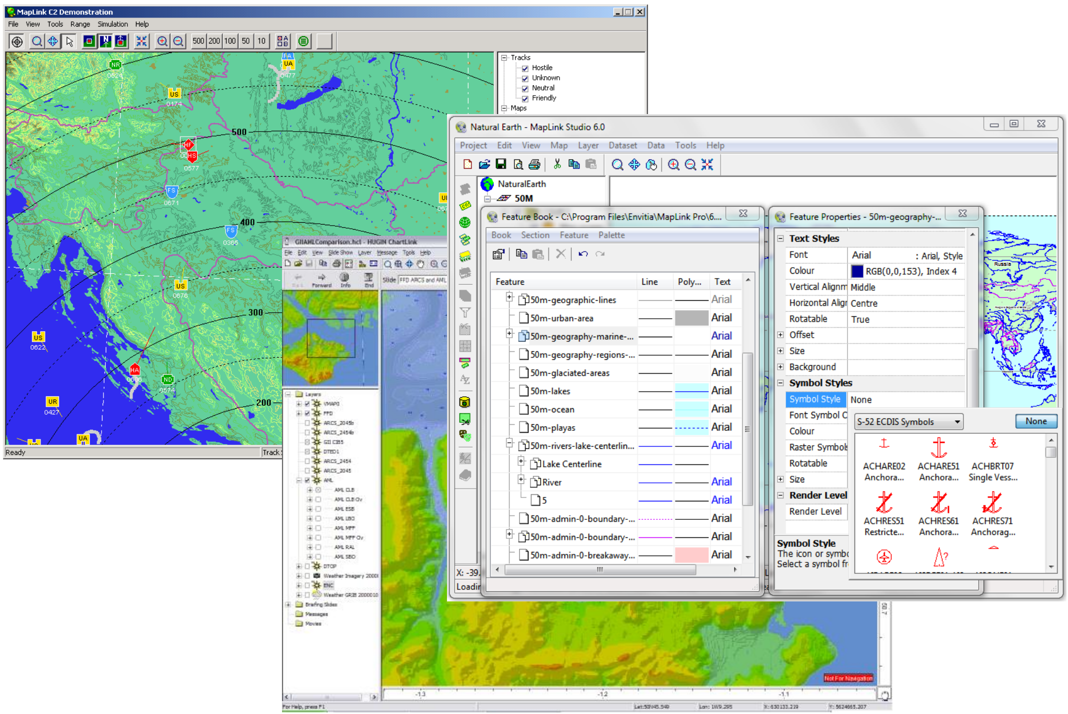
ENVITIA

DISCOVER > ANALYSE > ACT

# MapLink Studio User Guide

# MapLink Studio Features Overview with Exercises

AUM1115|21 January 2022| Status: Approved




North Heath Lane, Horsham, West Sussex, RH12 5UX, United Kingdom
Tel: +44 1403 273 173  Email: info@envitia.com
**www.envitia.com**

TABLE OF CONTENTS

# Table of Figures

# 1. INTRODUCTION

This guide is a key resource for all users producing maps for applications with MapLink Pro. It is aimed at both developers and end users, and at those starting with MapLink Pro Studio through to experienced users. It is designed to provide overview, tutorial and guidance information for those producing complex maps with the wide range of source data formats available and having to meet detailed system requirements. It is intended to complement the MapLink Pro Studio Help, the other key reference.

The Guide will describe:

- How MapLink maps are structured.
- How to build multi-layer vector, raster and gridded maps.
- How to build multi-layer terrain databases.
- Key concepts such as map coordinate systems.
- Map data support within MapLink Pro.
- Performance issues in relation to maps.
- How to optimise map structure visibility through an end-user application.
- Configurations file management (colours/styles/symbols).
- Options for providing users map import tools.

MapLink Studio User guide provides descriptive text, examples and tutorials. Sample data and output maps accompany the guide.

MapLink Pro Studio is primarily aimed at IT literate map users with some geographic background but assumes no essential knowledge of programming.

The requirements of map processing and the issues involved vary considerably from application to application. In some cases a simple map only is required. In others very high performance maps are required, very large maps must be created, or both.

Like most tools, the degree of expertise required depends on the complexity of the map requirement. As a further resource, Envitia can undertake training courses either at our Head office or on site for MapLink Pro Studio and the programming interface for MapLink Pro. To accelerate your development Envitia can undertake project work or consultancy. Please contact sales@envitia.com if you wish to take advantage of our expertise.

## 1.1. Typographical Conventions

The following typographical conventions apply:

Instructions to be carried out are described using the Courier New font

```
This is an exercise
```

Background information uses the Verdana font

This is background information

## 1.2. Sample Projects and Data

MapLink Pro is supplied with sample data, projects and generated maps to assist in gaining an initial competence in the product before embarking on a more advanced and specific processing task. Some sample source data, projects and output maps are installed with MapLink Pro. There is also free source data available on the internet.

## 1.3. Pre-requisites for the Examples

In addition to this guide and sample data you will need:

- A PC with MapLink Pro installed, with an evaluation or permanent licence for each of the data sources in which you are interested.
- Sample data installed with MapLink Pro.

# 2. Overview of the MapLink Pro Product Suite

The MapLink Pro software suite provides you with the ability to combine different map and chart data formats to create a high performance composite map. This map can then be displayed in an end user application built using the MapLink Pro libraries. The libraries contain functionality that enables both the map and real-time dynamic information to be displayed. It has been designed to meet the performance, data volumes and robustness requirements of modern military and civil systems.

MapLink is designed to be a very modular product. It provides the ability to display fast maps, imagery and overlay with no size limits, including rotating maps, layer and detail control and real time data on a wide range of platforms (from Windows (XP/Vista/Windows 7 and newer versions) and Unix variants (Solaris/Linux) through to fully embedded.

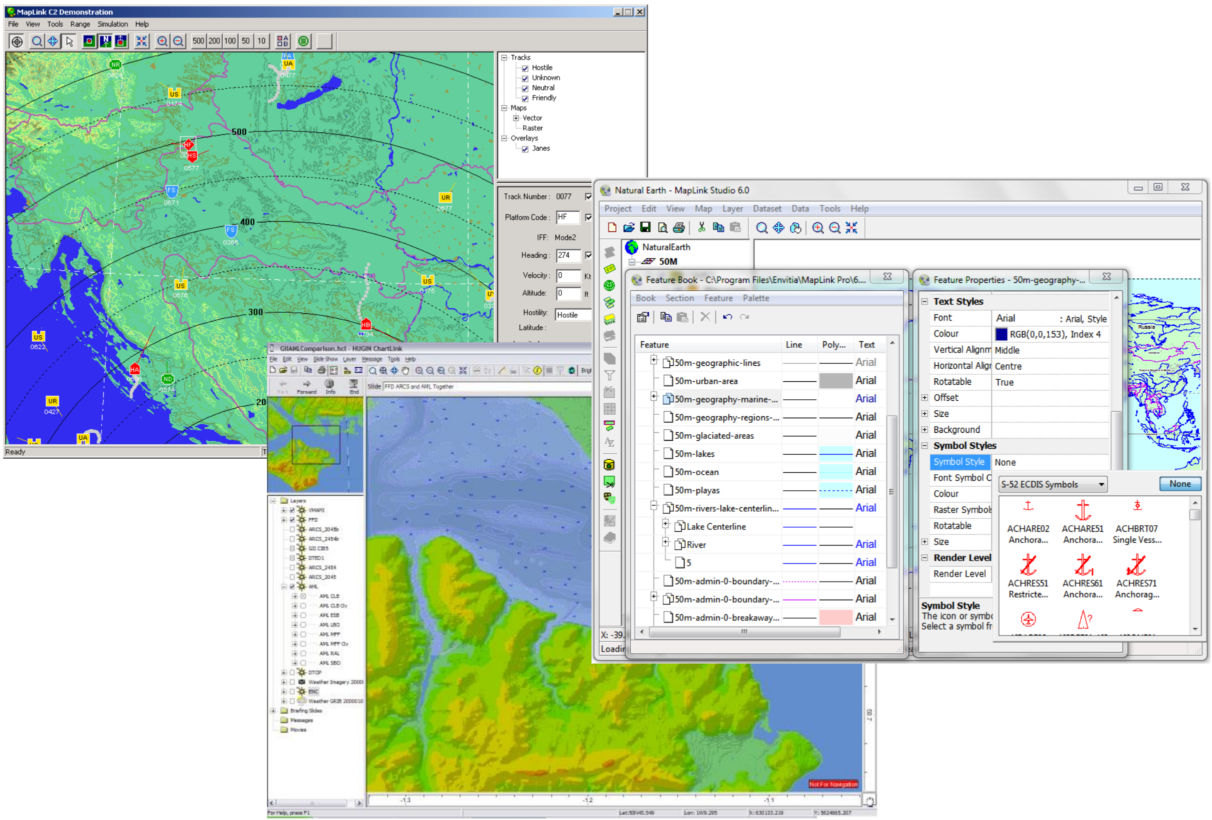


**Figure 1 MapLink Pro Screenshots**

The libraries allow the development of the application to be almost completely independent of the data that will ultimately be used and its visualisation, and in fact the map preparation requires no programming skill and can be put in the hand of geographic or domain experts or computer literate end users.

MapLink Pro also supports the automation of map processing using templates set up with MapLink Studio.

## 2.1. MapLink Pro Core Runtime Structure

It is essential to understand some aspects of the MapLink Pro runtime structure in order to effectively build maps for a MapLink Application. See Figure 2 below.

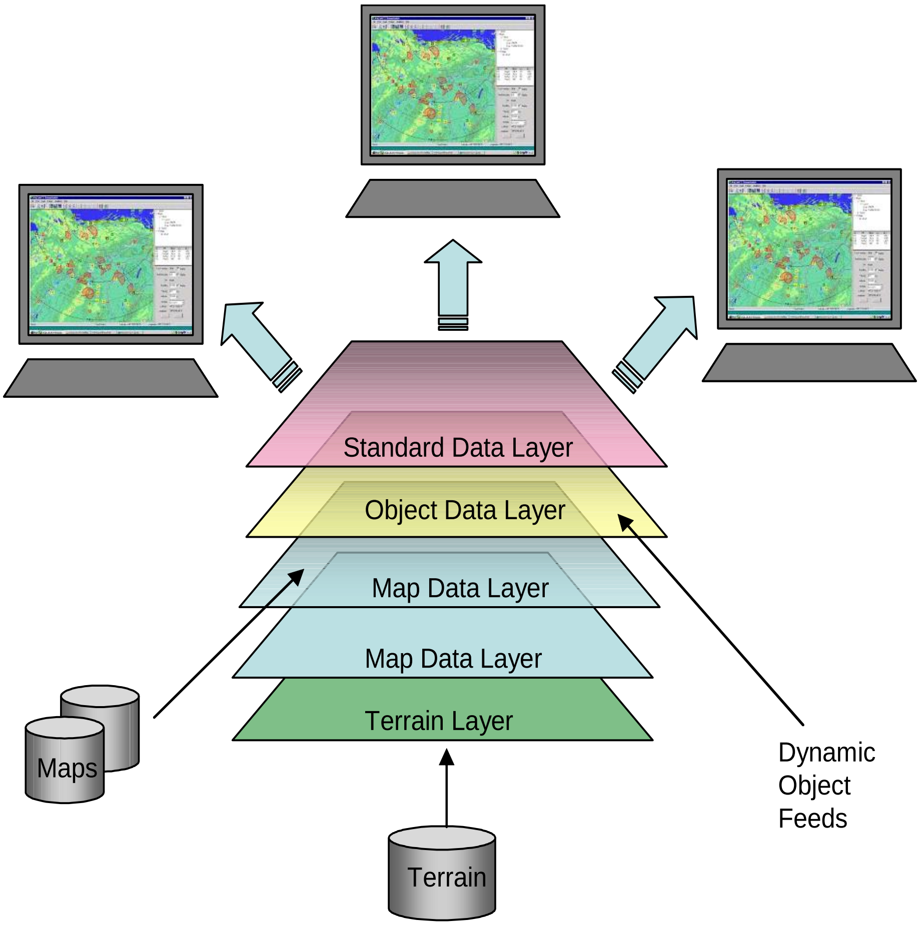



**Figure 2 MapLink Pro Runtime Structure**

The MapLink Pro runtime libraries are integrated into an application and the Application Programming Interface (SDK) is used for mapping related functions. MapLink manages the area of the map required for display and also allows the application to convert coordinates (for example from screen pixel to lat/long, or lat/long to screen pixel) using information in the map you create. This ensures very accurate alignment of objects with the map and ensures positions are still correct if the map projection or area of operation is changed.

With MapLink you can display multiple maps, overlays and dynamic objects together on multiple displays. MapLink helps to ensure that the update is very fast. MapLink ensures that only required data from the map is loaded rather than loading the entire map. This makes it possible to have very large maps and still pan/zoom very quickly across them. The map author sets up what is typically displayed at any zoom level. Although this can be varied at runtime, the map author's settings provide an essential default.

## 2.2. Additional MapLink SDKs

MapLink Pro supports additional Software Development Kits (SDKs) each with additional SDK calls and in some cases with complementary features in MapLink Studio.

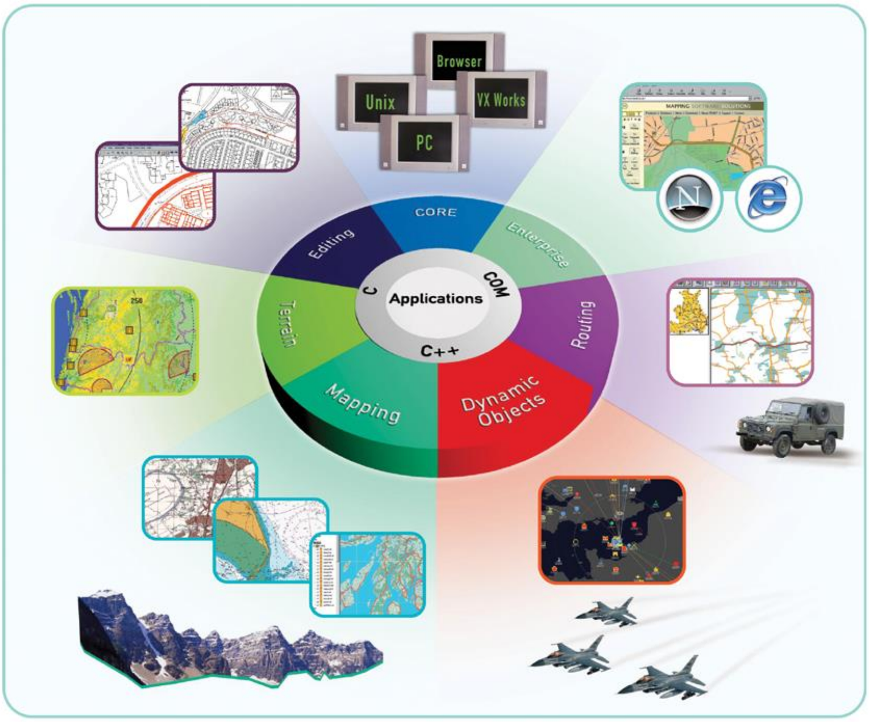



**Figure 3 The MapLink Pro SDKs**

Some of the additional SDKs are:

- Dynamic Data Object SDK. Manages the display of large numbers of very dynamic objects (for example fast moving aircraft or missiles) on top of maps in multiple displays (standard).
- Geometry SDK. Provides a set of programming functions to create a wide range of graphics objects at runtime (standard).
- Graphical Editor SDK. Provides a set of tools to construct complex runtime editing facilities, drawing objects, moving scaling and rotating and modifying their attributes e.g. colour, style (option).
- Terrain SDK. Allows querying of terrain height/slope information and derivation of areas of no-go, radar shadow etc at runtime (option).
- Network SDK. Allows the application to compute route based algorithms (shortest route, avoiding a given area etc) based on node and link data derived from map data sources (option).
- 3DSDK. Provides the capability of displaying 2D maps on a 3D Ellipsoid Earth, including WMS and CADRG runtime import.
- Accelerator SDK. Uses OpenGL or DirectX to harness the GPU to deliver ultra fast redraw times and rotation of raster maps.
- Satellite SDK. Render Satellite information in 2D or 3D.

## 2.3. Key Components of MapLink

The MapLink suite consists of both Tools (applications such as MapLink Studio) and programming libraries with Application Programming Interface. In principle the components split into three elements.

- The data preparation element (import of source data, fusion into a map, terrain or network topology). The key component of this is MapLink Studio, but other tools and utilities support this function.
- The resulting file-based Map/Terrain/Topology set which is in an optimised format for runtime access and is a hierarchy which can be many gigabytes in size.
- The runtime environment which uses the Map/Terrain/topology set and provides display, overlays, queries etc. This element is used by the application developer and can be accessed via C++, or .NET programming environments.

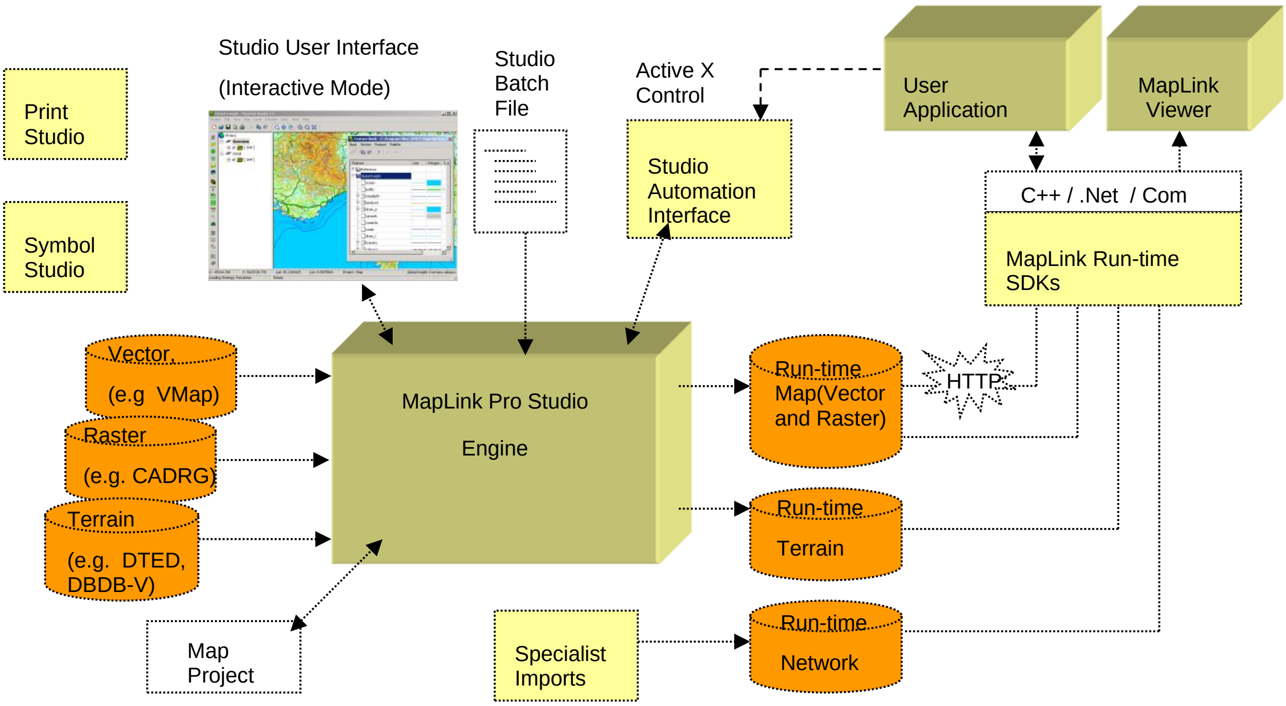



**Figure 4 MapLink Pro Components**

In addition to the three elements above the Studio Automation Interface capability, at present only under Windows, is able to control the preparation environment provided by MapLink Studio from the runtime application and so import data automatically. To assist with ease of data processing MapLink Studio can also be invoked from a batch file containing setup instructions. We will touch on this later in the guide but the primary aim of these tools is to avoid the need for expert users of MapLink Studio once the relevant templates are set up; allowing novice users to ingest new data into the map set via a simple interface.

## 2.4. Keys and Licensing

In order to use MapLink Pro you require a valid licence. The MapLink Pro Licence Agreement (both permanent and evaluation) is presented to you when you install MapLink. In order for MapLink Studio to operate you require one or more keys (depending on the capabilities you are using)

- **MapLink Pro Developer Licence** gives access to MapLink Studio and the developer libraries. It gives the user the ability to develop an end-user application based on MapLink. It includes the right to Developer and Studio Keys. The developer key allows the runtime to be run in Debug mode on the licenced machine.
- **MapLink Pro Studio Licence** gives a user access to the map preparation environment. Specific options e.g. map import filters, SDK options, are licenced separately. It includes the right to a MapLink Studio key.
- **MapLink Pro Runtime Licence** allows an end-user application linked in release mode against the MapLink libraries to be deployed. It includes the right to run an end-user application.

MapLink Studio deployments, with the exception of the end user application are protected by node locked licence keying. MapLink runtime licences are currently not keyed but Envitia reserves the right to do so in the future.

In order to obtain a licence key, open the Envitia Licence Key Administrator under MapLink in the start Menu. Any requests/questions related to licensing or any other questions can be put via phone or email to Envitia support on +44 1403 273 173 (support@envitia.com)

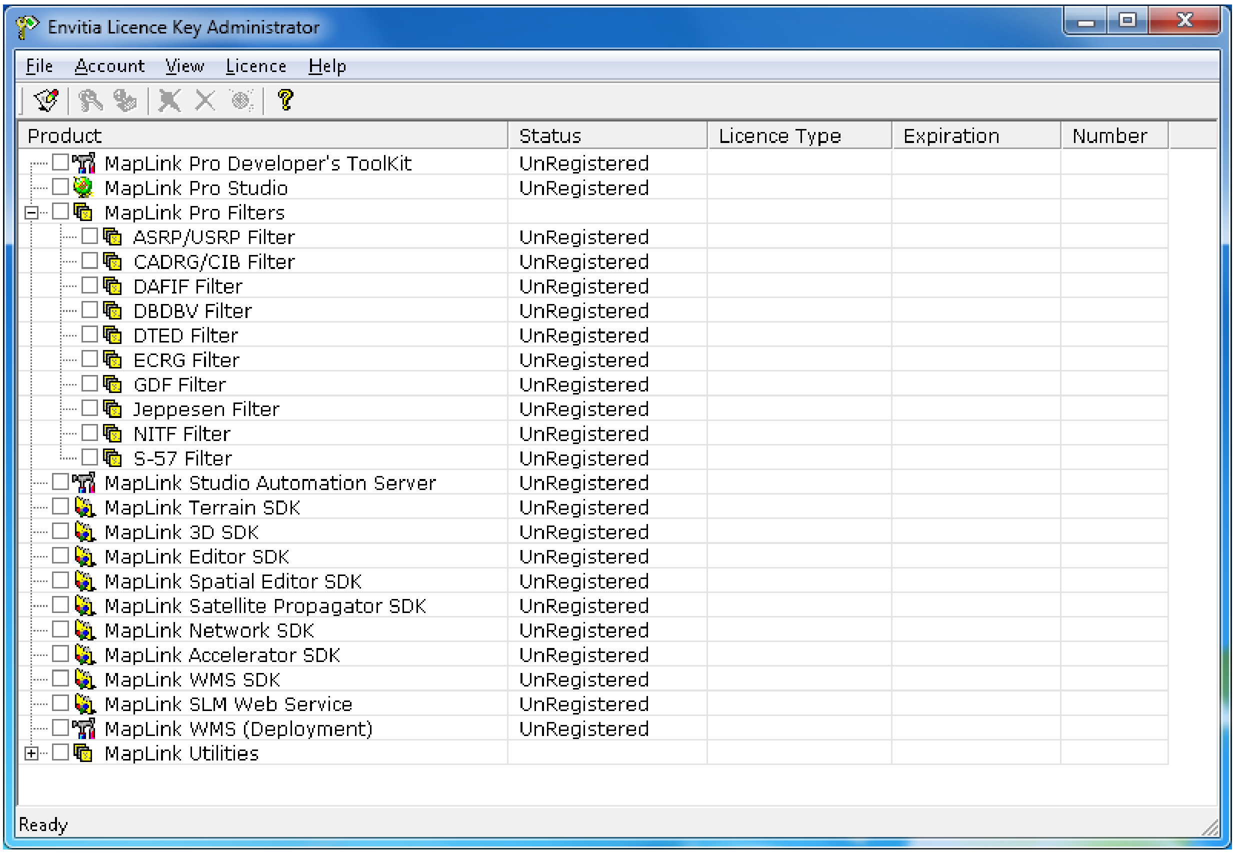



**Figure 5 The MapLink Pro Licence Key Administrator**

## 2.5. Documentation and Example Material

- MapLink Pro Studio Help.
- MapLink Pro Developers Guide.
- MapLink Studio Users Guide (this document).
- And don't forget you can email us at support@envitia.com or call the Support Helpline on +44 1403 273 173

## 2.6. System Requirements

The following are guide system requirements for both the MapLink Studio environment and the runtime.

| MapLink Studio | Runtime Environment |
|---|---|
| Available on Windows only (Windows 7, Windows 8.1, Windows 10) | Windows/Unix/Linux/Android |
| Operates best with significant amounts of disk/memory and with more than one processor core. | Windows most processors acceptable.<br>X11 Targets are specific to Release configuration. Specialised ports can be undertaken. |
| Minimum Specification: | Minimum of 65k colours on Windows |
| 1.8GHz Pentium® 4<br>1Gb on board memory<br>Local Disk<br>Plenty of swap space | Generated maps can be used on both Windows and X11 platforms. |
| | Maps aimed at being runtime friendly. |
| | Map Cache size controllable. |

**Table 1 Minimum System Requirements**

### 2.6.1. Supported Processors

The following processors are supported:

- Intel® Pentium® 4 processor family and higher.
- Non Intel® processors compatible with the above processor supporting the SSE2 instruction set.

#### 2.6.1.1. 64-bit MapLink

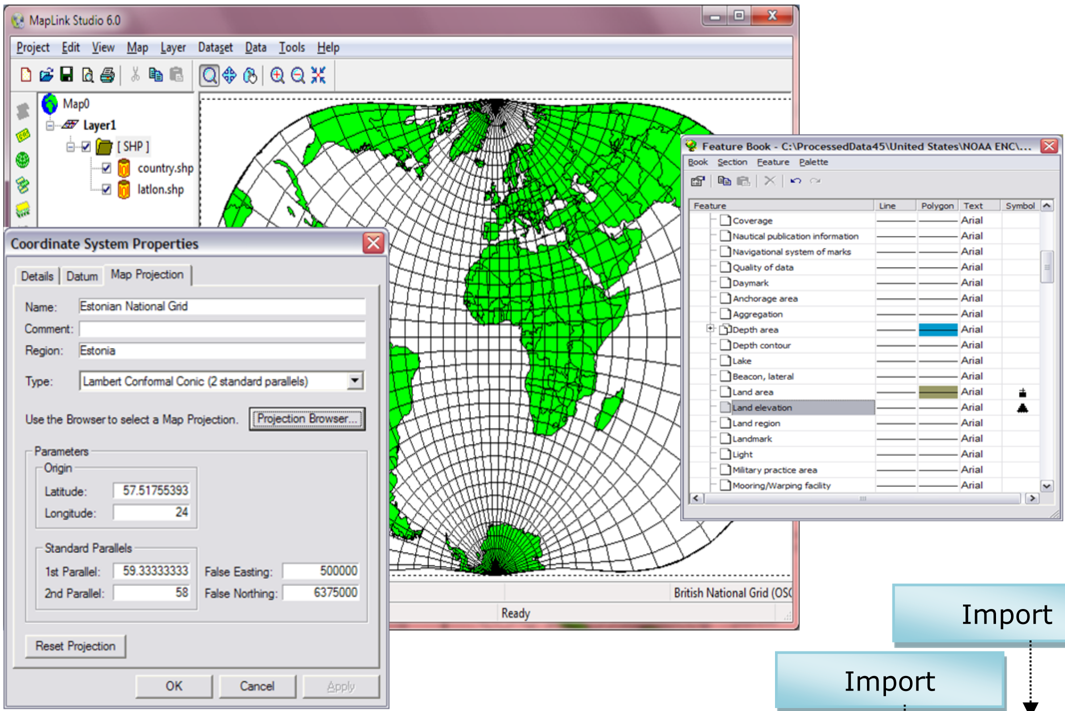
Early AMD x64 bit processors which lack support for `CMPXCHG16B` are not supported as this operation is required. These processors are the original "AMD Opteron Generation 1" (revision E and earlier).

### 2.6.2. Processing Large amounts of Vector or Raster

For processing large datasets or lots of raster we would recommend a 64bit Windows OS with more than one processor, 4Gb or more of memory and a hard disk large enough to hold the source data and the generated map.

# 3. MapLink Studio Overview

MapLink Studio is a powerful and flexible environment.




Some of the key capabilities are to:

- Fuse vector and raster maps, imagery and terrain, from disparate sources and on differing projection systems.
- Preview the input data to determine the quality/coverage.
- Define the default colour and style of the presented vector maps.
- Segment and filter the data for maximum performance.
- Prepare complex, multi-layer maps/terrain sets in a range of fixed and dynamic coordinate systems for use in runtime systems.
- Prepare templates for runtime map preparation environments.

All of these processes are dealt with within the MapLink Studio processing pipeline, a simplified view of which is shown here.

This guide covers the setting up of each of these conceptual elements to achieve the desired output from the process.

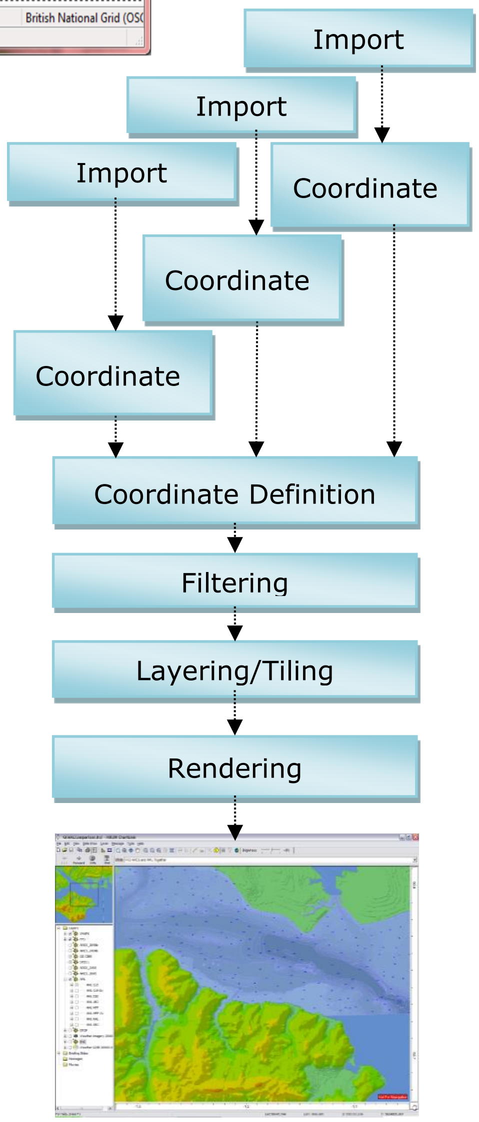



**Figure 6 MapLink Studio GUI and Processing Pipeline**

## 3.1. Data Types Supported

MapLink Pro supports many data formats and input standards. Data may be vector, raster, terrain or network and may be in one of many international, military or de- facto standard formats.

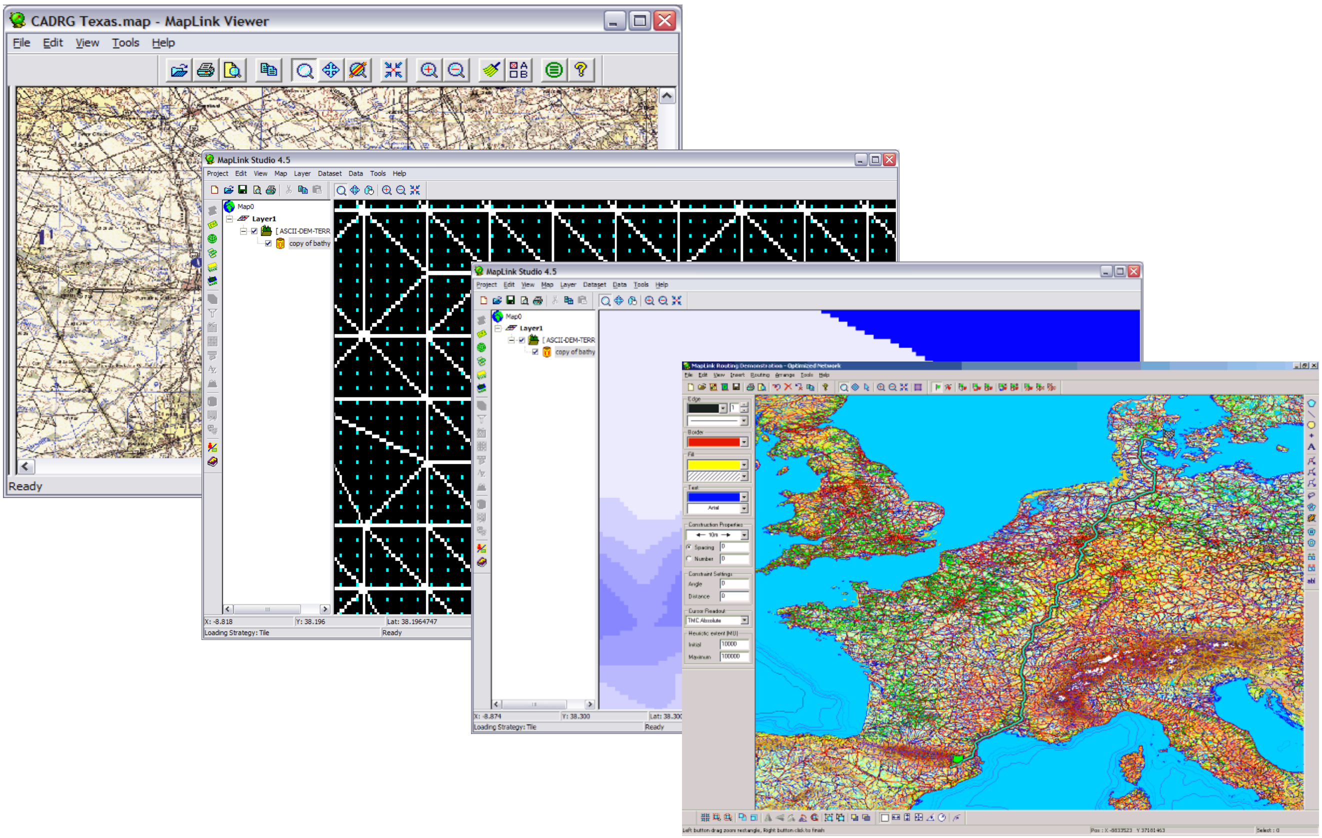



**Figure 7 Typical File Formats**

Geospatial data may be processed using MapLink Studio to create high performance maps with sophisticated rendering or in some cases imported directly into either specialised layers or into standard visualisation layers.

The direct import/display of data is supported in the MapLink Pro SDKs via:

- The `TSLKMLDataLayer` is designed for displaying KML/KMZ data in a 2D drawing surface.
- The `TSLS63DataLayer` is designed for handling S-63 or S-57 data correctly (including updates).
- The `TSLCADRGDataLayer` for drawing CADRG/CIB very fast at runtime.
- `The TSLFilterDataLayer` for loading and drawing GeoTIFF, NITF/NSIF, Raster.
- Vector Data may be imported via `TSLUtilityFunctions::importData/appendData` for S-57, Shapefile, MIF/MID, KML, OS MasterMap, OS Vector Map Local.
- General Raster Data may be loaded and displayed via `TSLRasterDataLayer` or `TSLRasterFilterDataLayer`.
- Terrain data may be loaded via `TSLDTEDTerrainDatabase`.
- Vector data in a database (Oracle) may be displayed using the `TSLDatabaseLayer`.
- The `TSLCustomDataLayer` can be used to create custom geospatial data layers.

- The `TSLWMSDataLayer` and `TSLWMTSDataLayer` for display of data from an OGC WMS or WMTS server

MapLink Studio can process significantly more geospatial data formats than can be loaded directly into the runtime. MapLink Studio will handle the re-projection of raster, vector and terrain data into a common coordinate system (see next section). This occurs when the map is created and not at runtime, saving a considerable amount of time to load and display a map.

A summary of the key formats which can be read by MapLink Studio (with the appropriate license/key) are given below. Filters shown in **bold** are standard, others are optional.

New filters are being added all the time or are created to meet custom requirements. Please contact Envitia if you have a specific requirement.

| Vector | Raster | Gridded |
|---|---|---|
| **DIGEST VPF** | **ADRG** | **ArcGrid ASCII (ESRI)** |
| **VMap0,1,2,UVMap,DNC etc.** | **CRP (Uncompressed)** | **ArcGrid Binary (ESRI)** |
| **Envitia ASCII** | **BMP/JPEG/TIFF/PCX etc.** | **ArcGrid Float (ESRI)** |
| **AutoCAD DXF** | **BSB Nautical Chart Format** | **OS Panorama (Ordnance Survey)** |
| **Shapefile (ESRI)** | **GeoTIFF** | **ASCII DEM (Generic)** |
| **MapInfo MIF/MID** | **Geospatial PDF** | DTED/DMED 0,1,2 |
| **NTF (Ordnance Survey)** | **MrSID[3]** | DBDBV 1,4,5,6 |
| **MasterMap (Ordnance Survey)** | **Other Raster[4]** | |
| **OS Vector Map Local** | DIGEST RPF | |
| **OS Vector Map District** | CADRG/CIB (Runtime & Studio) | |
| **OpenStreetMap** | ASRP (ONC, TPC etc.) | |
| **U.S. Census Tiger/LINE** | NSIF/NITF 2.1 | |
| **File Geodatabase (FileGDB)[1]** | ECRG | |
| **GML2/GML3 Simple Features** | USRP | |
| **KML Simple Features** | | |
| **Other Vector[2]** | | |
| DAFIF-Tabbed | | |
| DFAD | | |
| S-57 AML/ENC | | |
| S-57 AML/ENC for S-52 | | |
| GDF | | |
| NSIF/NITF 2.1 (CGM) | | |
| Jeppesen ARINC 424 NavData | | |

**Table 2 MapLink Pro Data Formats**

## 3.2. Coordinate Systems

A coordinate system defines a unit of measurement of position in most cases relative to a point on the earth. Coordinate systems can be spherical or geoid related (e.g. Lat/Lon WGS84) or represent positions of a flat surface via a map projection.

[1] The MapLink Studio user needs to install 3rd Party DLLs. Please refer to the MapLink Studio online help for additional information.

[2] This dataset type covers additional formats that are delivered by OGR. These have not been explicitly tested and may have format specific limitations

[3] The MapLink Studio user needs to install 3rd Party DLLs. Please refer to the MapLink Studio online help for additional information.

[4] This dataset type covers additional formats that are delivered by GDAL. These have not been explicitly tested and may have format specific limitations

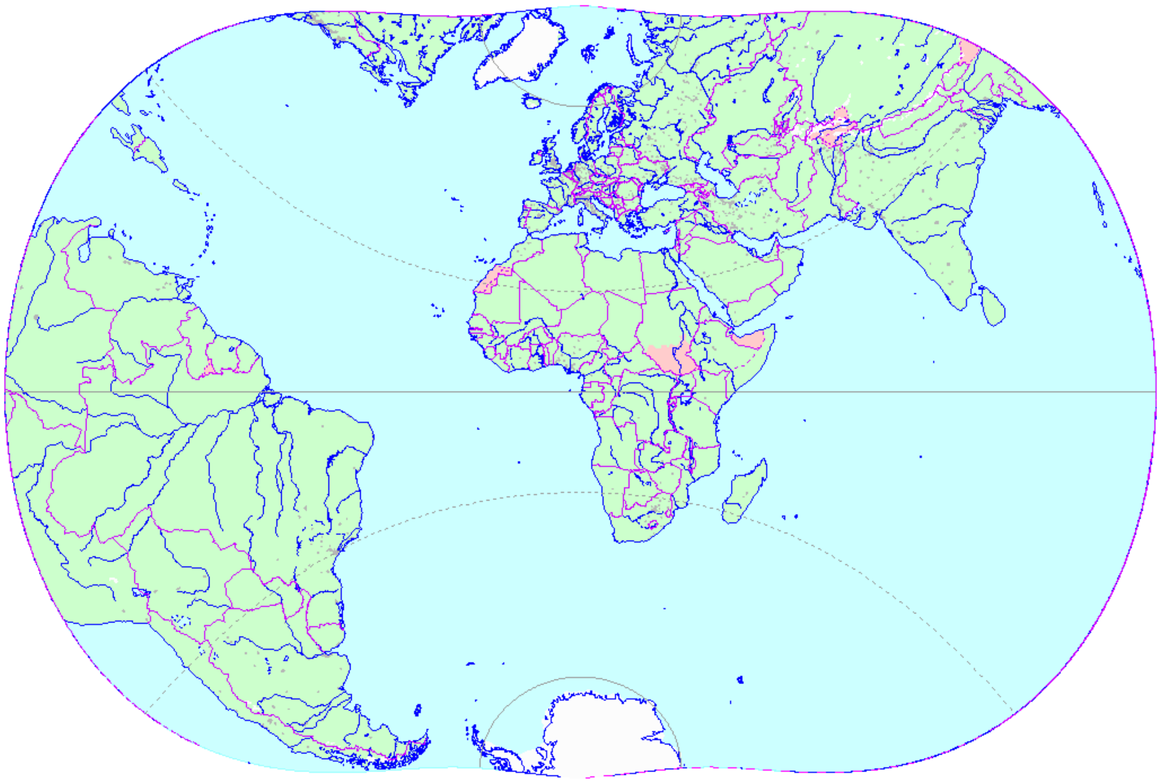


**Figure 8 The Earth in a Transverse Mercator Projection**

MapLink supports three key types of coordinate systems concurrently:

- Latitude/Longitude (typically WGS84 Spheroid/Datum model).
- Projected Coordinates (Map Units and User Units (scaled map units))
- Device Units (Pixels on screen or printer)

All of these can be used at runtime. The runtime map retains the conversion details.

MapLink supports a range of standard coordinate systems and custom defined systems based on supported projections and datums

Data can be sourced in any coordinate system. For example, you can mix data in British National Grid with data in WGS-84 and UTM and produce a map in any supported fixed, dynamic or semi-dynamic MapLink coordinate system.

Vector, raster and gridded data can be projected from any supported coordinate system to any other supported coordinate system.

## 3.3. Rendering Control

By rendering we mean defining the colour and style for features present in vector map data. This is because, unlike a drawing, map data typically does not contain specific details of colour and style. This detail is defined separately and so can be varied. For example minor roads may in one case be red with a black border, in another dotted green and so on.

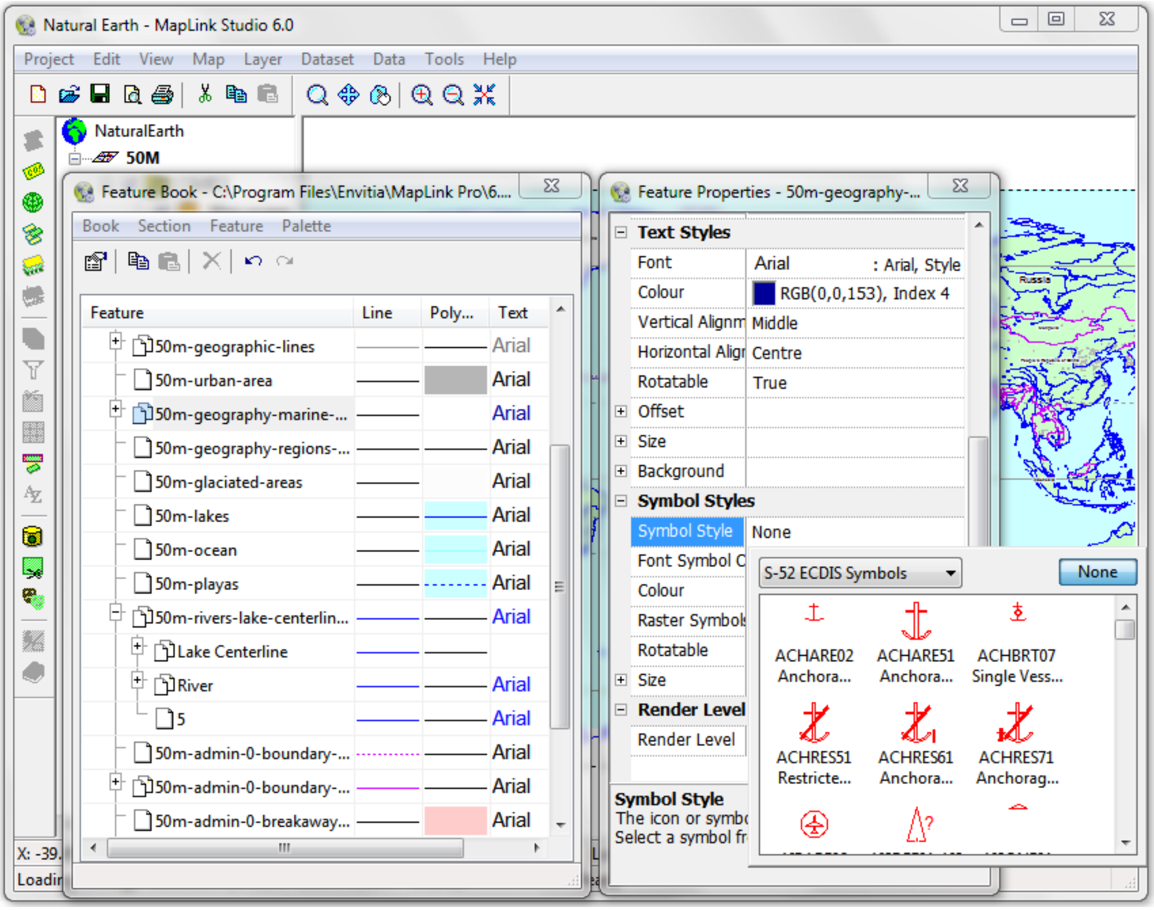



**Figure 9 The Feature Book**

MapLink Pro allows easy control of vector map display style both at processing stage and at runtime in an end-user application. The basic process is:

- Define custom symbology and linestyles
- Use the Feature Book to define vector map styles
- Create Feature Books for specific map types/uses
- Change the styles dynamically in your application as required

## 3.4. Unicode

MapLink Studio and the MapLink runtime support the display of multiple language scripts (from version 8.0 and newer). MapLink Studio converts on input all text data to UTF-8 from the source data's code page, so you do not generally need to worry about mixing source data files in multiple languages. If MapLink Studio is unable to correctly identify the code page the data is in you can alter the code page setting on a per file basis.

If you are writing an old map version and the input data has multiple code pages you have the option of over-riding what the input and/or output code page are. We would however recommend that you consider upgrading your runtime to MapLink 8.0 or newer as most Operating Systems only support a single code page at a time and thus only one set of text will display correctly in this situation.

For additional information please refer to sections 4.13 and 4.14.

## 3.5. Other Capabilities of MapLink Studio

MapLink Studio is also capable of the following:

- Feature Masking/Layering/Tiling/Clipping. MapLink can create maps with the appropriate resolution and content for the zoom factor and so allow fast pan and zoom to virtually infinite map data sets.
- Attribution. Can selectively embed source attribute data in maps, so that the application can query and display them (e.g. Fields such as **Date Sunk**, **Name of Object** etc.)
- Terrain Database Creation. Can create multi-resolution terrain datasets for runtime terrain queries using gridded data.

## 3.6. Accuracy and Resolution

The core projection mathematics is highly accurate. It is tested against independent data and subject to careful scrutiny. Taking a lat/long point, forward project and then inverse project the resulting lat/long position accurate to within 10-7 % of the original latitude and longitude.

With datum transforms, positional accuracy is typically a centimetre or better. All of these calculations though are limited by the accuracy of the model used. The map projections included have been derived from the "USGS Paper 1395. Map projections – A Working Manual, J P Snyder".

Datum Transforms and the coordinates used are based on European Petroleum Survey Group (EPSG) Geodesy Parameters (Release 6.2) www.epsg.org. Account has also been taken of "NIMA Technical Report TR8350.2 Edition 3 WGS 1984 Definition".

Coordinates are stored as fixed point numbers. This gives well defined accuracy and efficiency of storage (1cm resolution if global coverage assumed).

MapLink dynamically selects fixed point or floating point mathematics in display calculations to balance accuracy and performance of map display. The aim is always to maintain sub-pixel accuracy.

## 3.7. The MapLink Project Structure

A MapLink Studio project defines all of the processing required to create a single multi-layer, mixed vector, raster and gridded data map or terrain database. It consists of 4 files:

- .mlp - the project itself
- .mlc - information about the map data files referenced in the project
- .fbk - the Feature Book used by the project (more later)
- .mtf - stores details of any user defined coordinate systems

A project defines a single output coordinate system (static or dynamic) for example, the map is in British National Grid.

A project contains multiple layer definitions containing datasets and data.

The Project Hierarchy:

| | |
|---|---|
| - MAP | The project contains a single map. |
| - LAYER | The map contains one or more layers. |
| - DATASET | Each layer consists of one or more datasets. |
| - DATA | Each dataset contains one or more datafiles. |

The project is built using a standard Windows Tree, with levels corresponding to the hierarchy above (see below)

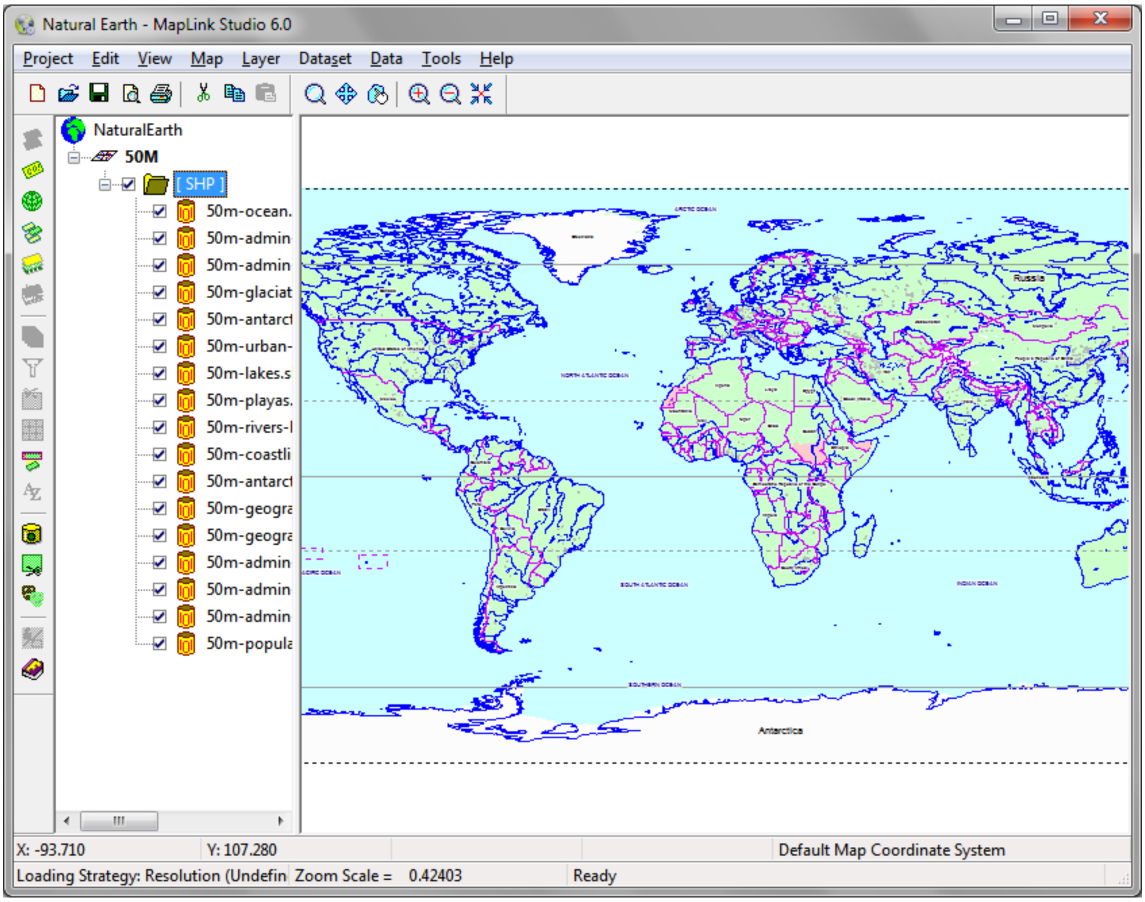



**Figure 10 The MapLink Pro Project Structure**

Let us examine each of these elements of the map hierarchy in a little more detail.

## 3.8. The 'Map'

The single map defines the following:

- The coordinate system, for example, British National Grid.
- The loading strategy. This defines when swapping between layers occurs.
- Some advanced options which will be dealt with later.

Each of these options is available from either the 'Map' menu or by right clicking on the map object in the tree.

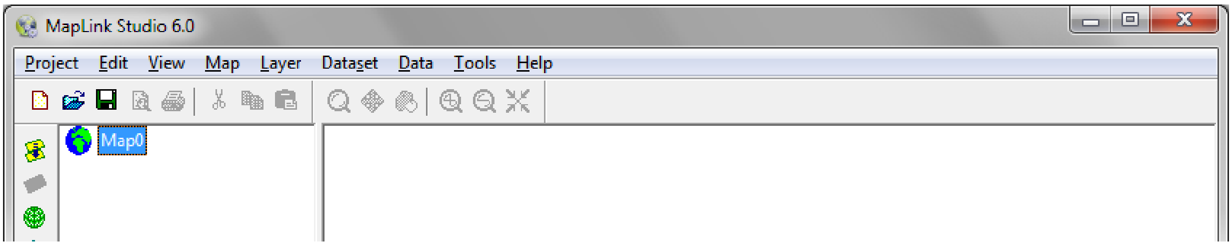



**Figure 11 MapLink Pro Map Object Selection**

## 3.9. The 'Layer'

Each layer defines a view of the map at a particular scale or style. For instance, one layer may be an overview of England, another more detailed overview of London, and another street level map of London. Each layer also has the following properties

- Clipping, to remove unwanted data.
- Filtering, to reduce the density of data.
- Tiling, segment the data and allow high performance.
- Terrain options for terrain database acquisition.

Each layer contains any number of Datasets which actually contain the data. Each of the options are available by selecting the layer in the tree and then either using the 'Layer' menu or by right clicking on the layer object itself in the tree.

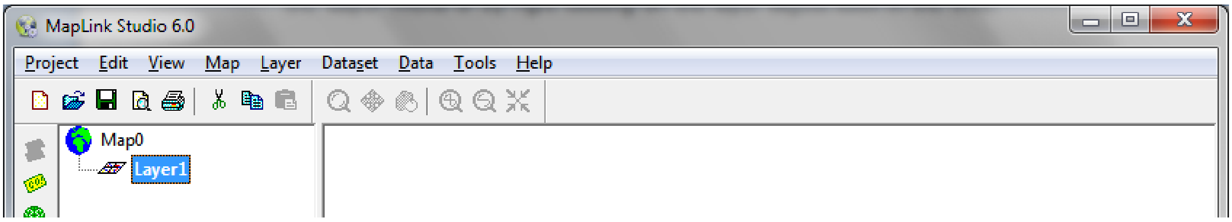



**Figure 12 MapLink Pro Layer Selection**

## 3.10. The 'Dataset'

The Dataset may hold any number of map data files. All files within each Dataset must be the same file format in the same coordinate system. Each Dataset has the following properties

- Clipping, to remove unwanted data.
- Input coordinate system information, to allow the data to be "unprojected".
- Feature masking, to remove unwanted map features.
- General raster, vector or terrain configuration options.

- Format specific configuration information.

Each of these options is available by selecting the Dataset in the tree and then either using the 'Dataset' menu or by right clicking on the relevant dataset object itself in the tree.

Each Dataset can contain any number of data files

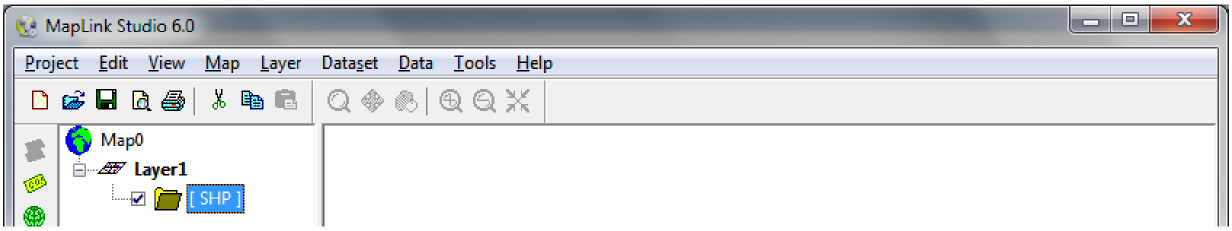



**Figure 13 MapLink Pro Dataset Selection**

## 3.11. The Product 'Dataset'

Product Datasets are preconfigured datasets stored in MapLink Studio template files. They are intended to simplify the configuration and rendering of common data products, such as; S57, VPF, OSMasterMap, OS VectorMap Local, etc.

To add a product dataset, select one of the items highlighted in Figure 14

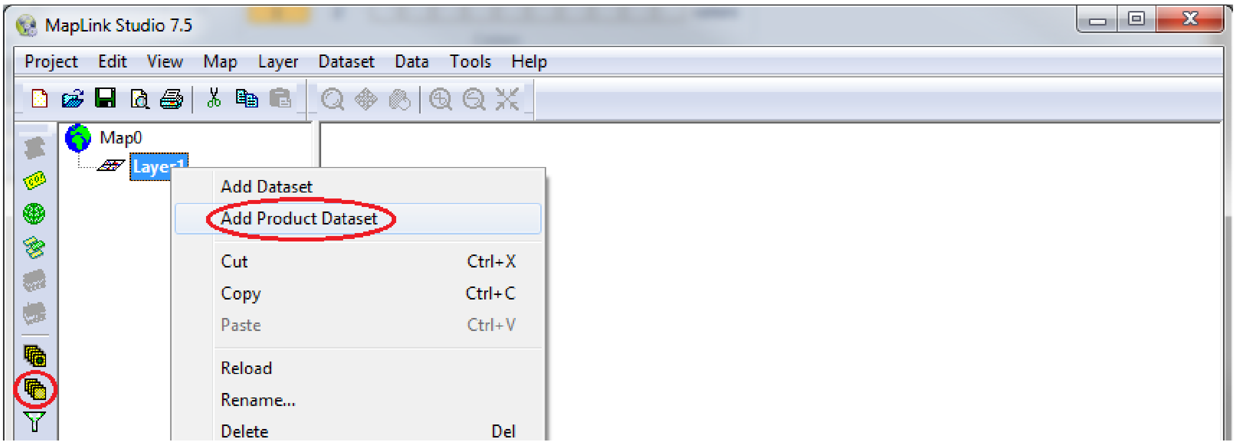



**Figure 14 MapLink Pro Product Dataset Selection**

These datasets have had any necessary subclassing already setup and are supplied with an associated Feature Book Section with rendering styles defined.

These datasets have been setup with a pre-determined TMC per MU value that the user should ensure is the same as that set for the project that the dataset is being added to.

The user also needs to ensure that the project and the dataset both use the same Palette.

The dialog is shown in Figure 15.

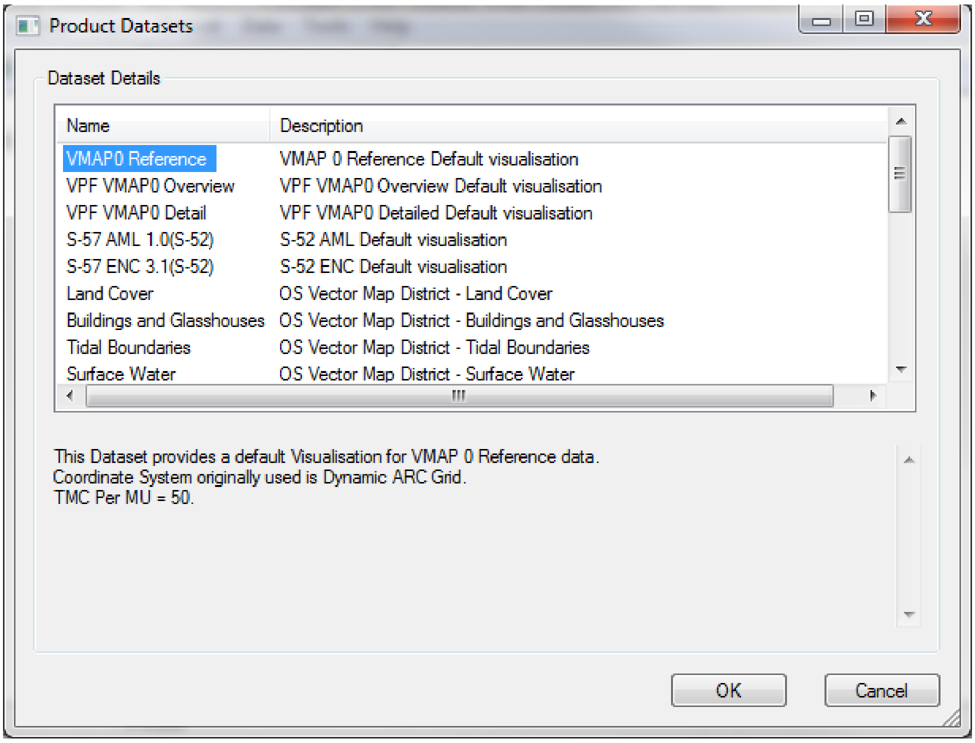



**Figure 15 MapLink Pro Product Dataset Dialog**

## 3.12. The 'Data'

A Data file is a single map data file, raster or vector. Each piece of Data has the following properties

- General information (extent of the data etc.).
- General raster or vector configuration options.
- Format specific configuration information.

Each data file can be a single flat map file e.g. OS NTF or S57 or the index to a more complex structure (e.g. VPF)

When a vector Data file is added to the project for the first time, it is analysed to find the extent of the features contained within the file.  When a raster Data file is added, it is analysed to find the size of the raster.

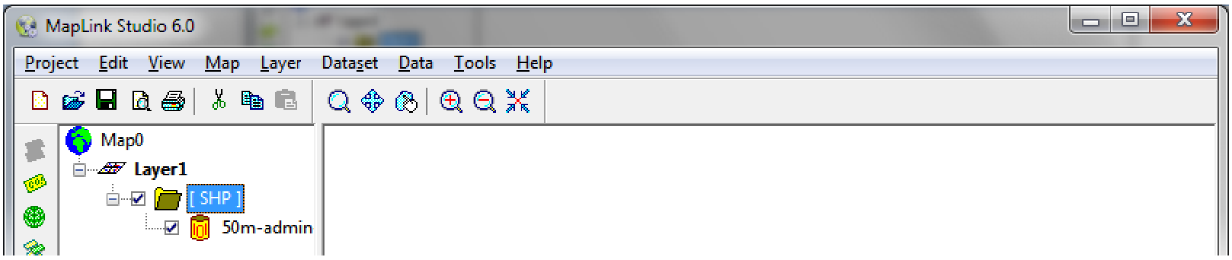



**Figure 16 MapLink Pro Data Item Selection**

## 3.13. The Output Map

Files created when a map is generated

The Map creates:

- 1 x .map	The control file to be loaded into the SDK

- 1 x .pal The colour palette of the map

Each Layer creates:

- 1 x .dtl The definition of the Layer
- n x .tmf Vector tiles
- n x .tpf, .tp1, .tp2 Raster Pyramid files/tiles
- n x .ttg, .tfg Terrain files

For Internet Deployments the file names change slightly. Depending on the setting in MapLink Studio (Write Layers into individual directories) a sub-folder will be created for each layer and all files for a layer will be created in this sub-directory.

NOTE: Since layer and dataset names do relate directly to file and directory names, saving a map in the same directory as another map even if it has a different name is ill advised and will almost certainly result in one of the maps being invalid. It is therefore strongly recommended that each map is created in a different directory.

A map can easily be previewed using the MapLink Viewer. This can be invoked either from the MapLink Studio Tools Menu or the Start Menu. In the former case the last map created in this invocation of MapLink Studio will be displayed automatically.

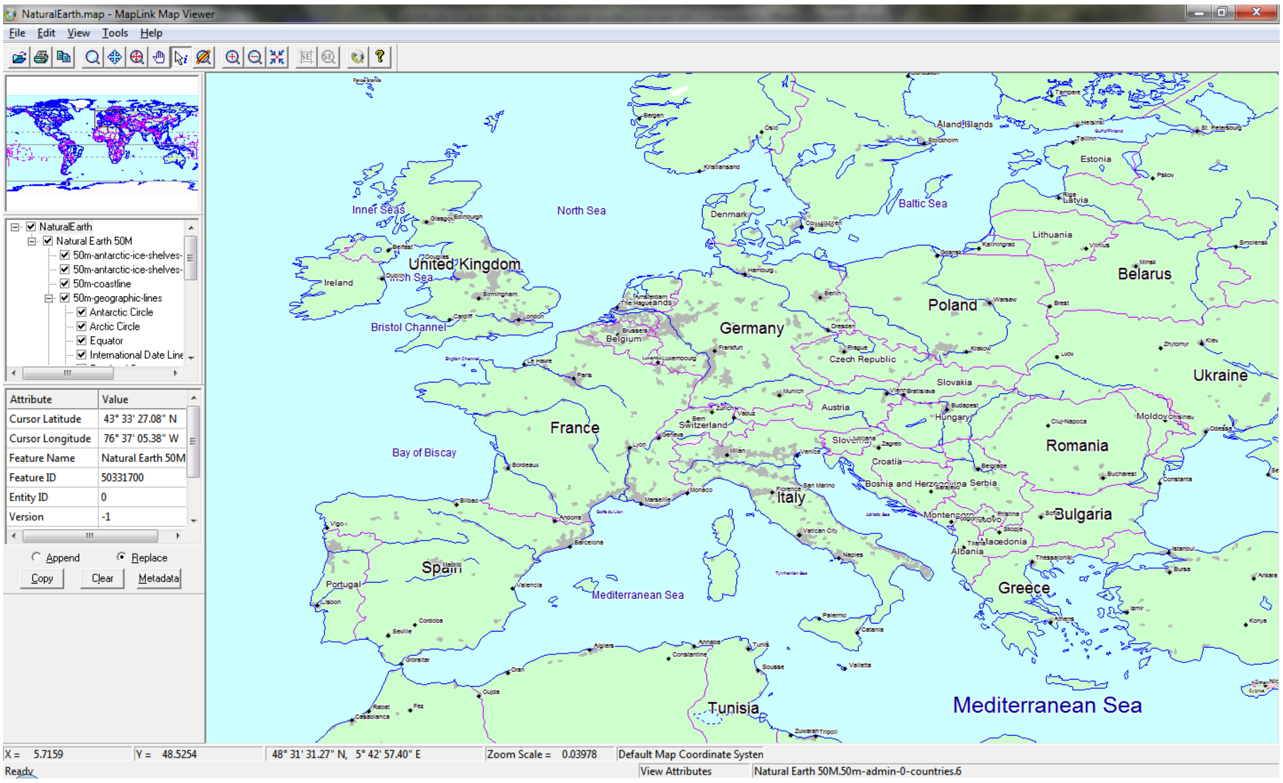


**Figure 17 MapLink Pro Map Viewer**

## 3.14. Exercise 1 Creating a Vector Map

When you have completed this exercise you will be able to

- Use MapLink Studio to define a simple map containing one layer of Shapefile data
- Create an output map from the MapLink Studio project
- View the output map using the MapLink Viewer

## 3.15. MapLink Software

The MapLink Software can be found under the Windows Start Menu.

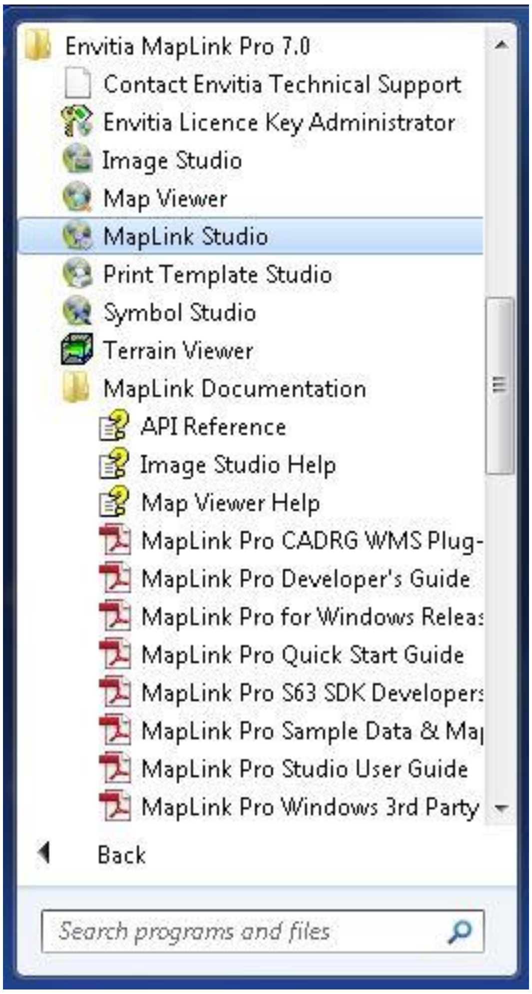



**Figure 18 Starting MapLink Pro Studio**

## 3.16. Creating a Map

MapLink Studio has a standard Windows interface with menus, toolbars and buttons. As with all Windows applications, options are only active when appropriate, and will otherwise be greyed out.

```
Start MapLink Studio. Create a new project, using the default
Project Template

Select Project | New
Accept the default Project Template when prompted
```

We will cover the Feature Book in detail in the next exercise.

```
Add a layer to the map
```

To add a new layer you can use:

- Menu option – Map | Add Layer
- Right mouse button on the Map icon to bring up the Map menu
- Top Button in left tool bar to add a layer

These three options exist for each MapLink Studio action – you can choose how you prefer to operate the application!

```
Select Dataset | New
Select SHP from the Vector Dataset list
Create a new Feature Book section called World
```

When you add a new dataset you can see all the different data formats that MapLink Studio supports.

```
Add the following Natural Earth 50M cultural Shapefile data
to your dataset.

As you are using Windows to select the data you can use
standard controls to open more than one file at once e.g. use
shift left mouse to select:

50m-admin-0-countries.shp
50m-urban-area.shp
50m-admin-1-states-provinces-lines-shp.shp
50m-populated-places.shp
```

Once you have added the data it will load automatically into the map display on the right. You will now be able to see the raw Shapefile data. MapLink uses a default colour and style to show the data. In this case black lines are used.

The top toolbar contains the navigation tools. These allow you to zoom and pan the map.

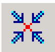
 Zoom Mode. The mouse is used to zoom in and out of the Map.

 Pan Mode. The mouse is used to zoom in and out of the Map View.

 Feature mode. This allows quick access to the relevant entries in the Feature Book.

 Zoom In. The mouse is used to zoom in to the Map by a single step.

 Zoom Out. The mouse is used to zoom in to the Map by a single step.

 Reset. Reset the Map to show the full extent of the layer.

```
Try using navigation tools to move around the map

Now is a good time to save the project. Use Project | Save as
you would with any Windows application

Look at how different items in the tree show their content
```

Selecting Map0 in the project tree shows an overview of the map. A rectangle is used to show the extent of each layer.

Selecting the Shapefile dataset shows the contents of that dataset.

Selecting `50m-admin-0-countries.shp` shows only data contained within that Shapefile data file.

The check boxes next to the files control whether or not the information is displayed. Notice the effects of turning these on and off. If an item is turned off then the data will not be included in the runtime map.

When a data file is added to the project for the first time, MapLink analyses its content. The information provided includes:

- The extent of the file
- Vector features contained within the file (if vector data)
- Raster file size (if raster data)

Vector data consists of separate geometrical primitives i.e. lines, polygons, text etc, that you can control the appearance of. Raster data is made up of one or more images, for example a photograph, which you cannot control the appearance of.

```
Select the 50m-admin-0-countries data file, and use the Data
Analysis option to view the information associated with that
file.
```

The top right and bottom left corner points show the extent of the data. The Vector Properties tab shows the contents of the data file. In this case, the `50m-admin-0-countries` file contains 1627 rings described by 99286 coordinates. We will look at Vector Configuration and Format Configuration in more detail later.

```
Use the Map menu to create an output map. Select a suitable
directory for the map to be created in.
```

This is the process where MapLink Studio takes the raw Shapefile data and the project configuration that you have set up, and from it creates an output map that can be displayed by a MapLink application.

## 3.17. The MapLink Viewer

```
Open the MapLink Viewer, located under the Tools menu.
```

This will load the most recently created output map. The MapLink Viewer is a sample application enabling users to quickly and easily test maps that are being created by MapLink Studio. Notice that this has the same navigation tools as MapLink Studio.

```
Use Windows Explorer to view the files that you have created.
```

Project files:

- **.fbk** The Feature Book.
- **.mlc** The MapLink catalogue, containing dataset information.
- **.mlp** The MapLink Project File.
- **.mtf** Contains user-defined transform information.

Output map files:

- **.dtl** Map detail layer.
- **.log** Log of any errors generated whilst processing the map.
- **.map** The map file. This defines the overall structure of the map.
- **.pal** The palette file. This contains the colours used within the map, including any user-defined.
- **.pth** Map path list file, detailing where to find output map files.
- **.tmf** The map data.

Congratulations, you have now created your first MapLink map!

# 4. Vector Map Handling

Vector map data has three main characteristics.

- It is made up of a series of geometric shapes e.g. lines, polygons, points, circles (Features).
- These geometric shapes do not usually in themselves have colour and style. This is implied from the type of data, e.g. roads, lakes (Feature Classes).
- Each geometric shape (primitive) may have data associated with it (road name, road width etc). This section describes the features of MapLink related to vector map data.

## 4.1. The Feature Book

Since in general, vector map data does not define the rendering styles that should be used to draw it, a mechanism is needed to define this style.

The MapLink Feature Book controls the rendering of Map Features. It is a library of rendering styles, typically for a specific format, and once defined can be transported across map projects. Therefore you can define your rendering styles once, and use them again and again. The Feature Book may be opened via Tools | Feature Book.  The dialog is shown in Figure 19 below.

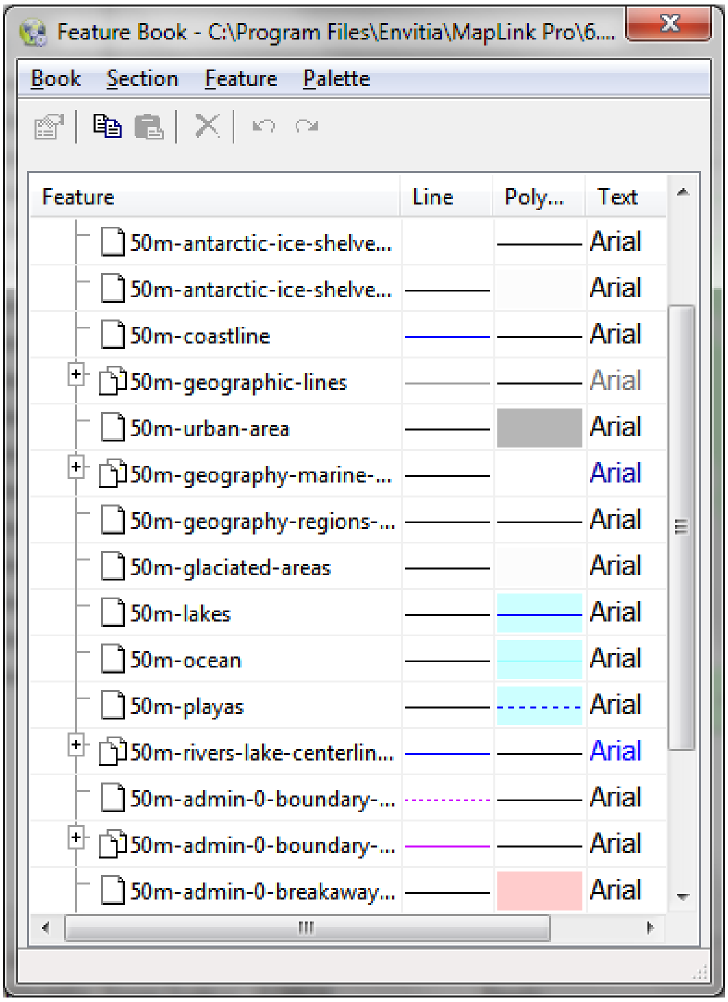



**Figure 19 Feature Book**

## 4.2. Key Characteristics of a Feature/Feature Class

The key characteristics of a 'Feature Class' that you can define in the Feature Book:

- Graphic attributes e.g. line, fill, text, symbol colour and style, style inheritance from a parent feature.
- Rendering order definition (which feature is on top). This is useful when data is not stored in rendering order e.g. lakes are required over land and therefore need to be drawn on top
- Specialist rendering choices and custom rendering e.g. lines made up of centreline, inner and outer edges, Geosym linestyles etc.
- Symbol and Text sizing (minimum and maximum size) width and Size in various units (map units, pixels). As we will see later the options in MapLink for these is quite extensive.
- Use of different colour palettes. It is worth noting that all maps to be displayed in the same application must use the same colour palette.

## 4.3. The Feature Book Structure

The Feature Book is broken into sections. Each Dataset in a map can use the same section, or a different section as required. In Exercise 1, we defined a Feature Book Section called World when we added the dataset. All of the feature classes in the data file were automatically added to the new blank section. Only feature classes missing from the Feature Book are added.

We could also have selected an existing Feature Book section if one was available, or imported one from another project that related to the data we were importing. Thus the Feature Book allows us to develop a library of visualisation characteristics for the map sources we intend to use and for the function for which we intend to use the map (we may have several visualisations for a map type).

The recommended approach is to build a Feature Book Section for each different map source type. For example, have a Feature Book Section for VMap0, one for DAFIF, one for Ordnance Survey NTF etc.

Feature class addition to the Feature Book is achieved when a Data file is added to the tree. It is analysed to obtain a list of all feature classes used in the file. These feature classes, if missing from the Feature Book section, are added to the Feature Book automatically.

If the feature class is already present in the Feature Book and already has colour and style settings associated with it, the feature will be rendered with this style. If it is a missing feature, it will have default rendering (linestyle thin black line, no fill style, no text style, no symbol style). These must be set up if different rendering is required.

## 4.4. Features in the SDK

As described above, feature classes from a specific Dataset are grouped into the chosen Feature Book section. At runtime individual features associated with the map can be accessed via the Application Programming Interface. This programming interface allows the developer to control the visibility of each individual feature class, and also to change the rendering style (colour/style font etc) if desired. Thus the Feature Book definition of the colour/style of each feature class represents the default/initial representation.

The Feature Book tree structure is visible in the same structure via the SDK for a generated map. For instance, using the following Feature Book tree:

World

countries

latlongrid

places

would result in the following features being available via the SDK:

"World.countries"

"World.latlongrid"

"World.places"

These feature codes can be used to declutter the map at runtime.

## 4.5. Feature Ordering for Display

MapLink allows you to control the order that features are displayed. The first element used to determine the order is the Dataset order within a layer, but where more than one feature class is present in a Data file (and therefore within a Dataset) the order the feature classes are drawn is from top to bottom in the Feature Book tree.

For example, a MIF file contains polygons for:

land area

desert

lake

To ensure that the desert and lake polygons are placed on top of the land area polygons, and therefore drawn on top, land area must be above desert and lake in the Feature Book tree. However, if land area was in a different file to desert and lake, then we would use the project tree to define the rendering order.

To re-order feature classes within a Feature Book section simply drag items up and down the tree. It is necessary in this case to press Reload in order to see the modified ordering.

There are two other characteristics that affect the order of drawing. The primitive type determines the order; polygons, lines, rectangles etc are all given the same priority, but symbols and text are always drawn last and therefore on top.

And lastly each feature class has a property called render level. This can be used to provide an additional degree of control. The rendering level ranges from (-5 to +5). The default rendering level is 0. This will normally have no effect. It is taken into account during drawing, with all items in the lowest render level (-5) being drawn first up to (5) which is drawn last. Although not covered in this chapter it is worth noting here that raster data always has a render level of 0.

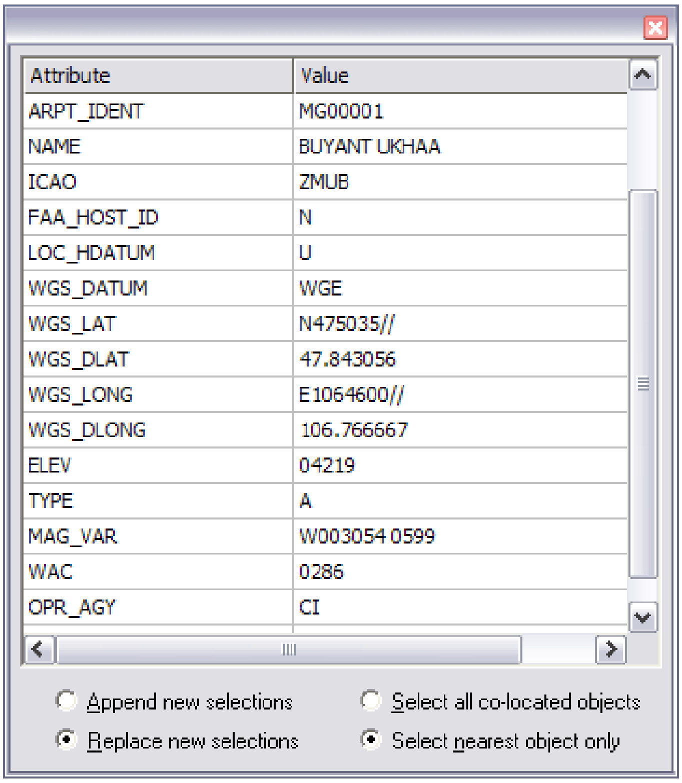
So to summarise the actual logic applied to drawing a layer with a number of datasets and associated Feature Book Sections, and a number of features in each is:

- Symbols/Text (drawn last) or other primitive (drawn first).
- Render level (Lowest to highest).
- Dataset order in the layer.
- Feature Class Order in Dataset's Feature Book Section.

## 4.6. Vector Data Attributes

Many vector data formats associate a set of data with each feature. For example, an Airport may have name, or a number of runways, type (military, civil, private etc). MapLink can use attributes in the following ways:

- To attach a name to an object
- To display text against an object
- To subclass a feature in the map
- To export attributes for use at runtime



| Attribute | Value |
|---|---|
| ARPT_IDENT | MG00001 |
| NAME | BUYANT UKHAA |
| ICAO | ZMUB |
| FAA_HOST_ID | N |
| LOC_HDATUM | U |
| WGS_DATUM | WGE |
| WGS_LAT | N475035// |
| WGS_DLAT | 47.843056 |
| WGS_LONG | E1064600// |
| WGS_DLONG | 106.766667 |
| ELEV | 04219 |
| TYPE | A |
| MAG_VAR | W003054 0599 |
| WAC | 0286 |
| OPR_AGY | CI |



**Figure 20 Typical Attributes for an Airport displayed in an Application**

## 4.7. Using Attributes to set the Name and Text Label

MapLink geometric objects have two fields with specific functions that can be set by attributes:

- The name field. If the name field is set to an attribute, MapLink Studio will display the name of the object in the status bar when you hover over it. The name is also available to a MapLink application and MapLink provides functions to easily search for it. The name field is commonly used as a key to some form of database in the application.
- The Text Label Field. This is the text that will be generated as a primitive for each point object in the map data. This is particularly useful for map formats where no text primitives are present in the data (e.g. Shapefiles).

Both of these fields can be set on either individual files or on the dataset as a whole (and will then apply to all files) by using the ‘Format Configuration’ Menu option. We will cover this dialog in a little more detail in the next section.

## 4.8. Using Attributes for Vector Subclassing

Subclassing allows attributes contained within vector datasets to be used to categorise data by creating new feature classes or by sub-dividing a parent feature. It is only available on vector datasets that include attributes for each feature.

MapLink normally allows two levels of subclassing to be defined for each feature. Any of the attributes that are available for a feature can be used for subclassing. For example, if the Shapefile "countries.shp" has the attributes "Name, Continent, Accuracy, Population, etc." then the two levels of subclassing could be:

```
Country -> Continent -> Name
```

This would group each country feature according to continent and then by name:

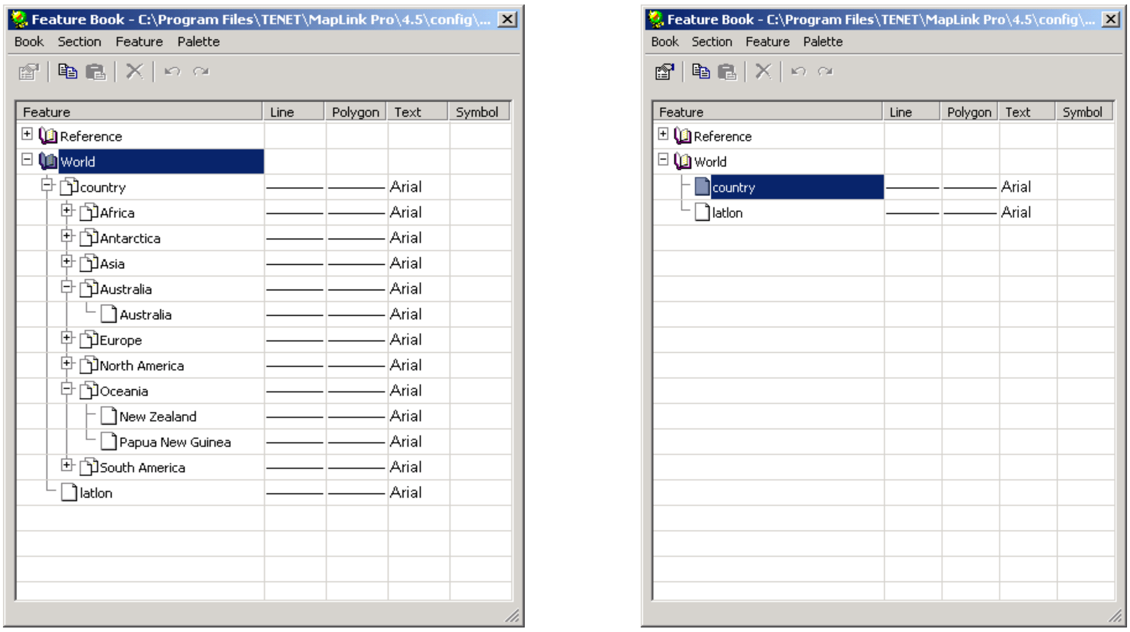



**Figure 21 Feature Subclassing**

Subclassing allows us to define separate representations (for example fill colour) not only based on the base feature class (for example country) but also based on an attribute of the country (such as its continent and country name); after subclassing both the base feature class (parent) and feature classes for each of the items present in the attribute (so individual continents and countries) are available in the Feature Book Section.

## 4.9. Configuring Subclasses

Subclassing configuration is similar in each vector format and is accessed by selecting Data File | Format Configuration or Dataset | Format Configuration.

Normally the panel includes “Level 1” and “Level 2” selection boxes. If “Level 1” is set and “Level 2” is empty, only one level of subclassing is configured. If both fields are defined (as they are in Figure 22) then the effect is to define a two tier feature class under the parent. The figure shows how the Feature Book structure shown in the previous example is achieved.

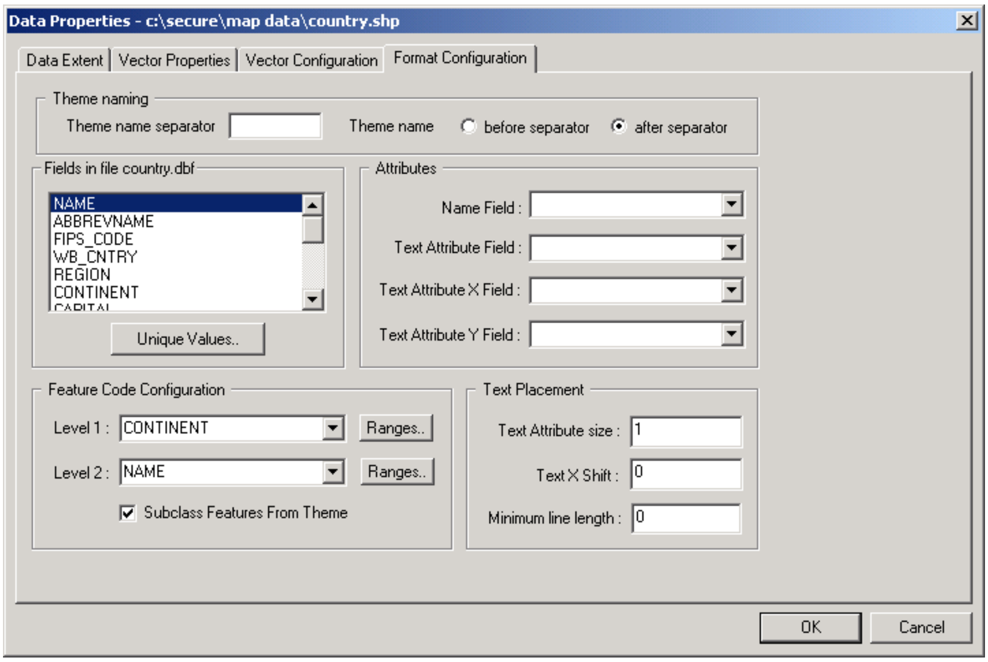



**Figure 22 Typical Vector Format Configuration Panel**

Most filters provide the option “Subclass from theme”. When not selected, “Level 1” will appear at the root of the Feature Book (and so not underneath country but completely independently at the end of the list). This allows greater control for reordering rendering within the Feature Book.

However, subclassing from the original theme has some benefit. In the example shown if you purely wish to highlight the UK in red while leaving other countries green, you can take advantage of inheritance of colour and style. If you have defined rendering at the parent level, when you subclass all sub-features rendering will be set to ‘Parent’. This means their representation will not change. You can then modify the rendering of the feature you specifically want to colour without having to define them all.

In order to see the visible effects of subclassing on the drawn map it is necessary to reload the layer after subclassing.

## 4.10. Range Based Subclassing Features

Subclassing can significantly increase the size of the Feature Book especially if two levels are selected. This can be a big problem with numeric values which can have a vast number of variations. For example, it is not uncommon to want to colour all contours over a certain height differently from others or highlight towns with a population greater than a certain level. To assist in doing this there is in most vector filters an additional button next to the Subclassing boxes with the label 'Ranges'. This invokes the Class Range Breaks dialog shown below.

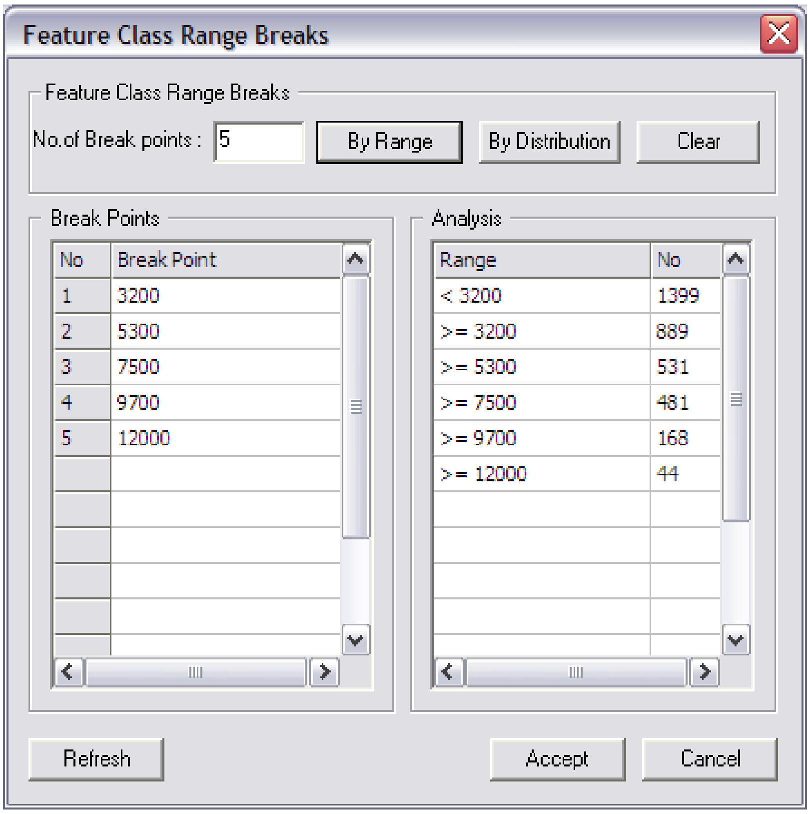



**Figure 23 Class Range Breaks Dialog**

This dialog allows you to analyse an attribute for appropriate ranges or manually enter them. Taking the population example, if a class range break was defined on 100000 for the attribute for population on towns, then under towns only two Level 1 Subclass feature classes would be present:

```
Population < 100000
Population >= 100000
```

This means that the Feature Book list is significantly smaller. Ranges can also be defined on Alphabetic values. If you manually edit the break points, then the Refresh feature allows you to see the populations of data present in the ranges you have defined. And once again you need to Reload Layer in order to see the effect of the changes made here.

The panel deals with text as a byte stream. Because text is UTF-8 we can make use of the UTF-8 encoding mechanism which means that if you sort by looking at the numeric value of each 8-bit encoded byte, you will get the **same** result as if you first decoded the string into Unicode and compared the numeric values of each code point.

## 4.11. Attribute Data

In addition to using attribute data for specific map visualisation functions it is also possible to embed the attribute data (field name/type and value) for each graphics primitive within the map. The fields present are assumed to be the same for a given feature class (e.g. roads) but the value will vary per primitive.

Why export attribute data in a map? It is sometimes desirable to retain this data for use in a MapLink application. This can be used to provide information to the user when he clicks on an object in the map, or it can provide extended information used to retrieve further data from an outside source such as a database. The display of the data is obviously dependent on the application. An example display format for DAFIF data is shown below.

MapLink Studio can be configured to include this attribute data within a generated map. At present attribute data can be obtained from most of the vector data types: S57, VPF, Shapefile/MIF, OS MasterMap, DAFIF and GDF to name the key ones.

MapLink will allow any vector filter to provide attribute facilities. The dialog within MapLink Studio which allows you to control what attributes are included is available under the Dataset menu or from the Right Mouse Button Menu options. The option is Attribute Collection.

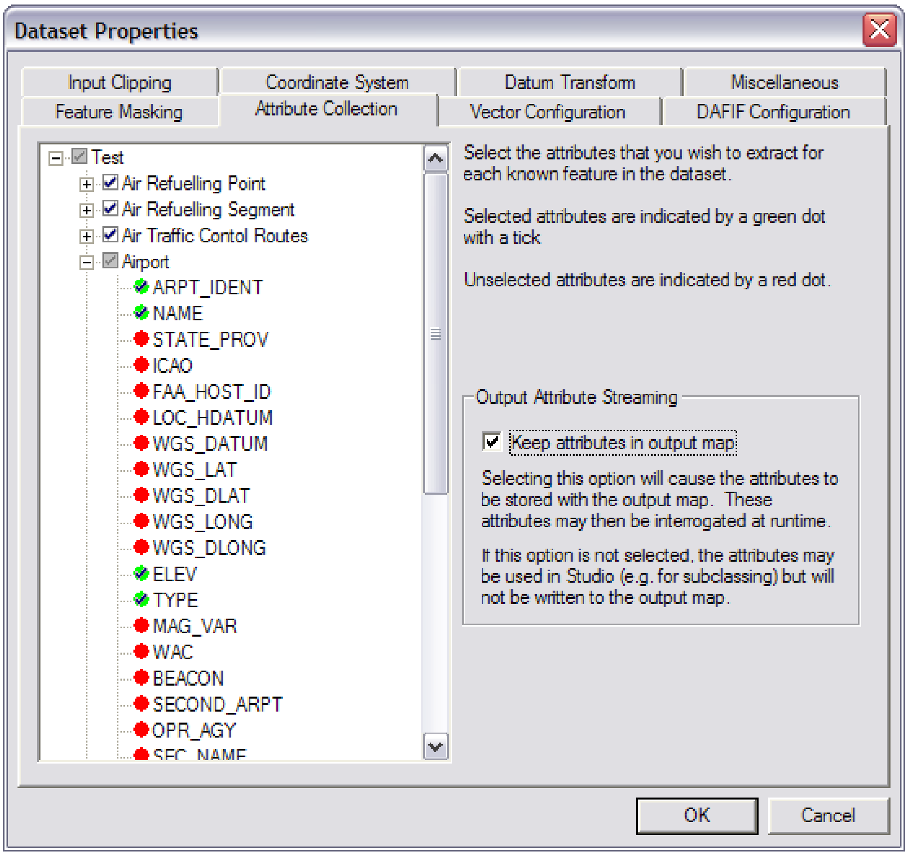



**Figure 24 Attribute Collection Dialog**

As can be seen there is an entry in the tree for each feature (including sub-classed features as described above). The green dots represent attributes which will be included and the red dots attributes which will be excluded. The status can be changed by clicking on the red or green dot.

If you wish the attributes to be included in the output map you need to tick the box on the right. For certain MapLink functions, e.g. line labelling, it is necessary to include attributes but you may not want to include these in the map.

It is advisable to not just include all attribute data without thought. Including attribute data within a map increases the size of the map on disk and in memory, and therefore potentially degrades performance. For this reason we recommend only including the required attribute data, and assessing the effect of including attribute data when defining the initial map structure. Sometimes it is best to include a few key attributes and then to use the runtime application to locate the rest based on the key in a separate database.

Another approach is to include only a few attributes on overview maps and provide more detailed attributes when the user is highly zoomed in. How to do this is described later.

## 4.12. Feature Masking

Feature Masking is the removal of unwanted features from a map. For example, the VMAP0 dataset contains many features. For a simple overview map most of these will not be required and their inclusion would seriously degrade the display performance of the map. In addition, in many cases the application requires that irrelevant detail is not displayed because it purely causes clutter on the display and degrades the function of the application.

Feature Masking allows you to exclude these unwanted features from the map. For a purely naval application all inland features could be excluded. It is also possible that we have two sources of data (in different datasets) with overlapping content. Both datasets may contain coastline and roads, but one may include more accurate roads and the other more accurate coastline. By using Feature Masking we can choose the best of both datasets for the output map.

Both feature classes and attribute derived sub-classes can be masked separately and so for example towns smaller than a population of 10000 could be masked out (in such a situation we would need a range class break for this value on population).

The Feature Masking Dialog is available on each Dataset, from the Dataset menu or right mouse button menu. The Feature Masking dialog is shown below.

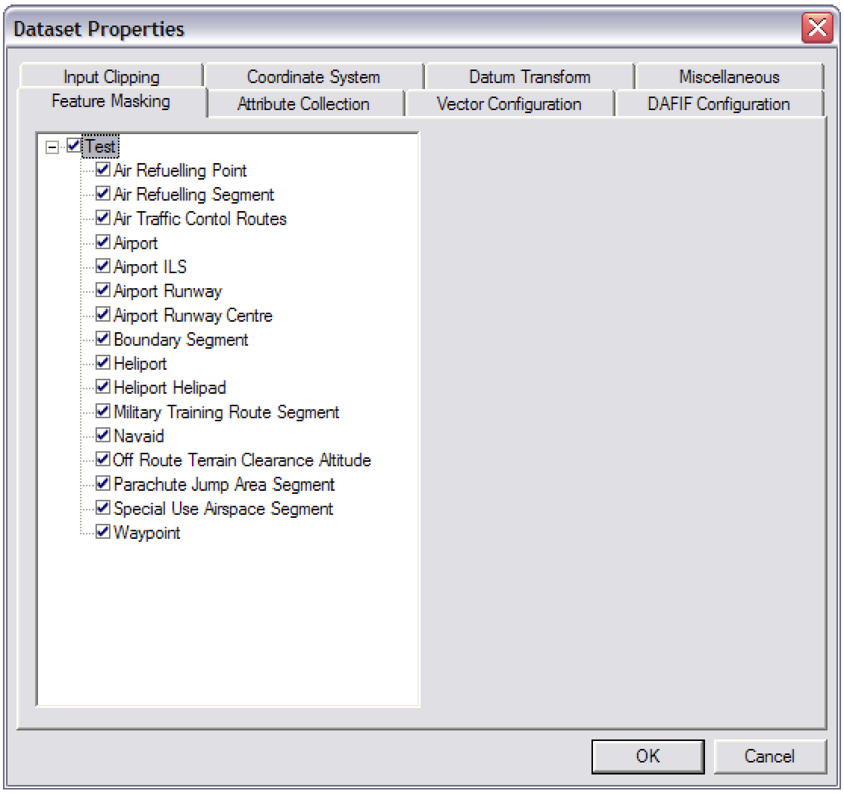



**Figure 25 Feature Masking Dialog**

For optimum performance it is highly desirable that Feature Masking is used instead of purely turning a feature off in the Feature Book (setting all its styles to none). The latter will still include the primitives in the output map and so degrade performance. Feature masking will actually ensure they are removed.

It is necessary to reload the layer in MapLink Studio before the effects of a change in the Feature Masking take effect.

## 4.13. Unicode/Code Page Configuration

A Data configuration panel exists at the data level to inform the user of the data's Code Page and to allow the user to over-ride the Code Page if the one reported by the data is incorrect.

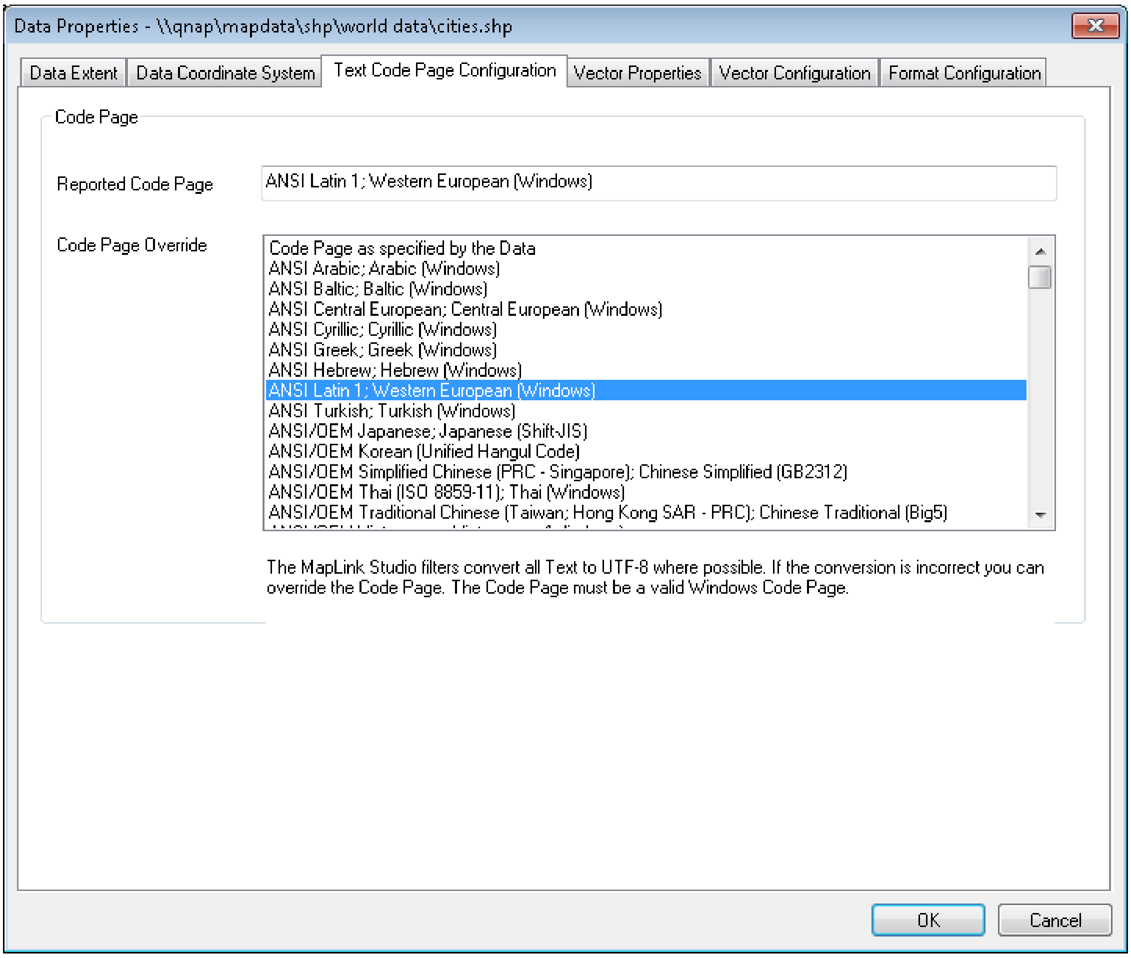



**Figure 26 Code Page Configuration Panel**

The list of code pages is defined in the file:

- `MapLinkStudioCodePages.csv`

## 4.14. Output Map Format

When you create a map and select a map format older than 8.0, MapLink Studio needs to perform a number of checks to make sure that creating the old file formats will work and to report what the problems may be.

With versions of MapLink Studio prior to 8.0, text was output in the Code Page it was read in. If text was read in multiple Code Pages then text was written out in multiple Code Pages. There was no way to tell which Code Page the text was in. Maps generated in this fashion would not normally display correctly since all the text would be displayed as if it was in the system's Code Page.

When creating maps for older versions of MapLink, using source data in multiple Code Pages is not supported due to the operating system locale setting limitations. Therefore if an old file format is selected for output, MapLink Studio will check to make sure that all data has the same Code Page.

If the Code Pages are not the same then the following warning will be displayed:

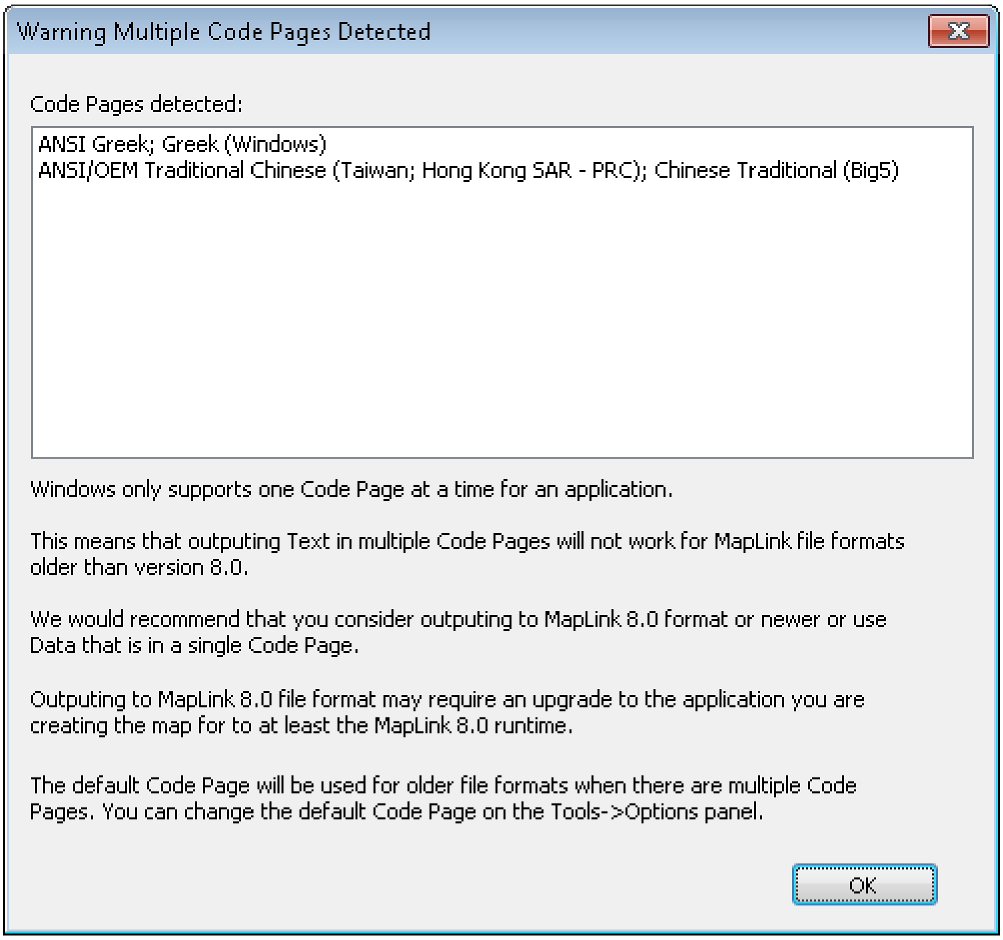



**Figure 27 Multiple Code Pages Detected Warning**

If the Code Page is not the same the override Code Page will be used (Tools->Options panel).

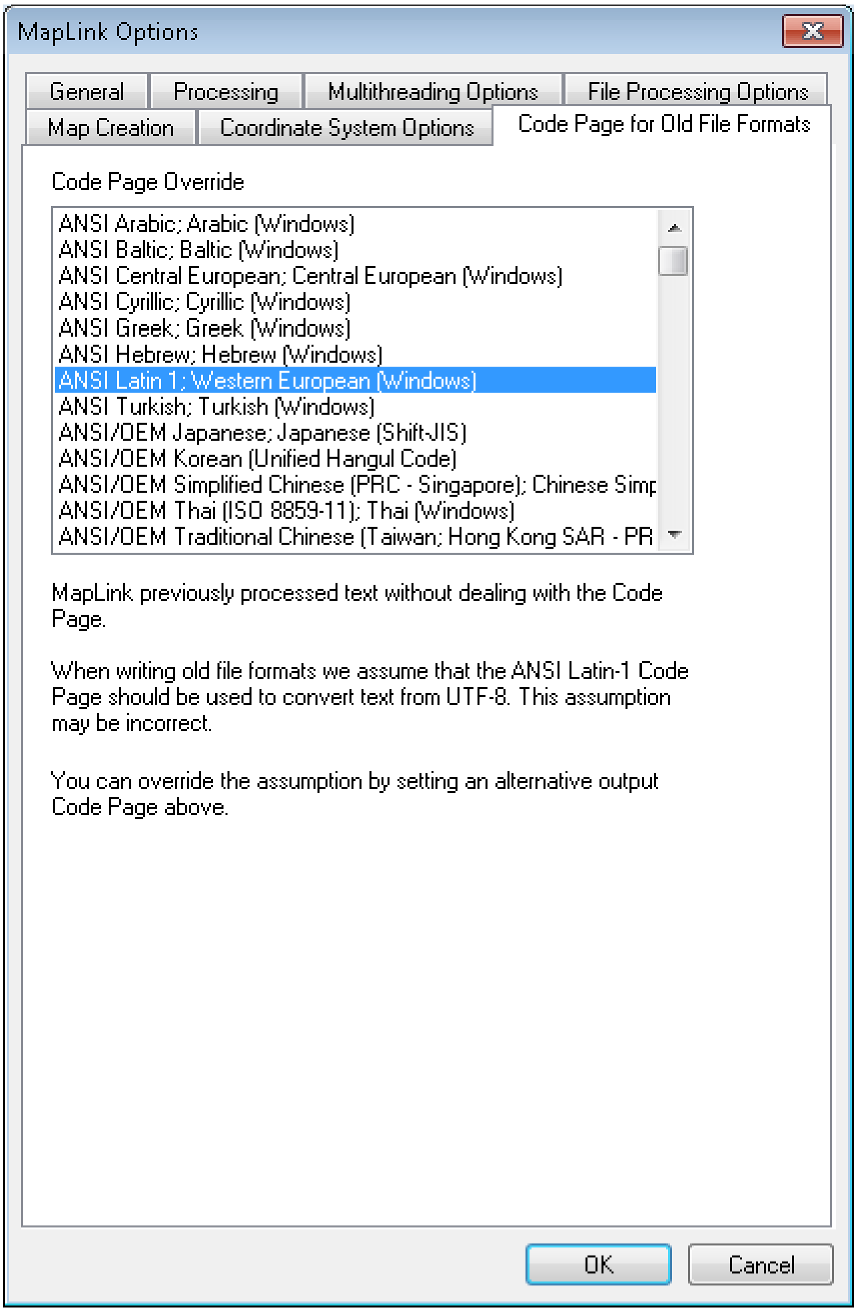



**Figure 28 Old File Formats Code Page Selection**

The OS Mastermap/Vectormap Local, Shapefile and MIF panels have a 'UTF-8 to Code Page' tick box (used to be Latin-1). This is ignored for the new file formats. However when outputting an old file format this will trigger the setting of the override code page setting.

If there is only a **single** Code Page in the data then that is the Code Page that will be used to output the old file format, not the override Code Page.

## 4.15. Exercise 2 Map Colours and Styles

When you have completed this exercise you will be able to

- Use the Feature Book to set up line, fill, point and text rendering styles.
- Use 3 methods to order primitives in the output map.
- Set up format configuration options specific to Shapefile data.
- Mask out unwanted features.

## 4.16. The Feature Book

The Feature Book is located under the Tools menu. It can also be accessed using the right mouse button clicking on any of the items in the project tree, or by pressing the  button in the left toolbar.

> Select the **50m-admin-0-countries** data file and open the Feature Book

The panel on the left side shows the features contained within the World Feature Book Section – in this case, the Shapefile data files. The panel on the right shows the available controls for setting the colour and style.

> Use the Feature Book to select a polygon style and colour, and an edge style and colour.

Remember that the data analysis showed that `50m-admin-0-countries` only contains polygon primitives. Therefore changing the Line, Text, or Symbol styles will not affect how the country data is displayed. The selected colour and style is immediately displayed on the map. You will see that only the `50m-admin-0-countries` polygon information is updated. This type of colour and style definition is known as "rendering".

The top left corner of the colour shows the most recently used colours. This is particularly useful when you are trying to create two identically coloured features. In this case you may decide to use the same colour for both the polygon fill and the polygon edges.

> Once you are happy with your polygon colour styles try setting line styles for **50m-admin-1-states-provinces-lines-shp**. Use Data Analysis to ensure that you are rendering the correct primitives for each feature.

As MapLink uses standard Windows functionality, you can use Copy and Paste to copy existing styles and colours from one feature to another. Notice that there is also an un-do and a re-do button to enable to you change your mind on the last rendering option you specified.

> Next choose a symbol style for **50m-populated-places**.

To see the symbols you have chosen expand the symbol size option, and set Size Units to be Pixels. Now set the Scale Factor to be 10. You will now be able to see your symbols on the map display.

You will notice that we haven't rendered any text. Shapefile data does not contain text features. We will use the vector configuration options in a moment to set up some text for rendering.

> Once you are happy with how the map looks close the Feature Book.

## 4.17. Ordering Data

We will now look at changing the data order. You may have noticed that on occasions the contents of one data file obscures the contents of another data file. The project tree defines the order in which data is drawn. Map0 represents the top of the tree. Within a Dataset, the data file closest to the Dataset itself will be drawn first. The next data file within that Dataset will be drawn on top of the first file. And so on. Problems occur when the second data file contains a feature that obscures the feature in the first Data file.

For example, if you have `50m-admin-1-states-provinces-lines-shp` at the top of your dataset, followed by `50m-admin-0-countries`, once you have rendered the features you will find that the country polygons hides the states-provinces-lines lines, and therefore your map will not show any lines.

MapLink provides you with full control over how the project tree is ordered, and therefore you are able to ensure that the map shows exactly what you want it to show.

```
Use the mouse to change the order of the Data files in the
project tree. In particular, notice what happens when you
drag states-provinces-lines above countries.

Add a new Shapefile Dataset, using the World Feature Book
section to your layer.

Drag the 50m-urban-area and 50m-populated-places data files
into the new Dataset.
```

You will notice that a * appears just below the Layer1. This indicates that the map display is out of date, and that you therefore need to reload the layer to view the latest information.

```
Reload the layer.
```

Even though the `50m-urban-area` and `50m-populated-places` are in a new Dataset, they have still retained the styles and colours that you previously selected. This is because both Datasets are using the World Feature Book Section.

```
Now drag the new Dataset above the old Dataset and notice the
result. Use this method to organise your data.
```

```
Save your project and create an output map.
```

You can see your rendered features in the MapLink Viewer.

## 4.18. Dataset and Data Configuration – Shapefile configuration

Shapefile data contains 3 files:

- .shp This is the Shapefile itself. It contains the primitives that define the data.
- .dbf This is a database file, containing extra information associated with the Shapefile.
- .shx This is an index file which indexes the Shapefile file.

We will now use the `50m-admin-0-countries.dbf` file to setup some text that can be displayed on the map.

```
Open the Format Configuration panel for the 50m-admin-0-
countries Data file.
```

This is the Shapefile Format Configuration panel. Each vector data format supported by MapLink has its own Format Configuration panel. We will cover these in more detail later in this guide.

The left side of the panel shows the features contained within the `50m-admin-0-countries` database. You will see a feature called COUNTRY. If you select this, and click on the Unique Values… button you will be able to see the contents of the COUNTRY field of the database.

```
Try looking at the unique values of the COUNTRY field in the
50m-admin-0-countries database.
```

We can use this information to add more information to the `50m-admin-0-countries data` file that we currently have in our project tree. The process that you are about to carry out is known as Subclassing.

```
Define some subclassing for the country datafile. Under
Feature Code Configuration select

Level 1 : COUNTRY

Enable Subclass Feature from Theme
```

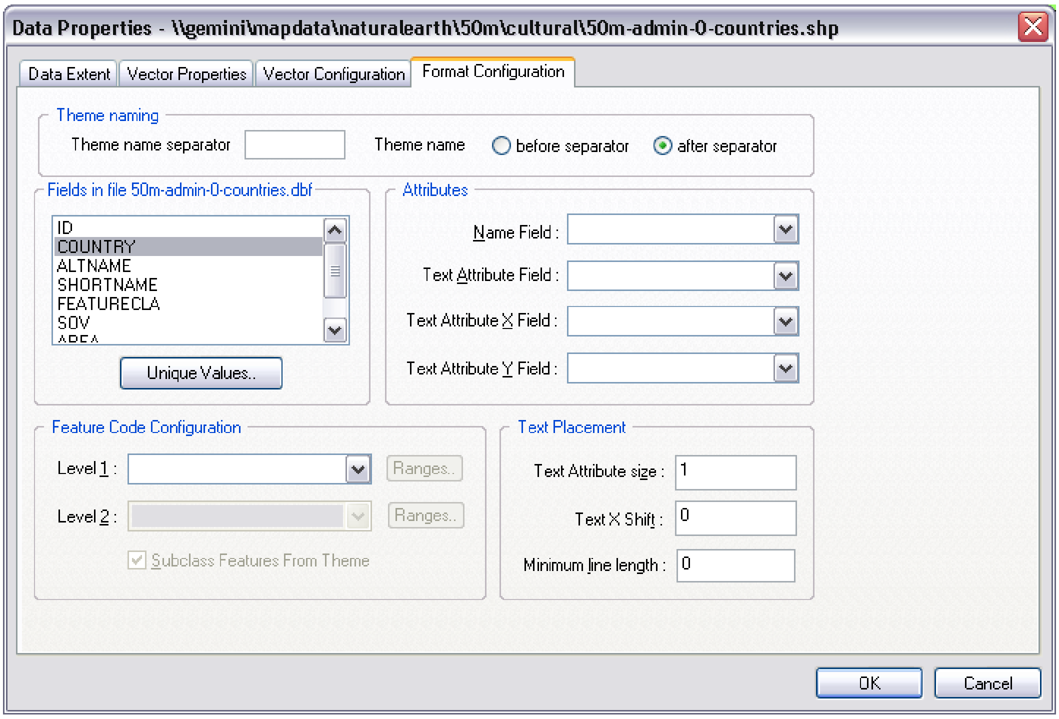



**Figure 29 Data Properties Panel**

`Press OK`

What does this do? Until now, each of the polygons in the data had a feature class of `50m-admin-0-countries`. We have now subclassed the feature codes. Each polygon will have a feature code of `50m-admin-0-countries.<level1value>`. These values are read from the `50m-admin-0-countries.dbx` database.

```
Reload the top layer

The display will not change. However if you now open the
Feature Book you will see that the 50m-admin-0-countries
feature has been extended to include country names.

Use the mouse to 'hover' over features in the map.
```

The display at the bottom right of the map view will show the name of each country you move the mouse over. Each country can now be uniquely identified.

This allows you to do the following:

- Render countries in different colours.
- Choose whether to display a continent or country now and at runtime in your application.

We'll look at both of these in more detail.

In the Feature Book you can see that at the moment the individual countries are rendered in exactly the same way – in fact, they all use the same colour and style as the original `50m-admin-0-countries` feature. In other words, rendering works on a hierarchical basis where each country inherits the `50m-admin-0-countries` rendering style. The top feature, in this case `50m-admin-0-countries`, is referred to as the Parent, and therefore the rendering of each of the subclassed features is currently set to 'As Parent.' However, we do have the ability to overwrite this rendering information using the Feature Book.

```
Try using the Feature Book to change the rendering of
individual continents and countries.
```

There is something missing from the map. At the moment we do not have any text displayed. Shapefiles do not contain any text geometry, however it is possible to use the database to add some labels to the existing data.

```
Close the Feature Book, and save your project.
```

## 4.19. Configuring Text

Reopen the Format Configuration panel for the **50m-admin-0-countries** data file.

In the Attributes section, set the Text Attribute Field to be COUNTRY. Press OK. Notice the * next to Layer1, and so reload the layer.

Each country now has a label associated with it. To display these labels we need to set the rendering using the Feature Book.

```
Change the text rendering for some of the labels. You may
need to use a Size Units of Pixels and a Size Factor of 10 to
see the text. You can change this to make sure that the text
is displayed at the correct size for your map, and to ensure
no overlapping of text.

When you are happy with the appearance of your map, save the
project, create an output map, and open the finished map in
the MapLink Viewer.

Use the Map Viewer to Declutter the map. If not already
displayed, you will find this option under the View menu
```

Declutter basically means to turn on and off features to change the appearance of the map. This is very useful as it enables users to look at the information that is only relevant to them at that time. The declutter functionality uses the configuration information that you supplied in MapLink Studio to allow you to turn off individual countries and their labels. The MapLink SDK will be able to access the feature codes that you have defined, at runtime and thereby provide full user control.

You have full control over the appearance of your map!

# 5. CONTROLLING MAP CONTENT

MapLink provides a number of methods to control the map content. These are Layering, Tiling, Masking, and Clipping. Why is it useful to control map content? To improve performance, to reduce disk space, to limit processing power needed, and probably most importantly to present clear and relevant data to the user for the current zoom level.

## 5.1. Layering

The MapLink map structure of Map/Layer/Dataset/Data has already been seen in the previous chapter. Unlike this simple example MapLink maps typically contain a number of layers. The concept of layer described here really describes a range of zoom and the detail relevant to that zoom range (what map features are displayed and how they are displayed).

Note: In the introduction the concept of types of layer in the application was introduced (Map Layer, Standard Data Layer etc). The layer within a map is a completely separate concept and for clarity we will call it a 'detail layer'.

Detail layering used in conjunction with a further feature (tiling), allows MapLink to display the relevant part of maps which in total are far too large to fit into memory almost instantaneously. A complex map can consist of many detail layers. Maps exist with many hundreds of layers. Several layers may cover the entire geographic area, but it is also possible for layers to describe specific areas.

For the simplest case a map may have several layers, each aimed at a specific zoom range and using appropriate data. Each successive layer in a project typically increases in content but is then typically segmented into smaller loadable units as its content (and therefore size) increases.

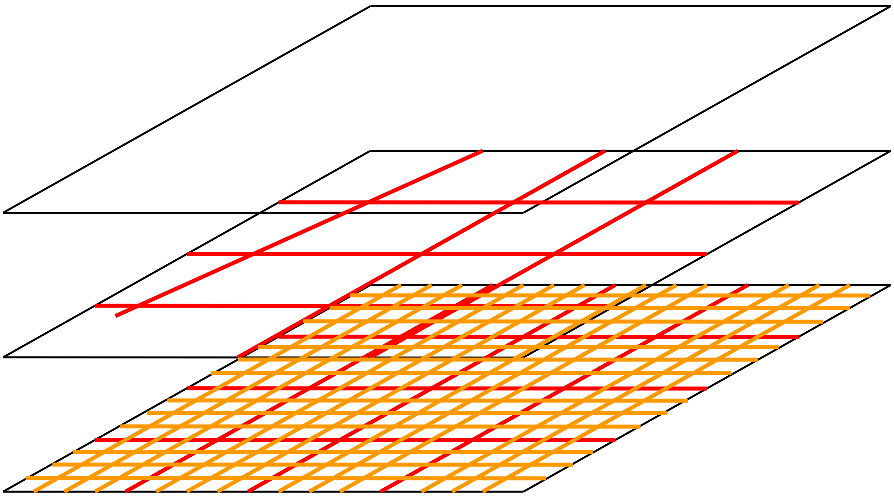


**Figure 30 Typical Map Layer Structure**

The more complex situation is shown in Figure 31 below. In this case there are two overview layers but also a series of discrete high resolution areas. Each of these has a pair of layers (low and high resolution). The sub areas typically cover urban detail, and are disjoint.

An example of a complex case is a map of Germany which contained approximately 600 layers (2 overview layers covering the country and the other layers being typically a pair of layers per town).

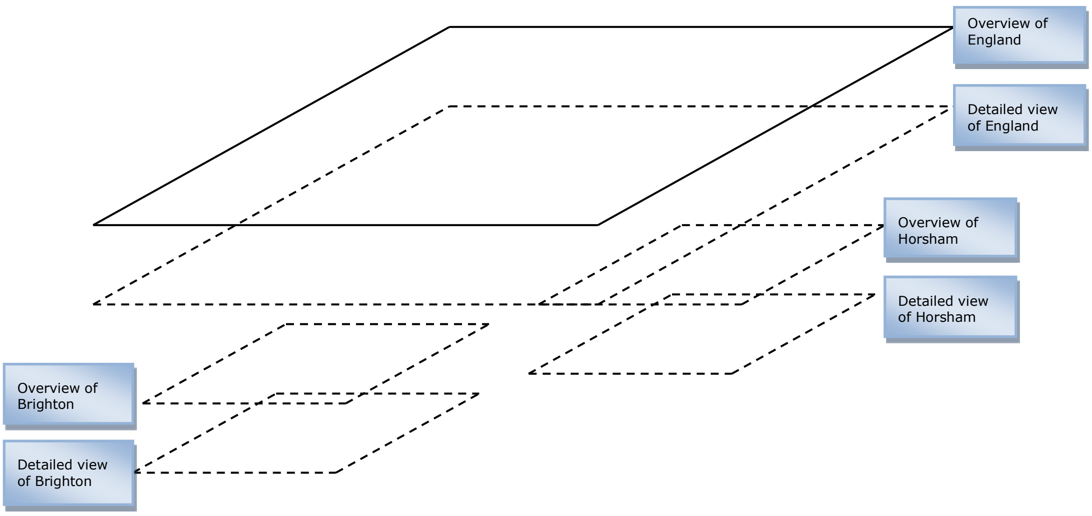



**Figure 31 Layering Example**

One map display window displaying a MapLink map will show only one layer at a time. MapLink will load the appropriate layer (using rules described later) and manage which parts of a layer are loaded to achieve the required display. Once defined the process is completely automatic based on zoom level. The application can also force the situation and decide when Layers are displayed at runtime.

## 5.2. Choosing Content for a Detail Layer

As an example of layering let us assume we require a map of France, which goes from country level down to town level. We want to use only VMap0 and coverage will be uniform (no pockets of specifically high detail).

We create three layers in MapLink Studio and add the data to each. We now mask out features from the top layer because we simply do not require them in an overview (Detailed contours, buildings, etc). We add 3 successive layers each with more detail until the lowest layer contains all of the features in the data. The map loaded is based on zoom factor so that:

- When the display is zoomed out, the top most layer will be visible. This includes wide coverage but little data.
- When we zoom in sufficiently, we will swap into the next layer. More data in the layer, but viewing a geographic subset of it.
- When we zoom in again, we will swap into the most detailed layer. More data still in the layer, but viewing an even smaller subset of it.

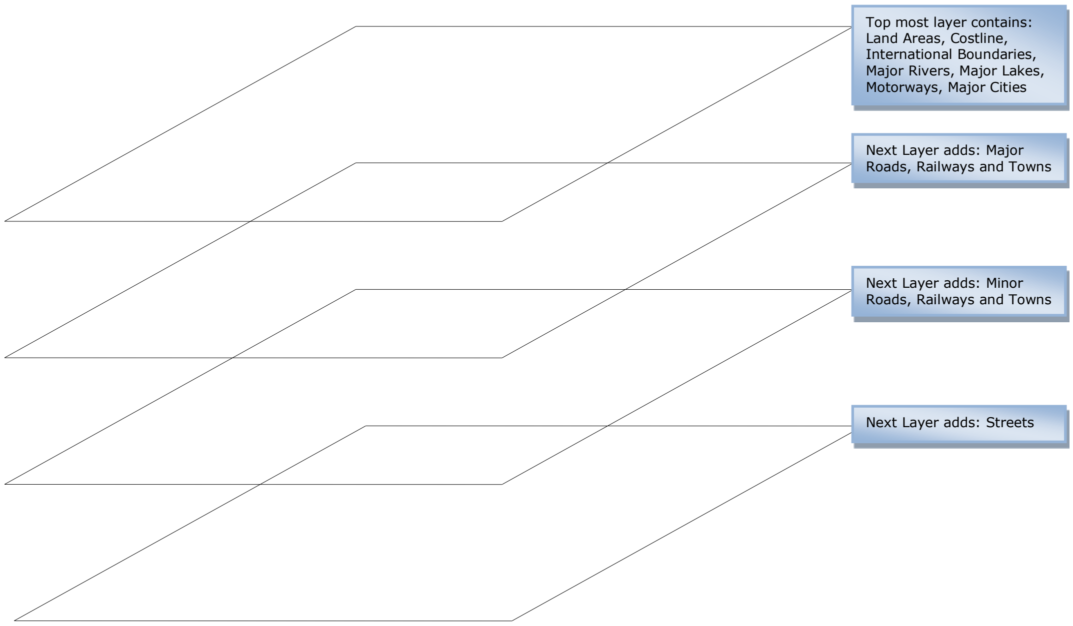



**Figure 32 Typical MapLink Layer Content**

## 5.3. Tiling

MapLink uses tiling in conjunction with layers to reduce the amount of information to be loaded into memory and drawn to screen. This makes it fast even with very large datasets.

The overview layer (say world map) has wide coverage but little detail. But when we zoom into the bottom, most detailed layer the detail is much higher. To avoid reading the whole layer the map is split into tiles. Only the tiles needed to update the area on display are loaded. This reduces memory and makes redraw quicker.

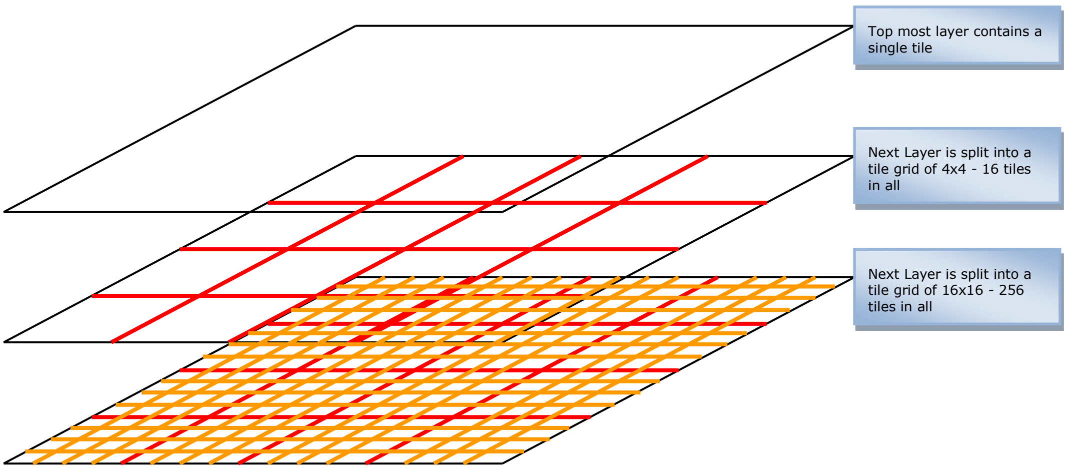



**Figure 33 Tiled Map.**

## 5.4. Vector Data Tiling

Vector data tiling is specified by defining the number of tiles in X and Y for each layer. When using Resolution based loading (section 6.3) MapLink will automatically choose sensible tiling settings based on when the layer will be visible. Guidelines are also available in the MapLink Studio Help to allow you to set up manual tiling settings if desired. It is a property of the layer and can be accessed from the Right Mouse Button menu on each layer. These properties apply only to Vector and Terrain Data (more later).

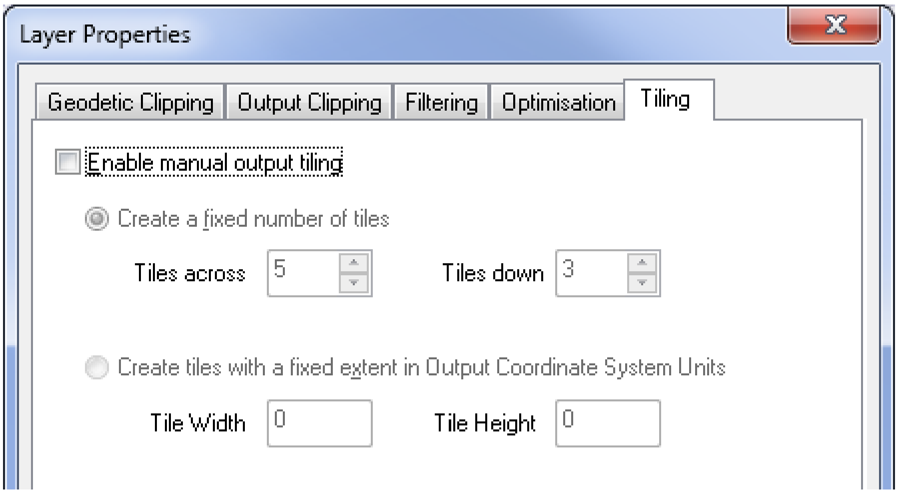



**Figure 34 Layer Properties**

## 5.5. Raster Tiling

Both vector and raster data is tiled but in subtly different ways. Raster data is still allocated to layers, but with raster data tiling is automatic. This is because the process is simply to create a series of reduced resolution images. This uses a process known as pyramiding.

## 5.6. Layer Loading Strategies

MapLink determines how to load a layer in one of two ways depending on the setting of an option in the map menu called 'Loading Strategy'. The two current strategies are tile based loading and resolution based loading.

## 5.7. Tile based loading

When using tile based loading MapLink will swap intuitively between Layers, without you having to define any swapping information (Tiled based loading). MapLink examines layers from the bottom of the tree defined in MapLink Studio. It will load from the Layer that would require the greatest number of tiles less than 10 to be loaded to show the required area. In practice this means that if it cannot display a layer using a 3x3 grid of tiles it will try for the next layer up. For example, given the zoom level, if:

- Layer1 requires 1 tile
- Layer2 requires 6 tiles
- Layer3 requires 10 tiles

MapLink will display Layer2. Therefore never more than 9 tiles are ever loaded.

Loading is fast, as long as the tile sizes are kept small (300-500k is ideal).

## 5.8. Resolution based loading

Resolution based loading allows the user to define the minimum zoom level at which the Layer should be displayed. Resolution based loading allows full user control of layer swapping. It is independent of tiling and straightforward to define (Map Properties):

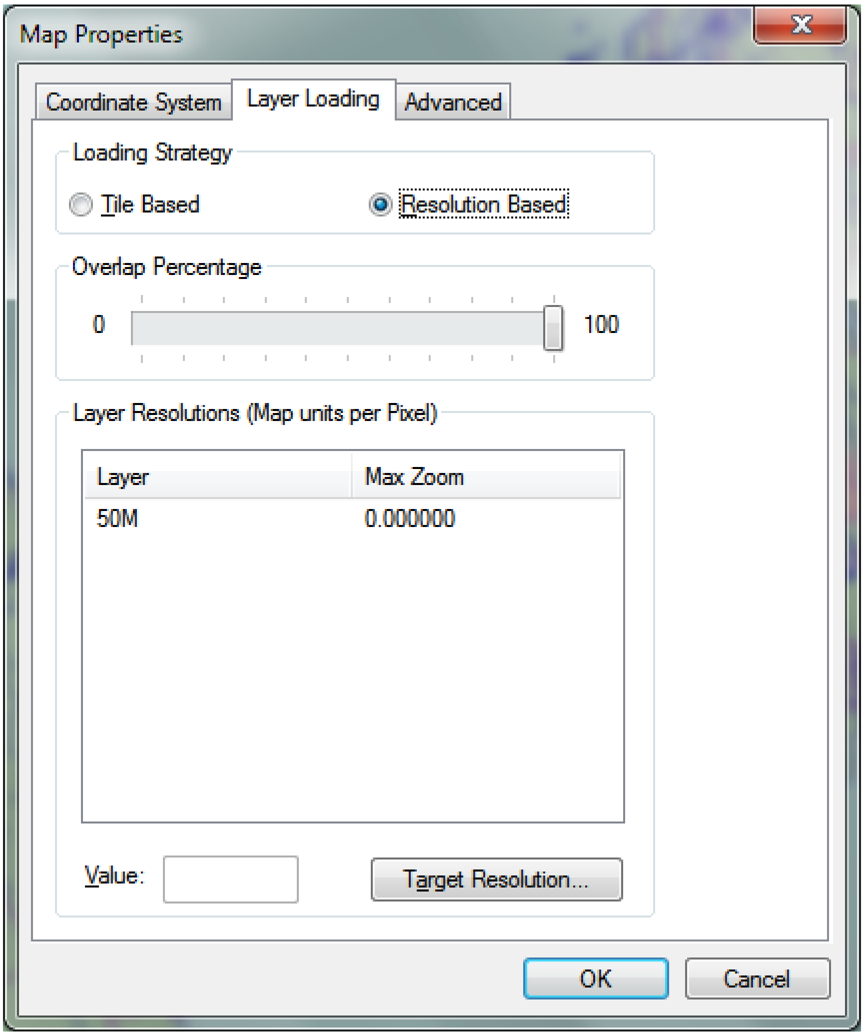



**Figure 35 Map Properties Window**

Resolution based loading while not directly related to tiling, does require that the tiling is taken into account. The same sort of relationship is required. This really relates to the memory requirement. So it is still best to keep the relationship of between 9 (3x3) and 16 (4x4) tiles in memory as a reasonable guide.

The point at which a layer loads for resolution based strategy is defined by the zoom level. This can be entered, as seen above, or can be defined graphically with the option 'Define Layer Zoom Range' found on each layer menu once resolution based loading has been enabled.

So to compare resolution loading to tile based loading, it is slightly more complex to set up (with tile based loading you simply set up the tiling and the rest is automatic) but resolution based loading gives more consistent results which makes it the best choice. A good approach is to work initially with tile based loading and then enable resolution based loading to fine tune.

# 6. EXERCISE 3 LAYERING AND TILING

When you have completed this exercise you will be able to

- Define a number of layers.
- Define tiles for each layer.
- Select appropriate detail for the layers.
- Set a zoom range for each layer.

## 6.1. Layering and Tiling

Open your Shapefile World project. This currently has 1 layer containing 2 Shapefile datasets.

Update your project to create a 2 layered map, each layer containing 1 Shapefile dataset both using the World feature book section. The structure should be as follows:

Layer1
  SHP
    **cultural/50m-admin-0-countries.shp**
    **physical/50m-graticules-all Folder/50m-graticules-10.shp**

Layer2
  SHP
    **cultural/50m-admin-0-countries.shp**
    **cultural/50m-admin-0-boundary-lines-land.shp**
    **cultural/50m-populated-places.shp**

Try using copy and paste to create the second layer.

Open the Format Configuration panel for **50m-admin-0-countries** in the 1st layer and clear the Text Attribute Field in order to disable the country name text fields for this layer.

You have now created two levels of detail for your map. The first layer shows just the countries and a latitude/longitude grid. The second layer shows additional Shapefile data. Selecting the map will show an outline of the extent of both of the layers. These extents are currently the same.

You will notice that at this stage as you click on each layer, the data is automatically reloaded to update the map display. You may wish to disable this option as it can become time consuming as you add more and more data to a project. Under the Tools menu, the Options panel allows you to disable Automatic Data Loading. You can always turn this on again if you want to automatically load data, but in general we recommend disabling this option.

Now to add some tiling. The top layer of the map is rarely tiled as most users will always want to see the complete layer on an initial view. In our case, this is certainly true. Therefore we will only setup tiling on the second layer. This will break down the amount of information that needs to be loaded to view the second layer – and therefore ensure that you map is displayed quickly within your runtime application.

Bring up the layer tiling options by selecting the 2nd layer and either selecting Tiling from the right click menu or clicking the  button.

Setup 4 x 4 tiling.

```
Open the Layer Loading page under the Map menu and select
Tile Based. The meaning of this option will be explained in
more detail in sections 6.2 and 6.3.


Reload the layer to view the tiling grid. If you can't see
the grid, turn off the 50m-graticules-10 file in the project
tree.

Save the project, and create an output map.

Now look at your map in the MapLink Viewer. The initial view
shows your top layer. As you zoom in, you will notice that
place names and populated areas appear. This is your 2nd
layer.
```

You now have a map that contains two levels of detail!

## 6.2. Tile Based Loading Strategy

At this stage MapLink has full control over when the 2nd level of detail is displayed. The Tile Based loading strategy is being used. This means that the 2nd layer will only be viewed when 9 tiles or less from layer 2 can be loaded into your view area. You can change this by altering the number of tiles in your layer – or better still you can use Resolution Based Loading.

To demonstrate these two loading strategies more effectively, we require a more complex map. Therefore we're going to add two more layers to the World project, adding some more detail to the map.

```
Add a third layer to you MapLink project. This time add a SHP
dataset and import an existing Feature Book section. Use the
Natural Earth Feature book that is included in the sample
project shipped with MapLink.

When prompted to select a section, use the Natural Earth 50M
Feature Book Section.
```

The Natural Earth Feature Book Section has been previously created and is provided with MapLink along with the data. Importing an existing Feature Book Section allows users to re-use existing Feature Book Sections – and to enable a standard map style to be defined.

```
Add the following Shapefile files to Layer 3:
   cultural/50m-admin-0-countries.shp
   cultural/50m-urban-area.shp
   cultural/50m-admin-0-boundary-lines-land.shp
   cultural/50m-populated-places.shp

Using the Format Configuration panel, set the Text Attribute
Field for 50m-populated-places to Name_Town so that text
labels appear for these features.
```

Once you have reloaded the layer notice that the rendering is already defined for this dataset using the Natural Earth 50M Feature Book Section. Therefore the features already appear with colours and styles.

```
Adjust the order of the files in the project tree as required
so that they match the order above.
```

Now to add the last layer to the project.

```
Copy and Paste the 3rd layer to create a 4th layer. Rename the
layer to Layer4.
Add the follwing Shapefile files to this layer:
   physical/50m-coastline.shp
```

**physical/50m-lakes.shp**
**physical/50m-rivers-lakes-centrelines-with-scales.shp**

Adjust the order of the items in the tree to make sure all the data is displayed.

Use the Format Configuration page to setup the following subclassing:

**50m-admin-0-boundary-lines-land**

Level 1 : FeatureCla
Enable 'Subclass Features From Theme'

Text Attribute Field : Name

**50m-rivers-lake-centrelines-with-scale-ranks**

Level 1 : FeatureCla
Enable 'Subclass Feature From Theme'
Text Attribute Field: Name1

Reload the layer. You will notice that the rendering is immediately updated from the information that is contained within the Natural Earth 50M Feature Book section.

Update the rendering if you wish.

Setup 32x32 tiles on layer 4.

Save the project, create an output map, and open it in the MapLink Viewer.

Note that as you zoom in you will swap through three levels of detail. You will not see the 3rd layer of your map. This is because MapLink is controlling the level at which the data is loaded using the 9 tile rule.

## 6.3. Resolution Based Loading Strategy

The way to force the map to show all four levels is to use Resolution Based loading.

Open the Layer Loading page under the Map menu.

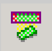
This is a property of the whole map, and therefore only appears under the Map menu.

Select the Resolution Based Loading Strategy.

You can now enter the resolution at which you want each layer to be displayed… however there is an easier way!

Press OK.

MapLink now knows that the Resolution Based loading strategy should be used.

Select and reload layer 2. Click in the map view, and then select the  button from the left toolbar. The Define Layer Zoom Range can also be found under the Layer menu.

In the map view, drag a rectangle the size of Europe. This will appear as a yellow rectangle. You will see that the resolution factor for layer2 has been updated.

What does this mean? MapLink now knows that the area viewed must be equal to or smaller than this area before this layer will be loaded. Therefore at runtime the map will only swap into this layer when the view area is smaller than the size of Italy.

```
Select and reload layer3. Define a zoom resolution area that
is approximately the size of Italy.

Select and reload layer4. Define a zoom resolution area that
is approximately the size of Corsica (off Italy).

Open the Layer Loading page to see how the resolution factors
have been updated according to the rectangle you have
dragged.

Save the project, create an output map and view it in the
MapLink Viewer. As you zoom and pan around the map you will
notice how the layer display has changed.
```

You have full control over when data is displayed!

# 7. Raster Data Handling

So far we have only used Vector Data. MapLink also deals equally well with raster data in many forms (raster maps, satellite imagery etc) and in complexity from small raster snippets through to 10s of GB of data.

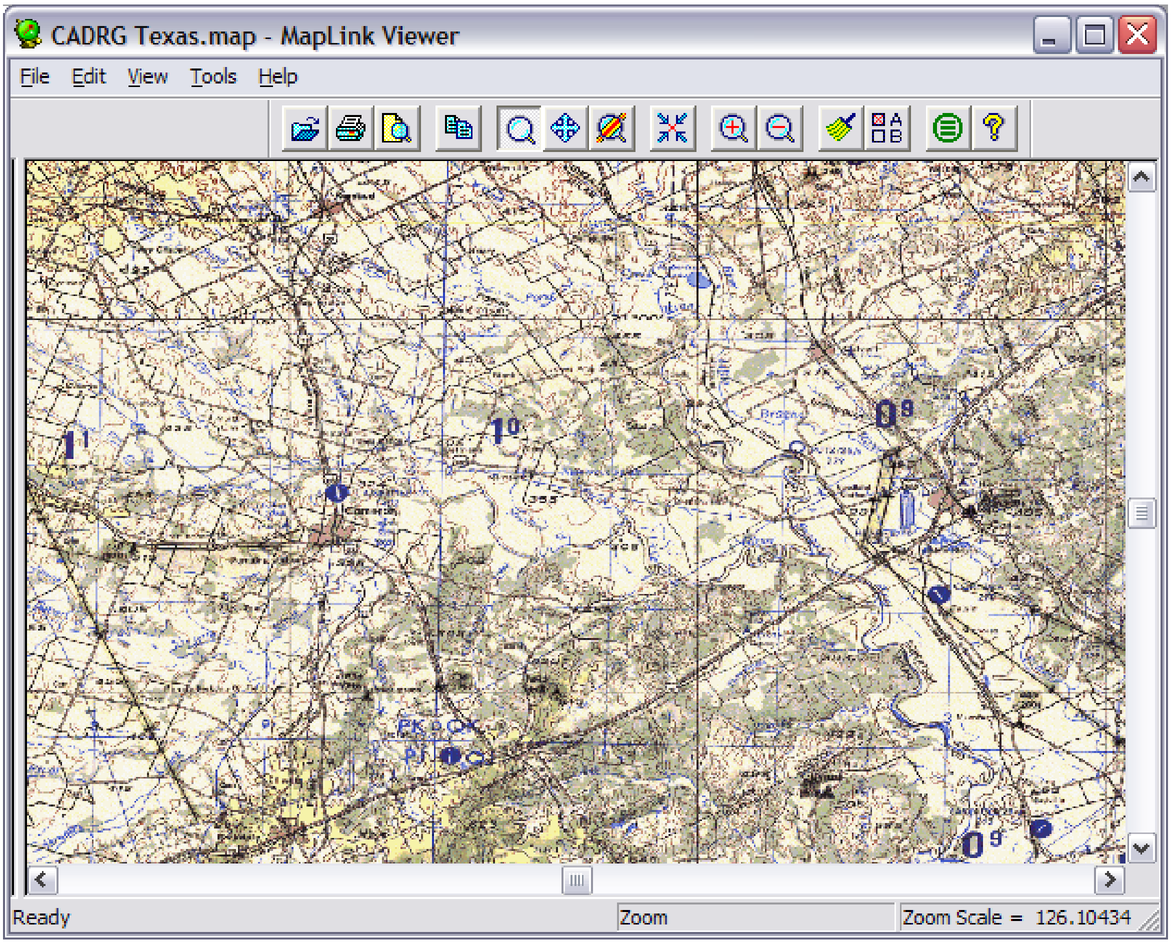



**Figure 36 Processed CADRG Data**

## 7.1. Using Raster Data

MapLink accepts many flat raster formats, e.g. TIFF, JPEG, BMP. It also accepts several structured raster formats e.g. ASRP, CADRG, CIB, GeoTIFF, LAN... These all contain embedded referencing information locating the raster data.

## 7.2. Raster Referencing with flat rasters

When loading a flat raster, referencing information is required. This tells MapLink where to locate the raster. MapLink reads the 3 standard referencing file types

- World files
- Tab files
- GeoTiff Tags

If no referencing information is supplied, MapLink will ask for it when the raster is added.

## 7.3. Common Raster configuration options

Raster Map Datasets and Data Files have a configuration panel which provides the following options:

- Raster downsampling; create reduced resolution raster datasets for fast loading
- Raster resizing; reduce memory usage by limiting pixels
- Raster projection; image interpolation
- Transparency
- Raster Colours
- Feature Name; use at runtime to switch rasters on and off individually.

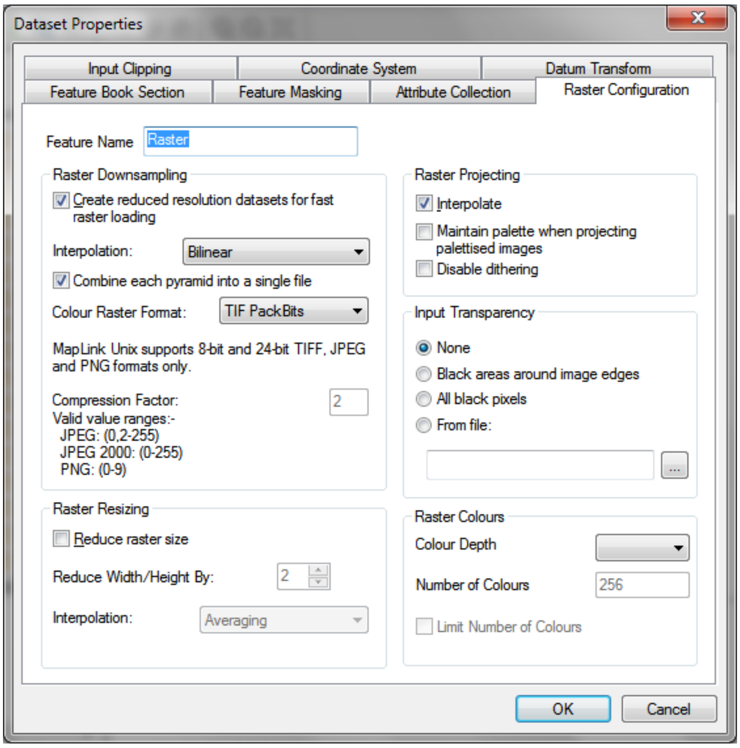



**Figure 37 Raster Configuration Panel**

## 7.4. Format Configuration Options

Raster data import filters, like vector import filters have specific configuration options depending on the data type. These specific options are accessed from the Format Configuration menu item on the Dataset or Data file. The options cover such items as:

- The map series to be displayed (where a format supports several). e.g. 1:1M, 1:500k.
- The brightness and contrast if available on a raster format.
- Which of a series of images present to use.
- The colours to be used for various heights in a terrain grid.

These options will be described for each raster format as it is covered later in this manual.

# 8. Exercise 4 Creating a Raster Map

When you have completed this exercise you will be able to:

- Add Raster data to a MapLink Studio project.
- Mix Vector and Raster data in one map.
- Define several levels of detail for Raster data.

## 8.1. Adding Raster Data to MapLink Studio

To look at how to add Raster data to MapLink Studio we'll use our existing project and add some additional sample data to it. As well as the Natural Earth sample Shapefile data there is are also two tif files – `california.tif` and `ocean_islands.tif` in the GeoTiff_SAMP directory. The California TIFF file has an associated world file (*.tfw). This file contains positional information describing exactly where the TIFF data should be positioned. MapLink can use this to accurately position the TIFF file in your map project and also at runtime in your application.

The existing project contains 4 levels of detail, all comprising of Shapefile data within Shapefile datasets. The Raster data we have covers a much smaller area than the existing layers 3 and 4. We will delete these layers, and instead use the Raster data.

```
Delete Layers 3 & 4 from the map.
```

To add Raster data you need to create a Raster dataset.

```
Add a new layer to the map, and add to this a Raster dataset.
Add the california.tif file to this dataset.
```

The Raster Dataset tab shows all the different types of rasters supported by MapLink Studio. In this case you need to use RASTER as your dataset type. RASTER is the generic dataset that supports common raster formats such as TIFF, .bmp, JPEG and so on. A list of the most common raster formats supported is shown below.

| | | | | |
|---|---|---|---|---|
| CALS | IBM MO | MS Paint | Photo Shop | Windows Bitmap |
| CIF | IFF | Multipage | PCX | PNG |
| Clip Board | JPEG | JPEG2000 | NCR | Postscript |
| DICOM | Kofax | PBM | PPM | Windows ICON |
| FlashPix | Laser View | PCT | Silicon Graphics | Windows metafile |
| Gem IMG | Mac Paint | PCX | Sun Raster | Windows DIB |
| Halo CUT | Mac Pict | PGM | Targa | X Pixmap/Bitmap |
| IBM IOCA | Meta File | Photo CD | Tiff | X Window Dump |

Notice that when you add a raster dataset you are not asked to choose a Feature Book section for the data. Unlike Vector data, the appearance of Raster data cannot be rendered. The data exists as a whole image and therefore you have no control over its appearance.

If the `california.tif` file did not have an associated world file you would be asked to enter the latitude/longitude corner points of the image. You will need to do this for any image that does not have any associated reference data – remember to note this information when you are generating images!

```
Open the Data Extent panel to see the extent of the file and
the size of the raster.
```

As you can see, primitives included in the file are not displayed. This information is only available for Vector data. We will look at the Raster configuration properties in more detail later.

```
Reload Layer3.
```

You will now see the raster data that you have added.

```
Now add another new layer to the map, and add a new GeoTIFF
dataset. Add the ocean_islands.tif datafile.

Reload the data to see the new image.
```

We now have a map that consists of 4 levels of detail.

As the project uses Resolution Based loading you need to define the Layer Zoom Range for each of the layers containing raster data.

```
Reload Layer3 and define a Layer Zoom Range rectangle
covering 3/4 of the raster area.
Reload Layer4 and define a Layer Zoom Range rectangle
covering 1/3 of the raster area.

Save the project, create an output map and use the MapLink
Viewer to view the new map.
```

You have now created a map containing both vector and raster data, of different levels of detail!

# 9. TERRAIN HANDLING

Terrain data is height information in a spatial form. As well as reading source map data and outputting fused runtime map sets, MapLink is also capable of processing and generating fused terrain sets. These are not directly displayed but provide terrain data to algorithms which will display or analyse terrain in some way (such as analyse for slope or display cross-sections). Terrain data can be provided in the form of a grid (very similar to a raster) or in vector form (contours).

## 9.1. Terrain Processing Features

MapLink supports the pre-processing of Terrain Data into visualisation combinations of vector and raster data. As with all visualisations in MapLink, retrieval at ideal level of detail from the map base helps achieve performance. Examples of gridded data formats supported are:

- DTED terrain
- DBDB-V bathymetry
- ASCII-DEM

An example of processing terrain data into a pre-defined visualisation is shown below. DTED data (Digital Terrain Elevation Data) is displayed with the terrain coloured by height and shaded by a defined sun angle to give an impression of the terrain. These features are specified via the format configuration panel for the DTED Raster Filter.

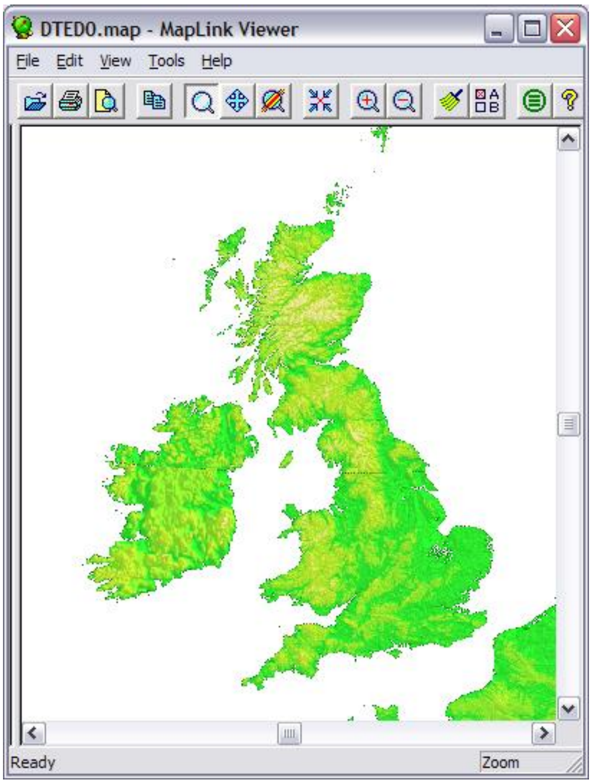



**Figure 38 Height Coloured and Light Shaded DTED0**

Another example output is DBDB-V Data, this time coloured based on depth.

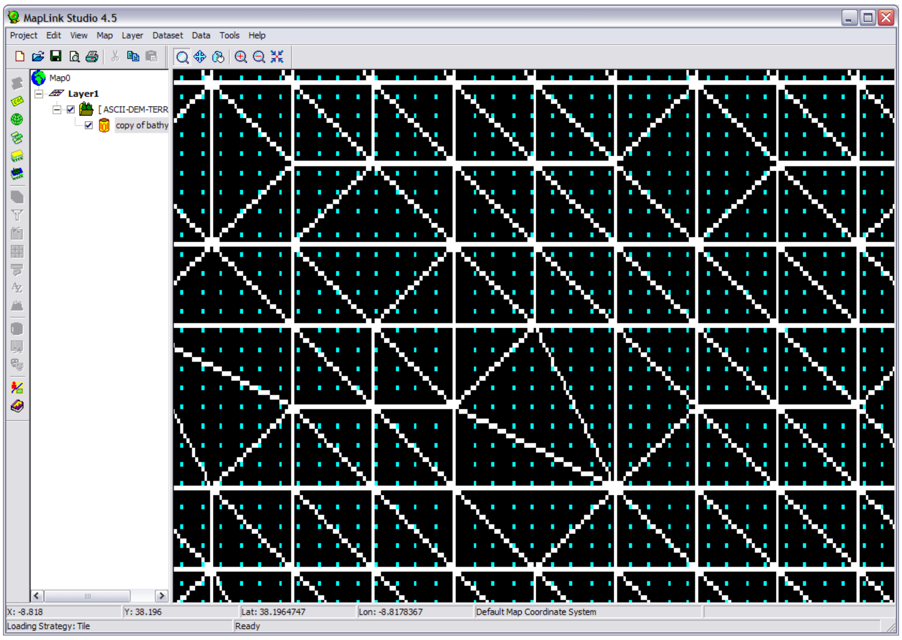


**Figure 39 DBDB-V Data**

## 9.2. Creating Terrain Databases

The same principle processes available for vector and raster maps, together with some additional features, can be applied to terrain data to create an output terrain database. Terrain and bathymetry can be imported, filtered, projected and exported in a multi-resolution terrain set.

The output terrain database then can be used for on the fly visualisation, simple terrain interrogation (height of a point) and return of cross-section or an area grid at the chosen resolution to support fast terrain calculation.

Methods for displaying terrain information vary - from a textual grid, display of terrain slope, cross sections etc. through to a 3D graph of the results. The Terrain database complements the map and has an SDK which is very similar to allow the application developer to exploit very large terrain sets efficiently.

## 9.3. The Terrain Viewer

The Terrain Viewer is supplied both as an application and as a source code example.

To use the Terrain Viewer, select a map and then load a terrain database. (.tdf) from the 'Load Terrain Option'

Supports:

- Point height display.
- Cross section display.
- Grid display.

The Terrain Viewer accepts maps in any coordinate system but presumes terrain databases to be in un-scaled latitude & longitude units.

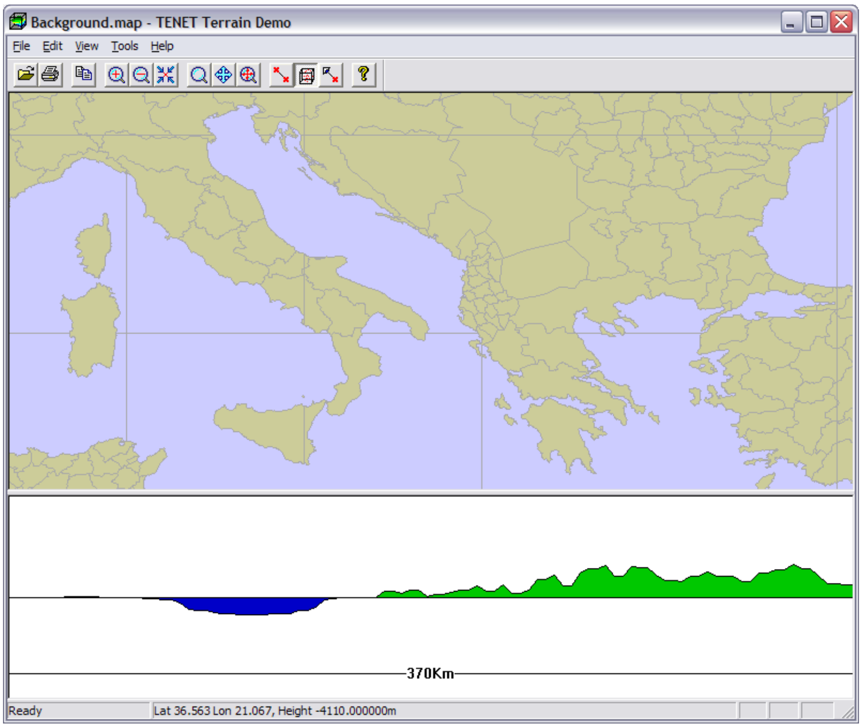



**Figure 40 The Terrain Viewer**

# 10. EXERCISE 5 CREATING A TERRAIN SET

When you have completed this exercise you will be able to

- Load DTED0 into MapLink Studio.
- Create a raster display shaded by colour.
- Create a multi-resolution terrain database to allow terrain queries.

## 10.1. Creating a contoured raster map using DTED

DTED, Digital Terrain Elevation Database contains data based on latitude longitude cells covering the earth. It consists of a set of regular gridded point height/depth values for each cell.

```
Create a new project, and add a layer.

Choose the Add Dataset option, and select the Raster tab.
```

This tab contains the filters of raster formats. For DTED there are two options. You can either select individual one-degree cells by choosing the DTED-RASTER option from the list, or you can load the DMED catalogue file which contains the entire DTED dataset.

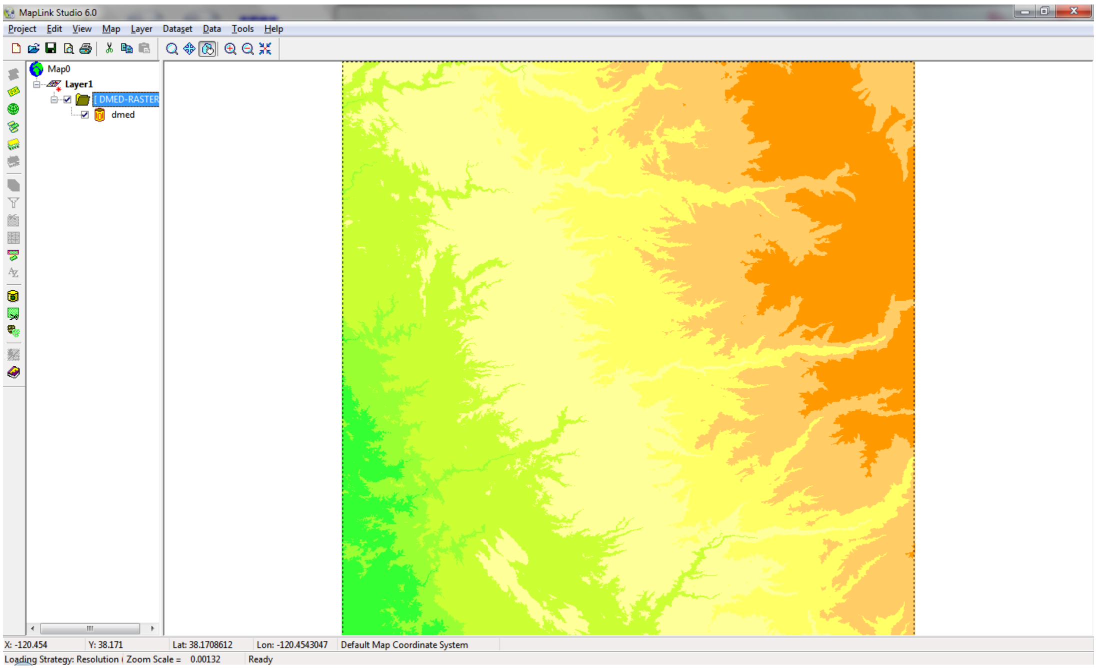
The data to be loaded contains global coverage. In the first instance you can use input clipping to select a subset of this data. This will limit the data to be processed, and therefore shorten the processing time. Processing time could take several hours for a high-resolution world terrain set.

```
Add a DMED-RASTER Dataset.

Apply the following input clipping to the DMED-RASTER
dataset:

   Minimum X   -121
   Maximum X   -120
   Minimum Y   38
   Maximum Y   39

Add a datafile and browse the source directory for the file
DMED in the DTED2_SAMP directory. Add this file to the DMED-
RASTER dataset.

Now open the Format Configuration panel for the dataset.
```

The DTED Format Configuration panel includes many options. At this point we will simply colour the heights present based on a set of ranges generated automatically.

```
Select Auto contour, and a range of colours will be
displayed. They appear blue, but there are actually terrain
and bathymetric values generated.

Enable Make Transparent

Accept the options and reload the layer. A series of coloured
bands will be displayed.
```

The result of the DTED output.



```
Create a directory called 'DTED2 Map'. Save the project and
create an output map.
```

## 10.2. Multi-resolution Terrain Database

We will now create a terrain database to go with the DTED map.

```
Create a new project and add a layer.

Choose the Add Dataset option, and select the Terrain tab.
```

This tab contains the formats that can be used to create a multi-resolution terrain database. For DTED there are two options. As with the Raster options you can either select individual one-degree cells by choosing the DTED-TERRAIN option from the list, or you can load the DMED catalogue file which contains the entire DTED dataset.

```
Add a DMED-TERRAIN dataset.

Add DTED2 DMED data to the DMED-TERRAIN dataset.
```

The same procedure applies to DTED levels 0 and 1.

Like the map we will restrict the data to a more limited area.

```
Apply the following input clipping to the DMED-TERRAIN
Dataset:

   Minimum X   -121
   Maximum X   -120
   Minimum Y   38
   Maximum Y   39

Now open the Format Configuration panel for the dataset, and
select the Land DTM Configuration (Raster) tab.
```

As well as creating a terrain database it is also useful to visualise the data. This is purely to preview the area and ensure you have selected the correct limits

```
Select Auto Contour
Enable the Make Transparent option
Accept the options and reload the layer.
```

You will now be able to see the contoured DTED data.

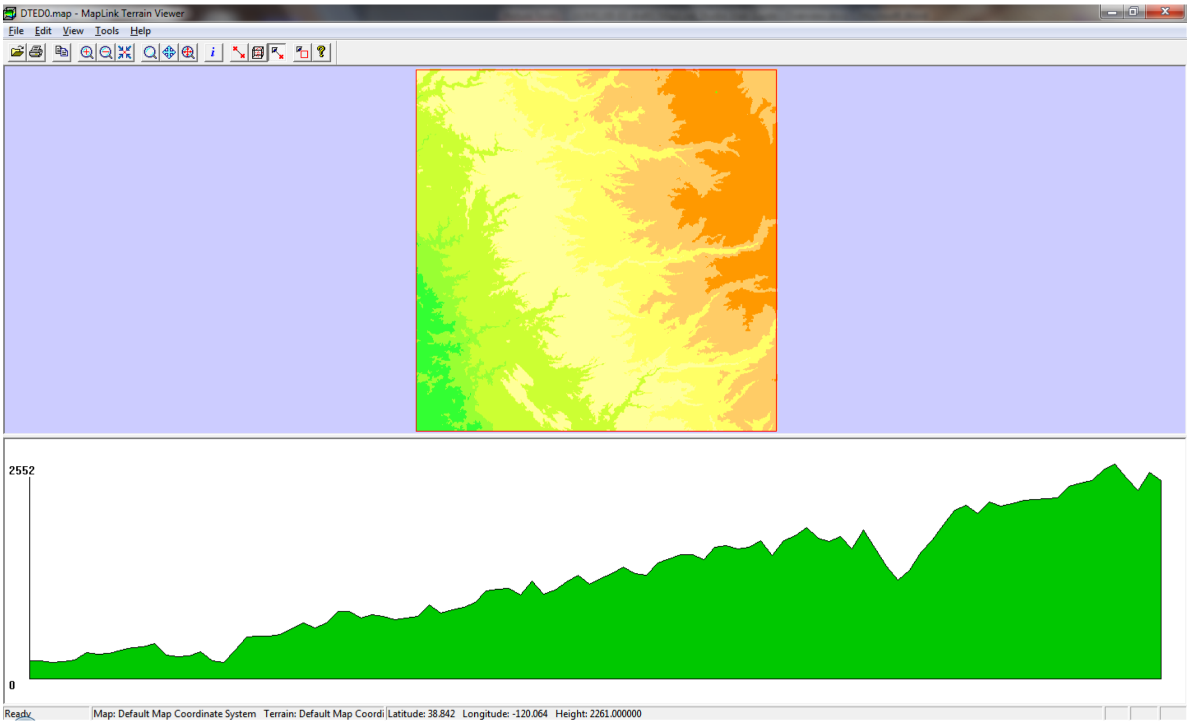
Now to export a terrain set. MapLink provides you with a number of options to control the way that terrain is displayed.

```
Open the Terrain Configuration panel on the DMED-TERRAIN
dataset

Set Z Offset to add to each data item : 0
Scale to apply to each data item : 1

Ensure that Display raster preview of terrain in layer reload
is ticke
```

This means that the raw data will be displayed without any correction, and that the terrain data will be displayed as a shaded raster.

```
Create a directory called 'DTED2 Terrain'.

Save the project and create a terrain database.
```

Use the MapLink Terrain Viewer to look at your map and terrain database. The output should be similar to the following:



# 11. COORDINATE SYSTEMS

A coordinate system is a multi-dimensional (typically 2D or 3D) reference allowing 'position' to be defined numerically. A geodetic coordinate system typically used in mapping is made up of a projection, a datum and possibly a series of linear corrections (offsets, scales, rotations).

## 11.1. Map Projections

Map projection is used to portray all or part of the round Earth on a flat surface. This cannot be done without some distortions. Every projection has its own set of advantages and disadvantages; there is no "best" projection.

The mapmaker must select the one best suited to the needs, reducing distortion of the most important features. Mapmakers and mathematicians have devised almost limitless ways to project the image of the globe onto paper. Scientists design projections for their own specific needs.

Which one best suits your need? Every flat map misrepresents the surface of the Earth in some way. No map can rival a globe in truly representing the surface of the entire earth. However, a map or parts of a map can show one or more – but never all of the following: True Directions; True Distances; True Areas; True Shapes.

MapLink provides a wide range of such map projections. There are many well written guides to map projections and so we will not try to repeat their contents here. It is recommended that you approach your own expert if available for guidance on the appropriate choice or Envitia can provide consultancy help and advice on the best choice. The following, then, provides an overview of the main concepts only.

Each of the main projection categories is covered, these being:

- Cylindrical Projections.
- Conic and Polyconic Projections.
- Azimuthal Projections.
- Isomorphic Projections.

Many but not all projections can be visualised geometrically. So for example a cylindrical projection can be viewed as a cylinder wrapped around the earth and by use of a ray from the earth's centre being used to project information from the globe onto the cylinder.

## 11.2. Cylindrical Projections

In Cylindrical Transforms (examples of which are Mercator, Oblique Mercator, Transverse Mercator) the earth is projected onto a cylinder which is wrapped around the earth either vertically, horizontally (transverse) or at an oblique angle. A typical example is shown in Figure 41 below. This is a transverse Cylinder wrapped around the Greenwich Meridian and then unwrapped.

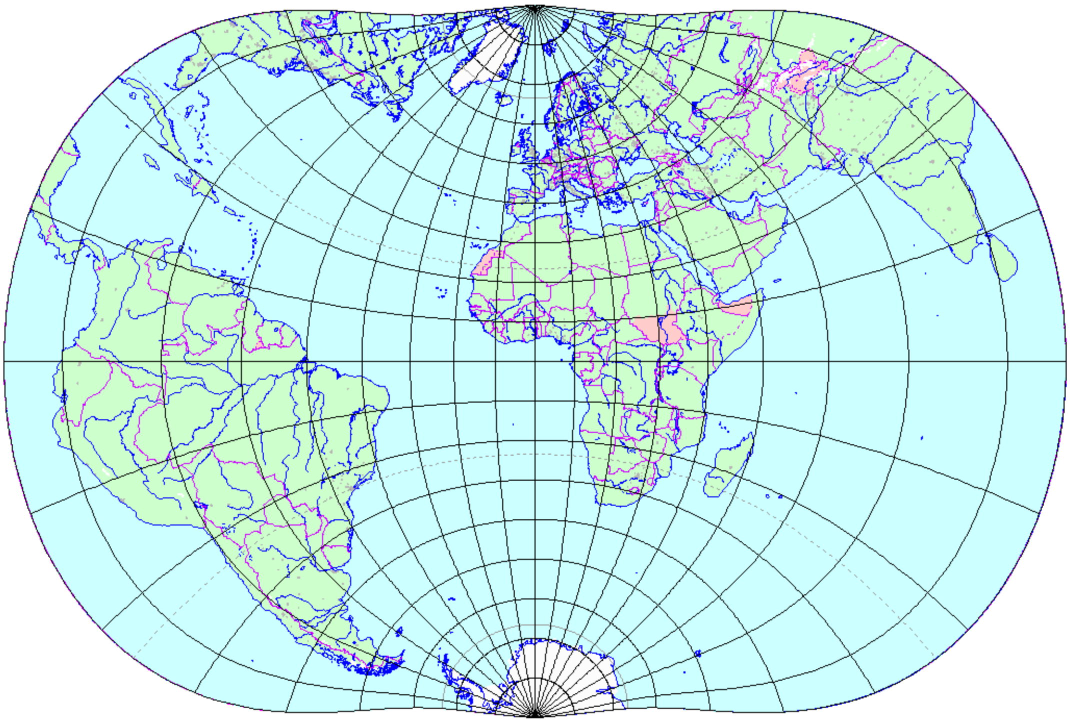


**Figure 41 Transverse Mercator Map**

Another example is Pseudo-Cylindrical (Eckert IV, Eckert VI, Mollweide, Sinusoidal) This is very similar to a Cylindrical transformation.

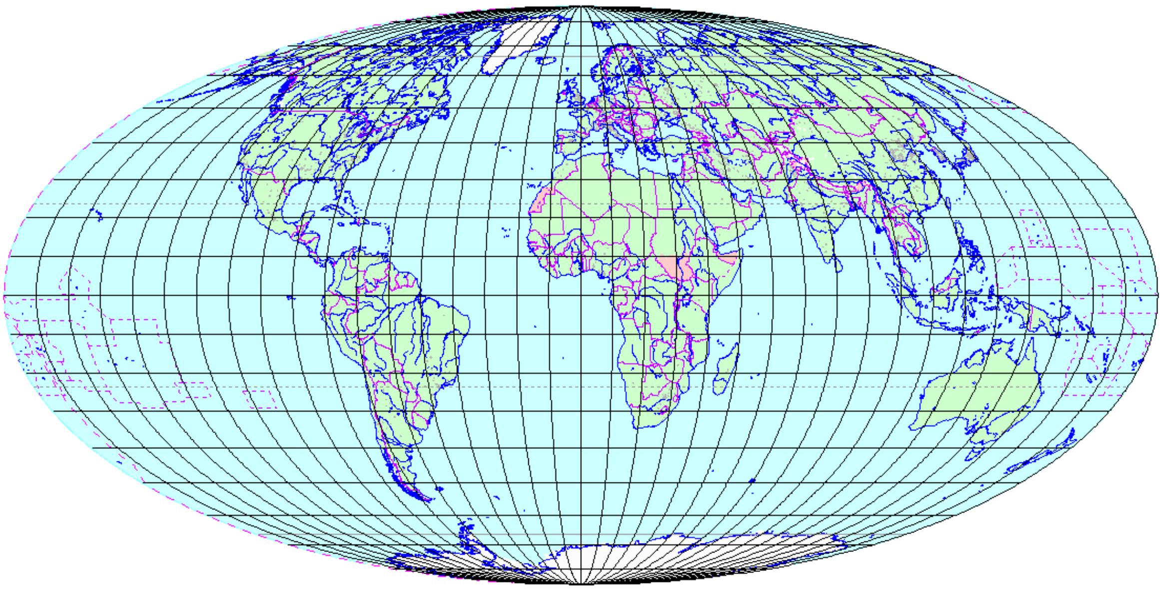


**Figure 42 Psuedo –Cylindrical (Spherical/Mollweide)**

### 11.2.1. Transverse Mercator

EPSG have changed the formula used for Transverse Mercator while retaining the EPSG IDs for the affected Coordinate Systems that use Transverse Mercator.

The original formula "USGS Snyder" and the "JHS" formulas produce similar results in a +-4 degree band around the central longitude. Outside this band the results diverge. The JHS algorithm is more accurate out to +-40 degrees.

EPSG recommend that the two formulas are not mixed.

EPSG recommend the use of the JHS formula.

The Snyder formula was used in MapLink 7.5 and older versions.

Both formulas are available in MapLink 8.0 and newer.

We would recommend that you consider the interoperability issues with other systems when you choose which formula to use.

## 11.3. Conic Projections

Conic Transforms are created by placing a cone onto the earth and projecting the earth's surface onto the cone from a point - typically the earth's centre. (Examples are Equidistant, Lambert Conformal, Alber's Equal Area) The earth is projected onto a cone which is once again unwrapped to provide the flat coordinate system.

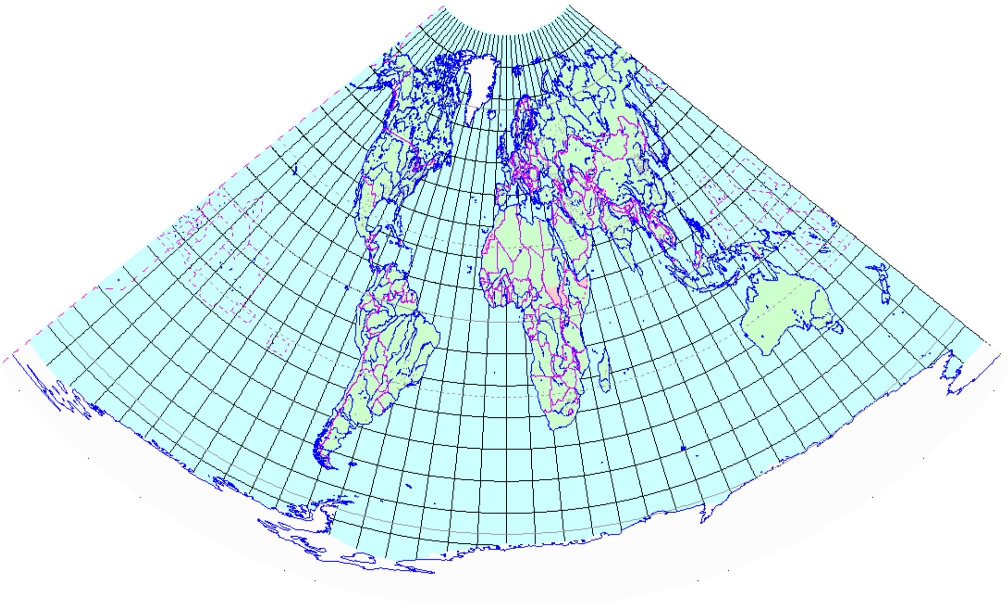


**Figure 43 Conic Projection (WGS84/Equidistant)**

A variation of Conic is the Polyconic projection in which a number of cones are used.

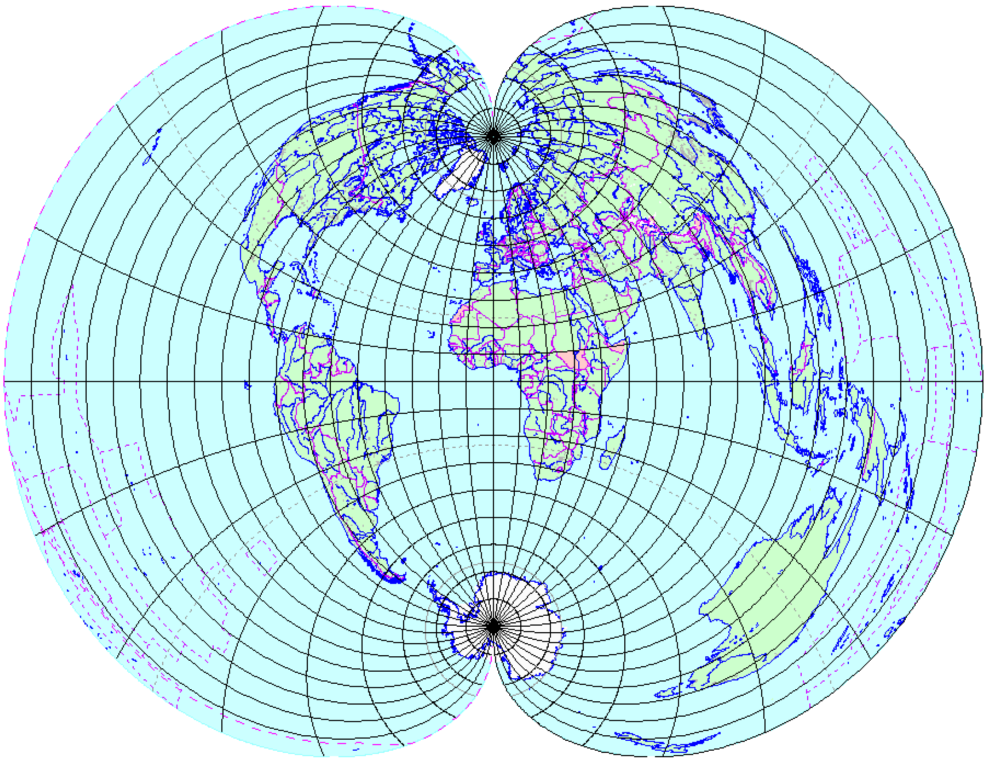


**Figure 44 Polyconic Projection (Spherical/Polyconic)**

## 11.4. Azimuthal Projections

In Azimuthal Projections (examples of which are Orthographic, Stereographic, Gnomonic) the earth is projected onto a plane which is placed against the surface at a given point.

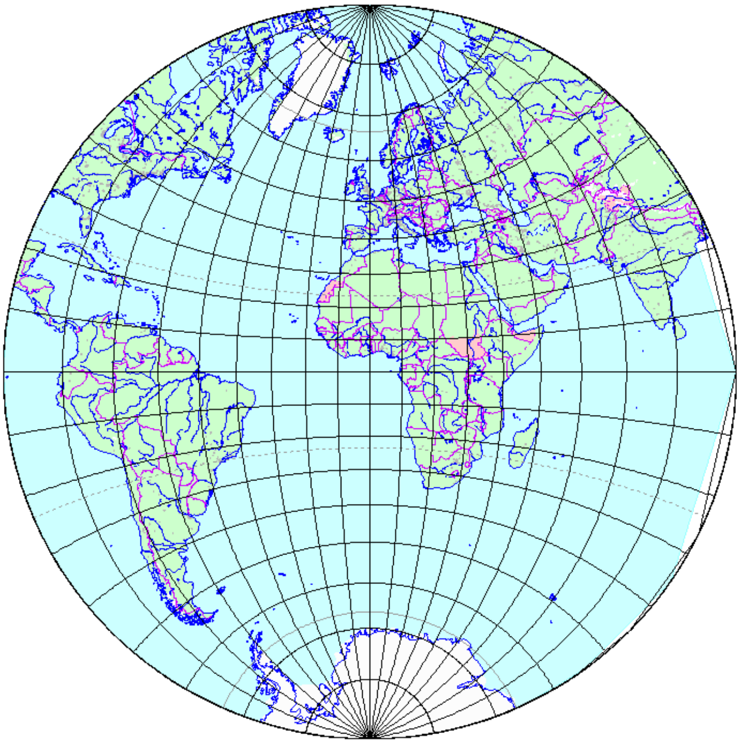


**Figure 45 Stereographic Projection**

## 11.5. Isomorphic Projections

Isomorphic projections are a special category in that it is no transform at all, the latitude longitude values are plotted directly in Y and X to give a flat projection. When displaying the entire earth the display may look a little strange, it is useful in some situations (e.g. to display a limited area, particularly if X/Y scales are set to approximate the distance in latlon at the area being displayed).

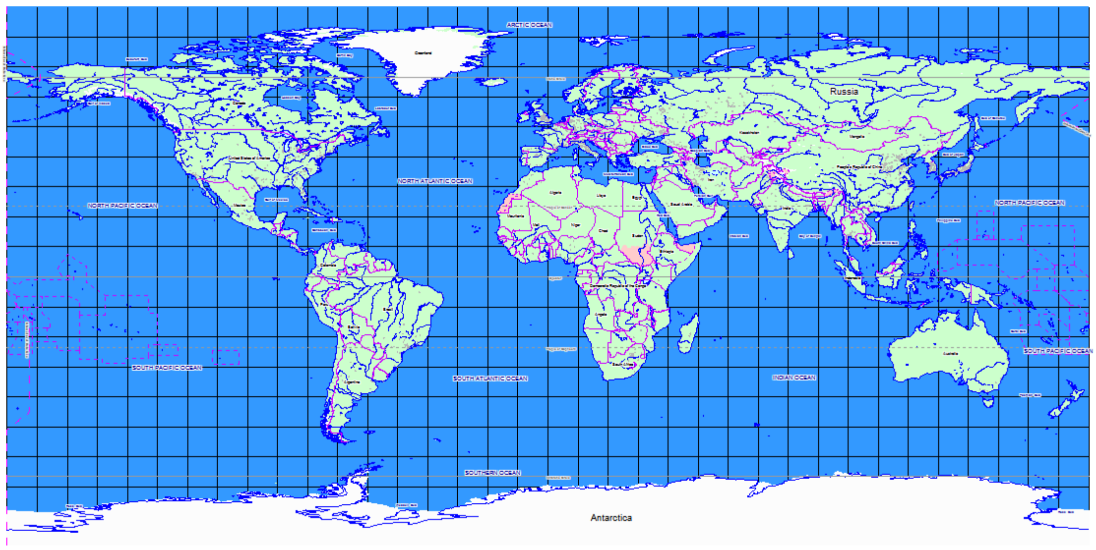



**Figure 46 Isomorphic Projection**

## 11.6. Reference Systems

A map is defined by a map projection but the projection assumes the shape/size and position of the earth itself. Of course the earth is far from a regular sphere and so a reference system is needed to model it.

Examples of reference systems are ED50 and WGS84. The reference system defines the model used to form the Earth the most common model of the earth is some form of ellipsoid. And because the Earth cannot exactly match the model, where the model is fixed to the Earth is critical.

Because the earth model was originally derived before satellite position fixing the best reference was to lock the model to the earth near the area that the projection was to be used. Thus there are reference systems with Datums in Europe, Japan etc. However with the advent of Satellites the WGS 84 System was born. WGS84 uses a conceptual reference point at the centre of the Earth. But other systems (such as ED50 which locks the spheroid to a point on the Earth surface in Europe) are still in use.

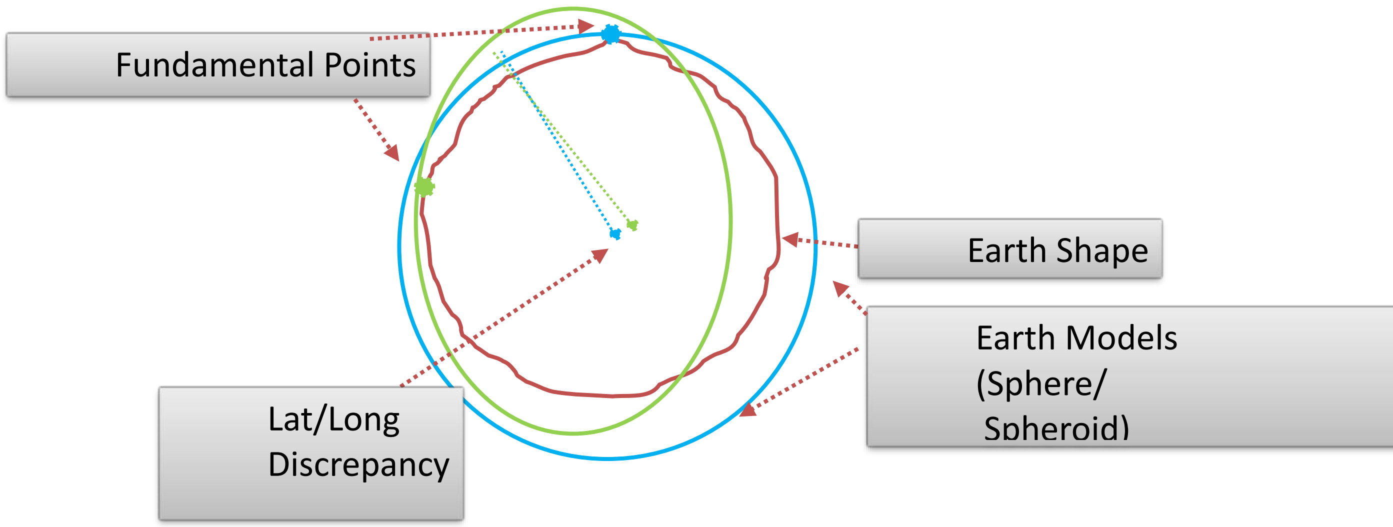



**Figure 47 Different Reference Systems/Datums**

## 11.7. Reference Systems in MapLink

MapLink Pro works internally in WGS84 (the common reference system) But it allows map data conversion to and from different reference systems.

A MapLink map supports a single output coordinate system i.e. a single map projection/reference system pair, however each Dataset can have a map projection/reference system pair defined to allow MapLink to project the data into WGS84. Therefore data in different coordinate systems can easily be merged into a single composite map.

Reference systems are known as Datums (as the point of contact with the earth is the key element). A well defined set of Datums exist for most places on the Earth. MapLink allows the selection of the datum for input and output coordinates.

## 11.8. The Coordinate System Model in MapLink

The entire transformation process involved in converting from one coordinate system to another is shown below:

- Each source dataset may have an input coordinate system. (Linear transform, projection and reference system /datum)
- There is a single output coordinate system, again with all three components.
- Some elements may not be required, for example for a lat/long source data input transforms is not required.
- This view is also simplistic in that datum conversion is often defined as a conversion directly from one datum to another.
- MapLink picks the most appropriate conversion.

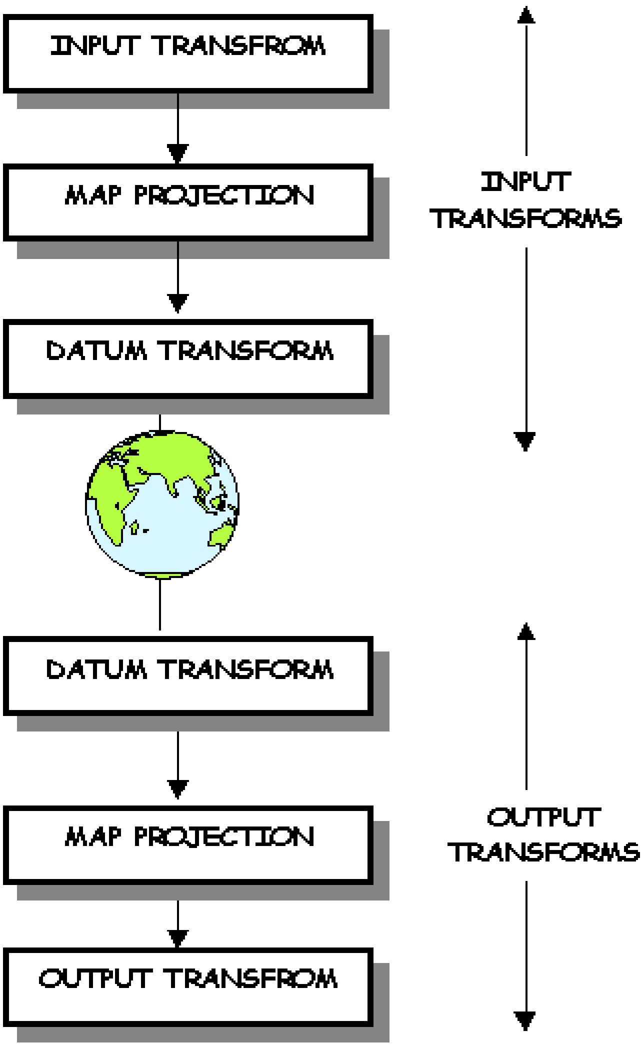



**Figure 48 Coordinate System Model**

In many cases only part of this model is used, as often the source data is in Lat/Long using the WGS84 Reference System and so no input coordinate conversion is necessary. Also it may be that output in lat/long is required, and so in this case no output coordinate system is required.

MapLink includes two default input and output coordinate systems (Default Dataset 'Coordinate System' and 'Default Map Coordinate System' respectively) that provide purely null conversions (input/output in Lat/Long WGS84).

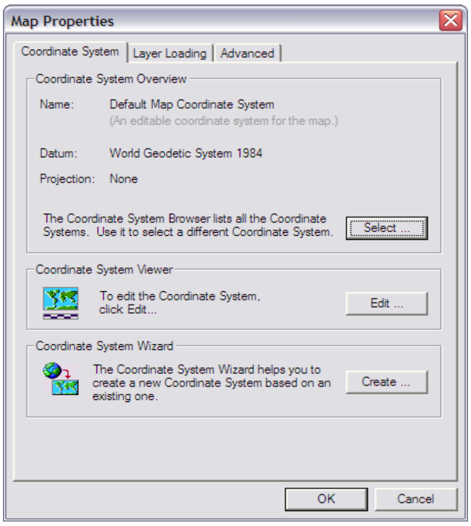
## 11.9. Map Coordinate System Interface

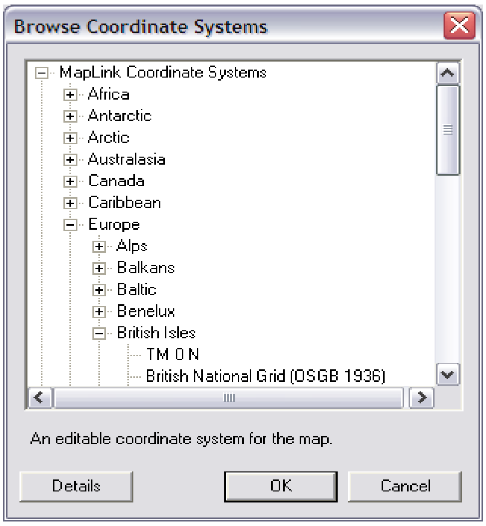
The coordinate system interface is made up of three dialog windows.

The Map Properties Dialog allows the user to see the output coordinate system in use. A summary of the coordinate system is also shown, but more details can be viewed by selecting View or Edit.

The user can then either select an existing coordinate system or create a new one by invoking the wizard.

The wizard guides the user through the options available including projection and datum choices.




'Select' invokes the Coordinate System Browser.

The Coordinate system browser allows the user to select a specific coordinate system for use. The browser displays both standard MapLink coordinate systems and coordinate systems defined by the user from the new coordinate system wizard.

The View or Edit Button (depending if the coordinate system currently selected is a standard MapLink one (read-only) or a custom one) allows the user to invoke the Coordinate System Properties Dialog.

The Coordinate System Properties dialog allows the user to view the details of a pre-defined coordinate system, and modify the details of a custom coordinate system.

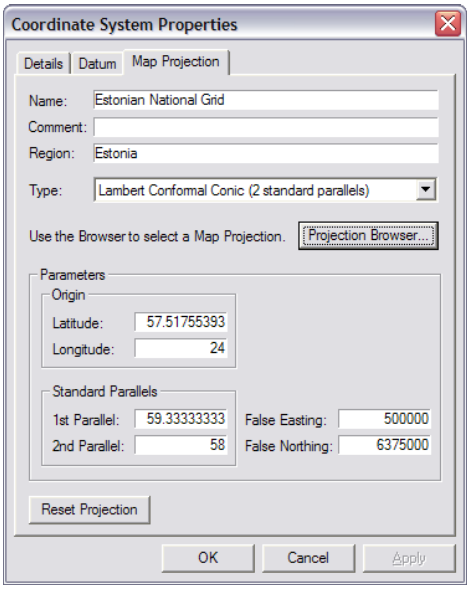



**Figure 49 Map Coordinate System Interface**

## 11.10. Map Coordinate System at Runtime

The coordinate system is not just used to produce the map. The output map stores the coordinate system details so that the application can accurately position objects on the map.

An important fact is if you wish to position objects in your application using latlon, an Output Coordinate System MUST be defined even if your data is already in the desired Output Coordinate System

For example, consider creating a map in British National Grid, using British National Grid (BNG) data. Without an Output Coordinate System definition, positioning could only occur in the local coordinate system which is X-Y, not in Latlon; MapLink knows nothing about the coordinate system. However if you correctly set both the input (Dataset) and output (Map) coordinate systems to BNG then MapLink will not convert the data (it will recognise that they are both in the same coordinate system) but it will be able to return for example lat/long as well as BNG Values.

The advanced tab of the Map Properties dialog allows the TMCPerMU value of the map to be modified. This value is used by MapLink in order to convert between the units of the map (MU) and internal geometry units (TMC). By default MapLink will automatically select an appropriate value.

If an application needs to overlay multiple maps, their output coordinate systems and TMCPerMU values must match. If the TMCPerMU value of a map is too low, some details of the source data may be lost when the map is processed. The upper limit of this value depends on the units used by the output coordinate system (X-Y/latlon), and the extent of the data.

## 11.11. ARC Coordinate System Data

The ARC Coordinate system is commonly used in military mapping. For example, both ASRP and CADRG maps use this system.

The projection uses the Equal ARC Second coordinate system. This is a multi-zoned coordinate system. MapLink understands the zoning and so can set up the input coordinate system correctly.

MapLink can only have a single output coordinate system in one map, therefore CADRG and ASRP data is usually projected.

It is however possible to operate with the ARC projection model in MapLink by projecting into a linear scale in X and Y and performing the X reduction in the SDK.

## 11.12. Output Coordinate System Choices

The output coordinate system is one of the following:

- The default output coordinate system '**Default Map Coordinate System**'. Pure Lat/long WGS84 with no coordinate conversion at all. For general reasons it is preferable to have a coordinate system set and so this option is really only used to initially view data where the output coordinate system has not yet been chosen.
- A standard, fixed MapLink Output Coordinate System (For Example **Europe/British Isles/British National Grid (OSGB 36)**. This is a standard fixed coordinate system.
- A custom coordinate system, defined with the Coordinate System Wizard. This is also a fixed coordinate system but is user defined.
- A fully dynamic projected coordinate system (**Default Runtime projection coordinate system**). This coordinate system is used with the full runtime projection feature in the MapLink SDK. Note though that this coordinate system cannot be used if you intend to use Raster data. It is sometimes also desirable to create a custom

variation of this coordinate system as the one aspect which can improve performance is to convert data to the datum that will be used at runtime. This coordinate system is defined as a scaled latlon coordinate system. In order to maintain source data resolution, the recommended TMCPerMU value for world coverage is 5,000,000.

- Dynamic ARC Coordinate System. (**Worldwide/Dynamic ARC/Dynamic ARC Grid**). This coordinate system is used in conjunction with the SDK and allows MapLink to support a semi-dynamic coordinate model. It operates at latitudes between 80S and 80N (i.e. not including the poles). It does support Raster Data.

## 11.13. Input Coordinate System Choices

There are three possible situations for the input coordinate system:

- Source data is in Lat/long and WGS84 and therefore the Default Dataset Coordinate System can be used (which defines purely a unity coordinate conversion).
- Source data is in a format for which MapLink automatically sets the input coordinate system (because it is available from the source). Currently this happens for ASRP, CADRG, CIB. In these cases you do not need to and must not set an input coordinate system. Check the MapLink Studio Help for the format you plan to use to see if this feature is supported.
- Source data is in a known coordinate system but it is not set automatically (for example for a TIFF, JEPG or NITF Image). In these cases you must define and select the correct Dataset Coordinate System. Note the coordinate system can only be set on the Dataset not on a data file and thus images or rasters in different coordinate systems need separate datasets.

## 11.14. Raster Projection Information

MapLink projects raster data in the same way as vector data. But certain sources, for example CADRG/ASRP, are already in a projection. MapLink automatically sets an input coordinate system to convert them to latlon. There is no need in these cases to set an input coordinate system.

GeoTIFF can be in lat/long or in a projection. Since it has not in the past been possible to reliably match the GeoTiff Tags for the projection you must select an appropriate input coordinate system yourself. MapLink reports the GeoTIFF string in the Data Analysis page available on the Data item in MapLink describing the projection to help you do this.

Note: A worthwhile option to set when previewing large images which need projection is under the menu option 'Tools->Options (Processing Tab). It is named 'Thumbnail Raster View'. This option when enabled will mean that for preview purposes a significantly reduced image resolution will be used. MapLink Studio will redisplay changes much more quickly. This option does not affect the output map which is still built at full resolution.

## 11.15. Clipping

Clipping is used to reduce irrelevant data outside the area of interest. This removes the need to store unwanted data, and therefore reduces the storage requirement and also speeds up processing.

Clipping can improve performance at runtime, but if properly optimised, the size of the map is largely irrelevant to the overall map performance as MapLink is designed to deal with large datasets in an incremental way.

Three types of clipping are available:

- Geodetic clipping Using the WGS84 coordinate system.
- Input clipping Clip the data in its original coordinate system.
- Output clipping Clip the data in the output coordinate space.

The various options are used in different circumstances. The way they are applied is shown in figure 34.

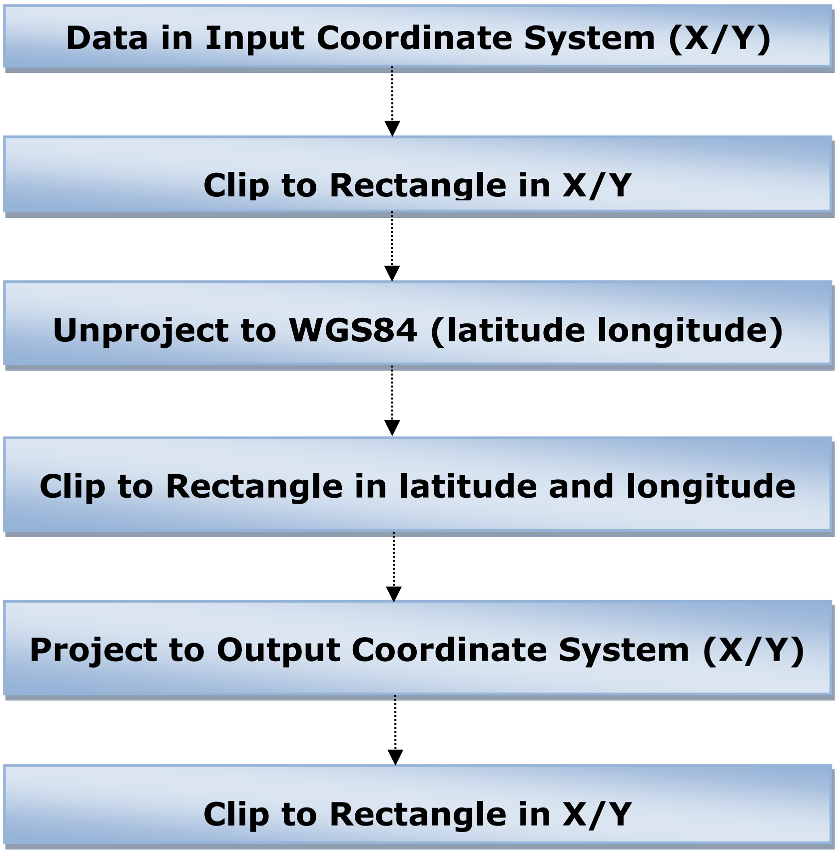



**Figure 50 The Structure of the Clipping Model**

Input Clipping is strongly advised for large datasets (such as VPF or DTED). It limits the amount of data that MapLink has to process and therefore significantly reduces the time to process a map.

Output Clipping is useful as it clips to a rectangle in the output map (the other forms of clipping may have odd shapes). Output Clipping can be defined graphically.

# 12. EXERCISE 6 COORDINATE SYSTEMS

When you have completed this exercise you will be able to

- Select an existing coordinate system.
- Show the effect of different coordinate systems.
- Define new coordinate systems.
- Convert input data from projected to un-projected format.
- Clip data in input units, latitude/longitude and output units.

## 12.1. Coordinate Systems

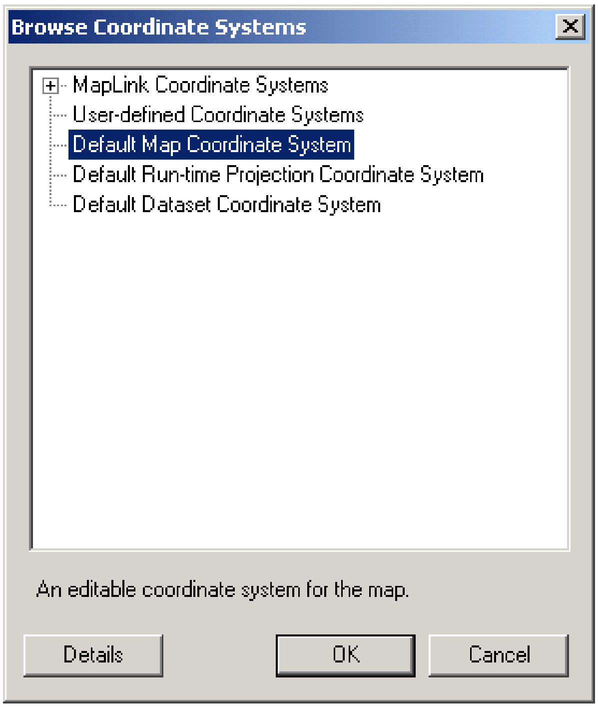
The MapLink projects that we have been working on currently have no projection information stored. This means that the output is exactly the same as the input. MapLink provides extensive coordinate system support to allow you to convert different types of data between projections and datums, and to project your final map into the required coordinate system.

You will find the Output Coordinate System page under the Map menu.

```
Open the Output Coordinate System page for your existing
project.
```

This shows that the project currently contains no projection information. The Datum is listed as World Geodetic System 1984 (WGS84). MapLink uses this datum internally to do all data conversions. This projection is the Default Map Coordinate System.

The Coordinate System page allows you to do three things. Firstly, you can choose an existing coordinate system and apply it your map. Secondly, you can take an existing coordinate system and modify some or all of the parameters to create a user-defined coordinate system. Finally you can create a completely new coordinate system. Taking each of these in turn, let's look at selecting an existing coordinate system.




The existing coordinate systems are listed under MapLink Coordinate Systems. These are all standard coordinate systems that you can choose to apply to your map.

```
Select an existing
coordinate system e.g.
The British National Grid
which can be found under
the Europe | British
Isles section.

Use the Details button to
look at the parameters of
thi coordinate system.
```

The coordinate system details contain the datum and projection information associated with the selected coordinate system.

```
Select OK to choose the coordinate system. You will need to
reload each layer to view the projected data.
```

Notice how the shape of the map changes once the data is projected. Notice also how, as you move your mouse across the map, the correct latlon as well as X-Y position is displayed on the bottom bar. Because the model of the Earth we are using is a spheroid or an ellipsoid both with a radius in metres, the X and Y coordinates now also correspond to metres.

```
Choose a number of the existing coordinate systems and apply
them to see how the data is updated.
```

## 12.2. User-defined coordinate systems

As stated previously, there are two mechanisms for defining a user-defined coordinate system. The coordinate system wizard helps you to create a user-defined coordinate system from the list of existing systems, and it also helps you to create a user-defined coordinate system from scratch.

```
Try using the coordinate system wizard to create a user-
defined coordinate system based on the British National Grid

Try using the coordinate system wizard to create a completely
new user-defined coordinate system.

Reload the layers each time to look at how the map shape
changes.
```

## 12.3. Input Coordinate Systems

All of these coordinate systems have been applied to the map as a whole. It is also possible to apply an Input Coordinate System to an individual dataset. This is used to convert input data from projected to unprojected. Remember MapLink works internally in WGS84 and therefore all input data must be converted to this system before an output projection can be applied.

```
Select a dataset and open the Input Coordinate System panel.
```

You will see that this is exactly the same as the Output Coordinate System panel – you only have to familiarise yourself with the layout once!

The Shapefile data and the raster data you have in the project are already in WGS84 and therefore in this case you do not need to define an Input Coordinate System. If the data was, for example, in British National Grid, then this is where you would select the British National Grid coordinate system. MapLink would then know how to unproject the data to be able to use it internally. This also allows you to combine data of different coordinate system together in one map.

## 12.4. Data Clipping

Often data will be provided covering a wider geographic area than you are interested in. Clipping allows you to remove extra data from around the edges of the area of interest. This ensures that unwanted data is not stored or loaded in your runtime applications.

Three are three forms of clipping in MapLink.

- Geodetic clipping This allows you to clip data at geodetic level.
- Input clipping This allows you to clip the data in its original coordinate system.
- Output clipping This allows you to clip the data in the output coordinate system.

## 12.5. Geodetic Clipping

Let's look at Geodetic clipping first. This allows you to clip data using a shape defined by latitude and longitude positions. Therefore if you know the exact latlon of the area that you need to cover then this is an ideal method to use. Geodetic clipping takes place on a layer – therefore all the datasets contained within that layer will be clipped to the values you enter.

```
Try applying Geodetic clipping to layer1 to only show the UK.
The latlon grid you have in layer1 will help you with the
values!

Reload the layer to look at the clipped data.

Once you are happy with this turn off Geodetic clipping and
reload the layer.
```

## 12.6. Input Clipping

We'll now look at Input clipping. This works on a dataset level and allows you to clip each dataset using a shape defined by the data input units. The bottom left and top right coordinates of the clipping rectangle need to be entered.

In the case of the Shapefile data that we are using, the data is already in latlon and therefore in this case you can simply enter the latlon coordinates as you did previously for the Geodetic clipping.

```
Try applying input clipping to layer1. This time, clip round
an area that contains Europe.
```

```
Once you are happy with this turn off input clipping and
reload the layer.
```

## 12.7. Output Clipping

Finally, output clipping. This works at a dataset level. This time you can define a shape in the output coordinate space. Again the bottom left and top right coordinate of the clipping rectangle are required. Output clipping makes this easier by providing a Define Extent button.

```
Try applying output clipping to layer1. This time use the
Define Extent button, and drag a rectangle to cover the area
that you wish to clip around.

Once you are happy with your clipped extent, save the
project, create an output map and view it using the MapLink
```

```
Viewer. You should see a projected map with a clipped
(reduced top layer)
```

You now have the ability to project your map data as well as reducing the amount of data displayed!

# 13. ADVANCED TOPICS

The following sections are for completeness and are covered in the Studio Training Course and Studio Help. The Studio Training Course covers the following and more, please contact sales for a course.

## 13.1. Filtering Vector Data

Filtering reduces the detail in a layer by removing points of a line or polygon

Useful when constructing overview layers from detailed data.

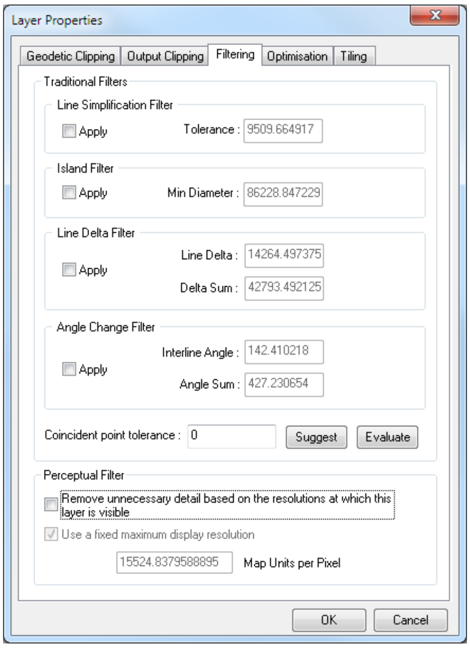
This can be done safely for overview layers where some points are imperceptible from each other when zoomed out.

This results in smaller tile sizes and consequently faster maps.

Filtering is applied to a complete layer of data.

MapLink supports the following filtering techniques:

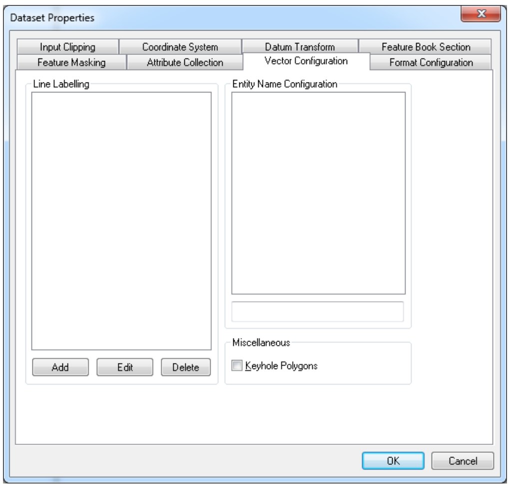
- Perceptual Filter. This is the filter recommended for general use. Removes points that have no visible effect on the shape of a line or polygon based on the resolutions at which the layer is visible.
- Line Simplification Filter. Uses only extreme points that define the shape of the line.
- Line Delta Filter. Removes points less than a specified distance from the last.
- Angle Change Filter. Removes points less than a specified angle change from the last.



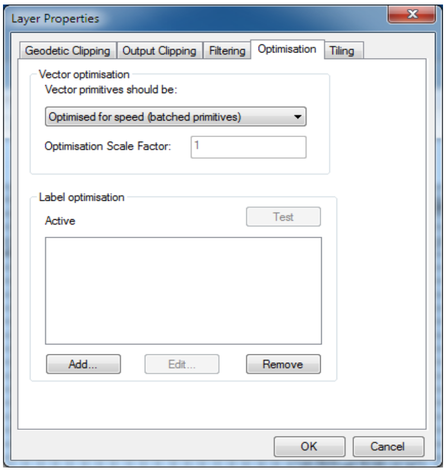

## 13.2. Keyholing Polygons

Replaces all input polygons with "keyholed" polygons. This improves performance on X11 platforms.

- Reduces the number of rings in the map.
- Some processing overhead, especially with large complex polygons – tree line in Canada, sea polygon near Denmark.
- Improved performance on some systems.
- Only when edge display not required, e.g. where there is another feature available for coastlines.



## 13.3. Vector Optimisations




- Reduces tile sizes.
- Batching is OK for typical map usage.
- Optimisation for compression is only really applicable to very poor bandwidths (<56k).
- Geometry in map does not reflect input geometry as information and attribute data is lost.

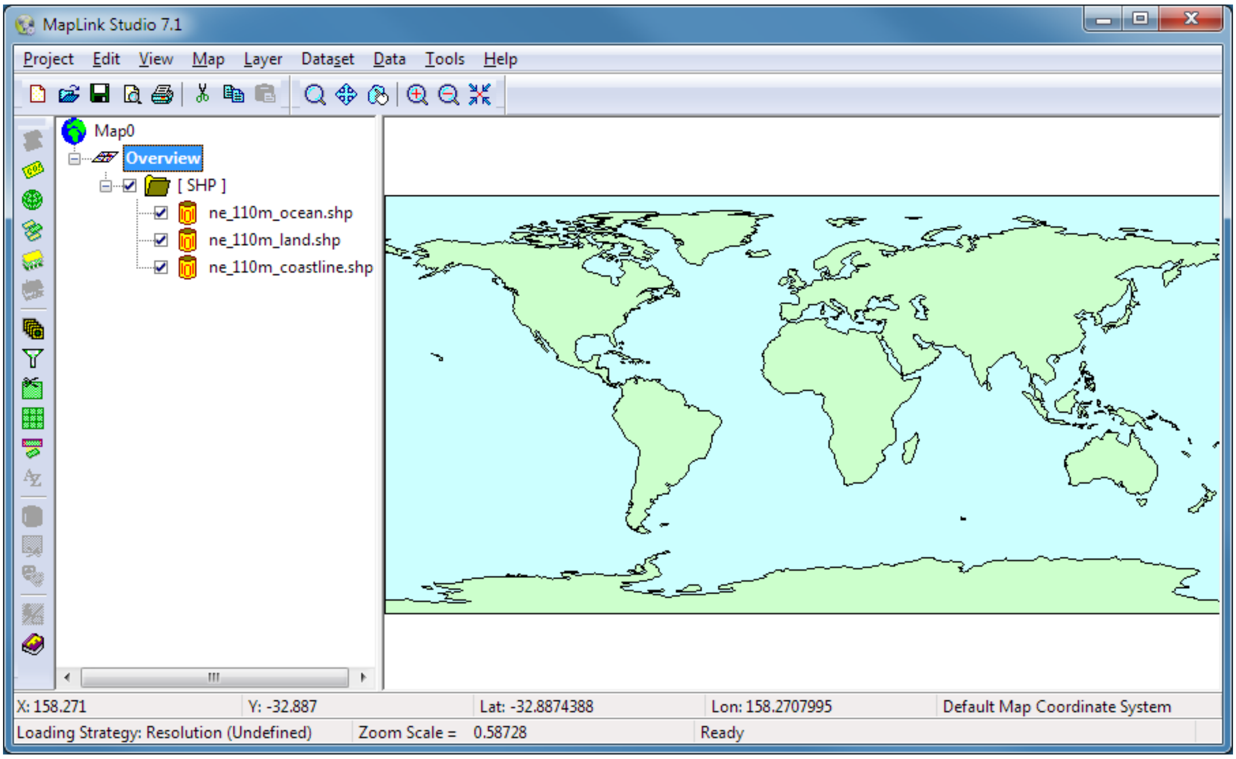
# 14. Optimising Maps for Performance

The topics covered in section 5 cover the basic techniques for creating efficient maps. Often these steps are enough for fast runtime display of the map, in which case nothing else needs to be done. However, if the runtime performance of the map still isn't fast enough what else can be done?

## 14.1. Remove Unnecessary or Duplicate Vector Data

Beyond using source data at an appropriate scale for each detail layer, it is also necessary to ensure that data that does not contribute anything to the appearance of the detail layer is not included in the map.

As a simple example, when creating a simple overview layer you may decide that you want to include oceans, land masses and coastlines, so each is added to the layer in the project. The fill colour of the land and ocean polygons is then set to give the desired appearance. Overall the layer would look like this:




While the layer gives the desired appearance, there are two things that could be changed to make the layer simpler, and thus faster, without affecting its appearance.

The first area for improvement stands out if each of the source items is viewed independently:

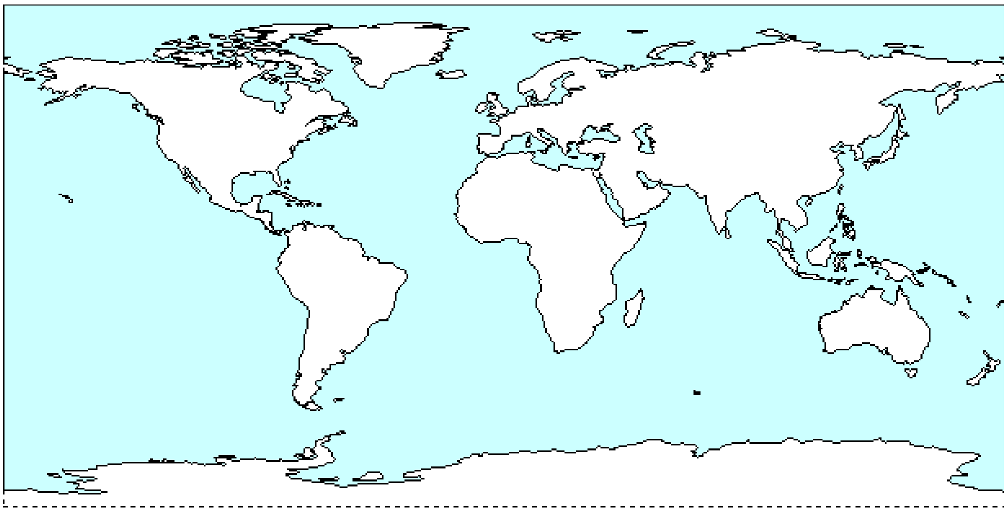


**Figure 51 - Ocean Features**

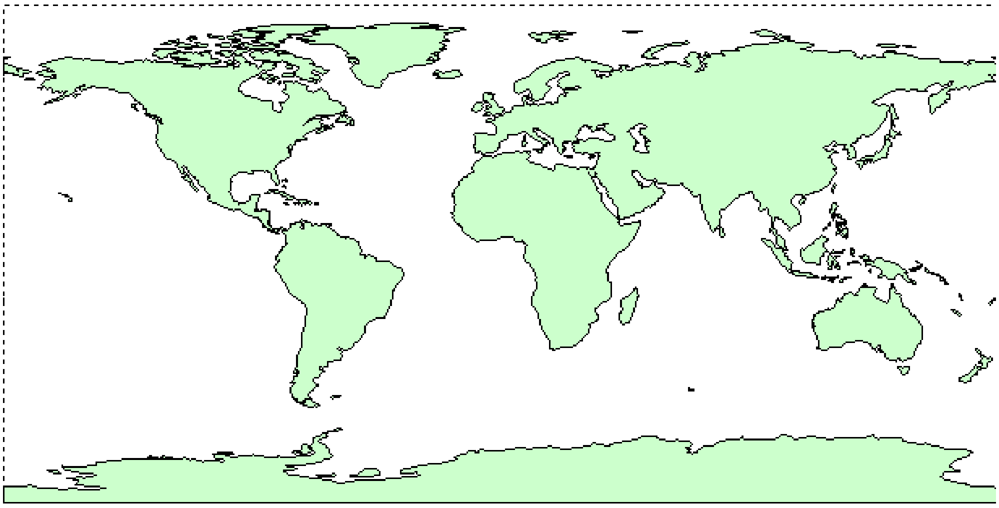


**Figure 52 - Land Features**

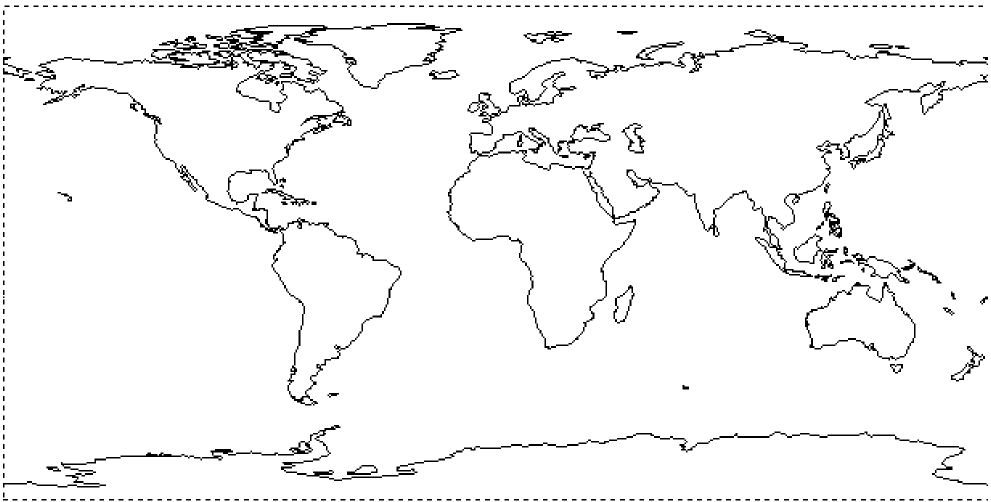


**Figure 53 - Coastline Features**

When the rendering for the ocean and land features was configured in the Feature Book, the default line rendering was left with the default setting of one pixel wide black lines. This means that when the map is displayed the coastline edges are actually rendered three times (once from each feature type), whereas this only needs to happen once. This could be done by turning off line rendering for the ocean and land features - however with this data there is no difference between the coastlines features and the edges polygons from the land features. This means that if the line rendering for the land features is left enabled the coastlines features can be removed from the layer entirely without affecting how the layer looks.

The second area for improvement is in the data used to visualise ocean areas, in which the data consists of polygons that have the land areas cut out. Instead of using these potentially quite complicated polygons it would be much more efficient to use a simple rectangle that covered the globe for the ocean areas, and have the land features drawn on top. A simple source file to use for this purpose is provided with a MapLink installation in the following folder:

```
data/ascii/Background.ascii
```

In combination these changes have reduced the amount of data in the layer to one third of what it was originally without changing the appearance of the layer at all, which gives a corresponding increase in runtime performance of the map.

Even better, the background colour of the map could be set without needing any data at all, using the settings in the Options panel.

In summary:

- Remove line data whose visual effect can be duplicated by enabling polygon edge rendering in another feature.
- Use simple rectangles that appear behind other data in the layer to fill large background areas instead of many complex polygons, or simply set the background colour of the map.

## 14.2. Enable Perceptual Layer Filtering for Vector Layers Where Possible

Maps that contain multiple vector layers can often have overly complex features present in the overlay layers, especially when using data that was not created for use within the scales that the layer will be shown at.

The perceptual filter can automatically reduce the complexity of these features, which results in significant improvements in both drawing performance and reductions in both memory usage and loading times due to the smaller amount of data present in the layer.

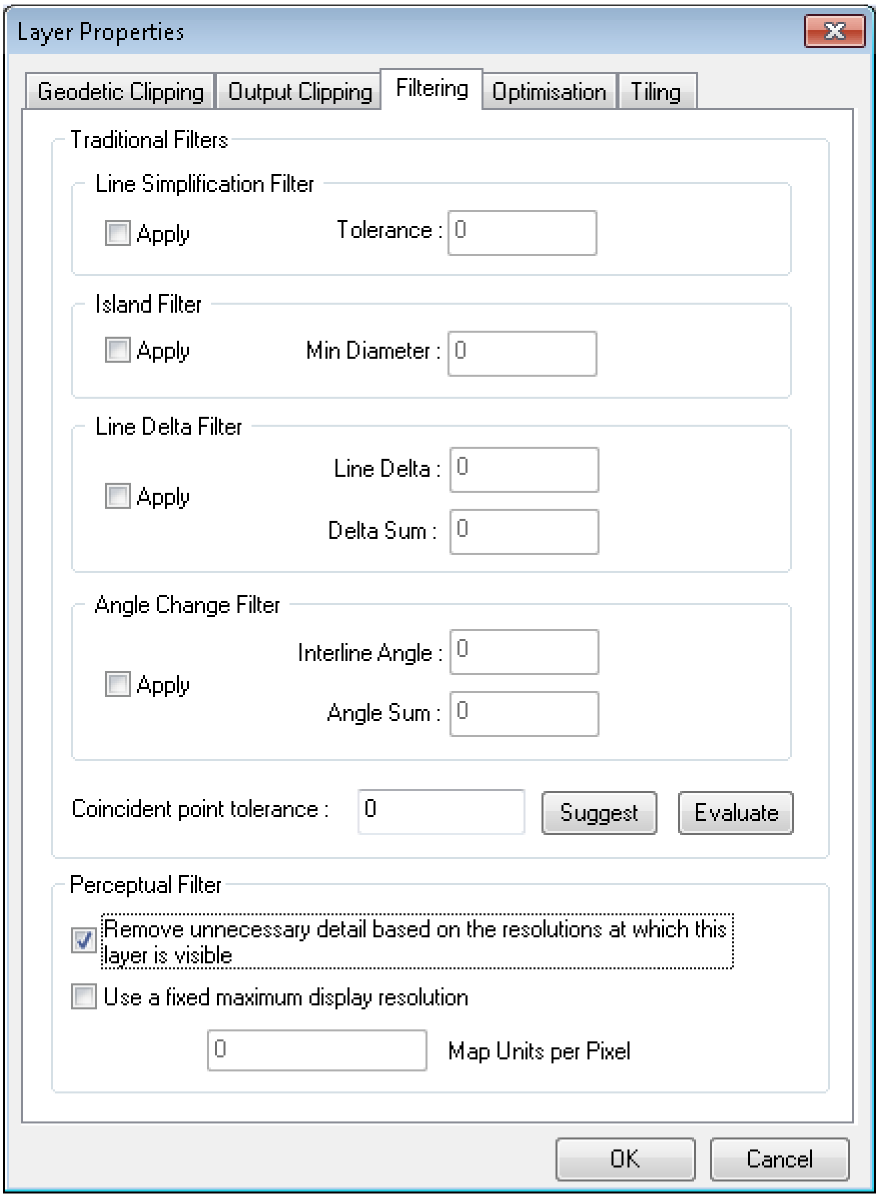



**Figure 54 – Perceptual Filter**

The perceptual filter requires that resolution based layer loading is used.

## 14.3. Use Feature Masking to Remove Unwanted Features

When using data types that include many features within a single source file, such as VPF, to build layers it is frequently the case that not all features should be included in the layer. Making these features invisible by clearing the rendering styles through the feature book does not stop them from being included in the map, which means that the map requires more disk space and memory.

In high level overview layers this can include text and point features whose size is usually specified relative to the scale of the map being displayed as they are often too small to be seen when viewing the overview layer.

Feature masking stops these features from being included in the map, which means that they do not have to be considered at runtime for display thus increasing performance.

## 14.4. Use Output Clipping to Produce Straight Boundaries

When reprojecting a map containing vector data, features around the edge of the map have additional points inserted in order to ensure they appear correctly in the output projection, as often straight lines in the source data are curved after the projection has taken place. When the extreme edges of the layer are not of interest output clipping removes these curved edges, reducing the complexity of background areas of the map.

Geodetic clipping is still required where the data crosses the dateline boundary or to improve processing performance, but can be replaced or supplemented by output clipping to improve run-time performance.

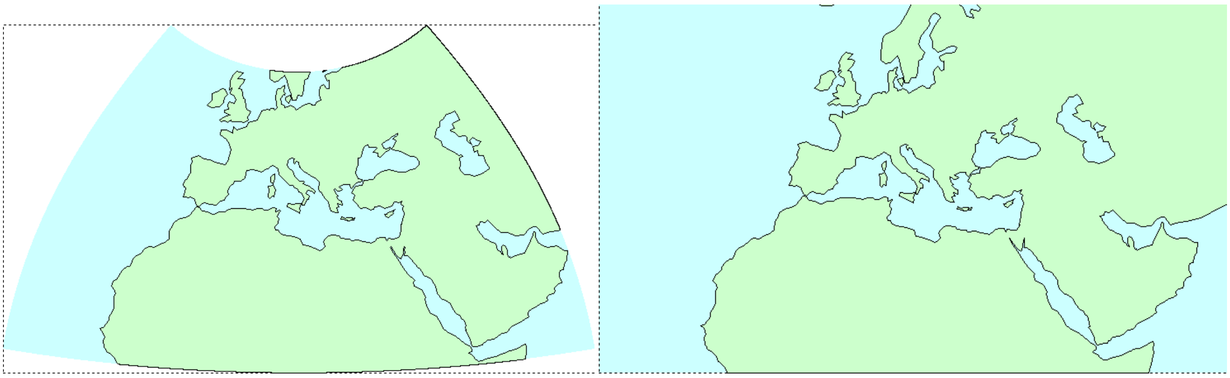


**Figure 55 - Geodetic Clipping vs Output Clipping**

## 14.5. Enable Raster Downsampling for Raster Datasets

Also referred to as raster pyramiding, this option creates a pyramid of tiles from the original raster(s) of decreasing resolution. At runtime the level of the pyramid most suitable for the current display scale will be shown, which substantially reduces the memory requirements for displaying large rasters while still allowing them to be shown at their full resolution when zoomed in.

This option is enabled by default for raster datasets. Note that this is different from 'Raster Resizing' which reduces output quality and is generally not recommended.

## 14.6. Avoid Unnecessary Use of Render Levels

Using render levels can be invaluable for visualising complex features, however they come at a runtime cost to do so. Render levels should not be used to set the drawing order of data if the same effect can be achieved by reordering the features that make up the layer as described in section 4.5.

## 14.7. Enable Automatic Overview Layer Generation

If the map does not naturally have several detail layers that provide an increasingly detailed view of the same data, consider enabling automatic overview layer generation. This ensures that detailed maps will contain suitable detail layers for use when viewing large extents. Without these overview layers a detailed map would require significant amounts of memory to display in its entirety and would correspondingly be slow to draw.

Automatic overview layer generation can be enabled from via the Map Properties options page.

## 14.8. Enable Vector Optimisation for Complex Vector Layers

The use of vector optimisation (where objects of the same type are batched together for rendering purposes) will give a significant improvement in run-time performance. Vector

optimisation can be used in any layer of the map where you do not need to interrogate individual objects within the layer, and where you do not need to apply complex line styles to any of the line features in the layer. The improvement may be particularly noticeable if optimisation for compression is used in conjunction with the **Map | Advanced** option to control the internal coordinate space. This can allow some or all of the objects in the map to be coded into a smaller number of bits in the output file than would otherwise be the case. The resultant reduction in file sizes is particularly valuable when the data is being downloaded across a low bandwidth connection.
This option should be used with caution, however, as it is easy to degrade the precision of the data by choosing inappropriate conversion factors.

## 14.9. Platform Specific Tips

The steps listed above apply to maps generated for any MapLink platform. Depending on the target environment of the map there are additional steps that can be taken to improve map runtime performance.

Which set of guidelines applies depends on the MapLink Drawing Surface used by the end application. If you don't know which platform your maps will be used with, ask your application's development team.

### 14.9.1. Windows using GDI

- **Cosmetic vs Geometric pens**. Windows provides a small number of built-in line styles (the so-called cosmetic pen styles), and also allows a wide range of line styles to be defined using geometric pens. The first five line styles in the Line Rendering list in the Feature Book are the cosmetic pen styles. All other line styles are provided using geometric pen facilities. The rendering performance will be greatly improved when cosmetic pens are used, so geometric pens should be used sparingly, and avoided altogether where possible.

- **Line thicknesses**. The use of line thicknesses greater than 1 will affect the rendering performance. Of the cosmetic pens, only the solid line style can be used in conjunction with line thicknesses greater than 1. It is therefore tempting to combine the use of geometric pens with the use of thick lines. This has a very significant impact upon the performance, and the impact will increase as the line thickness increases. Rendering of this sort should therefore be used only in very limited cases.

- **Complex line styles**. Complex line styles are invaluable for providing clear natural representations of roads and similar linear features. By its nature, it inevitably makes use of line thicknesses greater than one for the standard rendering, the background rendering or both. In addition the number of lines to be rendered for the given feature class doubles or trebles depending upon whether centre rendering, background rendering or both are enabled. It is therefore sensible to limit the use of multi-part rendering and to follow the guidelines above concerning the use of geometric pens and thick lines when multi-part rendering is used.

- **Fill styles**. The standard fill styles provided by Windows are the solid, hatched and hollow patterns that comprise the first eight entries in the Polygon Rendering fill style list in the Feature Book. The other fill styles are instances of the so-called user-defined fill style that is supported by Windows. User-defined fill styles have a significant impact on rendering performance, so they should be used sparingly. Instead of using a user-defined fill style to simulate a colour that is not present in your palette, consider the use of the Pastel palette.

- **Fonts**. Limit the number of separate fonts used, as there is a cost involved in setting up each font. Each combination of text style, scale, colour and alignment constitutes a separate font. Reducing the number of separate fonts used also usually leads to clearer, more readable maps.

- **Hershey font**. If text orientation issues are causing a performance degradation, consider the use of the Hershey font. This is a single vector font that is rotated as necessary according to the text orientation. The text rotation operation has a much smaller impact on performance than does the creation of a separate font for each separate orientation. The Hershey font of course does not have the visual appeal of the other fonts.
- **Keyholing**. Use keyholing in cases where the dataset contains polygons with holes. Keyholing will always improve performance, even if relatively few holes exist, but the improvement will be greatest in the presence of large numbers of holes.
- **Custom Linestyles**. These are expensive to draw as each line segment has to be drawn separately.

### 14.9.2. X11

- **Line thicknesses**. The use of line thicknesses greater than 1 may affect the rendering performance. This is X-Server dependent.
- **Complex line styles.** Complex line styles are invaluable for providing clear natural representations of roads and similar linear features. By its nature, it inevitably makes use of line thicknesses greater than one for the standard rendering, the background rendering or both. In addition the number of lines to be rendered for the given feature class doubles or trebles depending upon whether centre rendering, background rendering or both are enabled. It is therefore sensible to limit the use of multi-part rendering and to follow the guidelines above concerning the use of geometric pens and thick lines when multi-part rendering is used.
- **Fonts**. Limit the number of separate fonts used, as there is a cost involved in setting up each font. Each combination of text style, scale, colour and alignment constitutes a separate font. Reducing the number of separate fonts used also usually leads to clearer, more readable maps.
- **Keyholing**. Use keyholing in cases where the dataset contains polygons with holes. Keyholing will always improve performance, even if relatively few holes exist, but the improvement will be greatest in the presence of large numbers of holes.
- **Raster symbols.** Avoid the use of Raster symbols as these consume large amounts of X-Server resources.
- **Raster Data**.
  - 8bit data is significantly quicker to draw and process than 24bit depths.
  - If performance is key than TIFF is faster to decode than PNG which is faster than JPEG to decode.
  - 24bit raster data cannot have the brightness, contrast or gamma changed at runtime.
- **Custom Linestyles**. These are expensive to draw as each line segment has to be drawn separately.

### 14.9.3. Embedded, Mobile and OpenGL

- **Line thicknesses**. Solid lines with a thickness of 3 pixels or greater are slightly slower to render than solid lines that are 1 or 2 pixels, therefore rendering performance can be improved by reducing the number of features using thicker lines on complex layers. Patterned line styles are not affected.

- **Complex line styles.** Complex line styles are invaluable for providing clear natural representations of roads and similar linear features. However, they effectively multiply the number of lines that need to be rendered for the map by the number of components the line style is made from. Overuse of these line styles can therefore have a negative impact on rendering performance.
- **Fonts**. Limit the number of separate fonts used, as each font used requires additional system resources to render. When using dynamically sized fonts (e.g. when a feature's text size is specified in Map Units), setting the minimum size to the smallest desired display size will reduce the amount of resources consumed by the font for text sizes too small to be readable. Unlike the other targets, using different colours, alignments and rotations for a particular font does not require additional resources or noticeably impact rendering performance.
- **Raster Nugget Size.** This option is accessible in the Map Creation tab of the MapLink Options panel, located under Options in the Tools menu. The default value of 512x512 is suitable for other platform targets, but is smaller than necessary for this target. Values of 1024x1024 will generally improve rendering performance on mobile or embedded platforms, while 2048x2048 is suitable for desktop systems.
- **Map Post Processing.** This option is accessible on the Map Post-Processing tab of the MapLink Options panel, located under Options in the Tools menu.

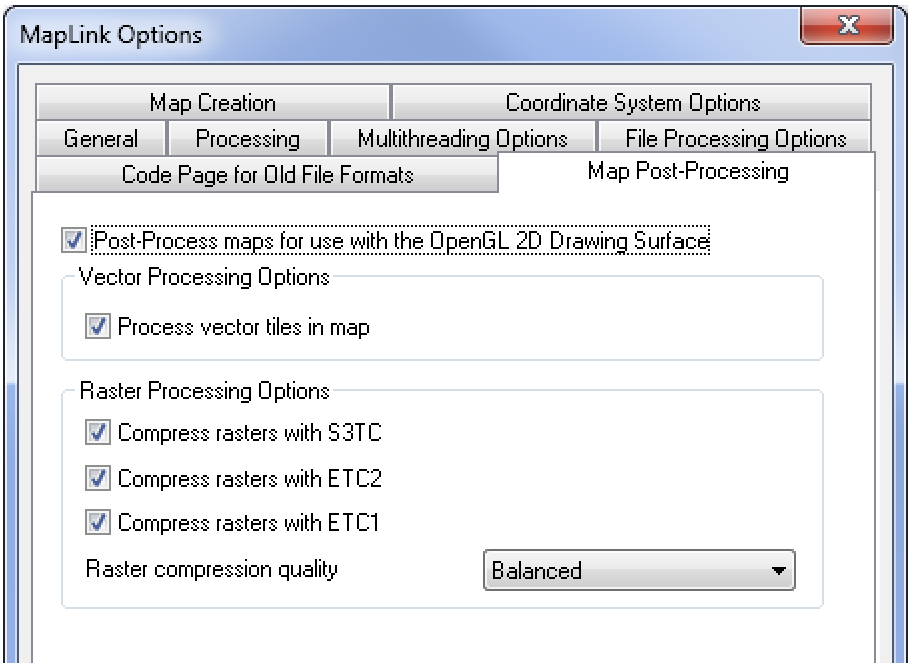



**Figure 56 - Map post processing options**

When creating maps for use with the OpenGL drawing surface, certain processing steps can be done at the time of map creation in order to reduce the amount of calculation that must be done at runtime, controlled through this dialog. Maps containing vector data intended for use with the OpenGL drawing surface should generally always have the vector processing option enabled.
Maps containing raster data should normally have one or more of the raster processing options enabled depending on how they will be used:

- Maps created for desktop systems should have the S3TC compression option enabled.
- Maps created for mobile or embedded systems should have the ETC1 compression option enabled.

- If the target device for the map is recent for either desktop, mobile or embedded devices, the ETC2 compression option can be enabled instead of either of the other two options.

If multiple formats are selected, the MapLink runtime will choose the most appropriate format to use when displaying the map based on the capabilities of the device it is running on. There is no runtime penalty for enabling multiple raster formats.

The post-processing settings are stored as part of the MapLink Studio configuration and not the project file. These settings therefore apply to all maps created with MapLink Studio.

## 14.10. Post Optimise Phase

When creating Maps you now have the option to re-process a map to optimise the contents by removing all features that are not visible and to remove any redundant group levels. This can reduce the size of a map and improve the drawing performance.

You should consider carefully if you should do this for a runtime projected map because of the removal of features which are not drawn as this may change the output extent.

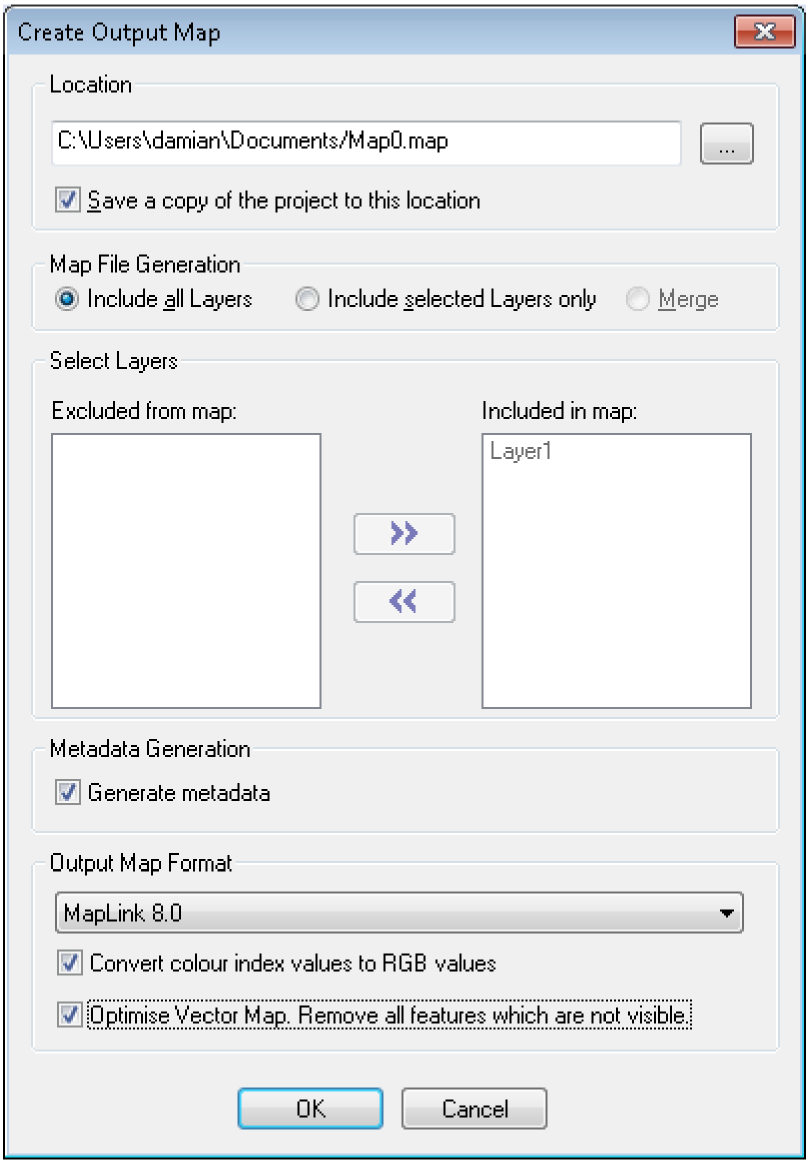



**Figure 57 – Post Optimisation Phase**