_Highway_Two-Lane_Asphalt  pavedRoad
   Match      Road_Street_Single-Lane        pavedRoad
}
```

The center column in each entry is an OpenFlight texture. The right column is a MÄK soil type.

i The list of supported soil types is fixed. Therefore, you cannot add new soil types.

2.5.3. Restricted and Unsupported Features

When you import OpenFlight terrain databases:

- The following nodes are not imported or displayed:
  - Light points
  - Light point system
  - DOF nodes
  - Text nodes
  - Sound.
- The following features are not imported:
  - UV map coordinates
  - Normals
  - LOD centers
  - Geospecific RGB attributes.
- The following face attributes are not supported:
  - Multi texture
  - Polygon type wireframe or closed wireframe
  - Shade
  - Line style
  - Transparency
  - Material
  - IR color
  - Feature ID
  - SMC
  - Cultural footprint
  - Roofline
  - Terrain.
- Billboards are converted to static polygons.

2.6. Importing MetaFlight Files

The TDB Tool can load MetaFlight files. The MetaFlight reader has the following abilities and restrictions:

- Loads maps with both localized and unlocalized origins.
- Coordinate system information is extracted from the first OpenFlight file loaded.
- Only loads files that contain a GeometryGridDataset node. FileListDatasets, VirtualTextureDataset, ImageGridDataset, VectorGridDataset, and GeneralGridDataset are not supported.
- Loads the highest level of detail available.
- Does not support paging.
- Solves filename patterns that contain DSNAME, DSTITLE, LEVEL, COL, and ROW entries.
- If the <Volume> and <Rootpath> entries in the <GeometryGridDataset> tag are set, the loader uses their values to build a full path to search for the files. If the resulting path is a relative path, it is relative to *./data/terrain*, in which case the files referenced in the *.mft* file do not get loaded. Therefore, make sure that the <Volume> and <Rootpath> entries resolve to a fully qualified path. If the <Volume> and <Rootpath> entries are empty, the loader looks for the files relative to the location of the *.mft* file.

2.7. Optimizing Databases Using the TDB Tool

The TDB Tool has options for optimizing GDB files for both memory usage (relevant to all applications) and runtime performance (relevant to VR-Forces), although optimizing one usually happens to the detriment of the other. The options are:

- OPCODE: A third-party collision library that is used by VR-Forces. It is the fastest intersection library, but also uses the most memory. It is particularly good when there are a lot of small polygons close together, such as detailed urban terrains. However, it cannot work with databases in which the coordinates have very large values, such as geocentric databases, or projected terrains that cover a large area.
- Spatial Subdivision: Provides an even grid index into the terrain. You can configure how many subdivisions are in each axis. It requires some extra memory, and makes terrain intersection calls faster, though typically not as fast as OPCODE. All coordinate sizes are supported, so it is a good choice for geocentric terrains or other terrains with large coordinate values.
- Simple: This method uses the native GDB calls for intersections. They are not optimized, so it runs the slowest of these options. It does not use any additional memory, so it is also the smallest. It is useful for very large databases when memory is the main concern.

2.7.1. Choosing the Terrain Intersector Type for a Terrain

You can select the terrain intersector type or let the TDB Tool choose. If you let the TDB Tool choose, it tries to optimize the database for speed. It will usually use OPCODE for those databases that support it.

After you generate a terrain database, the intersector type is listed in the Terrain Database Information dialog box. It is not part of the project file.

To choose the terrain intersector:

1. Choose **Edit** → **Choose Terrain Intersector**. The Choose Terrain Intersector dialog box opens (Figure 2-8).

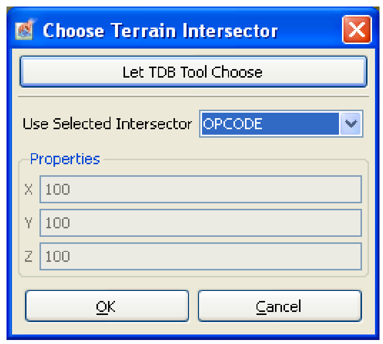


Figure 2-8. Choose Terrain Intersector dialog box

2. Select an intersector type from the drop-down list (Simple, Spatial Subdivision, or OPCODE). For a description of these options, please see "Optimizing Databases Using the TDB Tool," on page 2-20.
3. If you selected Spatial Subdivision, optionally specify the X, Y, and Z values. For details about specifying these values, please see "Creating Spatial Subdivisions," on page 2-22.
4. Click OK.

2.7.2. Creating Spatial Subdivisions

A spatial subdivision is a spatially sorted subdivision of all the elements of the terrain database's terrain group. It is designed to increase performance of terrain queries. The general algorithm is that the entire database is divided up into X by Y by Z "buckets." The relevant nodes are "placed" into each bucket and stored for quick lookup when needed.

In the best case scenario, the spatial subdivision sorts the nodes – mostly triangles, polygons, coordinate systems, and groups – into small groups that can be sorted and tested quickly. However, if improper settings are used, the spatial subdivision may not help performance or may require too much memory. To avoid this, try to set the number of divisions to be roughly associated with the average size and density of the polygons in the database. When memory is not a constraint, erring on the side of finer division is recommended. The default settings of 100, 100, 1, add on average, only one or two megabytes (MB) of size to the GDB files and improve performance for most terrain databases.

As a result of this complexity and the differences between terrain databases, you may need to try different settings before you find the best values. A guideline is to try to have 10 to 20 polygons per cell. For a relatively evenly distributed terrain, use the following formula:

```
square root (number of polygons/desired number per cell)
```

Use the result as the X and Y values. Z should be 1. For scenarios with many entities, large numbers of polygons, or other types of complexity, the improvements in performance can be dramatic.

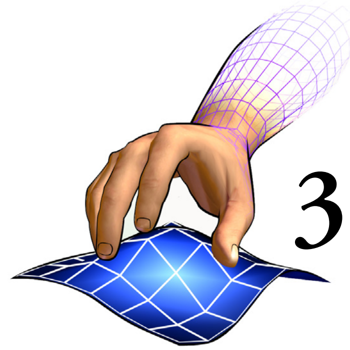
2.7.3. Reduction

Reduction minimizes memory usage of the database by collapsing vertices within a tolerance, distance of one another, into a single instance of the vertex, and all identical surfaces into a single instance of the surface. All triangles and polygons that reference the vertices and surfaces are adjusted appropriately. This reduces memory use, sometimes dramatically for large terrain files, with no degradation in performance other than the cost of the reduction.

You can specify the value of the tolerance. It defaults to a conservative value of 0.0, to ensure that incorrect adjustments are not made by default. For most databases, a value of 0.0001 or 0.001 will remove most duplicated vertices with no unintended changes to the terrain. However, since it is possible to incorrectly weld vertices that were not intended to be shared, be cautious about overzealous reduction.

Because surfaces are either identical or not, no tolerance is necessary for their reduction.

> For large numbers of surfaces and vertices, the reduction may be processor intensive.



3

Editing GDB Terrain Databases

This chapter explains how to add features, soil types, and polygon colors to GDB terrain databases, and how to associate images with terrain databases.

3.1. Loading a Terrain Database

You can load a terrain database for editing by opening its MTD file or loading the database directly.

---

i The TDB Tool only edits GDB terrain databases. If you have a terrain in another format, you must create a project and build a GDB version of the database before you can edit it.

3.1.1. Loading a MÄK Terrain Descriptor (MTD)

To load an MTD file:

1. Choose **File** → **Open** → **Terrain Descriptor**. An Open Terrain Descriptor dialog box opens.
2. Select the MTD file to open.
3. Click Open.

3.1.2. Loading a Terrain Database

When you load a terrain database, if there is not an MTD file with the same name as the database, the TDB Tool automatically creates one; otherwiseTDB Tool opens the existing MTD file.

To load a terrain database:

1. Choose **File** → **Open** → **Terrain Database**. An Open Database dialog box opens.
2. Select the GDB terrain to open.
3. Click Open.

3.1.3. TDB Tool Toolbars

The TDB Tool has several toolbars for quickly accessing frequently used tasks. In each of the following lists, the numbers reference the buttons in the corresponding figures.

Figure 3-1. Main toolbar

The Main toolbar (Figure 3-1) has the following buttons:

- New Terrain Project or Terrain Descriptor (1)
- Open Terrain Project, Terrain Descriptor, or Database (2)
- Save File to Disk (3)
- Edit Properties (4)
- Delete Selection (5)
- Select Objects (6)
- Select Features (7)
- Select Polygons (8)
- Polygon Selection Filter (9).

i Some buttons display menus when you click them.

Figure 3-2. Drawing Options toolbar

The Drawing Options toolbar (Figure 3-2) has the following buttons:

- Enable/Disable Polygon Drawing (1)
- Enable/Disable Feature Drawing (2)
- Enable/Disable Contour Line Drawing (3)
- Enable/Disable Wire Frame Drawing (4)
- Enable/Disable Terrain Shading (5)
- Enable/Disable Raster Maps (6)
- Enable/Disable Polygon Coloring Using Soil Type (7).

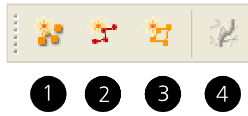


Figure 3-3. Features toolbar

The Features toolbar (Figure 3-3) has the following buttons:

- Add Point Feature (1)
- Add Linear Feature (2)
- Add Areal Feature (3)
- Selects polygons under selected features (4).

3.2. Adding Vector Features to a Terrain Database

You can add point, linear, and areal features to a terrain database. The TDB Tool includes a default set of configured features of each type.

3.2.1. Adding Point Features to a Terrain

To add a point feature to a terrain database:

1. Load a terrain database or MTD file.
2. Choose **Feature** → **Add Point Feature**. The Add Point Feature dialog box opens (Figure 3-4) and the cursor changes to draw mode.

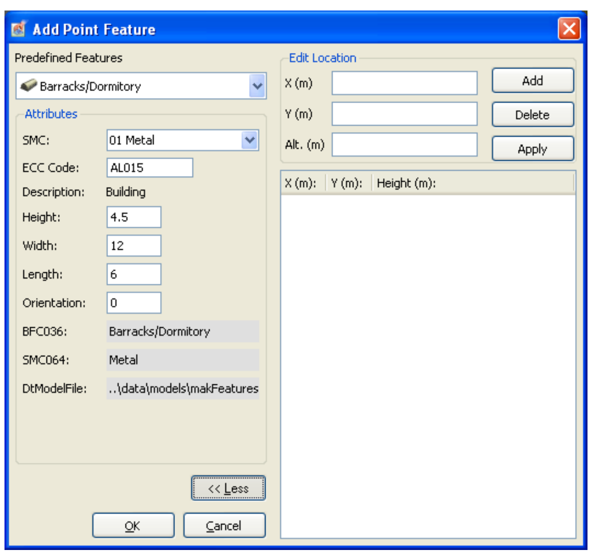


Figure 3-4. Add Point Feature dialog box

3. In the Predefined Features list, select the type of feature you want to create.
4. Click on the terrain where you want to add the point. You can add multiple points and you can change the feature type being created. You can also specify the location of a point in the Edit Location group box.

> **i** The orientation of a point increases in a counter-clockwise direction. A value 0 (zero) for orientation is aligned along the database X-axis. Orientation is specified in degrees.

5. Click OK.
6. Save the terrain database.

3.2.2. Adding Linear Features to a Terrain

To add a linear feature to a terrain database:

1. Load a terrain database or MTD file.
2. Choose **Feature** → **Add Linear Feature**. The Add Linear Feature dialog box opens and the cursor changes to draw mode.
3. In the Predefined Features list, select the type of line you want to create.
4. Click on the terrain where you want to specify the vertices of the line. Right-click the final vertex. You can also specify the location of the vertices in the Edit Location group box.
5. Click OK.
6. Save the terrain database.

3.2.3. Adding Areal Features to a Terrain

To add an areal feature to a terrain database:

1. Load a terrain database or MTD file.
2. Choose **Feature** → **Add Areal Feature**. The Add Areal Feature dialog box opens and the cursor changes to draw mode.
3. In the Predefined Features list, select the type of area you want to create.
4. Click on the terrain for each of the vertices you want to specify for the area. Right-click the last vertex. Do not try to close the area with the last vertex, the TDB Tool will draw a line between the starting vertex and the last one you specify. You can also specify the location of the vertices in the Edit Location group box.
5. Click OK.
6. Save the terrain database.

3.3. Editing Vector Features

You can edit the properties of new and imported vector features.

3.3.1. Selecting Terrain Features

Some terrain databases contain feature data, such as roads, buildings, trees, and so on. You can select features and display information about them. You can also create routes and points based on feature data.

> **i** Among the databases provided with TDB Tool, the *makland.mtd*, *VR-Village.mtd*, and *berkeley.mtd* databases contain feature data.

Selecting Features

To select a feature:

1. Choose **Select** → **Features**, or select the Select Features button on the Main toolbar.
2. Click the feature you want to select. It is highlighted (Figure 3-5).

Figure 3-5. A selected feature

Selecting All Features of A Particular Type

You can select all terrain features of a selected type in the Terrain Features dialog box.

To select all features of a given type:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens.
2. On the Drawing Mode tab, click Features. The Terrain Features dialog box opens (Figure 3-6). It lists each of the features in the terrain database.

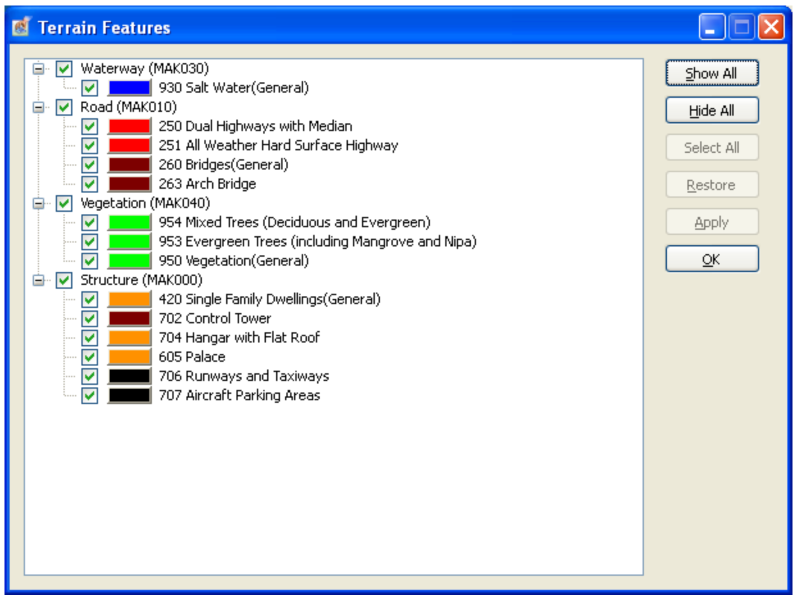


Figure 3-6. Terrain Features dialog box

3. Select the check boxes for the feature types that you want to select on the map. The Select All button is enabled.
4. Click Select All. All features of the selected type are highlighted on the map.
5. Click Apply or OK.

3.3.2. Editing the Properties of a Feature

You can edit the properties of features.

You cannot edit multiple features at one time.

To edit the properties of a feature:

1. Select the feature you want to edit and choose **Edit** → **Properties**, or double-click the feature. The Properties dialog box opens.

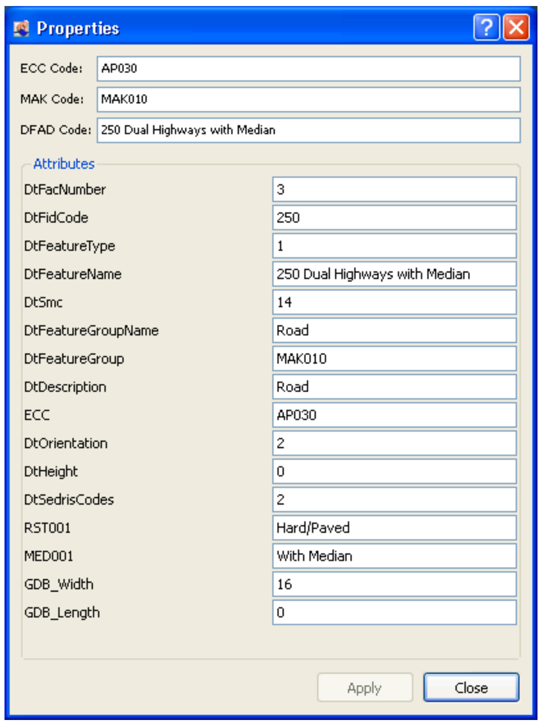


Figure 3-7. Properties dialog box

2. Edit the properties as desired.
3. Click Apply to apply the changes. The dialog box remains open.
4. Click Close to close the dialog box. If you did not apply the changes before closing the dialog box, your changes will not be saved.

3.4. Editing Polygon Soil Type and Color

You can edit the polygon soil type and color for GDB terrain databases. You can specify that a soil polygon be one of the types listed in Table 3-1.

Table 3-1: Polygon soil types supported

Ground	Water	Road	Other
Boulder	Ocean	Asphalt	Building
Dry Ground	Deep Lake	Dirt	Cultivated fields
Grass	Shallow Lake	Gravel	Forest
Mud	Deep River	Paved	Tree
Orchards	Shallow River	US Railroad	
Rock	Shallow Pond		
Sand	Shallow Stream		
Soft Soil			
Swamp			

You can assign specific colors to polygons or you can color them by soil type or elevation.

In VR-Forces, the soil types affect how the entities move on them. The soil types are configured as a list of soil roughness types in the object parameter database entry for each entity. Table 3-2 lists the soil types that you can apply to polygons in terrain databases and how they map to the soil roughness types configured for VR-Forces entities.

Table 3-2: Soil type/roughness type mappings

Soil type	Soil roughness type
ocean	deep-water
deepLake	deep-water
shallowLake	shallow-water
deepRiver	deep-water
shallowRiver	shallow-water
shallowPond	shallow-water
shallowStream	shallow-water
dryGround	hard-packed
pavedRoad	paved-road
gravelRoad	gravel
dirtRoad	gravel
asphalt	hard-packed

Table 3-2: Soil type/roughness type mappings

Soil type	Soil roughness type
USRailroad	hard-packed
softSoil	sand
swamp	muck
mud	muck
forest	hard-packed
grass	hard-packed
cultivatedFields	hard-packed
orchards	hard-packed
rock	rocks
boulder	rocks
sand	sand
building	hard-packed
tree	hard-packed

3.4.1. Setting the Terrain Soil Type

To set the soil type for a polygon:

1. Load a terrain database or MTD file.
2. Choose **Select** → **Polygons**. The cursor only selects polygons now.
3. Select the polygons whose soil type you want to change. For details, please see “Selecting Polygons,” on page 3-12.
4. Choose **Polygon** → **Set to** *category* → *soil type*, where *category* is Soil, Water, Road, or Other, and *soil type* is one of the options on the category sub-menu, such as Swamp (Table 3-1).

   The soil type is changed. The color is not automatically changed. To confirm the change, click on a polygon and look at the soil type in the Terrain Information toolbar (Figure 3-8). You can change the color to any arbitrary color, based on soil type, or based on elevation. For details, please see “Setting the Color of Polygons,” on page 3-12.

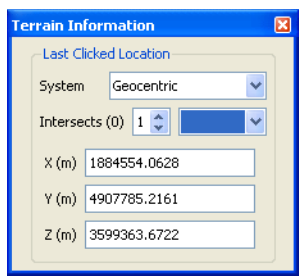


Figure 3-8. Information toolbar

5. Save the terrain database.

3.4.2. Setting the Color of Polygons

You can set the color of polygons to any arbitrary color, based on soil type, or based on elevation.

To set the color of a polygon to any color:

1. Load a terrain database or MTD file.
2. Choose **Select** → **Polygons**.
3. Select the polygons whose color you want to set. (Please see "Selecting Polygons," on page 3-12.)
4. Choose **Terrain** → **Set Color to**. A color picker dialog box opens.
5. Select the color you want to assign the polygons.
6. Click OK.
7. Save the database.

The Set Color Based on Soil Type command uses a predefined palette of colors, which you cannot change.

➤ To set terrain color by soil type, select the polygons you want to color, then choose **Terrain** → **Set Color Based on Soil Type**.

➤ To set terrain color by elevation, select the polygons you want to color, then choose **Terrain** → **Set Color Based on Elevation**.

3.4.3. Selecting Polygons

To select polygons, choose **Select** → **Polygons**, then do any of the following:

- Click an individual polygon
- Draw a bounding outline around an area of contiguous polygons
- **`Shift`**+**`Ctrl`**-click or **`Shift`**+**`Ctrl`**-draw a bounding outline to add polygons to the selection.
- **`Alt`**-click or **`Alt`**-draw a bounding outline to remove polygons from the selection.
- **`Ctrl`**-click or **`Ctrl`**-draw a bounding outline to toggle the selection of multiple individual polygons, areas, or both

Filtering the List of Selectable Polygons

By default, when you draw a bounding outline to select polygons, all soil types within the outline get selected. You can filter out soil types that you do not want to select.

To filter the list of selectable polygons:

1. Choose **Select** → **Polygon Selection Filter**. The Polygon Selection Filter dialog box opens (Figure 3-9).

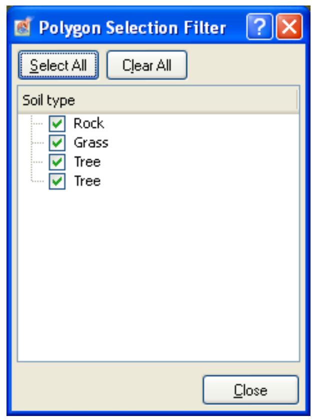


Figure 3-9. Select Polygons dialog box

2. In the Soil Type list, select the check boxes to include those soil types and clear the check boxes to exclude them.
3. Click Close.

Selecting the Polygons Under a Feature

You can select the polygons under a vector feature. For example, you might want to specify the soil type for the polygons under the feature, such as adding a vector feature for a river and setting the soil type to deepRiver.

Make sure the terrain intersector is set to Spatial Subdivision; not OPCODE. (Please see "Editing the Terrain Intersector Type," on page 3-14.)

To display the polygons under a feature:

1. Select the feature.
2. Choose **Feature** → **Select Underlying Polygons**. The polygons are selected.

- **Select Underlying Polygons** is not available unless a feature is selected.
- You cannot edit or hide the polygons that are displayed unless you change the selection mode to Polygons.

3.5. Editing the Terrain Intersector Type

You can change the type of terrain intersector used in your terrain database. You can select the new terrain intersector type or let the TDB Tool choose. If you let the TDB Tool choose, it tries to optimize the database for speed. It will usually use OPCODE for those databases that support it.

For details about terrain intersector types and optimizing terrain databases, please see "Optimizing Databases Using the TDB Tool," on page 2-20.

To change the terrain intersector type, in descriptor mode:

1. Choose **Edit** → **Choose Terrain Intersector**. The Choose Terrain Intersector dialog box opens (Figure 3-10).

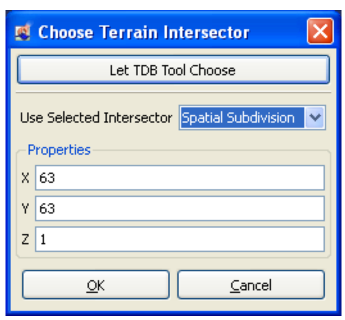
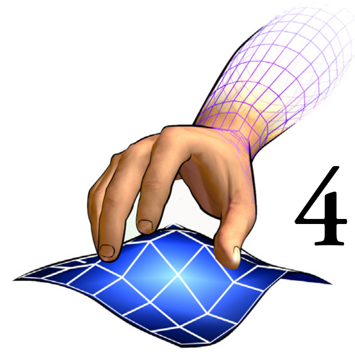


Figure 3-10. Choose Terrain Intersector dialog box

2. To let the TDB Tool choose the terrain intersector type, click Let TDB Tool Choose. The dialog box displays the intersector type that the TDB Tool has chosen (which might be the type that was already selected).
3. To choose the terrain intersector type, select an intersector type from the Use Selected Intersector drop-down list (Simple, Spatial Subdivision, or Opcode). For a description of these options, please see "Optimizing Databases Using the TDB Tool," on page 2-20.
4. If you or the TDB Tool chooses Spatial Subdivision, you can optionally specify the X, Y, and Z values. For details about specifying these values, please see "Creating Spatial Subdivisions," on page 2-22.
5. Click OK.



4

Editing Images

This chapter provides detailed instructions about how to edit images for your terrains.

4.1. Introduction

The TDB Tool has several options for manipulating images to properly align them with the terrain and to provide appropriate levels of resolution. The options are available in both project mode and edit mode. In project mode, you cannot manipulate an image until you view the project. In edit mode, you can edit images as soon as you load them.

4.2. Adding Images to a Terrain

You can associate images with a terrain by adding them to the terrain descriptor file. You can split images to create multiresolution papermaps and add them to the terrain. For details about multiresolution papermaps and splitting images, please see "Multiresolution Papermaps," on page 4-5.

To add an image to the terrain:

1. Click the Add Data icon on the Descriptor tab of the Utility bar. The Add Image Data dialog box opens.
2. Select the image to load.
3. Click Open. (If the image is too large to load into memory, you are prompted to split the image or scale it to fit. For details, please see "Adding Large Images to a Terrain Descriptor," on page 4-2.) The image is displayed on top of the terrain. Its filename is displayed at the Northwest corner. (If the image is the same size as the database, zoom out to see the file name.)

4.2.1. Adding Large Images to a Terrain Descriptor

The options for loading large images are explained in "Loading Large Images," on page 4-3. If you add a large image to a terrain descriptor, the TDB Tool detects that it is too large and provides an option to split the image, scale it to fit, or skip it. If you skip the image, a record is added. You can split the image at a later time, using the procedure described in "Splitting Images," on page 4-7.

4.3. Loading Large Images

Sometimes TDB Tool users try to load an image that is too large to fit in the available memory. To avoid this problem, the TDB Tool checks the size of an image when you view a project, build a project, or add an image to an MTD file.

! Loading large images is only supported for GeoTiff and JPG images.

When you are viewing a project or adding an image to an MTD file, if the TDB Tool detects that an image is too large for memory, it displays the Image Too Large dialog box (Figure 4-1).

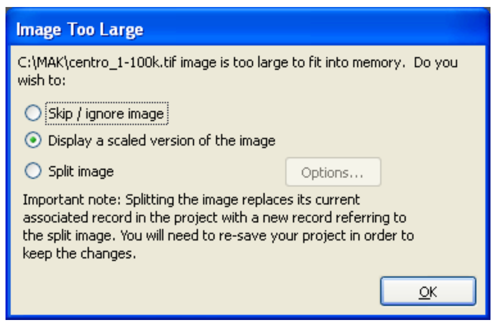


Figure 4-1. Image Too Large dialog box

The Image Too Large dialog box has the following options:

- Skip/ignore image. The TDB Tool adds a record for the image, but does not display it. It displays an outline where the image would be located.
- Display a scaled version of the image. The TDB Tool reduces the resolution enough to fit into memory and displays the image.
- Split image. The TDB Tool processes the image into a set of larger, lower resolution images, and smaller, higher resolution images, and configures them in a multiresolution papermap file (in MTL format). For an explanation of multiresolution papermap files and image splitting, please see "Multiresolution Papermaps," on page 4-5.

When you build a terrain project, if the TDB Tool detects that an image is too large, you are only given the option to split the image (Figure 4-2). If you choose No, the build process ignores it. However, if you load the MTD file at the conclusion of the build process, the TDB Tool displays the Image Too Large dialog box (Figure 4-1).

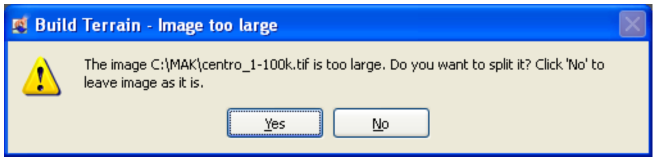


Figure 4-2. Build Terrain - Image Too Large dialog box

If you are prompted to choose an option for loading a large image:

➤ Select an option and click OK.

If you choose Split Image, you can specify options for the image splitting process or accept the default values. For details, please see step 8 in “Splitting Images,” on page 4-7.

! The TDB Tool detects that images are too large after the build or view process begins. Therefore, you cannot cancel the build or view process. You must choose an option.

4.4. Multiresolution Papermaps

The TDB Tool can split a large image into smaller images for use in multiresolution papermaps. Multiresolution papermaps let you:

- Divide image files into segments and load only the image file segments that need to be displayed on the screen at a particular time. This feature lets you use large, high-resolution images without using large amounts of memory.
- Specify different images at different screen zoom resolutions. For example, you can use lower resolution images for zoomed-out views, and higher resolution images for zoomed-in views.

The TDB Tool also generates the configuration file needed to display the set of images as if they were the original high-resolution image.

4.4.1. Multiresolution Papermap Levels

When you split an image, you must specify the number of levels of resolution. Level 1 does not split the image. Level 2 splits it into quarters, level 3 into sixteenths, and so on.

The TDB Tool provides images for level 1 through the chosen level. For example, if you choose three levels of resolution, the TDB Tool provides a level 1 image (which shows the entire original image), a set of level 2 images (each shows 1/4 of the original image), and a set of level 3 images (each shows 1/16 of the input image).

When the TDB Tool processes an image, all of the output images are the same size, which is determined by the resolution level chosen. Figure 4-3 shows the result of splitting an image into 1 level, 2 levels, and 3 levels.

- At one level of resolution, there is only one output image, which retains the dimensions of the input image.
- At two levels of resolution, the output is:
  - Four images, each showing one quarter of the input image
  - One image that is the result of resampling the input image into an image one quarter the size of the original.
- At three levels of resolution, the output is:
  - Sixteen images, each showing one sixteenth of the input image
  - Four images, each showing one quarter of the input image and are the same size as a one sixteenth image
  - One image that is the result of resampling the input image into an image one sixteenth the size of the original.

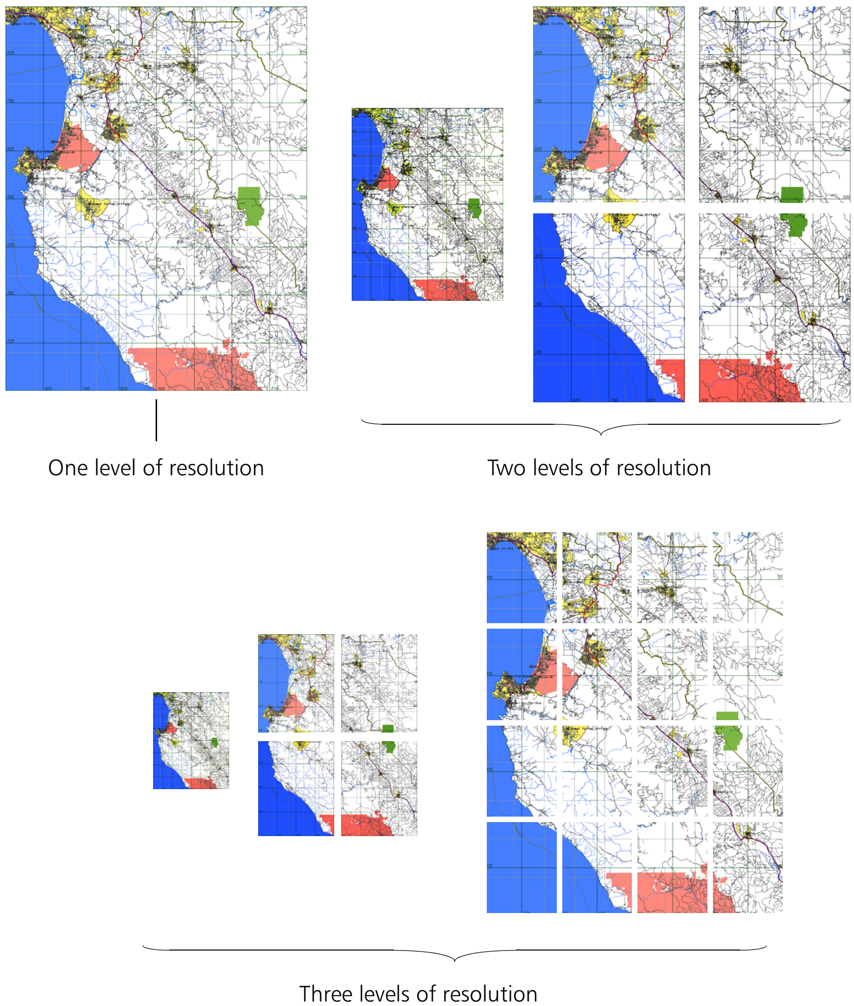


Figure 4-3. TDB Tool output at three different levels of resolution

4.5. Splitting Images

You may want to split images even when the TDB Tool can load them in their original size. This allows you to take advantage of the use of multiresolution papermaps to manage memory use and vary the resolution of the bitmap being viewed at any given time.

To split an image and create a corresponding configuration file:

1. Create or load a project or load an MTD file.
2. If the image you want to split is not part of the project or MTD, add the image, as described in "Adding Images to a Project," on page 2-10 or "Adding Images to a Terrain," on page 4-2.
3. If you are in project mode, view the project. If you are in edit mode, you can split the image any time.
4. Select the image you want to split by clicking on the image. The image displays on top of the terrain (Figure 4-4). The filename displays at the Southwest corner. (If the image is the same size as the database, zoom out to see the filename.)

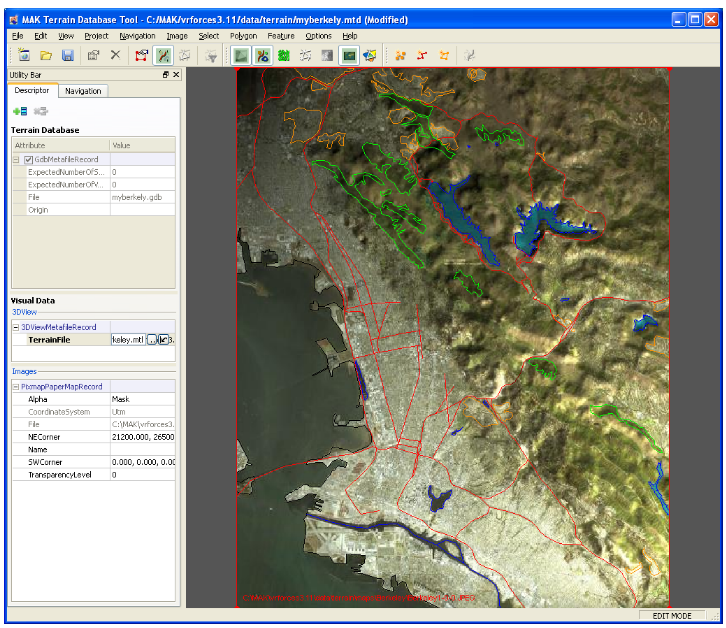


Figure 4-4. TDB Tool in edit mode with image displayed

You can load multiple images. They are split individually.

5. Position the image where you want it is to be located relative to the terrain. For more details, please see "Positioning Images," on page 4-9.

6. Choose **Image** → **Split Image**. The Split Image dialog box opens (Figure 4-5). The title bar displays the name of the image being split. If you have multiple images loaded, it indicates that this image is 1 of *n* images to be split.

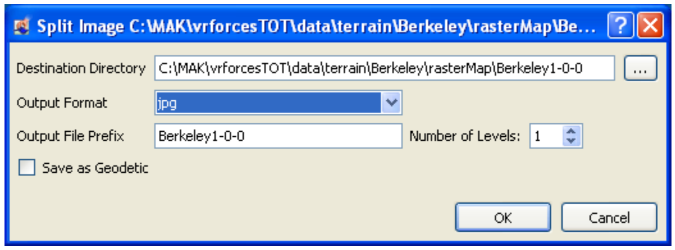


Figure 4-5. Split Image dialog box

7. Optionally, change the default options for splitting the image, as follows:
    a. Select the directory where you want to save the images into by clicking the ... button, or type a directory path in the Destination Directory text box. If you leave this text box empty, the images are saved in the *terrain* directory.
    b. Select the directory in which you want to save the images, or type a directory path in the Directory text box. If you leave this box empty, the images are saved to the *terrain* directory.
    c. If you want to save the files as geodetic, select the Save As Geodetic check box. For details, please see "Saving Images in Geodetic Format," on page 4-11.
    d. From the Output Format list, select a format.
    e. In the Output File Prefix box, type a name to use for the multiresolution papermap configuration file and the image files.
    f. In the Number of Levels box, specify the number of levels of resolution to split the image into. For more details, please see "Multiresolution Papermap Levels," on page 4-5.

If the source image is not evenly divisible into the number of parts determined by the level, the resulting extents in the output configuration file may not match the extents given to the TDB Tool due to round-off errors. The images may not display properly.

---

9. Click OK. The images and configuration file are created in the designated directory. The TDB Tool removes the image record from the project or MTD and replaces it with a multiresolution papermap record. If you have more than one image loaded, the Split Image dialog box opens again and indicates which image it is ready to split.

4.6. Positioning Images

After you load an image, you can resize it, move it around on the terrain, overlap images and align them with markers on the terrain, and set the Northwest (NE) and Southwest (SW) corners in any of the supported coordinate systems. This allows you to align the image with the proper portion of the terrain database.

i You must view a project before you edit an image.

4.6.1. Resizing Images

You can resize an image, as you might in a graphics program.

➤ To resize an image, drag one of its corners to the desired location.

4.6.2. Moving Images

➤ To move an image, drag it to the desired location.

4.7. Creating Markers

The TDB Tool lets you create markers to align an image precisely with the terrain.

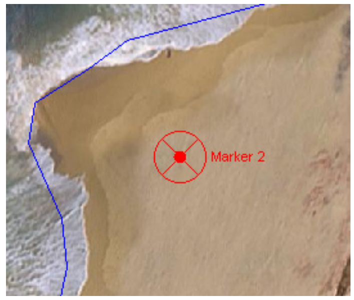


Figure 4-6. A marker

To create a marker:

1. Load a terrain database or MTD file.
2. Choose **Image** → **Create Marker**.
3. Click on the terrain where you want the marker to be located. To specify the exact location of the marker:
   a. Select the marker.
   b. Click the Navigation tab if it is not already displayed.
   c. In the Image Location toolbar (Figure 4-7), select a coordinate system from the list.
   d. Enter the appropriate location values for the coordinate system.
   e. Click Apply.

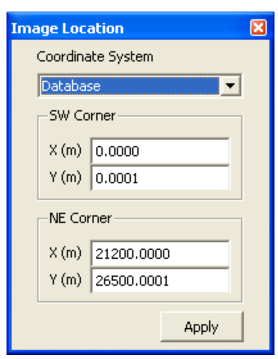


Figure 4-7. Image Location toolbar

4.7.1. Aligning an Image with a Marker

➤ To align an image with a marker, drag the image over the marker until a known location in the image is at the same location as the marker.

4.8. Specifying an Image's Extents

Rather than moving an image by dragging it, you can specify its NE and SW corners.

i You must view a project before you edit an image.

To specify an image's extents:

1. Select the image.
2. In the Image Location toolbar (Figure 4-7), select the coordinate system from the list.
3. Enter the appropriate location values for the SE and NE corners.
4. Click Apply.

4.9. Saving Images in Geodetic Format

By default, the TDB Tool saves images using the extents of the loaded UTM database. The resulting images are only valid with the database used to save them. If you want to create images that you can use with other UTM databases that cover the same geographic area, save them in geodetic format, as described in "Splitting Images," on page 4-7.

4.10. Converting Images to Different Coordinate Systems

If you have an image that was created using a particular coordinate system, such as UTM, and you want to display it with a database that uses a different coordinate system, such as flat earth, you can convert the image to the other coordinate system.

Figure 4-8 illustrates a geodetic terrain database for a portion of California, USA. The two images below the terrain show a raster image before and after conversion using the Map Image to Terrain option.

The image does not have any coordinate information. However, the configuration file that was created when you split the image, contains the coordinate information that allows a MÄK application to display the images correctly over the database.

Figure 4-8. Converting an image

- You must view a project before you edit an image.
- To use the Convert dialog box, you must understand terrain coordinate systems and your goal for converting an image.

To convert an image:

1. In the Images list on the Descriptor panel, select the image you want to convert.
2. Choose **Image** → **Convert Image**. The Convert dialog box opens (Figure 4-9).

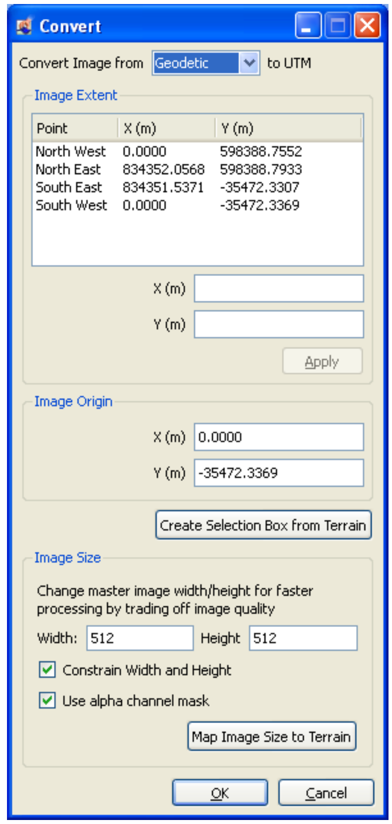


Figure 4-9. Convert dialog box

3. In the Convert Image From list, select the coordinate system for the image you are converting. (In the label, the *to* coordinate system is the coordinate system of the database that you loaded.)

4. The input options in the dialog box are optional. To ignore them, click OK. The TDB Tool does its best to convert the image based on the information available. The conversion options are as follows:

   – Specify the image extents. For each extent in the Image Extent list:

     a. Select the extent.

     b. Enter the X and Y coordinates to convert it to.

     c. Click Apply.

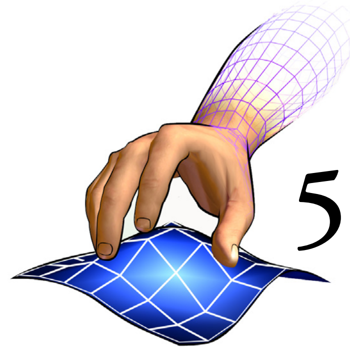
   – Specify the origin. In the Image Origin box, specify the X, Y values for the origin of the converted image.

   – Specify the Phi values for LCC images.

   – Crop the image to the extents of the terrain. If the terrain extents do not exactly match the image, click Create Selection Box from Terrain.

- Map the image size to the terrain. To shrink or enlarge the image to fit the terrain, click Map Image Size to Terrain.
- Reduce the image resolution. If you have a high resolution image, you can speed up the conversion process by reducing its resolution. To do so:
  a. In the Image Size box, enter the width and height, in pixels, to resize the image. To maintain the proportions of the image, select the Constrain Width and Height check box and change only one of the dimensions.
  b. To assign an alpha channel, instead of a black border, to the image border, select the Use alpha channel mask check box. The TDB Tool creates a PNG image instead of a JPG.

5. Click OK.



5

The TDB Tool Command Line Interface

This chapter explains how to use the TDB Tool command line interface to convert terrain databases to GDB format.

5.1. The TDB Tool Command Line Interface

The TDB Tool has the following command-line options:

```
tdbtool [options...] [-o output_file] input_file
tdbtool [options...] [-o output_file] -m input_file ...
tdbtool [options...] [-o output_file] -k levels
  [-a min_lod] input_file
tdbtool [-o output_file] -i input.gdb
  vectorDataFile.{dfad|dfd} vectorDataFile.{dfad|dfd}
tdbtool [options...] [-o output_file] metafile.mtp
```

where options are any of the following:

```
-b br,lf
-c colormap_file
-d data_definition_file
-e map_file
-f input_format
-g
-h | --help
-i GDBfile vectorfile1 ... vectorfilen
-k levels
-l lod_dist
-m {GDBfile1 ... GDBfilen | pattern*.gdb}
-moveorigin -lat lat -lon lon -zone zone
-n
-noconsole
-numvertices number
-numsurfaces number
-numspecs number
-o output_file
-p tolerance[, excessive_edge_count]
-r {tolerance|n}
-setorigin -lat lat -lon lon -zone zone
-switchchild node
-t
--terrainIntersector {basic | spatial | opcode}
-v
-w
-zone zone
```

Arguments

`-a` *`minLodDistance`* sets the minimum LOD distance (double). It is only useful when importing OpenFlight files.

`-b` *`br`*,*`lf`* balances the tree, with branching (*`br`*) and leafing (*`lf`*), for example:

```
-b 4,4
```

`-c` *`colormap_file`* specifies a color map file for SEDRIS feature codes. Use only with the `-i` option.

`-d` *`data_definition_file`* specifies a data definition file for importing DFAD or DFD data. Use only with the `-i` option.

`-e` *`map_file`* specifies a SEDRIS DFAD or DFD map file. Use only with the `-i` option.

`-f` *`input_format`* specifies the input file format where *`input_format`* can be `gdb`, `flt`, `c4b`, `c7b`, or `dtd` (the default is to guess from the filename extension).

`-g` specifies that the terrain database should be converted to the geocentric coordinate system.

`[-h | --help]` displays a description of TDB Tool command line options.

`-i` *`GDBfile vectorfile1 ... vectorfilen`* specifies the names of vector files to import. All files must be of the same type – SHP, DFD, or DFAD files.

i When importing SHP files, all features are imported as type Roads (fcc 240).

`-k` *`levels`* specifies the number of levels of externally referenced files to keep as externally referenced files. This argument allows you to set up a structure for paging terrain files. The number must be a non-negative integer. Any referenced file nested deeper than the specified limit is loaded into the file in which it is referenced. Use this option only when importing OpenFlight files.

`-l` *`lod_dist`* sets the level-of-detail distance to use for OpenFlight files. Default: 0.

`-m {`*`GDBfile1 ... GDBfilen | expression.gdb`*`}` specifies a list of GDB files for which you want to create a master GDB file. The TDB Tool creates a GDB file (with the name specified by the `-o` argument) that references all files specified as parameters to this argument. You can specify a list of GDB files by name, or you can specify an expression (for example, *makland*.gdb*) that the TDB Tool will use to identify the files to import. (The asterisk (*) is the only wild card supported.)

`-moveorigin -lat` *`lat`* `-lon` *`lon`* `-zone` *`zone`* moves the database to the new origin location, changing the coordinates within the database. (This is useful for moving a database from one zone to another.)

`-n` collapses all coordinate system nodes into the terrain database by applying the transforms on all children. Automatically done when transforming the database coordinate system.

`-noconsole` hides the output console and does not ask you to press any key once the command has finished. This is useful when you want to run multiple commands inside a batch file.

! If you collapse coordinate system nodes, and you are using spatial subdivisions, the spatial subdivision is automatically rebuilt.

`-numvertices` *`number`* specifies the expected number of vertices in an imported terrain (any format). This can result in significant memory and performance savings for very large terrain files.

`-numsurfaces` *`number`* specifies the expected number of surfaces in an imported terrain (any format). This can result in significant memory and performance savings for very large terrain files.

`-numspecs` *`number`* specifies the expected number of specifications in an imported terrain (any format). This can result in significant memory and performance savings for very large terrain files.

`-o` *`filename`* specifies the filename of the GDB file to be created.

`-p` *`tolerance`*`[,` *`excessive_edge_count`*`]` planarizes vector data, where:

- *`tolerance`* specifies the required tolerance attribute. The tolerance is used to determine if points are close enough to be considered the same. Default: 0.0001.
- *`excessive_edge_count`* stops processing when more than a certain number of edges are generated. Default: 50000.

! The TDB Tool always planarizes vector data when it loads it, so if the MTP file also specifies planarizing, it is done twice. This has no effect on the data, but may be a performance issue.

`-r` *`tolerance`* reduces the complexity of the terrain by combining vertices within a *`tolerance`* distance of one another and combining all identical surfaces. If you do not specify `-r`, vertex and surface reduction takes place using the default tolerance value (0.0). If you know that your database will not benefit from vertex reduction, specifying `-r n` (or `-r no`) turns it off, which can improve the speed of the conversion process.

`-setorigin` `-lat` *`lat`* `-lon` *`lon`* `-zone` *`zone`* moves the origin of the database. It does not reproject the database to the new origin (so, basically, it moves the local origin.) Latitude and longitude must be set in degrees:minutes:seconds.

`--switchchild` specifies which child node of a switch node to import for that node. Default: the first node. Only relevant when importing an OpenFlight file.

`-t` converts a database to flat earth format.

`--terrainIntersector {basic | spatial | opcode}` specifies which terrain intersector type to use for terrain intersection calculations.

- The basic intersector does not use special intersection optimization and does not allocate additional memory. It is useful when reducing memory is more important than intersection speed.
- The spatial intersector uses the spatial subdivision methodology to provide fast terrain intersections. It uses more memory to increase intersection speed.
- The opcode intersector uses the external OPCODE library to provide very fast terrain intersections. It takes a little while to initiate and uses more memory, but greatly increases runtime speed for intersection checks.

`-v` clips vector network data to the underlying terrain extent.

! The TDB Tool always clips vector network data when it loads it, so if the MTP file also specifies clipping, it is done twice. This has no effect on the data, but may be a performance issue.

`-w` specifies the level of notification (logging) that the TDB Tool uses. The level must be an integer between 0 and 4. A level of 0 reports fatal errors. A level of 4 reports every message. The default value is level 1. Setting the notification level to 4 may affect program performance due to the large amounts of logging that will be performed.

`-zone` *`zone`* specifies the UTM zone for the terrain, where *`zone`* is a value from 1 to 60.

5.1.1. Examples

The following example shows how to convert a FLT file to GDB.

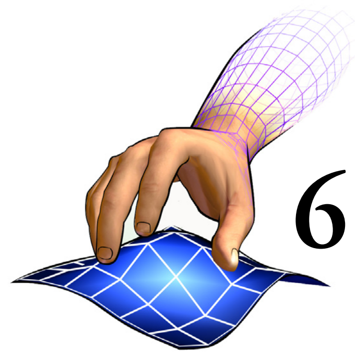
```
tdbtool -o mygdbfile.gdb mydatabase.flt
```

The following example shows how to override the default reduction attribute (0.0) with a attribute of 0.0001:

```
tdbtool –r 0.0001 –o myGdbFile.gdb mydatabase.flt
```

The following example shows how to import DFAD/DFD data into a GDB file. For more information, please see "Importing Vector Data," on page 2-14.

```
tdbtool -o myvectordatabase.gdb -i myexistingdb.gdb
  input1.dfd input 2.dfd -c colormap.txt -d features.txt
  -e mapfile.txt
```



6

Navigating and Configuring the Map

This chapter explains how to use basic GUI features.

6.1. The Utility Toolbar

The Utility toolbar has two tabs, Project, and Navigation. The Project tab is where you manage terrain projects. The Navigation tab contains toolbars that allow you to navigate and manipulate the terrain map.

The Utility toolbar and the View menu are not displayed unless a project, MTD file, or database is loaded.

6.1.1. Displaying and Hiding Toolbars

You can hide the display of any of the toolbars, including the main menu.

- To hide or view a toolbar, select it or clear it on one of the **View** → **Toolbars** submenus.
- To hide all toolbars except the Menu toolbar, choose **View** → **Hide All Toolbars**.
- To show all toolbars, choose **View** → **Show All Toolbars**. Toolbars are displayed in their previous docked or undocked state. If the menu bar is hidden (as the result of viewing full screen), click the middle mouse button anyplace in the window for a menu that allows you to display toolbars.

6.1.2. Full Screen View

You can display the map so that it covers the entire screen of your monitor. In full screen mode, no toolbars are displayed. You can zoom in and out using the mouse wheel and can pan the map using the keyboard.

- To view the map full screen, choose **View** → **Show Full Screen**.
- To exit full screen mode, press the **`Ctrl+Enter`** keys.

6.2. Navigating the Map

You can change your view (the extents of the database visible in the window) of the map by panning, zooming in and out, or selecting a specific area in the Minimap.

6.2.1. Panning the Map

You can pan the map in eight compass directions. In the Terrain View Options dialog box, you can set the increment by which the map is panned. For details, please see "Setting Panning Increments," on page 6-4.

To pan the map, use one of the following methods:

- Click on the appropriate pan button in the Navigation toolbar.
- On the Navigation menu, choose the Pan command for the direction in which you want to pan the map.
- On your keyboard's keypad (Num Lock must be on), press the number keys that correspond to the compass points of the Navigation toolbar (Figure 6-1).

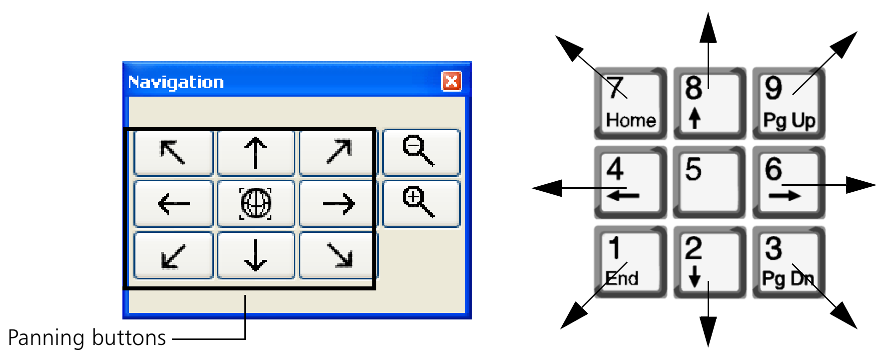


Figure 6-1. Panning with the Navigation toolbar and the keyboard

- Hold down the active mouse button and drag the mouse to the edge of the map in the direction that you want to pan. The map pans as long as you hold down the button.

Setting Panning Increments

Panning is specified as a percentage of the map. The default increment for panning the map is 20% (1/5 of the window).

To change the pan increment:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens.
2. Select the Navigation tab (Figure 6-2).

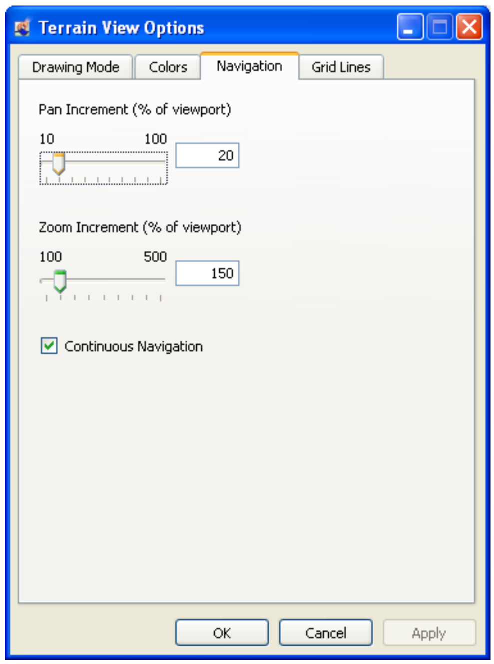


Figure 6-2. Terrain View Options dialog box, Navigation tab

3. Change the increment by moving the Pan Increment slider or entering a value in the text box.
4. Click OK.

6.2.2. Zooming the Map

You can zoom in and out on the map to vary the level of detail that you see.

To zoom the map, use one of the following methods:

- Click the Zoom In or Zoom Out buttons on the Navigation toolbar (Figure 6-3).

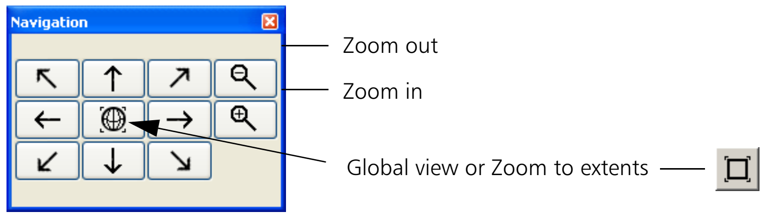


Figure 6-3. Zoom buttons

- To show the entire database, click the Global View button on the Navigation toolbar. If the Zoom to Extents button is displayed, click the terrain to show the Global View button.
- In the Navigation menu, choose the Zoom command for the direction you want to zoom.
- On your keyboard, press **+** (plus key) to zoom in; press **–** (minus key) to zoom out.
- Rotate the scroll wheel on the mouse forward to zoom in, backward to zoom out.

---

i When you use the mouse wheel to zoom in or out, it zooms one increment for each click of the wheel.

The map zooms in or out around the center point of the map.

In the Terrain View Options dialog box, you can set the increment by which the map is zoomed. For details, please see "Setting Zoom Increments," on page 6-6.

Zooming in on an Area of a Map

➤ To zoom in on a specific area, hold down the **`Shift`** key and drag a bounding outline around the area on the map to zoom in on.

Zooming to Extents

You can select an object or objects and zoom the map to the extents of those objects. If no objects are selected, Zoom to Extents acts as Reset to Global View.

To zoom the map to the extents of the selected objects, use one of the following methods:

- Choose **Navigation** → **Zoom to Extents**.
- Press **5**.
- Click the Zoom to Extents button in the Navigation toolbar (Figure 6-3).

Setting Zoom Increments

The zoom increment is specified as a percentage of the screen. The default increment for zooming the display is 200, which increases or decreases magnification by two times the image size.

To change the zoom increment:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens.
2. Select the Navigation tab (Figure 6-2).
3. Change the increment by moving the Zoom Increment slider.
4. Click OK.

6.2.3. Enabling and Disabling Continuous Panning and Zooming

By default, when you press a navigation control on the Navigation toolbar, the TDB Tool keeps panning or zooming while you press the control. You can enable and disable this feature.

---

i The keyboard controls for panning and zooming always support continuous navigation.

To enable or disable continuous navigation:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens.
2. Select the Navigation tab (Figure 6-2).
3. To enable continuous navigation, select the Continuous Navigation check box. To disable continuous navigation, clear the check box.

6.2.4. Using the Minimap to Change the View

The Minimap shows a thumbnail version of the database, including where entities are located. It shows you the portion of the database that is currently displayed in the window, and it allows you to select a portion of the database to view.

As you resize the TDB Tool window, the rectangle in the Minimap adjusts to match the aspect ratio of the map.

➤ To select a portion of the map to view, draw a rectangle in the minimap that defines the area you want to view (Figure 6-4).

Figure 6-4. Viewing area rectangle in the Minimap

➤ To change the region being viewed without changing the size of the view, in the Minimap, click the center point of the area you want to view.

---

i If you try to change the view with the Minimap and the view does not change, check to be sure you are not tracking an entity.

6.2.5. Changing the Color of the Minimap Rectangle

To change the color of the Minimap rectangle:

1. Choose **Options** → **Minimap Color**. A color picker dialog box is displayed.
2. Select the color you want to use for the rectangle.
3. Click OK.

6.2.6. Resizing the TDB Tool Window

In addition to zooming and panning the canvas, you can resize the TDB Tool window to change your view of the exercise. If a window is showing the entire database and you resize the window, it may not show the entire database, depending on how the proportions of the resized window compare to the proportions of the database.

6.3. Changing Display Settings

You can change settings that control which information gets displayed in a TDB Tool window and how the information is displayed. You change settings in the Display Options dialog box and the Terrain View Options dialog box. When you close the TDB Tool, the settings get saved to the *tdbtool.mtl* file.

6.3.1. Specifying the Coordinate System for the Terrain Information Toolbar

The Terrain Information toolbar (Figure 6-5) displays the location of the selected terrain point or entity in one of the following coordinate systems:

- Geocentric
- UTM
- Latitude and longitude
- MGRS
- Database.

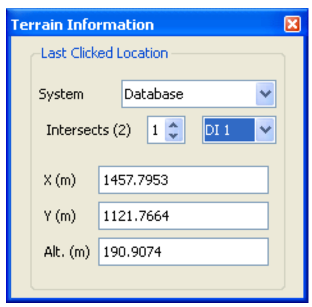


Figure 6-5. The Terrain Information toolbar

➤ To specify the coordinate system used in the Terrain Information toolbar, select an option from the System list.

6.3.2. Specifying the Drawing Mode

The drawing mode determines how a database is displayed. You can turn the following modes on and off:

- Polygons
- Features
- Wireframe
- Raster maps
- Terrain shading
- Contour lines.

To turn display of a drawing mode on or off:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens.
2. Select the Drawing Mode tab (Figure 6-6).

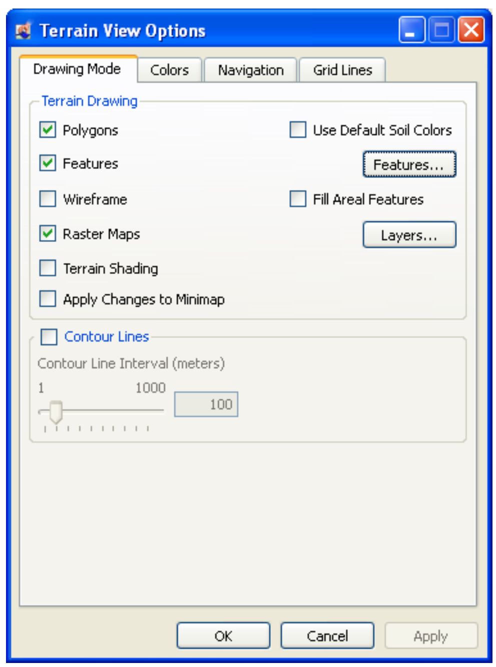


Figure 6-6. Terrain View Options dialog box, Drawing Mode tab

3. Select or clear the check box for the item you want to display or not display.
4. To apply the current changes to the Minimap, select Apply Changes to Minimap.
5. Click OK.

Drawing Modes

This section illustrates the effect of displaying or hiding different drawing modes.

Polygons

Polygons describe the terrain.

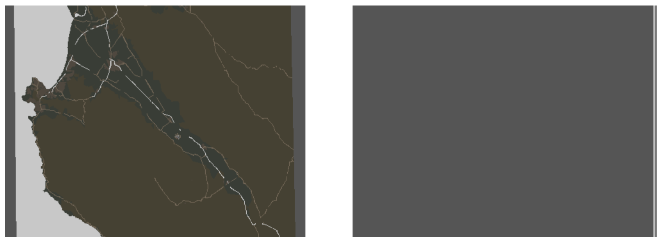


Figure 6-7. Polygons on and off

Features

The Features check box turns the feature nodes in the terrain database on or off, if the database has them.

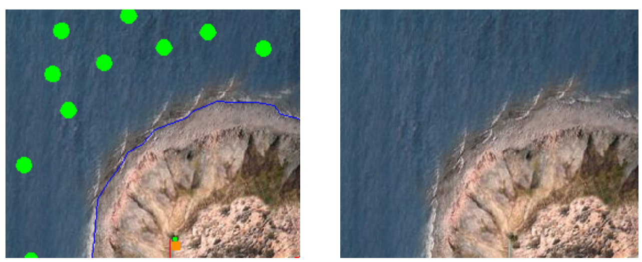


Figure 6-8. Features on and off

Wireframe

Wireframe mode shows the outline of the polygons.

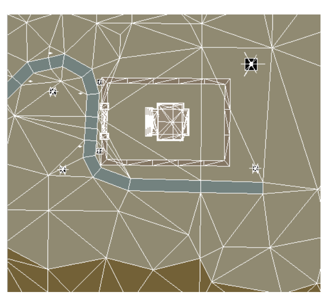


Figure 6-9. Wireframe on

Raster Maps

When raster maps are enabled, the bitmap picture of the map is displayed, if one is configured in the MTD file.

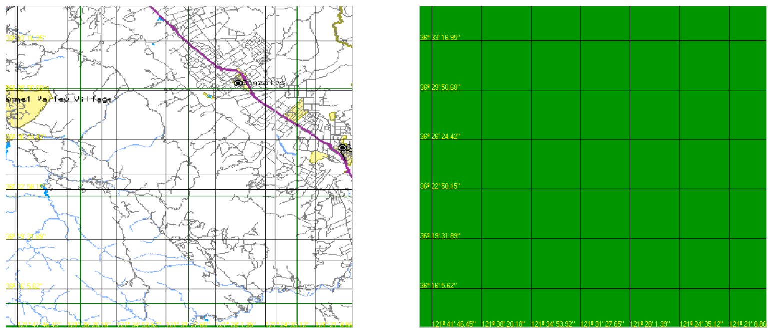


Figure 6-10. Papermap on and off

Terrain Shading

Terrain shading shows the contours of the terrain by varying the color according to height. Polygons must be displayed to display terrain shading.

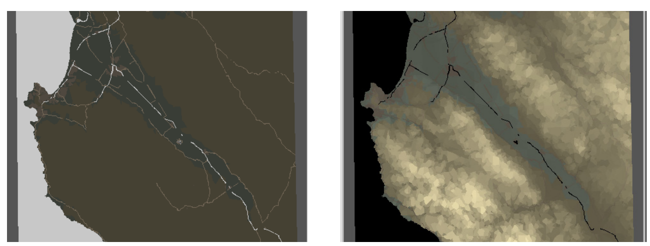


Figure 6-11. Terrain shading off and on

Contour Lines

Contour lines show changes in the height of the terrain. The interval between contour lines is determined by the Contour Line Interval slider.

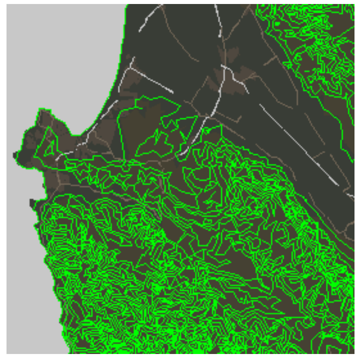
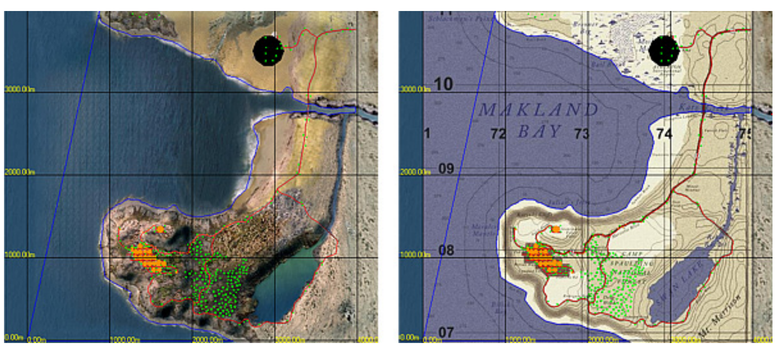


Figure 6-12. Contour lines

6.3.3. Specifying the Display of Raster Map Layers

If you are displaying raster maps and there is more than one layer, you can specify which layers to display. You can also specify the order in which layers are drawn. For example, the *Makland.mtd* terrain (provided with VR-Forces and the PVD) has four layers (Figure 6-14). Figure 6-13 illustrates the difference in the display between having the Makland Raster Image layer on top (the default), and having the Makland Map Image layer on top.



Makland Raster Image　　　　Makland Map Image

Figure 6-13. Effect of changing layer order

To display or hide raster map layers:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens (Figure 6-6).
2. On the Drawing Mode tab, click Layers. The Raster Map Layers dialog box opens (Figure 6-14). It lists each of the features in the MTD file.

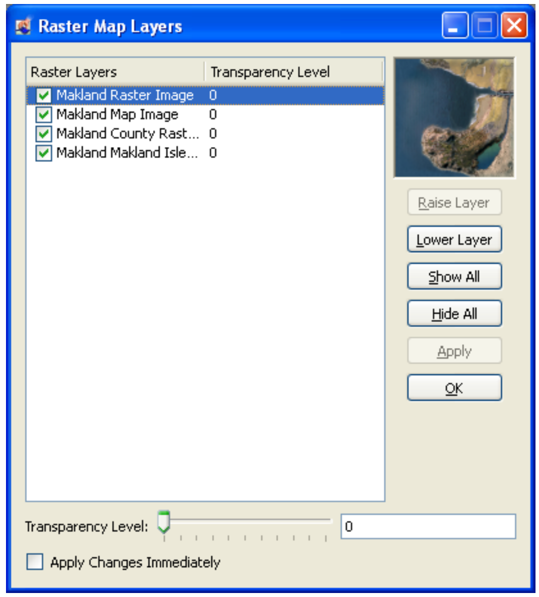


Figure 6-14. Raster Map Layers dialog box

3. To hide or display a particular layer, select or clear the check box next to the layer's name in the list.
   - To hide all layers, click the Hide All button.
   - To show all layers, click the Show All button.
4. To change the order in which layers are drawn, select a layer and click the Raise Layer or Lower Layer button.
5. To have changes take effect immediately, select the Apply Changes Immediately check box.
6. If you are not applying changes immediately, click Apply or OK.

6.3.4. Specifying the Transparency of Raster Map Layers

You can specify a degree of transparency for a raster map layer when you load raster map files, including multiresolution papermaps. This feature allows you to see the polygon data underlying the map. Figure 6-15 illustrates the *Berkeley.mtd* terrain at three levels of transparency. (Feature data is enabled in all three pictures.)

> **i** Use of transparency can degrade performance.

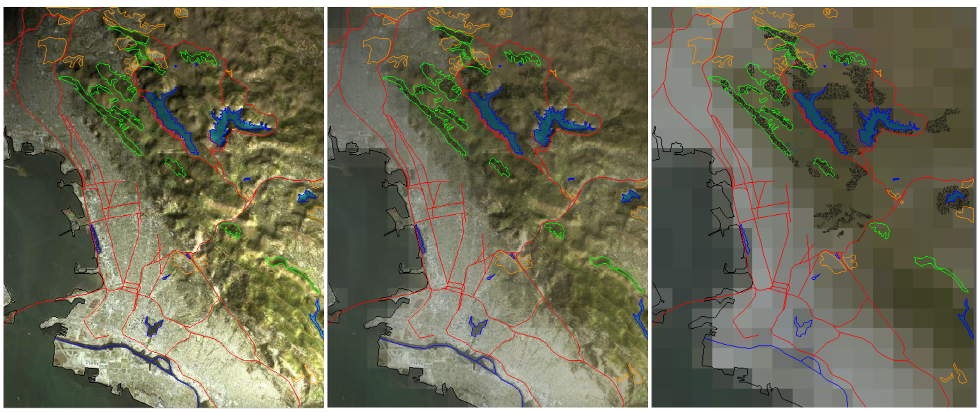


Figure 6-15. Setting transparency levels

To specify transparency of map layers:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens (Figure 6-6).
2. On the Drawing Mode tab, click Layers. The Raster Map Layers dialog box opens (Figure 6-14). It lists each of the layers in the raster map file.
3. Select the layer whose transparency you want to change.
4. Click the value in the Transparency Level column of the layer that you want to edit. The value becomes highlighted to indicate that it is editable.
5. Type a transparency value between 0 (opaque) and 100 (fully transparent).
6. Press **`Enter`**.

> **i** As an alternative to steps 3 through 5, you can adjust transparency using the Transparency Level slider (at the bottom of the dialog box).

7. Click Apply or OK.

6.3.5. Specifying the Display of Features

If you are displaying a database that has feature data, you can specify which features to display.

To display or hide feature data:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens (Figure 6-6).

2. On the Drawing Mode tab, click Features. The Terrain Features dialog box opens (Figure 6-16). It lists each of the features in the raster map file.

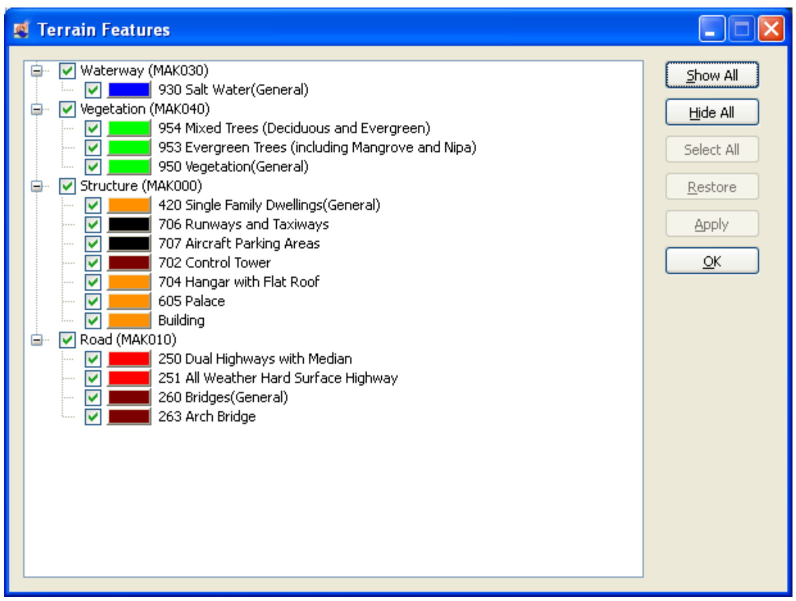


Figure 6-16. Terrain Features dialog box

3. To hide or display a particular feature, select or clear the check box next to the feature's name in the list.

   – To select all features, click the Show All button.

   – To unselect all features, click the Hide All button.

4. Click Apply or OK.

6.3.6. Using the Default Soil Colors

By default, terrain polygons are displayed using the color that was assigned to them when the database was imported to GDB. These hard-coded colors do not necessarily correspond to the soil type and polygons with the same soil type may have different hard-coded colors. You can specify that TDB Tool display polygons with the color associated with the soil type.

To display polygons with the color associated with their soil type:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens (Figure 6-6).
2. On the Drawing Mode tab, if the Polygons check box is cleared, select it.
3. Select the Use Default Soil Colors check box. (To display polygons with their hard-coded color, clear the Use Default Soil Colors check box.)
4. Click OK.

6.3.7. Specifying the Spacing of Contour Lines

To change the spacing of contour lines:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens.
2. Select the Drawing Mode tab (Figure 6-6).
3. Change the interval by moving the Contour Line Interval slider.
4. Click OK.

6.3.8. Specifying the Display of Grid Lines

You can toggle the display of grid lines on and off. You can also specify the spacing between grid lines.

To specify the display of grid lines:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens.
2. Select the Grid Lines tab (Figure 6-17).
3. Select or clear the Grid Lines check box.
4. Click OK.

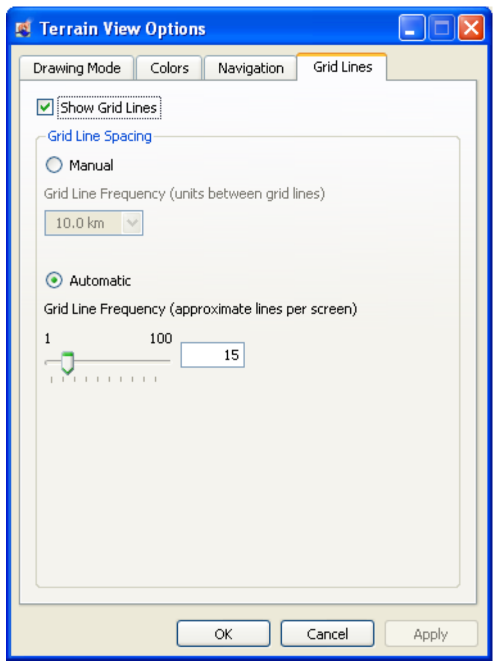


Figure 6-17. Terrain View Options dialog box, Grid Lines tab

Changing the Grid Line Spacing

The TDB Tool maintains the distance between grid lines at powers of 10 of the current measurement unit, for example, every 10 meters, every 100 meters, and so on. (For information about measurement units, please see "Changing Measurement Units," on page 6-20.) You can change the grid line spacing using manual mode or automatic mode. In either case, the number of grid lines in the display varies depending on the extent of the database that is displayed in the window.

In manual mode, you select the power of 10 to use for grid line spacing.

In automatic mode, you choose the number of grid lines you want to display along the longest dimension of the map. TDB Tool chooses the power of 10 that lets it display a number of grid lines that is as close to the amount specified as possible.

To change the spacing of grid lines:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens.
2. Select the Grid Lines tab (Figure 6-17).
3. If grid lines are not enabled, select the Grid Lines check box.
4. Specify automatic grid lines spacing as follows:
   a. Select the Automatic option button.
   b. Change the frequency by moving the Grid Line Frequency slider.

   Specify manual grid lines spacing as follows:
   a. Select the Manual option button.
   b. Select a frequency from the list.
5. Click OK.

6.3.9. Changing Terrain-Related Colors

You can change the colors assigned to:

- Contour lines
- Background
- Wireframe
- Grid lines
- Grid line labels.

To change a color setting:

1. Choose **Options** → **Terrain View**. The Terrain View Options dialog box opens.
2. Select the Colors tab (Figure 6-18).
3. Click the color button to the left of the option that you want to change. The Select Color dialog box opens. This is a color picker for your windowing system.
4. Select one of the existing colors or click Define Custom Colors to select any RGB value.
5. Click OK.
6. Click OK in the Terrain View Options dialog box.

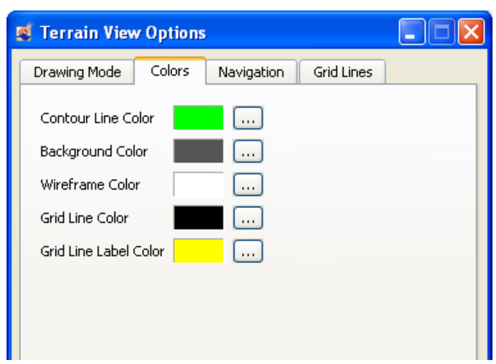


Figure 6-18. Terrain View Options dialog box, Colors tab

6.3.10. Changing Measurement Units

You can specify the units used to display coordinate and distance measurements. Display units are configured by measurement type. This means that you can use different measurement units for different types of data. You can also specify the number of decimal places for measurement values. When you change a measurement display unit, the changes take effect immediately. They are saved and applied the next time you run TDB Tool.

The types of data and the display objects that they affect are as follows:

- Distance: Applies to task and set dialog boxes, terrain information, intervisibility lines, entity information, symbol information, and grid lines.

---

i If you open a geodetic terrain, distance only applies to range information. Grid lines for geodetic databases take their precision from the Lat/Lon DMS coordinate system selection.

- Altitude: Applies to task and set dialog boxes, terrain information, intervisibility lines, entity information, symbol information, and grid lines.

To specify measurement units:

1. Choose **Options** → **Display**. The Display Options dialog box opens.
2. Select the General tab (Figure 6-19)
3. In the Display Units group box, select a measurement unit for each category from the drop-down list.
4. Type a precision for each measurement unit in the corresponding box in the Decimal Places column.

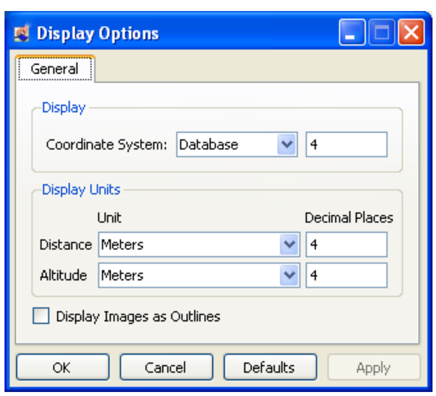


Figure 6-19. Display Options dialog box, General tab

5. Click OK.

6.3.11. Specifying the Coordinate System for Dialog Boxes

Some dialog boxes require you to enter the coordinates of a point on the map. You can specify which coordinate system these dialog boxes should use.

i If you select the UTM coordinate system, grid lines are displayed using UTM coordinates. Otherwise, grid lines are drawn using the coordinate system of the terrain database.

To specify the coordinate system used in dialog boxes:

1. Choose **Options** → **Display**. The Display Options dialog box opens (Figure 6-19).
2. On the General tab, in the Display group box, select an option from the Coordinate System list.
3. Type a value for the number of decimal places to use in coordinate values in the box next to the Coordinate System list.
4. Click Apply or OK.

7

Metafile Records

This chapter describes the metafile records used in MÄK Terrain Project (MTP) files.

7.1. Terrain Metafile Records

When you add data to a Terrain Project, a record is added to the project and a set of attributes is displayed. When you save the project, this information is read to a text file in which it is stored as a series of metafile records. Similar records are written to MTD files. This chapter explains the syntax of these records. Although you do not need to know the exact syntax, if you use the TDB Tool to create terrain projects, you may want to understand the meanings of the various attributes.

The TDB Tool supports metafile records for the following database formats:

- Digital Terrain Elevation Data (DTED)
- MultiGen-Paradigm OpenFlight (FLT)
- MultiGen-Paradigm MetaFlight (MFT)
- Compact Terrain Database (CTDB)
- MÄK Terrain Format (GDB)
- ESRI Shapefile (SHP)
- Vector product format (VPF)
- Extent.

The following is a simple MTD file. It instructs the TDB Tool to load the *world.gdb* database and to overlay it with the *world.bmp* raster file.

```
GdbMetafileRecord
{
  Import True
  ExpectedNumberOfVertices 0
  ExpectedNumberOfSurfaces 0
  File world.gdb
}
MultiResPaperMapRecord
{
  Import True
  File .\maps\World_Raster\World_Raster.mtl
  Name "World Raster Rendered Image"
  Cache_Images True
}
```

The following MTD file imports an OpenFlight database:

```
FltMetafileRecord
{
   Import True
   ExpectedNumberOfVertices 0
   ExpectedNumberOfSurfaces 0
   File HL_94.flt
   Reference_Ellipsoid_Semi_Major_Axis 6.37814e+006
   Reference_Ellipsoid_Semi_Minor_Axis 6.35675e+006
   Inverse_Flattening 0
   DatumShift 0 0 0
   Projection_Central_Scale 0.9996
}
```

- Although it is possible to create and edit MTP files using a text editor, we strongly recommend that you use the TDB Tool to create and edit them. The TDB Tool handles all formatting and you only need to enter the appropriate attribute values.
- In MTP files, boolean attributes can be lowercase or initial capitals (for example, true or True.) They should never be all capitals (for example, TRUE.)

7.1.1. DTED Terrain Metafile Records

The PVD and VR-Forces show a DTED database as projected onto FlatEarth, Geodetic, or UTM coordinates with white-on-green relief shading.

In a DTED Metafile Record, you specify the conversion attributes, and one or more DTED files. The syntax is as follows:

```
DtedMetafileRecord
{
  [Import {True | False}]
  PostSpacing rate
  Projection FlatEarth | Geodetic | UTM
  LowColor red green blue
  HighColor red green blue
  LowColorHeight height
  HighColorHeight height
  ReliefShaded flag
  Origin latitude longitude
  MinMaxLatitude min_latitude max_latitude
  MinMaxLongitude min_longitude max_longitude
  SeaRectangleAdded flag

  File filename_1.dt0 (or .dt1, .dt2, .dt3, .dt4, or .dt5)
  File filename_2
  .
  .
  .
  File filename_n
}
```

Table 7-1: DTED Metafile Record attributes

Attribute	Description
File	Specifies a DTED filename. DTED files typically have extensions *.dt0* (for level 0 DTED), *.dt1* (for level 1 DTED), and so forth. Multiple DTED files can be added by clicking the Open button.)
HighColor	Specifies the RGB values of the high color to be displayed. For relief shading, the high color is used for northeast facing polygons, and for height shading, the high color is used for polygons at the high color height and above. Default: 250, 250, 255, 0.

Table 7-1: DTED Metafile Record attributes

Attribute	Description
HighColorHeight	Specifies the height, in meters, to use with the HighColor value in height shading. An interpolated value between the low color and the high color is used for polygons between the low-color height and the high-color height. Default: 3000.
Import	Specifies whether or not to include the data when the terrain is built. This attribute is not displayed on the project tab. It is set according to the state of the check box next to a record's name when a project is saved. Default: True.
LowColor	Specifies the RGB values of the low color to be displayed. For relief shading, the low color is used for southwest facing polygons, and for height shading, the low color is used for polygons at the low color height and lower. Default: 0, 100, 0, 0.
LowColorHeight	Specifies the height, in meters, to use with the LowColor value in height shading. An interpolated value between the low color and the high color is used for polygons between the low-color height and the high-color height. Default: 0.0.
MinMaxLatitude	Allows you to clip the database from the north and south. No grid point with a latitude, in degrees, larger than the maximum or less than the minimum is loaded.
MinMaxLongitude	Allows you to clip the database from the east and west. No grid point with a longitude, in degrees, larger than the maximum or less than the minimum is loaded.
Origin	Specifies the latitude and longitude, in degrees, for the database origin. Default: 0.
PostSpacing	Specifies the subsampling rate, as an integer. The default value, 5, loads every fifth value along each dimension from the raw DTED data.
Projection	Specifies what coordinates (FlatEarth, Geodetic, UTM) the database should be projected onto. Default: UTM.
ReliefShaded	Specifies whether to use relief shading (True) or height shading (False). Default: True.
SeaRectangleAdded	Specifies whether (True) or not (False) to add a surface at sea level across the entire database. Default: False. This can be useful when DTED data is unavailable for an ocean area. Figure 7-1 illustrates the SeaRectangleAdded attribute.

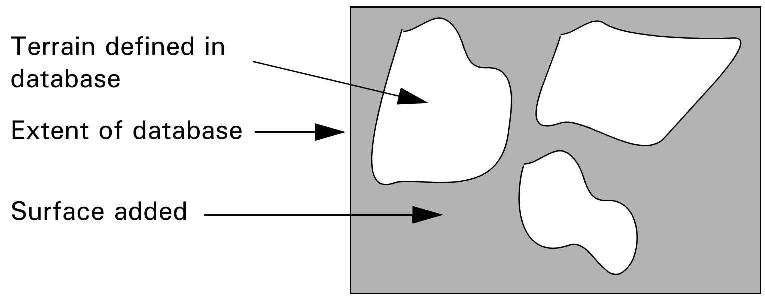


Figure 7-1. Use of the SeaRectangleAdded attribute

Creating Large Area Terrain Databases from DTED Data

You can create terrain databases from one or more DTED files for simulating over large geographic areas.

To create a large area database from DTED data:

1. Create a project that specifies the DTED file(s) to import. If the DTED data spans multiple UTM zones, set the Projected flag to False (keep the data in geodetic coordinates). Use the TDB Tool to generate a geodetic GDB database.
2. Load the geodetic database into the TDB Tool to generate a geocentric GDB version of the terrain.

7.1.2. OpenFlight Terrain Metafile Records

The OpenFlight Metafile Record specifies the name of a MultiGen-Paradigm OpenFlight database. The syntax is:

```
FltMetafileRecord
{
  File filename.flt
  [Import {True | False}]
  [Reference_Ellipsoid_Semi_Major_Axis axis]
  [Reference_Ellipsoid_Semi_Minor_Axis axis]
  [Inverse_Flattening value]
  [DatumShift x y z]
  [Projection_Central_Scale scale]
}
```

FLT files do not load textures.

Table 7-2: OpenFlight Metafile Record attributes

Attribute	Description
DatumShift	Specifies the offset from the current location that is sent out over the network, in geocentric coordinates.
File	Specifies the OpenFlight terrain database file to load.
Import	Specifies whether or not to include the data when the terrain is built. This attribute is not displayed on the project tab. It is set according to the state of the check box next to a record's name when a project is saved. Default: True.
Inverse_Flattening	Specifies the ratio of major to minor axis. If you specify Inverse_Flattening, you must also specify Reference_Ellipsoid_Semi_Major_Axis.
Projection_Central_Scale	Specifies a ratio applied to both the easting and northing values of UTM coordinates when converting from UTM to geodetic or from geodetic to UTM. Default: 0.9996.
Reference_Ellipsoid_Semi_Major_Axis	Specifies the semi-major axis of the reference ellipsoid to use.
Reference_Ellipsoid_Semi_Minor_Axis	Specifies the semi-minor axis of the reference ellipsoid to use.

If you specify both **Inverse_Flattening** and **Reference_Ellipsoid_Semi_Minor_Axis**, the **Inverse_Flattening** value takes precedence and the minor axis is calculated as:

```
semiMinor = semiMajor - (semiMajor / Inverse_Flattening)
```

7.1.3. CTDB Terrain Metafile Records

The TDB Tool can import CTDB version 4b, 7b, and 7l databases. Import of version 7 databases has the following conditions:

- The TDB Tool can import gridded and triangulate irregular network (TIN) data.
- The TDB Tool provides limited support for multi-elevation surface (MES) data, representing it as feature data. It does not import the geometric model data associated with MES. For example, the TDB Tool represents a house imported from MES data as a point feature (it does not import the polygon data representing the structure itself).

The CTDB terrain database record specifies the name of a Compact Terrain Database version 4b, 7b, (big endian) or 7l (little endian) database. The syntax depends on the version.

CTDB Version 4

```
C4bMetafileRecord
{
   File filename.c4b
}
```

The **File** attribute specifies the file name of a CTDB version 4 database. The filename must end with *.c4b*.

CTDB Version 7

```
C7bMetafileRecord
{
   File {filename.c7b | filename.c7l}
}
```

The **File** attribute specifies the file name of a CTDB version 7 database. The filename must end with *.c7b* or *.c7l*.

7.1.4. GDB Terrain Metafile Records

The GDB Metafile Record specifies terrain databases that are in MÄK Terrain Format. The terrain database record specifies the name of the terrain database file. The syntax is:

```
GdbMetafileRecord
{
  File filename.gdb
  [Import {True | False}]
  [ExpectedNumberOfVertices number]
  [ExpectedNumberOfSurfaces number]
}
```

Table 7-3: GDB Metafile Record attributes

Attribute	Description
ExpectedNumberOfVertices	Specifies the number of vertices you expect to have in the database.
ExpectedNumberOfSurfaces	Specifies the number of surfaces you expect to have in the database.
File	Specifies the GDB terrain database file to load.
Import	Specifies whether or not to include the data when the terrain is built. This attribute is not displayed on the project tab. It is set according to the state of the check box next to a record's name when a project is saved. Default: True.

7.1.5. GDB Processing Metafile Record

The GDB Processing Metafile Record makes most TDB Tool command-line options available for metafile records. It initializes a *DtTerrainDatabaseToolConfiguration*, which is passed to the TDB Tool to process the terrain database after all imported data is in GDB format.

Table 7-4: GDB Processing Metafile Record attributes

Attribute	Description
ApplyCoordSysNodes	Specifies whether or not coordinate system nodes in the terrain database should be applied.
BalanceBranching	Specifies the branching factor to use when balancing the terrain. Default: 4.
BalanceLeafing	Specifies the leafing factor to use when balancing the terrain. Default: 4.
BalanceTerrain	Specifies whether or not to balance the terrain node tree. Default: false. Should almost never be true.
ConvertToGeocentric	Specifies whether or not to convert the terrain to geocentric coordinate system. Default: false.
File	Specifies the terrain database file to load.
Import	Specifies whether or not to include the data when the terrain is built. This attribute is not displayed on the project tab. It is set according to the state of the check box next to a record's name when a project is saved.
PlanarizeExcessiveEdgeCount	Specifies the planarization excessive edge count. Default: 50000.
ReduceTerrain	Specifies whether or not to eliminate duplicate vertices and surfaces and remove other redundant or unnecessary terrain information. Default: True.
ReductionTolerance	Specifies a tolerance when eliminating duplicate vertices. Default: 0.0001.

7.1.6. 3D View Metafile Record

The 3D View Metafile Record specifies the terrain that the VR-Forces 3D Front End will open for display. Use of this record in an MTD file, along with the GDB Metafile Record allows a scenario to specify one MTD file that transfers seamlessly between the 2D front-end and the 3D Front End. If an MTD file does not include a 3DViewMetafileRecord or has a blank 3DViewMetafileRecord, a scenario must explicitly state (in the scenario file) which terrain database to load for the front-end, or the 3D Front End loads the GDB terrain specified in the MTD file.

When you create a new project, it automatically includes a blank 3DViewMetafileRecord. If the GDB file created by a project is created from an OpenFlight or MetaFlight database, when the TDB Tool creates the MTD file, it automatically specifies the source terrain in the TerrainFile attribute. If you create a GDB from some other terrain type or if you do not want to use the FLT file in the 3D Front End, and if you want to specify a terrain for use in the 3D Front End, then you must edit the 3DViewMetafileRecord to specify the terrain. You can edit the record in the TDB Tool project view or the edit view. The syntax is:

```
3DViewMetafileRecord
{
  TerrainFile filename
}
```

Table 7-5: 3DViewMetafileRecord attributes

Attribute	Description
TerrainFile	Specifies the terrain database file to load.

7.1.7. DFAD Metafile Records

You can import DFAD data into the TDB Tool using DFAD Metafile Records. The syntax for the record is:

```
DfadMetafileRecord
{
  [Import {True | False}]
  [ImportDescriptorFile import_desc_filename]
  [FidSmcFile fid_smc_filename]
  [DfadSedrisMapFile dfad_sedris_map_filename]
  [ColorMapFile color_map_filename]
  [Clip {True | False}]
  [Planarize {True | False}]
  [Tolerance tolerance]
  [ExcessiveEdgeCount edgeCount]

  File filename
  [File filename]
  ...
}
```

Table 7-6: DFAD Metafile Record attributes

Attribute	Description
Clip	Specifies whether (True) or not (False) to clip the incoming vector data to the extent of the terrain. Default: True.
ColorMapFile	Specifies the name of a file that defines the color representation for each subcategory in the SEDRIS code, in the format RGBA. By default it loads the file *../data/importDfad/sedris_colormap.txt*.
DfadSedrisMapFile	Maps each type of DFAD/DFD feature to a default SEDRIS feature. You can edit this file or specify a different map file with the `tdbtool -e` option. By default it loads the file *../data/importDfad/dfad_sedris_map.txt*.
ExcessiveEdgeCount	Specifies the maximum number of edges to process while planarizing. This attribute is useful as a performance cap for planarizing. Default: 50000.
FidSmcFile	Specifies the name of a file that defines the list of FID and SMC codes for DFAD and DFD basic features. By default it loads the file *../data/importDfad/FID_SMC.txt*.
File	Specifies the name of a DFAD file to load. You can specify any number of files.
Import	Specifies whether or not to include the data when the terrain is built. This attribute is not displayed on the project tab. It is set according to the state of the check box next to a record's name when a project is saved. Default: True.
ImportDescriptorFile	Specifies the name of a file that defines the list of DFAD/DFD features to import. Each possible feature Specifies IMPORT or NOIMPORT. By default it loads the file *../data/importDfad/importFile.txt*.

Table 7-6: DFAD Metafile Record attributes

Attribute	Description
Planarize	Specifies whether (True) or not (False) to planarize incoming vector data. Except in limited circumstances, this should always be True.
Tolerance	Specifies the allowable distance at which two separate points can still be considered to be the same point. The higher the tolerance, the less rigorous the planarizing. Default: 0.0.

DFD Records (Windows Only)

On Windows PCs, you can import DFD using a DFD DFAD Metafile Record. This record is almost identical to a DFAD Metafile Record, except that you can specify MultiGen-Paradigm DFD files to load in addition to DFAD files. The syntax for the record is:

```
DfdDfadMetafileRecord
{
  [Import {True | False}]
  [ImportDescriptorFile filename]
  [FidSmcFile filename]
  [DfadSedrisMapFile filename]
  [ColorMapFile filename]
  [Clip {True | False}]
  [Planarize {True | False}]
  [Tolerance tolerance]
  [ExcessiveEdgeCount edgeCount]
  [Reference_Ellipsoid_Semi_Major_Axis axis]
  [Reference_Ellipsoid_Semi_Minor_Axis axis]
  [Inverse_Flattening value]
  [DatumShift x y z]
  [Projection_Central_Scale scale]
  File filename
  [File filename
  ...
}
```

Table 7-7: DFD DFAD Metafile Record attributes

Attribute	Description
Clip	Specifies whether (True) or not (False) to clip the incoming vector data to the extent of the terrain. Default: True.
ColorMapFile	Specifies the name of a file that defines the color representation for each subcategory in the SEDRIS code, in the format RGBA. By default it loads the file *../data/importDfad/sedris_colormap.txt*.
DatumShift	Specifies the offset from the current location that is sent out over the network, in geocentric coordinates.

Table 7-7: DFD DFAD Metafile Record attributes

Attribute	Description
DfadSedrisMapFile	Maps each type of DFAD/DFD feature to a default SEDRIS feature. You can edit this file or specify a different map file with the `tdbtool -e` option. By default it loads the file *../data/importDfad/dfad_sedris_map.txt*.
ExcessiveEdgeCount	Specifies the maximum number of edges to process while planarizing. This attribute is useful as a performance cap for planarizing. Default: 50000.
FidSmcFile	Specifies the name of a file that defines the list of FID and SMC codes for DFAD and DFD basic features. By default it loads the file *../data/importDfad/FID_SMC.txt*.
File	Specifies the name of either a MultiGen-Paradigm DFD file or a DFAD file to import. You can specify any number of files.
Import	Specifies whether or not to include the data when the terrain is built. This attribute is not displayed on the project tab. It is set according to the state of the check box next to a record's name when a project is saved. Default: True.
ImportDescriptorFile	Specifies the name of a file that defines the list of DFAD/DFD features to import. Each possible feature Specifies IMPORT or NOIMPORT. By default it loads the file *../data/importDfad/importFile.txt*.
Inverse_Flattening	Specifies the ratio of major to minor axis. If you specify Inverse_Flattening, you must also specify Reference_Ellipsoid_Semi_Major_Axis.
Planarize	Specifies whether (True) or not (False) to planarize incoming vector data. Except in limited circumstances, this should always be True.
Projection_Central_Scale	Specifies a ratio applied to both the easting and northing values of UTM coordinates when converting from UTM to geodetic or from geodetic to UTM. Default: 0.9996.
Reference_Ellipsoid_Semi_Major_Axis	Specifies the semi-major axis of the reference ellipsoid to use.
Reference_Ellipsoid_Semi_Minor_Axis	Specifies the semi-minor axis of the reference ellipsoid to use.
Tolerance	Specifies the allowable distance at which two separate points can still be considered to be the same point. The higher the tolerance, the less rigorous the planarizing. Default: 0.0.

Example:

```
DfdDfadMetafileRecord
{
  File prague1.dfd
  File prague.dfad
  Clip True
  Planarize True }
```

7.1.8. VPF Metafile Records

VPF format is a database format similar to SQL used to store vector feature data. Each "table" in the database is stored in its own file or can be broken down into multiple tiles, which are sorted and stored in multiple directories. The entire system uses the file system directories and files to organize the database. A database has several themes or layers, each holding related information. For example, the VMAP standard defines a transportation layer that defines the roads and railways on the map. The TDB Tool supports VMAP 0, VMAP 1, and DNC.

Elevation data is supported to a limited extent. The contour lines stored in the VPF data are imported as features. They will be drawn, but they will not affect the polygonal data in the terrain.

You can import vector product format (VPF) vector data using a VPF Metafile Record. The syntax for the record is:

```
VpfMetafileRecord
{
  [Import {True | False}]
  [Clip {True | False}]
  [Planarize {True | False}]
  [Tolerance 0.0001]
  [ExcessiveEdgeCount 50000]
  [LibraryPath directory]
  [Library libraryName]
  [ThematicLayer layer]
  [FeatureClass featureClass]
  [SWCorner (-54, -36, 0)]
  [NECorner (-52, -30, 0)]
}
```

Table 7-8 describes the attributes for VPF Metafile Records.

Table 7-8: VPF Metafile Record attributes

Attribute	Description
Import	Specifies whether or not to include the data when the terrain is built. This attribute is not displayed on the project tab. It is set according to the state of the check box next to a record's name when a project is saved. Default: True.
Clip	Specifies whether (True) or not (False) to clip the incoming vector data to the extent of the terrain. Default: True
Planarize	Specifies whether (True) or not (False) to planarize incoming vector data. Except in limited circumstances, this should always be True.
Tolerance	Specifies the allowable distance at which two separate points can still be considered to be the same point. The higher the tolerance, the less rigorous the planarizing. Default: 0.0

Table 7-8: VPF Metafile Record attributes

Attribute	Description
ExcessiveEdgeCount	Specifies the maximum number of edges to process while planarizing. This attribute is useful as a performance cap for planarizing. Default: 50000.
Library	Name of the VPF Library to use. This matches the library directory name in the library path.
LibraryPath	Name of the top-level directory where VPF data is located (the directory that contains the library attribute table (lat) file for the dataset).
ThematicLayer	String identifying the thematic layer to import. VPF data is organized into thematic layers, which are groupings of related feature classes. This matches the string name of the thematic layer defined in the VPF database standards.
FeatureClass	String identifying the specific feature class of data to import. This matches the string name of the feature layer defined in the VPF standards.
SWCorner NECorner	Specifies the X, Y, and Z coordinates for the southwest corner and the northeast corner of the region to import map, in the geodetic coordinate system. The X and Y values define the corners in latitude and longitude (degrees).

Adding Support for New Types of VPF Databases

Support for VPF databases is configured in *./data/importvpf/vpfSpecificationMap.csv*. To add support for new types of VPF databases, such as Urban VMAP, add entries to *importVpf.csv*. The syntax of entries is documented in the file.

The following is the first few lines of the specification for VMAP0:

```
# VMAP level 0 features
point,bnd,polbndp,FA001,691,Political Boundaries,Other
  Feature,0,10,10,0,10,source=VMAP0
linear,bnd,barrierl,AL260,922,Walls,Feature,0,10,0,0,10,source=VMAP0
linear,bnd,coastl,BA010,1220,Shoreline,Waterway,0,10,0,0,10,source=VMAP0
linear,bnd,depthl,BE015,1216,Bay/Estuaries/Gulfs/Oceans/Seas,Waterway,0,
  10,0,0,6,source=VMAP0
linear,bnd,polbndl,FA000,691,Administrative Boundary,Other
  Feature,0,10,0,0,10,source=VMAP0
linear,bnd,polbndl,FA020,691,Armistice Line,Other
  Feature,0,10,0,0,10,source=VMAP0
linear,bnd,polbndl,FA030,691,Cease-Fire Line,Other
  Feature,0,10,0,0,10,source=VMAP0
```

! It is possible to specify one library and a lat/long for data that are mutually exclusive. This would result in no data loaded. There would be little feedback to help debug the problem.

7.2. Extent Metafile Records

An Extent Metafile Record specifies a region of the world to view in geodetic, UTM, or Lambert Conformal Conic (LCC) coordinates. It is expressed using two points, a southwest corner and a northeast corner, to construct a "dummy" database that consists of a single, flat polygon. Use an Extent Metafile Record to specify a coordinate system when there is no GDB file present. You can then overlay a bitmap on this coordinate system using the PixmapPaperMapRecord. There can only be one Extent Metafile Record per project and it is not valid when used with a GDB file record (unless the GDB record is not selected for import on the Project tab.)

The syntax for each type of extent record is as follows:

```
ExtentMetafileRecord
{
  CoordinateSystem Geodetic
  SWCorner (latitude, longitude, 0.0)
  NECorner (latitude, longitude, 0.0)
}

ExtentMetafileRecord
{
  CoordinateSystem Utm
  Origin (latitude, longitude, altitude)
  SWCorner (X, Y, Z)
  NECorner (X, Y, Z)
  Color 150 150 150
}

ExtentMetafileRecord
{
  CoordinateSystem Lambert
  Lam0 longitude
  Phi0 latitude
  Phi1 latitude
  Phi2 latitude
  SWCorner (X, Y, Z)
  NECorner (X, Y, Z)
}
```

Table 7-9: Extent Metafile Record attributes

Attribute	Description
Color	Specifies the color used to draw the extent. Colors are values from 0-255, listed as red green blue.
CoordinateSystem	Specifies the coordinate system of the extent you are defining. The SWCorner and NECorner must be expressed in this coordinate system.
Lam0	Specifies the central longitude (degrees) for the LCC projection. (LCC only)

Table 7-9: Extent Metafile Record attributes

Attribute	Description
Origin	Specifies the UTM origin for the coordinate system in latitude (degrees), longitude (degrees), and altitude (meters). (UTM only)
Phi0	Specifies the central latitude (degrees) for the LCC projection. (LCC only)
Phi1	Specifies the lower latitude (degrees) for the LCC projection. (LCC only)
Phi2	Specifies the upper latitude (degrees) for the LCC projection. (LCC only)
SWCorner NECorner	Specifies the X, Y, and Z coordinates for the southwest corner and the northeast corner of the paper map. In the geodetic coordinate system, the X and Y values define the corners in latitude and longitude (degrees). In the UTM and LCC coordinate systems, the X and Y values define the corners in meters. For all coordinate systems, the Z value is not used, but you must include it. Setting it to zero satisfies the requirement.

7.2.1. Z Values in Extent Records

In extent records, the altitude of vertices is determined by the altitude of the left, back, bottom vertex and the right, front, upper vertex. These are used as the SW and NE vertices respectively. The altitude of the SE vertex matches the altitude of the SW vertex and the altitude of the NW vertex equals the altitude of the NE vertex. Figure 7-2 illustrates the polygons that get created from the extents.

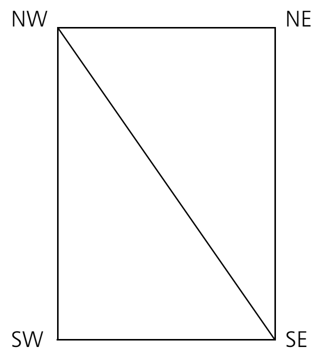


Figure 7-2. Extent polygons

The altitude can be non-zero, but it must be the same for all vertices or the terrain will be malformed.

You can specify a maximum latitude or longitude that is smaller than the minimum latitude or longitude. TDB Tool wraps around until it is larger. The extent initializes the maximum to be within +- 180 or 90 respectively (for example, min = 150, max = -150).

For example, this allows simulation of the area covered by +150 degrees to -150 degrees.

7.3. Papermap Records

A papermap is either a raster file or a collection of Raster Product Format (RPF) files. A papermap file has no three-dimensional information. It is the equivalent of a papermap that you might display on a wall or use to plan an automobile trip. Papermaps are useful because they may display details not available in a terrain database, such as identifying text and information about terrain details and man-made structures. Figure 7-3 compares the display of a GDB database to a papermap.

A MÄK Terrain Descriptor (MTD) file consists of one Terrain Database Record or Extent Record, and zero or more Papermap Records. An extent specifies a region of the world to view in geodetic, UTM, or LCC coordinates. An extent is defined by two points, a southwest (SW) corner and a northeast (NE) corner, to construct a "dummy" database that consists of a single, flat polygon. This extent can be used to provide underlying support for papermaps or to view entities in a particular section of the world in the absence of a database for that area.

---

i You cannot display a papermap by itself. It must be associated with a terrain database or an extent.

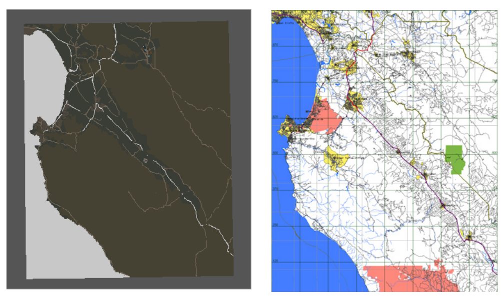


Figure 7-3. Database and papermap

7.3.1. Specifying the Coordinate System for Papermap Records

The coordinate system in which the papermap is defined must be aligned with the display coordinate system of an application GUI for the features on the map to line up correctly with their locations in the simulated world. (The display coordinate system of the GUI is dictated by the coordinate system indicated in the Terrain Database Record or Extent Record that precedes the papermap record in your MTD file).

For example, if your underlying terrain database or extent is defined in UTM coordinates, the X and Y axes of your papermap image must be aligned with the X and Y axes of the UTM coordinate system (lines of constant UTM easting and northing should be vertical and horizontal lines respectively, in your map.) On the other hand, if your underlying terrain database is defined in geodetic coordinates, then lines of latitude should run exactly horizontally, and lines of longitude should run exactly vertically in your papermap.

If you are using a papermap with the VR-Forces or PVD GUI, you must use a UTM or geodetic terrain database or extent, which means that your papermap image must be aligned with the proper axes for the coordinate system that you are using.

7.3.2. Pixmap PaperMap Records

The Pixmap PaperMap PaperMap Record specifies the raster file used to overlay the terrain database and its location. The syntax is:

```
PixmapPaperMapRecord
{
  File filename
  Name "name"
  CoordinateSystem {Geodetic | Utm | Lambert}
  SWCorner (X, Y, Z)
  NECorner (X, Y, Z)
  Alpha {Blend | Mask}
  TransparencyLevel level
}
```

Table 7-10: Pixmap paper map record attributes

Attribute	Description
Alpha	Specifies the alpha mode to be used when drawing a paper map. The modes are `Blend` and `Mask`. ◆ `Blend` causes the paper map to be drawn using full alpha blending. ◆ `Mask` causes the paper map to be drawn using a simple 1-bit alpha mask, that is, fully transparent, or fully opaque. Mask mode does not look as good as Blend mode, but it is much faster. The default mode is Mask. Multiresolution paper maps do not support blending. These options are only useful if the image has alpha data saved with it.
CoordinateSystem	Specifies the coordinate system, relative to the origin of the database, along which your paper map image is aligned. The SWCorner and NECorner extents must be expressed in this system. Furthermore, this coordinate system must match the coordinate system of the underlying terrain database or extent record. The TDB Tool supports the following coordinate systems: ◆ Geodetic ◆ UTM ◆ Lambert Conformal Conic (LCC, Lambert). UTM and LCC are only supported if the underlying terrain or extent is in the same coordinate system.
File	Specifies the name of the raster file to load. The file can be any of the supported raster formats.
Name	Specifies a name for the paper map. This name is used in the Raster Map Layers dialog box. If you do not specify a name, the filename is used in the dialog box.
SWCorner and NECorner	Specify the X, Y, and Z coordinates for the southwest corner and the northeast corner of the paper map. These coordinates must be expressed in the coordinate system designated by the CoordinateSystem attribute. In the geodetic coordinate system, the X and Y values define the corners in latitude and longitude (degrees). In the UTM and LCC coordinate systems, the X and Y values define the corners in meters. The Z value is not used, but you must include it. Setting it to 0.0 satisfies the requirement.
TransparencyLevel	Specifies the degree of transparency of the paper map layer. The range is 0 (opaque) to 100 (completely transparent.)

7.3.3. RPF Paper Map Records

For CADRG or CIB terrain databases, use RPF Paper Map Records, as follows:

```
RpfPaperMapRecord
{
  File "pathname"
  CoordinateSystem Geodetic
  Type type
  Resolution "resolution"
}
```

Table 7-11: RPF Paper Map record attributes

Attribute	Description
CoordinateSystem	Always Geodetic.
File	Specifies the pathname of the directory that contains the CIB or CADRG data.
Resolution	Specifies the resolution to use. The default values are 1:1M for CADRG and 5M for CIB.
Type	CIB or CADRG.

7.3.4. Multiresolution PaperMap Records

Instead of specifying an image file in the paper map entry, you specify a configuration file (MTL format) that contains image information. You can generate the configuration file with the TDB Tool (see "Splitting Images," on page 4-7) or create it by hand. The configuration file specifies sets of images to use at different screen resolutions, and the database coordinates of each image. Only the files that need to be drawn on the screen are read from the disk, and when an image is no longer needed, it is removed from memory.

A collection of images specified in a configuration file (and therefore one MultiResPaperMapRecord map file entry) are treated as a single papermap. For example, there is one entry in the Raster Map Layers dialog box. You can load more than one set of multiresolution papermaps, and you can load other types of papermaps at the same time.

The following code (*Makland.mtd*) is an example of multiresolution papermap records in an MTD file:

```
GdbMetafileRecord
{
  File Makland.gdb
}
MultiResPaperMapRecord
{
  File ".\maps\Makland_Isle_Raster\Makland_Isle_Raster.mtl"
  Name "Makland Makland Isle Raster Image"
  Cache_Images True
}
MultiResPaperMapRecord
{
  File ".\maps\Makland_County_Raster\Makland_County_Raster.mtl"
  Name "Makland County Raster Image"
  Cache_Images True
}
MultiResPaperMapRecord
{
  File ".\maps\Makland_Raster\Makland_Raster.mtl"
  Name "Makland Raster Image"
  Cache_Images True
}
MultiResPaperMapRecord
{
  File ".\maps\Makland_Map\Makland_Map.mtl"
  Name "Makland Map Image"
  Cache_Images True
}
```

When this file is read by the TDB Tool, the layers are listed in the Raster Map Layers dialog box, as shown in Figure 7-4.

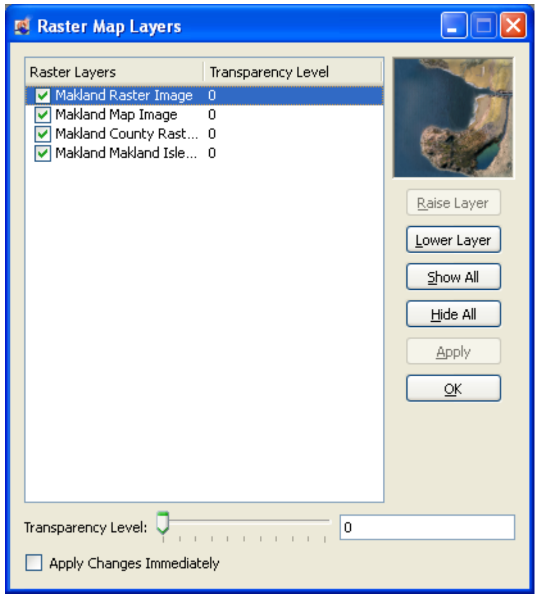


Figure 7-4. Raster Map Layers dialog box

Table 7-12: Multiresolution paper map record attributes

Attribute	Description
Cache_Images	Specifies whether or not to cache images. When set to True, images are kept in memory for faster drawing performance. When set to False (the default), images are discarded when not being used and must be read if the layer selection changes. Drawing performance can improve significantly for smaller maps. However, for larger images, caching may consume too much memory.
File	Specifies a multiresolution papermap configuration file.
Name	Specifies a name for the layer, which is displayed in the Raster Map Layers dialog box.

7.4. Shapefiles

A shapefile (*.shp*) stores non-topological geometry and attribute information for the spatial features in a data set, using a format created by Environmental Systems Research Institute, Inc. (ESRI). The format is based on shapes comprised of a set of vector and dBASE format files holding information about attributes. Each record in a shapefile has a one-to-one relationship with a record in a dBASE file. For complete technical information, please see the ESRI white paper, "ESRI Shapefile Technical Description," which is available at: http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf.

The data records in a shapefile describe geographic information using a type of geometric figure (point, line, polygon). The data in shapefiles can be structured as follows:

- Polygon-based describes the terrain as 2D polygons.
- ZPolygon-based describes the terrain as 3D polygons.
- postElevationData-based describes the terrain as a set of points. It contains information about elevation in an arbitrary (irregular) distribution of points.

If a shapefile has polygon or zpolygon data, use the file in that format. Polygon-based files load much faster than postElevationData files. However, you can use any shapefile as postElevationData.

You should specify a shapefile as postElevationData for any of the following reasons:

- It is an unknown format file. If you do not know the type of entries in the shapefile, you can describe it as postElevationData.
- It is a polyline or point based file. When the files have data based on lines or points, the data must be used as postElevationData.
- It is a polygon-based shapefile that does not contain information about how to connect polygons with different elevations. In polygon-based shapefiles, there is only one elevation for each polygon and there may not be any information about how to connect polygons with different elevations. If you know that your database is like this and you do not want discontinuities between polygons, load the database as postElevationData to obtain a fully connected GDB database. In Figure 7-5, each bar represents a polygon at a different elevation. There is no transitional data.

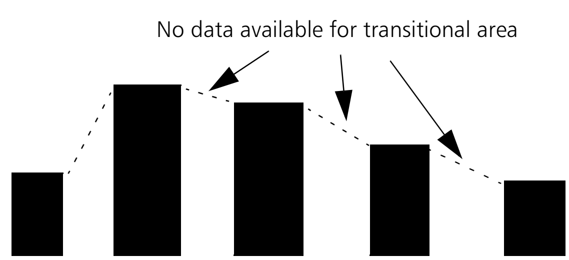


Figure 7-5. Polygon-based database with discontinuities

The way that TDB Tool converts a shapefile into GDB depends on the structure of the file.

- For polygon and zpolygon shapefiles, TDB Tool creates an identical internal set of polygons, representing the terrain. For polygons it extracts the elevation value for each polygon from the dBASE file or sets a value by default.
- For postElevationData there is no information about how to connect two different levels. The algorithm uses the data as posted data, creating an irregular grid and connecting adjacent vertices with triangles.

The border terrain values are calculated based on the closest point in the grid created with the data. This may show some inconsistencies in the corners, but in general will do a good job.

To use shapefiles, you define a Shape Metafile Record, a Shape Vector Metafile Record, or one of each.

Shapefiles may have an associated *.prj* file. For details about the *.prj* file, please see "Loading Shape .prj Files," on page 2-15.

7.4.1. Shape Metafile Records

A Shape Metafile Record entry has terrain information and one or more File entries.

> The zpolygon shapefile is supported only for terrain shapefiles. 3D features are not supported.

The format for a Shape Metafile Record entry is as follows:

```
ShapeMetafileRecord
{
  TerrainType {postElevationData | poly | zpoly}
  [SeaRectangleAdded | GroundRectangleAdded]
  [Origin lat lon]
  [MinimumDistanceBetweenPoints distance]
  [ThresholdForSeaLevel threshold]
  [ExcessiveEdgeCount count]

  File filename
  elevationFactorToMeters elevation
  {elevationInData | elevationField field_name |
    elevationValue elevation}
  ...

}
```

Table 7-13: Shape Metafile Record attributes

Attribute	Description
ExcessiveEdgeCount	Specifies a threshold number of edges, as a positive integer, beyond which planarizing stops. Optional if Planarize is True; otherwise, meaningless. Default: 50000.
File	Specifies the terrain database file to load.
GroundRectangleAdded	Adds a surface at the lowest ground elevation across the entire database. Use this attribute if the shapefile does not include information for the entire area covered by the extent of the database. This attribute is optional and is mutually exclusive with `SeaRectangleAdded`. For more information, please see Figure 7-1.
Import	Specifies whether or not to include the data when the terrain is built. This attribute is not displayed on the project tab. It is set according to the state of the check box next to a record's name when a project is saved. Default: True.
MinimumDistanceBetweenPoints	Specifies the minimum distance between points, in meters. This attribute is used with postElevationData. If two points are closer than this distance, the second one is omitted. Default:150 m.
Origin	Specifies the latitude and longitude of the origin of the database, in degrees.
SeaRectangleAdded	Adds a surface at sea level across the entire database. Use this attribute if the shapefile does not include information for the entire area covered by the extent of the database. This attribute is optional and is mutually exclusive with GroundRectangleAdded. Please see Figure 7-1.
TerrainType	Defines how to interpret the data stored in the shapefiles. The possible values are: ◆ `postElevationData` specifies that data in the shapefiles is used as points in an irregular grid. The type of shapefile is ignored. ◆ `poly` specifies that data is stored as polygons. Valid only with polygon or multipatch shapefiles. ◆ `zpoly` specifies that data is stored as polygons with elevation. Valid only with zpolygon or multipatch shapefiles.
ThresholdForSeaLevel	Specifies the elevation below which terrain is considered to be at sea level. This attribute is useful when you use postElevationData to set new points used to build the GDB and for setting soil types. Default: 0.

The File Entry in A Shape Metafile Record

A Shape Metafile Record must have at least one File entry. The first File entry is used to calculate the origin of the database, if it is not specified with the Origin attribute. If two or more File entries overlap, the TDB Tool uses the highest elevation.

The format of the File entry is as follows:

```
File filename
elevationFactorToMeters elevation
{elevationInData | elevationField field_name | elevationValue elevation}
...
```

Table 7-14: File attribute subattributes

Attribute	Description
elevationFactorToMeters	Specifies a factor to use to transform the elevation units into meters. Default: 1.
elevationInData elevationField elevationValue	Specifies how to calculate the elevation data, as follows: ◆ `elevationInData` specifies that the shapefile contains elevation information, for example, zpoly files. ◆ `elevationField` specifies the name of the field in the *.dbf* file that contains the elevation data. ◆ `elevationValue` specifies an arbitrary elevation.
File	Specifies the name of the shapefile. You must have a shapefile (*.shp*) and dBase file (*.dbf*) named *`filename`*.

7.4.2. Shape Vector Metafile Records

A Shape Vector Metafile Record allows vector data to be imported from a shapefile and specification attributes to be imported from an associated dBASE file. According to the ESRI standard, both files must have the same prefix. Multiple shapefiles can be imported using one Shape Vector Metafile Record, however the reader will attempt to import all the fields in the attributes for each file. So unless all the shapefiles have the fields listed, it is more efficient to put them into separate Shape Vector Metafile Records.

A Shape Vector Metafile Record has one or more **File** entries. You can specify one or more Shape Vector Metafile Records to provide feature data for any of the supported database types, for example, GDB Metafile Record, Dfad Metafile Record, or Shape Metafile Record.

! The origin in a Shape Vector Metafile Record must match the origin in the corresponding Shape Metafile Record.

The format for a Shape Vector Metafile Record entry is as follows:

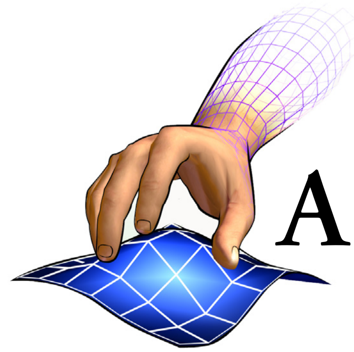
```
ShapeVectorMetafileRecord
{
  Projected { True | False }
  [ColorMapFile file]
  [SedrisFile file]

  [File filename
  type type]
  ...
  [Attribute attributeName (integer | real | string) fieldName]
  [Clip {True | False}]
  [Planarize {True | False}]
  [Tolerance tolerance]
  [ExcessiveEdgeCount count]
}
```

For an example of ShapeVectorMetafileRecords that import infrastructure, vegetation, roads, and lakes, please see *LittlePondInset.mtp* in *./data/terrain*.

Table 7-15: Shape Vector Metafile Record attributes

Attribute	Description
Attribute	Allows a dBASE field (referenced by *`fieldName`*) to be imported as a specification attribute (named by *`attributeName`*). The type of attribute added is specified as `integer`, `real`, or `string`. If the attribute type does not match the dBASE type, a conversion will be attempted, if possible. For example, a dBASE field of type 'F' (floating point) can be converted to an integer attribute by specifying integer. Not all dBASE field types are supported, but custom import routines can be added using the TDB Tool API. Contact MÄK technical support for more information.
Clip	Specifies whether or not to clip vector data to the extents of the terrain. Default: True.
ColorMapFile	Specifies the file that contains color information. Default: *sedris_colomap.txt*. For more information, please see "Importing Vector Data," on page 2-14.
ExcessiveEdgeCount	Specifies the maximum number of edges to process while planarizing. Default: 50000.
File	Specifies a file to be imported as feature data. The file must be a *.shp* file.
Import	Specifies whether or not to include the data when the terrain is built. This attribute is not displayed on the project tab. It is set according to the state of the check box next to a record's name when a project is saved.
Planarize	Specifies whether or not to planarize vector data. Unlike DFD and DFAD data, this attribute can be False. If in doubt, set it to True.
Projected	Specifies that the data is UTM (True) or lat, lon (False).
SedrisFile	Specifies a file with feature data specifications. Default: *dfad_sedris_map.txt*. For more information, please see "Importing Vector Data," on page 2-14.
Tolerance	Specifies the distance at which two separate points can be considered to be the same point. The higher the tolerance, the less rigorous the planarizing. Default: 0.0.
type	Specifies the type of information in the feature data file, according to the description in *data/importDfad/dfad_sedris_map.txt*.



The MÄK Specification Model

This appendix describes the specification model used for feature data.

A.1. The MÄK Specification Model

When feature data is imported into MÄK GDB format, a particular set of attribute label/value pairs are always assigned to feature specifications. These attributes include DFAD codes and SEDRIS enumerations. The rationale behind this approach is that there is a considerable amount of data already available in the DFAD format, and the SEDRIS enumerations are flexible and extensible.

The MÄK specification model also includes a classification called *DtFeatureGroup*. The DtFeatureGroup attribute organizes features into one of several broad categories. A corresponding DtFeatureGroupName attribute is also assigned to each feature. It contains a human-readable string representation of the *DtFeatureGroup* value. Features are categorized according the attributes listed in Table A-1.

Table A-1: Feature attributes

DtFeatureGroup	DtFeatureGroupName
MAK000	Feature
MAK010	Road
MAK020	Railroad
MAK030	Waterway
MAK040	Vegetation
MAK050	Power and Communications
MAK999	Other Feature

When you import vector data into a *DtTerrainDatabase* (from DFAD, DFD, SHP, or C7B sources) a vector network is created that uses the MÄK specification model. The TDB Tool takes advantage of this design. During the import process, it reads the following files and uses them to specify how incoming DFAD and SEDRIS codes should be mapped to various feature attributes:

- *dfad_sedris_map.txt* – maps each type of DFAD/DFD feature to a default SEDRIS feature.
- *importFile.txt* – defines the list of DFAD/DFD features to import into the GDB file. Each possible feature specifies IMPORT or NOIMPORT.
- *sedris_colormap.txt* – defines the color representation for each subcategory in the SEDRIS code, in the format RGBA.
- *FID_SMC.txt* – defines the list of FID and SMC codes for DFAD and DFD basic features.

A.2. MÄK Specification Attributes

Table A-2 lists attributes that MÄK uses in its specification model that are always present for each feature in the vector network.

Table A-2: Specification model attributes

Label	Type	Description
GDB_Width	real	Width of the feature. ◆ Point – If the width > 0, actual width ◆ Point – If the width = 0, a circular feature with radius length ◆ Linear – actual width ◆ Areal – 0 (not used).
GDB_Length	real	Length of the feature. ◆ Point – actual length ◆ Linear and Areal – 0 (not used).
DtOrientation	integer	Rotation about the database Z-axis. Starts with a value of 0 at the X-axis and increases in the counter-clockwise direction. ◆ Point: 0 – 360 degrees.
DtHeight	integer	Predominant elevation in meters.
DtFeatureType	integer	Defines the “shape” of the feature. ◆ 0 – Point ◆ 1 – Linear ◆ 2 – Areal.
DtFidCode	integer	DFAD code for that type of feature. Please see *./data/importDfad/FID-SMC.txt*.
DtFeatureName	string	DFAD definition for corresponding DtFidCode code. Please see *./data/importDfad/FID-SMC.txt.*
DtSmc	integer	Soil type based on SMC codes. Please see *./data/importDfad/FID-SMC.txt.*
DtFeatureGroup	string	Defines a grouping based on a MÄK extension of SEDRIS (see Table A-1): ◆ MAK000 ◆ MAK010 ◆ MAK020 ◆ MAK030 ◆ MAK040 ◆ MAK050.

Table A-2: Specification model attributes

Label	Type	Description
DtFeatureGroupName	string	String name of DtGroup (see Table A-1): • Feature • Road • Railroad • Waterway • Vegetation • Power and Communications.
ECC	string	Main SEDRIS identification code mapped from DFAD. Please see *./data/importDfad/dfad_sedris_map.tx.*
DtDescription	string	Main SEDRIS identification code description. Please see *./data/importDfad/dfad_sedris_map.txt.*
DtSedrisCodes	integer	Number of additional SEDRIS codes specifying the feature. Please see *./data/importDfad/dfad_sedris_map.txt.*
SEDRISCODE#	string	Where 1 <= # <= DtSedrisCodes. Please see *./data/importDfad/dfad_sedris_map.txt.*
i-SEDRIS code	string	i-SEDRIS code description. Please see *./data/importDfad/dfad_sedris_map.txt*.

Table A-3 lists commonly used attributes that are not necessarily present in every specification.

Table A-3: Commonly used attributes

Label	Type	Description
DtFacNumber	integer	Number of the feature, used to provide unique specification for a feature
The following optional attributes apply to area features only:		
DtStructsPerKm	Integer	Number of structures per kilometer squared.
DtPctRoofCoverage	Integer	Percentage of areal covered.
DtPctTreeCoverage	Integer	Percentage of area covered with trees.
DtXMin	Real	Minimum X coordinate of bounding outline, in meters.
DtYMin	Real	Minimum Y coordinate of bounding outline, in meters.

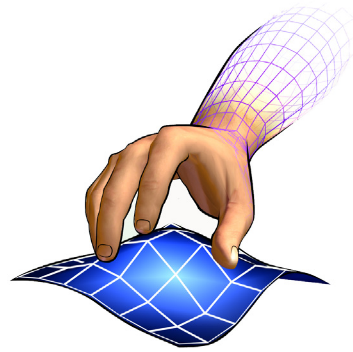
Table A-3: Commonly used attributes

Label	Type	Description
DtZMin	Real	Minimum Z coordinate of bounding outline, in meters.
DtXMax	Real	Maximum X coordinate of bounding outline, in meters.
DtYMax	Real	Maximum Y coordinate of bounding outline, in meters.
DtZMax	Real	Maximum Z coordinate of bounding outline, in meters.

MÄK implements this specification model using the following rules:

- Each area has its own specification.
- All of the segments, nodes, and edges making up a feature such as a road or river must share the same specification.
- If two features are different by even a single attribute value, they have different specifications.

! Using the DtFacNumber attribute, each feature can have its own unique specification. Another way to achieve unique specifications is to add additional attributes, such as street name or river name.



Index

B

C

E

F

G

N

O

P

T

U

V

W

X-Y-Z

A company of VT Systems

Link - Simulate - Visualize

TDB-4.2-1-090707

68 MOULTON STREET CAMBRIDGE, MA 02138 617.876.8085 www.mak.com