Debrief NG User Documentation

Ian Mayo

Debrief NG User Documentation

Ian Mayo

Published August 2007
Copyright © 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007 Maritime Warfare Centre Ltd. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation. A copy of the license is included in the section entitled "GNU Free Documentation License".

Contact details

Should you wish to learn more about Debrief please contact one of the following:

Debrief Project Sponsor.

Lt Cdr Andy Evans
SM OA
Maritime Warfare Centre
Marlborough Building, HMS Collingwood
Fareham
HANTS
UK
PO14 1AS
e-mail: smtac.oa@mwc.rn.mod.uk

Debrief Project Manager.

Mr Ian A Mayo
Debrief Project Manager
PlanetMayo Ltd
34 Shearwater Avenue
Fareham
HANTS
UK
PO16 8YE
e-mail: ian@planetmayo.com

Table of Contents

Introduction

1. Welcome

Welcome to the documentation for Debrief NG. Debrief has been produced by the Maritime Warfare Centre in Portsmouth UK, to support the analysis and reporting of maritime tactical exercises.

In December 2000 the Maritime Warfare Centre decided to give Debrief an Open Source status, opening the application and its source code into the public domain. Debrief has been made Open Source to facilitate its wider use, encouraging adoption of standard file formats, presentation and practices between analysis agencies.

Debrief provides the following features:

- 2-Dimensional (top-down) view of vessel tracks
- Ability for user to *step-through* exercise serials
- Full formatting of data presented on screen
- Palette of tactical, vector map and chart-related features for insertion onto *plot*
- Export of plot images for insertion into word processor
- X-Y plotting of relationships between tracks
- 3-Dimensional view of tracks using height/depth data
- Gridded database of bathymetry/elevation
- Display of time-stamped text and sensor data

2. What's new?

Adoption of Eclipse framework. 2006 has seen a change in direction for the delivery of Debrief. Instead of being a standalone application build from scratch (albeit with a couple of third-party components), Debrief NG has been rehosted into an existing framework; Eclipse [www.eclipse.org]. Adoption of Eclipse has bought the following benefits:

Functionally rich user interface	Eclipse has provided a wealth of trees, tables and slider controls to make your analysis job easier. You are now able to drag & drop any window anywhere on the screen.
Persistent window layout.	Your modified window layout is remembered when you re-open Debrief NG.
Online updating.	If you're connected to a suitable network (MWC or the Internet) Eclipse is able to download & install it's own updates. Agencies with large numbers of Eclipse users can provide their own internal update site for Eclipse installations to download updates from.[1]
Extensive guidance.	Since Debrief sits "on top" of Eclipse, the broad spectrum of Eclipse online & printed guidance is available. You rely less on Debrief-specific guidance to resolve your problems. Don't

	worry, all of the guidance is presented together and covered by a single search engine.
Modular infrastructure.	It is now easier to create Debrief plug-ins, and applications based on Debrief. Developers no longer have to learn the full depth and breadth of Debrief prior to adding new features. They just need to learn about the interfaces presented by Debrief, and how the Eclipse infrastructure works. Eclipse development is documented online and in many off-the-shelf books, and it has a strong support community

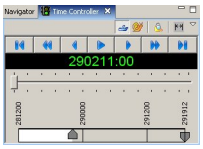
Editing multiple items. It is now easy to edit multiple items at once. As you select them on the layer manager, their common properties are shown in the properties window. Then just edit away...

Polished user interface. All of the user interface have received a refresh. The layer manager includes contextual icons giving an extra indication of what's in each layer.

Figure 1. Contextual icons

The time-controller has also received some TLC, it is now tidier, more capable, and will act as a firm foundation for more time-related operations in the future.

Figure 2. Improved time controller



Bookmarks. You can now bookmark significant events within an exercise. The list of bookmarks across all of your Debrief plots are managed from the Bookmarks dialog, and generated from the drop-down menu of the improved Time Controller.

Figure 3. Adding a bookmark

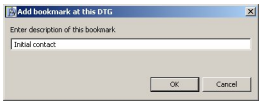
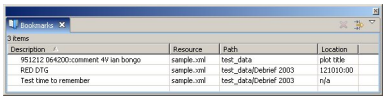


Figure 4. Managing bookmarks



3. History

3.1. Pre-history

Debrief was originally produced in 1995 in the *Maritime Warfare Centre* to act as a desktop viewer for results produced by the MWC's ASSET submarine simulator. In use it quickly became apparent

that real exercise data could also be viewed in the application removing the requirement for clerical staff to produce paper plots for use in analysis. The initial version of Debrief was a 16-bit MS Windows C++ application.

Debrief was updated in late 1996 to 32 bits, in order to exploit the richer user interface components available for 32 bit Windows applications. It was at this stage that the application was demonstrated and subsequently issued under license to COMSUBDEVRON 12 of the US Navy.

In 1999 development of an updated version of Debrief, Debrief 2000 was started. Over the previous four years a number of fresh requirements had arisen, requirements which could not be economically met using the existing architecture. Accordingly development of the Debrief 2000 application started from a fresh-whiteboard, adopting a modern modular approach to allow incremental implementation and insertion of future modules as they were required. The rapid maturity experienced by Java together with the availability of cheaply available development environments, rich application libraries (serialisation, Java3D and XML in particular) and its platform independence made Java the natural choice for the application.

During 2000 Debrief gained wider use within the Maritime Warfare Centre, being used for more varied types of submarine exercise analysis together with analysis in surface-related warfare areas.

3.2. The switch to Open Source

In Summer 2000 the Maritime Warfare Centre committed to switching the Debrief application to Open Source status. The adoption of the Open Source licensing model affects MWC in a number of ways:

1. Organisations that are currently using Debrief will have full access to the source code of Debrief, allowing them to identify and correct bugs (provided they have sufficiently trained staff). The licensing is such that these modifications can be again made public through the re-insertion into the central, online "code base".

2. Organisations that are not currently using Debrief also have full access to the application and its source code. Since the application and its supporting documentation clearly describe its origins in MWC this will spread the name of the organisation together with enforcing its reputation as a centre of maritime tactical analysis.

3. Any organisation using Debrief that identifies a bug/algorithmic problem is able to independently correct the problem and submit the corrected code back into the central "code base". In time, this will greatly increase the accuracy and reliability of the application. MWC may then freely utilise these improvements, only incurring the administrative overhead of "checking-in" code modified by third party organisations.

4. The free, open source status of the application makes it easier for third party, commercial organisations to bid for development contracts to maintain or extend Debrief. This wider availability will only bring economic advantages to MWC and fellow organisations.

5. The wider national/international use of Debrief will also lead to easier exchange of exercise data between nations (through common file formats) and potentially offer an increase in efficiency and the general quality of naval exercise analysis

3.3. Debrief 2001 onwards

At the end of 2000, Ian Mayo, the developer and project manager of Debrief, left full-time employment at the Maritime Warfare Centre to setup his own software development consultancy, PlanetMayo Ltd.

A competitive open tender process was conducted during late 2001 to supply the Maritime Warfare Centre with Debrief support. The contract was won by PlanetMayo, who grouped up

the implementation of the MWC's fresh requirements in a major update to Debrief, titled Debrief 2001. This update bought new, large areas of functionality to Debrief including vectored chart data, display of narrative text, and display of sensor-data.

Debrief development continued in 2002, with the update to Debrief 2002, which bought greatly improved 3-d plotting facilities and a number of incremental improvements to other areas of functionality.

The year 2002 also saw the first conference paper extolling the virtues of Debrief and the Open Source principles behind it, at Undersea Defence Technology 2002 in La Spetzia, Italy. The paper itself is available for viewing at the: Debrief web site [http://www.debrief.info].

Another significant step forward for Debrief in 2002 was the contribution from NUWC [x] of an algorithm and code suitable for shifting tracks. NUWC developed a set of experimental classes used to perform track-shifting. The algorithms used in these classes were taken and modified to complement the Debrief look and feel, and to provide greater usability resulting in the track-shifting editor included in Debrief 2002.

Through 2003-2007 PlanetMayo has continued to provide MWC with contracted support for Debrief. This support has included user guidance, bug fixes and addition of new features. Additionally this support contract was used to deliver Debrief 2003; (including provision of a bathymetric bottom in 3-d views, presentation of TMA data, and improved time-variable graphs), and the fundamental rebuild of Debrief into Debrief NG (providing a modular architecture ready to seamlessly accommodate ad-hoc analysis tools).

4. Debrief users

Whilst Debrief was originally developed for use within the Maritime Warfare Centre, the switch to Open Source licensing has led to its adoption by the following organisations and companies:

Table 1. List of acknowledged Debrief users

	Debrief project sponsor. Lt Cdr Andy Evans RN Maritime Warfare Centre Marlborough Building, HMS Collingwood Fareham, HANTS PO14 1AS United Kingdom <smtac.oa@mwc.rn.mod.uk>
PlanetMayo	Debrief project manager. Mr Ian Mayo Marlborough Building, HMS Collingwood Fareham, HANTS PO14 1AS United Kingdom <ian@planetmayo.com>
	COMSUBDEVRON 12 Submarine Development Squadron Twelve Naval Submarine Base New London Groton, CT, 06349-5200 USA
	Naval Undersea Warfare Center (NUWC) Roger Howlett Code 2212 Combat Systems ICE Development Team Naval Undersea Warfare Center (NUWC) 1176 Howell St Bldg 1171 CSTL Laboratory Newport, Rhode Island, USA <HowletRW@npt.nuwc.navy.mil>
	Joe Sileo COMSUBPAC Tactical Analysis Group (N7231) Commander Submarine Force, U.S. Pacific Fleet 1430 Morton St, Bldg 619 Pearl Harbor HI 96860-4664
	Jay Spry COMSEVENTHFLT/COMSUBGRU 7

If your organisation or company uses Debrief and wishes to be included as an acknowledged user please forward your details to the Debrief Project Manager as recorded above.

5. Document conventions

This document uses the following conventions

Table 2. Document conventions

Descriptions	Appearance
Warnings	 **Caution** Warnings.
Hint	 **Tip** Hint.
Notes	 **Note** Note.
Information requiring special attention	 **Warning** Warning.
File Names	`file.extension`
Label of a screen item	View Toolbar
On-screen button	Auto Generate
Name of an applications	Internet Explorer
Emphasized text	*word*
An entry in the glossary (click to view)	*Plot*
Source Example	`<para>Beginning and end of paragraph</para>`

Part 1. Getting started

This tutorial will lead you through Debrief right from the installation through to more advanced topics such as creating custom sets of layers for annotation fresh plots.

Chapter 1. First steps

1. Before installing Debrief

1.1. Obtaining Debrief

A number of project support activities (news, feature-requests, bug-reports) for Debrief are conducted online, at a web-site known as SourceForge. SourceForge is also used to store the newest versions of Debrief, ready for download.

The home page for Debrief on SourceForge is at http://sourceforge.net/projects/debrief. From this home you can read the latest news on Debrief, request new features, report bugs, and download the most recent version.

To obtain the latest version, examine the table titled "Latest File Releases", and click on the Download link next to Debrief-Installation-Set. Then click on install.exe for a windows version of Debrief, or install.bin for a unix version.

Tip

If you already have DebriefNG installed, but just want to install an update, look at the files listed in the Debrief-Updates section of the Latest File Releases. For each new release of Debrief, the significant files are located individually in the Latest File Releases. If you do not wish to download the full Debrief installation, just download these individual files and replace your existing copies with these. It is important to download and use all of the files provided to ensure compatibility.

Note

Debrief is also available in source and binary distributions from the SourceForge web-site, in the Debrief-Developer package. These distributions are suited to the software developer wishing to learn more about the structure of Debrief or wishing to extend its functionality. Once either of these distributions are downloaded, unzip them and examine the `readme.txt` file in the top level directory.

1.2. Where to get Help on Debrief

If you get stuck with Debrief, the following sources of information are available:

1. **This document.** The Debrief NG User guide is a useful reference for determining where to find something, the overall capabilities of the tool, and assorted reference guidance. It's presented online as part of Debrief NG help within a help browser (accessed via Help Contents on the Help menu.

2. **Cheat sheets.** A series of online cheat sheets is distributed with Debrief. These sheets guide the user through a series of complex tasks to achieve an overall goals such as: *loading data into Debrief* or *analysing narrative data*. The cheat sheets are accesssed by selecting Debrief Cheat Sheets from the Help menu in Debrief NG.

3. **Welcome Page.** Debrief NG's welcome page provides a high level overview of Debrief, and guides the user to further sources of information, tutorials (cheat sheets), and samples.

1.3. Debrief on CD-Rom

Note

If you are unable to download Debrief from the Internet, please contact the Project Manager, and a copy of Debrief on CD-ROM will be forwarded to you.

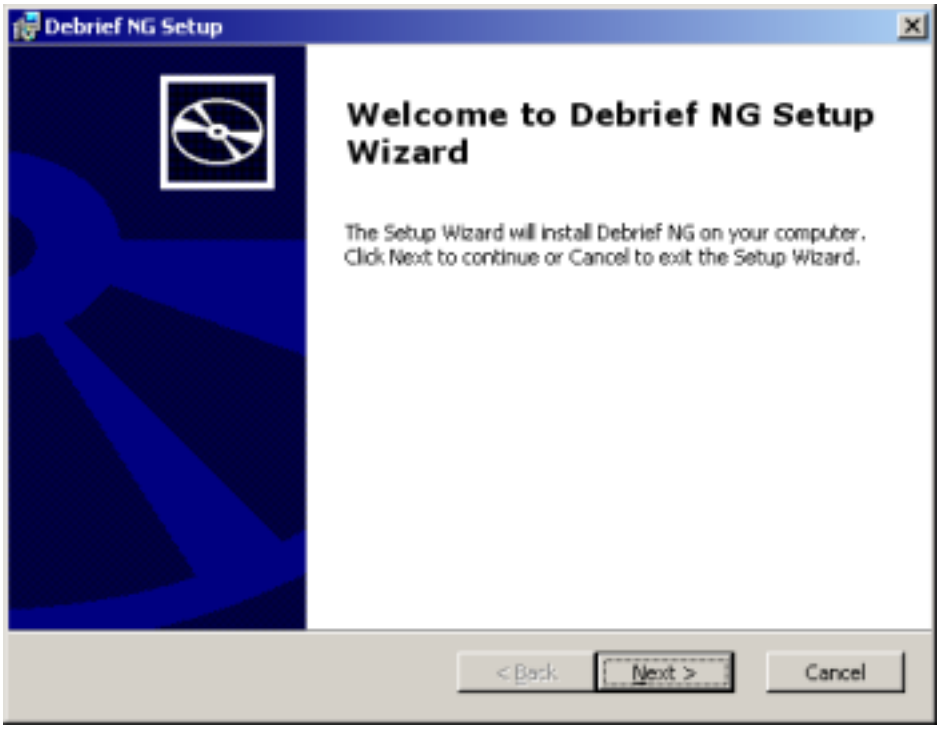
1.4. Running installation

Warning

If you already have Debrief NG installed on your machine you should remove it using the Add/Remove Programs button from the Settings button on the Windows Start menu.

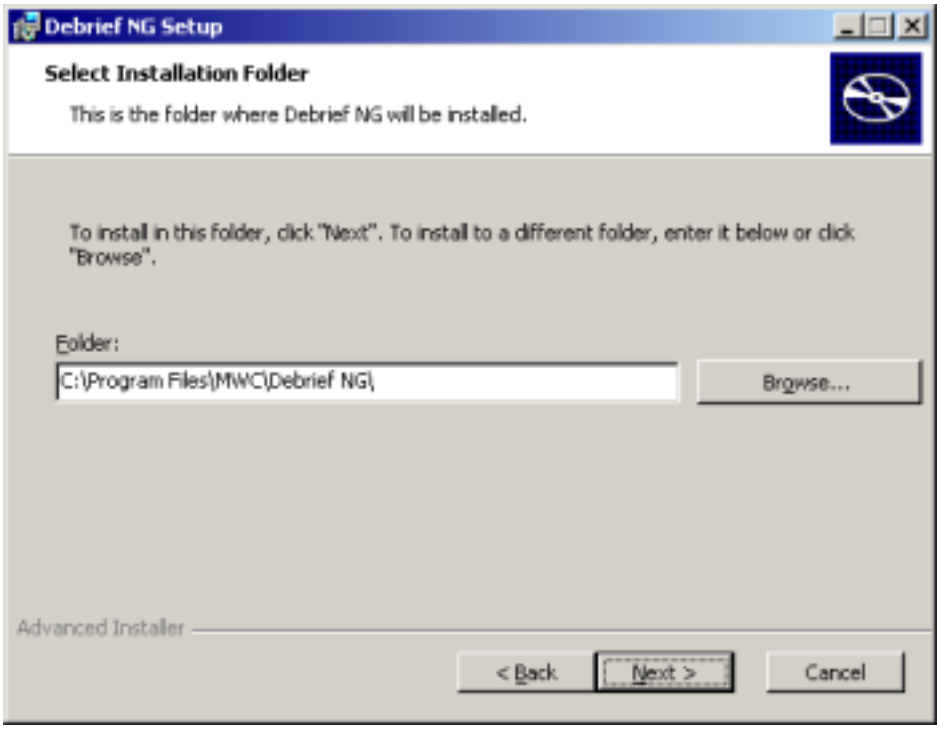
Once you have downloaded Debrief NG, double-click on `install.exe` and follow the steps provided. You are first met with the welcome screen:

Figure 1.1. Welcome view of the installer



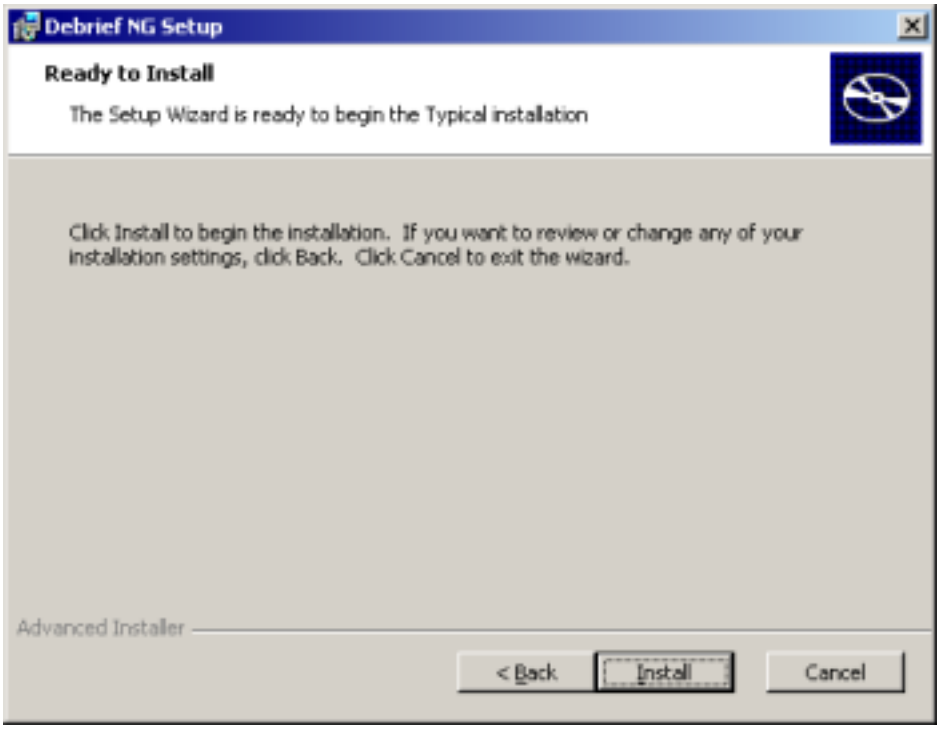
Click on the Next button to move forward through the screens. The first choice you are offered is the destination of the Debrief files. Unless you have any strong objection this default destination is normally acceptable.

Figure 1.2. Select Debrief location



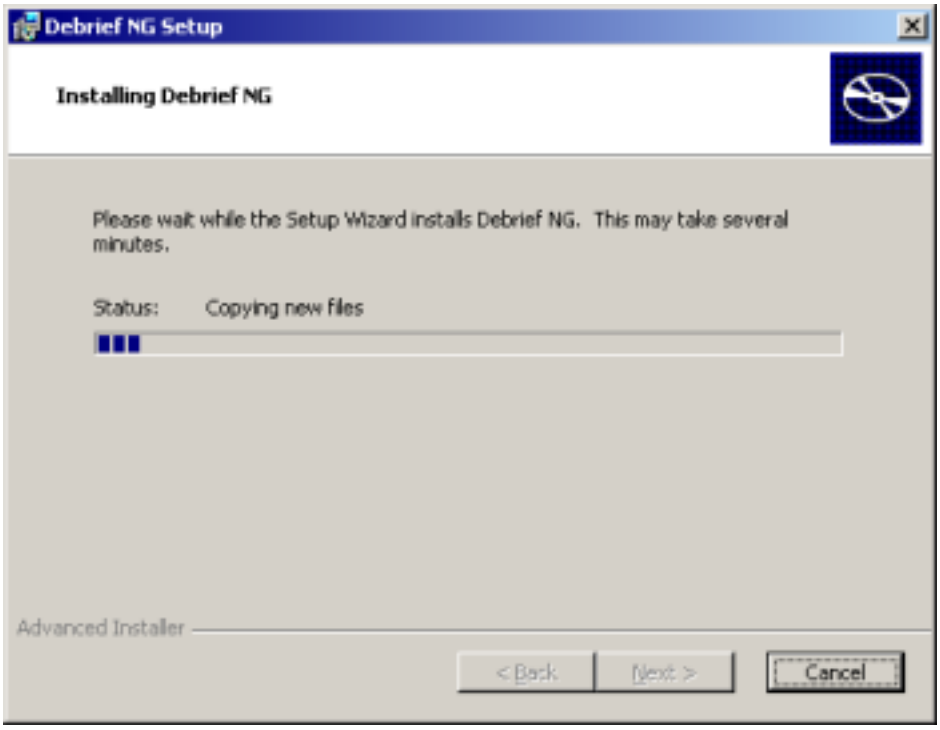
Next you will see the Pre-Installation Summary dialog which will allow you to step backwards to review installation settings prior to conducting the install.

Figure 1.3. Pre-installation summary



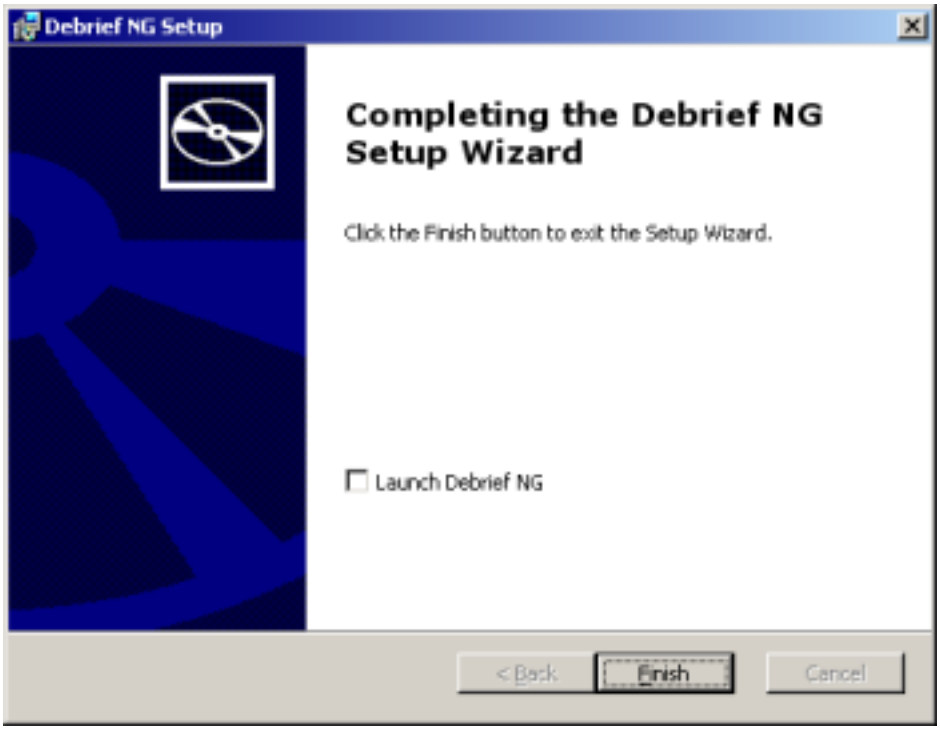
After this you will see Debrief being installed onto your machine

Figure 1.4. Debrief installation progress



Finally you will be greeted with the Installation Complete dialog informing you that everything went ok.

Figure 1.5. Debrief installation complete



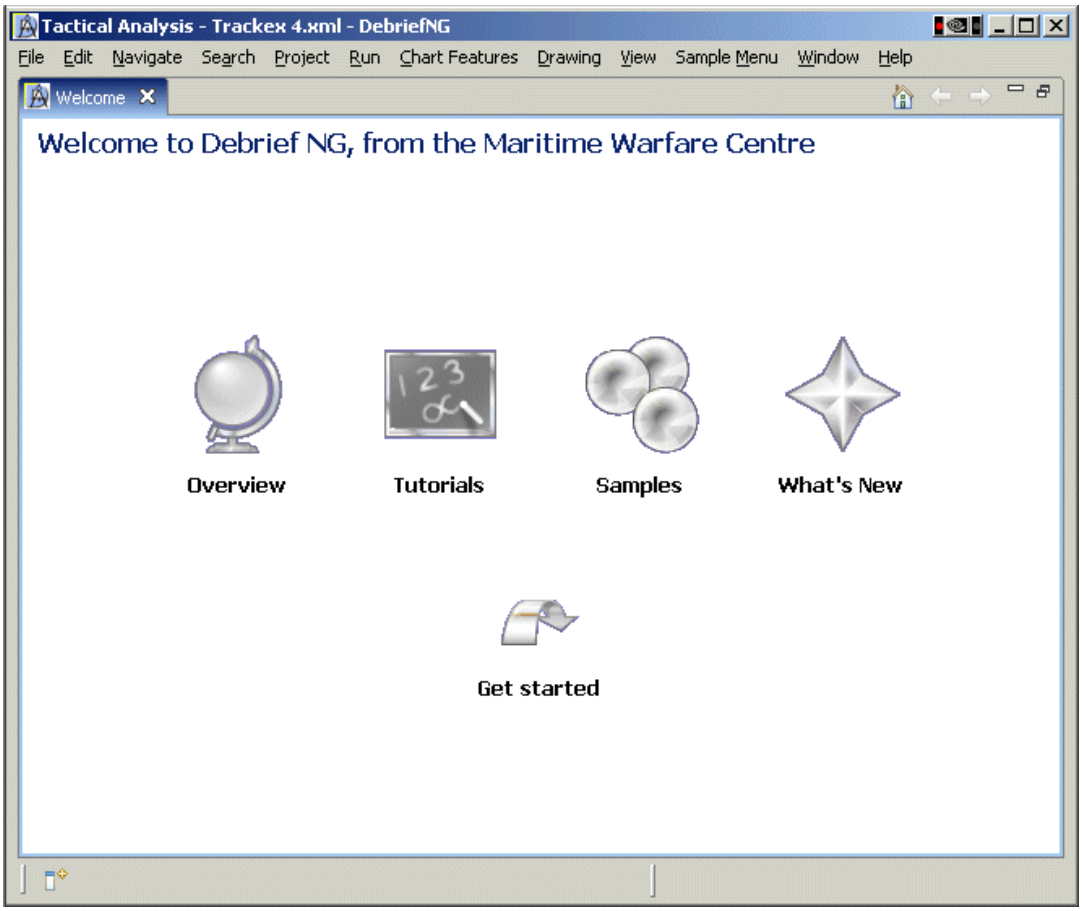
It's now time to start Debrief

2. Start Debrief

2.1. Opening Application

Open the application using the shortcut placed in the Programs section of your Start menu. The application may take a couple of seconds to load, but don't worry about the speed; once the windows are open Debrief runs quite quickly. If you are viewing this tutorial online, arrange this window and the Debrief window so that both are visible (although you may need to let them overlap if you have limited screen space).

Figure 1.6. Debrief's Welcome view

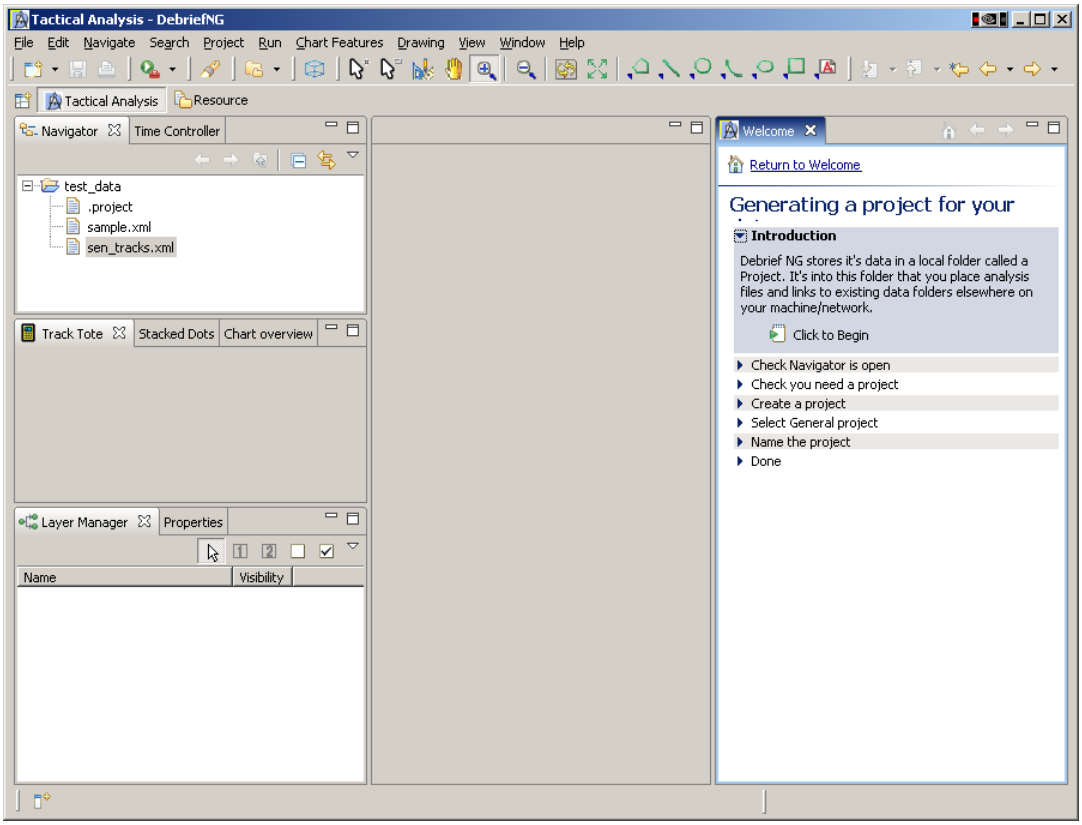


When opened for the very first time Debrief shows its Welcome page. This page (shown above) includes a series of links to high-level introductory information. The Overview page contains guidance for new-adopters of Debrief, whereas the Tutorials page contains leads users through more specific tasks through the use of *cheat-sheets* - small pages breaking a task down into a series of tasks. The buttons at the top-left of each page navigate you around the welcome guidance. On the Tutorials page one cheat-sheet in particular will help new Debrief-adopters:

Getting started with Debrief NG Walking you through loading your first data into Debrief

Starting any of the cheat-sheets will shrink the Welcome guidance to a smaller pane exposing the rest of Debrief NG as shown in the following diagram.

Figure 1.7. Debrief's default view

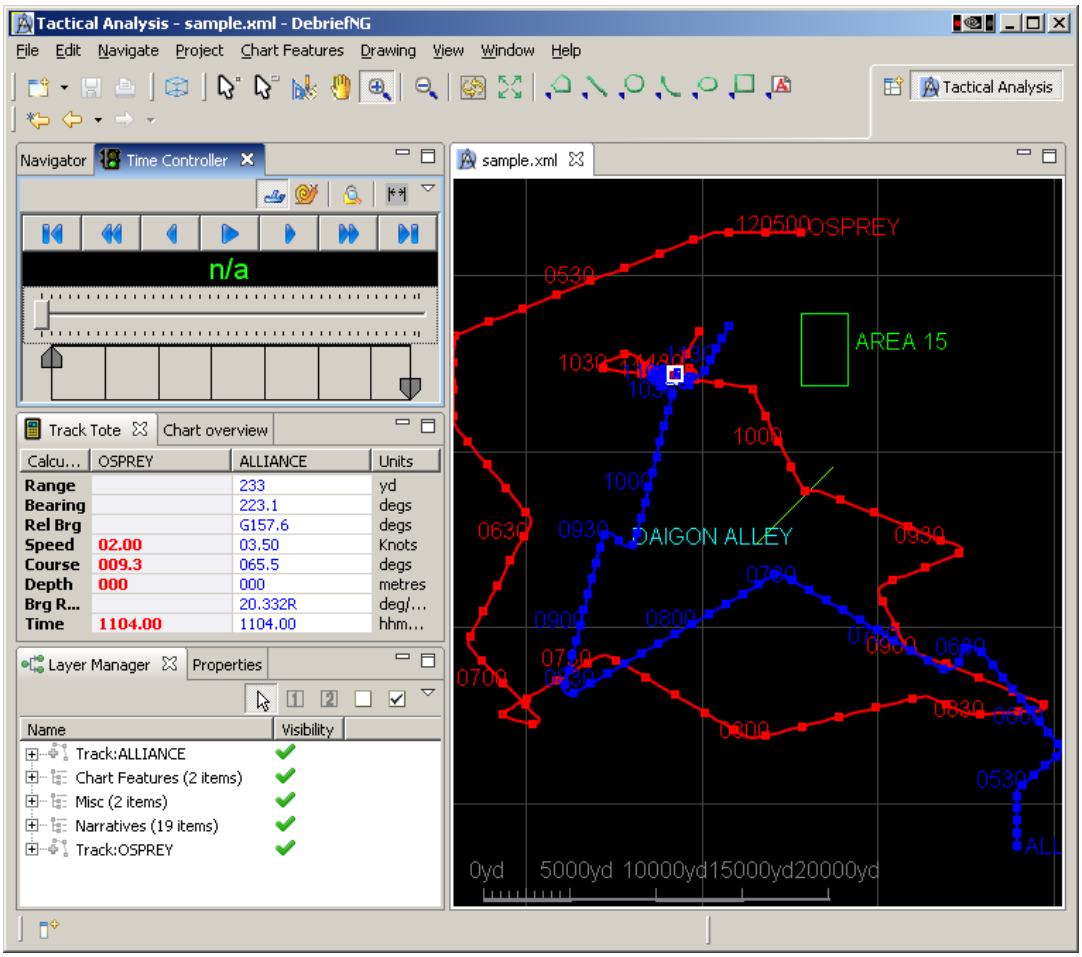


The Debrief user-interface follows the Workbench convention, whereby a users edits individual files in an *Editor* assisted by a series of supporting information panes called *Views*. The specific arrangement of editors and views offered by Default is called a *Perspective*, in this instance the *Tactical Analysis* perspective. The Tactical Analysis perspective arranges the views into sets of grouped tabs, though views can be dragged to any other location on the workbench, or floated above the workbench. Views frequently have their own toolbar and drop-down menu providing actions related to that particular task. Once selected, views are closed by clicking on the white diagonally oriented cross. To re-open a view, or open a fresh view select Show View from the Window menu. A list of Debrief-related views will be shown. Other views can be selected from the Other... menu item

2.2. Navigating around Debrief

Fundamentally, the Debrief user interface (generically termed the Workbench) is constructed from a plot editor, surrounded by a series of support panes (called views), both underpinned by a main menu and a series of toolbars. These elements are described further below.

Figure 1.8. A Debrief plot



2.2.1. Editors

Any number of editors can be open at once, but only one can be active at a time. The main menu bar and toolbar for Debrief contain operations that are applicable to the active editor. Initially the active editor will always be the Debrief plot, but the extensibility of Debrief NG means other types of editor can be implemented. Tabs at the top of the editor area indicate the names of files that are currently open for editing (`Sample.xml` in the previous screenshot). An asterisk (*) indicates that an editor has unsaved changes. By default, editors are stacked in the editor area, but you can choose to tile them (by dragging the tabbed plot-name) in order to view source files simultaneously.

2.2.2. Views

Views support editors and provide alternative presentations as well as ways to navigate the information in your Workbench. For example, the Navigator displays projects and other resources that you are working with, and the Layer Manager shows a list of selectable items within the current plot.

Views also have their own drop-down menus. To open the menu for a view, click the icon at the right end of the view's title bar. Some views also have their own toolbars. The actions represented by buttons on view toolbars only affect the items within that view.

A view might appear by itself, or stacked with other views in a tabbed notebook. You can change the layout of a perspective by opening and closing views and by docking them in different positions in the Workbench window. We've attempted to provide a logical arrangement of Debrief views, but feel free to experiment. Debrief will remember your settings, but you can return to the defaults at any point by selecting Reset Perspective from the Window menu.

2.2.3. Toolbars

There are four kinds of toolbars in Debrief.

The main toolbar, sometimes called the Workbench toolbar, is displayed at the top of the Workbench window directly beneath the menu bar. Items in the toolbar might be enabled or disabled based on the state of either the active view or editor. Sections of the main toolbar can be rearranged using the mouse.

There are also individual view toolbars, which appear in the title bar of a view. Actions in a view's toolbar apply only to the view in which they appear. Some view toolbars include a Menu button, shown as an inverted triangle, that contain actions for that view.

Whilst Debrief doesn't currently make use of Perspectives, a third type of toolbar is the perspective switcher. The perspective switcher allows quick access to perspectives that are currently open. It also has a button that can open new perspectives. The perspective switcher is normally located in the top-right, next to the main toolbar. However, it is also possible to position it below the main toolbar ("top-left"), or to position it vertically on the left-hand side of the workbench ("left"). The name of the perspectives is shown by default, but it is possible to hide the text and show only the icons. To reposition the perspective or hide the text, right-click on it and choose the appropriate item from the context menu. When Debrief contains modules for Track Reconstruction it would be understandable for a Track Reconstruction perspective to provide a suitably tailored set of views.

Finally, the fast view bar is a toolbar that contains icons representing the current set of fast views. A fast view is a shortcut to a view that is frequently used - generated by right-clicking in a view's title bar. The fast view bar appears in the bottom left corner of the workbench by default. However, it is possible to position it on the left or right as well. In all cases, you can find out what toolbar buttons do by moving your mouse pointer over the button and reading the tooltip that opens. .

Note

Perspectives. Each Workbench window contains one or more perspectives. A perspective defines the initial set and layout of views in the Workbench window. Within the window, each perspective shares the same set of editors. Each perspective provides a set of functionality aimed at accomplishing a specific type of task or works with specific types of resources. For example, the Tactical Analysis perspective combines views that you would commonly use while editing analysing tactical exercises. Perspectives control what appears in certain menus and toolbars. They define visible action sets, which you can change to customize a perspective. You can save a perspective that you build in this manner, making your own custom perspective that you can open again later. You can use the General > Perspectives preference page to open perspectives in the same window or in a new window.

2.3. Loading your first plot

See the Debrief Cheat-Sheet for guidance in creating a new Debrief plot, or loading an existing plot.

3. Importing Track Data

3.1. Introduction to Replay files

Debrief reads text files using a format named the Replay file format (*Replay* was an application used in Debrief-style tasks in Royal Navy analysis in past years).

Replay files are ASCII-files containing vessel position data in a flat-file format similar to that below:

Example 1.1. Sample Replay file contents

```
951212 050000.000 CARPET @C 12 11 10.63 N 11 41 52.37 W 269.7 2.0 0
951212 050100.000 CARPET @C 12 11 10.58 N 11 42 2.98 W 269.7 2.0 0
;NARRATIVE: 951212 095700.000 TOMATO SUSPECTED DETECTION OF RED
951212 050200.000 CARPET @C 12 11 10.51 N 11 42 14.81 W 269.9 2.0 0
951212 050300.000 CARPET @C 12 11 10.51 N 11 42 27.27 W 268.7 2.0 0
951212 050400.000 CARPET @C 12 11 10.28 N 11 42 40.33 W 270.6 2.0 0
```

Note

Since Autumn 2004 multi-word track names can now be read in if they are surrounded by quotation marks.

This data has a single vessel location per-line, with white-space separated columns containing this data:

- Date (year, month, day)
- Time (hours, minutes, seconds, decimal seconds))
- Vessel Name (single-word))
- Formatting to apply (see the maintainer documentation for details of this, but experiment with the second character to change the default colour of the *track*).)
- Latitude (deg, min, sec, hemisphere))
- Longitude (deg, min, sec, hemisphere))
- Course (degrees))
- Speed (knots))
- Depth (m))

Note the third line, which contains a *narrative* entry. This entry represents one of a group of types of data called *Annotation* Entries. These entries allow inclusion of data other than vessel locations, and can be interspersed with positional data throughout the file.

Multiple vessel tracks can be contained sequentially in a single Replay file.

For more detail regarding this file format, including how to represent annotations which are only visible for a defined time period, together with lines, ellipses and rectangles, examine the maintainers section of the this document.

3.2. Import

See the cheat-sheet covering importing existing data into Debrief.

When data is imported, Debrief applies default values to the time labels for positions. The first point in a *track*, together with the first point every day includes the day, otherwise just hours and minutes are displayed.

Of course though, you haven't learnt how to show the labels, so that's still to follow - but remember, when you switch on the labels for a *track*, it was at the import stage that the default labels were set.

4. Moving around the view

4.1. Introduction

Once data is loaded into Debrief, there are a number of ways of controlling the view of the track data using the buttons on the View toolbar.

Figure 1.9. View toolbar

Tip

For most of Debrief's operations (such as adding drawing features) it needs to have an area to look at. When re-opening an existing plot file, or importing data, Debrief can determine the area covered by the plot. However, when starting a new, blank session, Debrief does not know what area to cover. In this circumstance Debrief sets the origin to its own origin, at Fort Blockhouse, HMS Dolphin, Portsmouth, United Kingdom - the plot will be centred approximately on 50 degrees 49 minutes North, 1 degree 19 minutes West.

4.2. Mouse mode buttons

The first five buttons represent modes of use and are selectable individually, with only one selected at a time. When one does get clicked, it remains depressed, popping up the button previously in use.

 Drag Component

This setting lets you drag a single point within a large feature. For example, after switching to Drag Component mode you can drag the corners of a rectangle or a single point within a vessel track.

 Drag Whole Feature

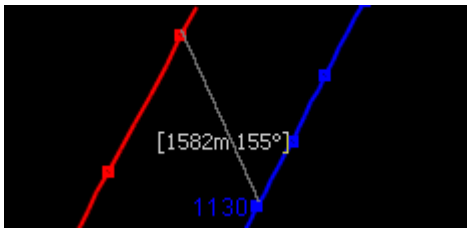
This setting lets you drag a complete feature. In Drag Whole Feature mode you can drag a rectangle around the plot, or move a complete vessel track.

 Pan

The Pan control lets you move your current viewpoint. Click on the Pan button, and then drag the mouse around the *plot* -- you will see the view change as you do it.

 Range Bearing

Whilst the Range and Bearing control does not allow you to move around the view, it does merit discussion at this point. Select the Range and Bearing control, then drag the mouse to measure the range and bearing between two points - you will see the calculated results displayed at the mid-point of the connecting, as shown below. The default units are yards, but different units may be selected from the CMAP section of the properties shown in the Window/Preferences dialog.



 Zoom in

You are in the Zoom In mode by default. So, click and drag the mouse across the area where the two *track*s are located near the centre of the *plot*. When you release the mouse button you will see a zoomed in view of the data.

Tip

In addition to the mouse mode buttons described above, the middle mouse button may be used to navigate irrespective of the current mouse mode. Hold the middle mouse

button down and drag the cursor to pan around the plot, or hold down the CTRL key and roll your mouse backwards and forwards to zoom in and out of the plot

4.3. Click buttons

The next set of buttons do not involve mouse interaction with the plot, they just require mouse clicks

 Zoom out

If you now click on the Zoom out button, you will "zoom out" of the *plot* (surprise, surprise). This button works well in conjunction with the Zoom in button, since this does not take the mouse out of "zoom in" mode -- therefore you can zoom in on areas, click zoom out a couple of times, then zoom in again, all with the minimum of mouse clicks. The wheel-mouse may also be used to zoom out as described above in ???

 Fit to Window

Whilst the meaning of the icon on this button is not all that clear (it's meant to be a combination of zoom in, zoom out, and all of the pan/move buttons), it fits the view to display all of the currently visible data. Note that there may be data on the *plot* which is not visible (hidden layers), the area covered by the hidden items of information is not included in the "Fit to window" operation. Additionally this phenomenon relates to the data once you have pressed "Filter to time period" from the Time Controller - the *plot* is resized to fit the data contained in the indicated time period (yes, you'll learn about this later in Section 2.1.8, “Filter to time period” [34]).

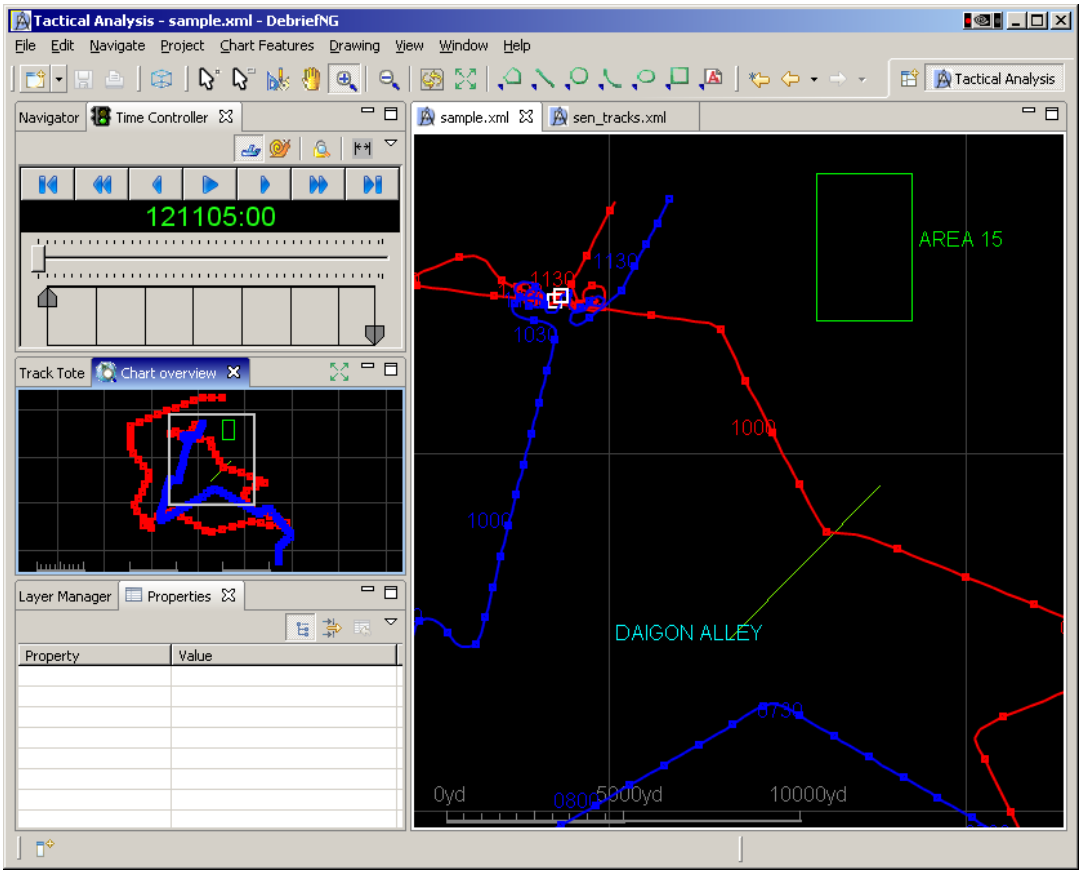
 Refresh View

This button refreshes the current view. Refreshing is occasionally required when switching between display modes

4.4. Chart Overview

The Chart Overview provides an overview of the full exercise area. This view is opened from the Window/Show View... drop down list. Using it is quite simple. Drag a region in the overview and the current plot will quickly zoom to show that area. A highlight rectangle shows the current plot view. Once you've zoomed in a little you can double-click in the overview to recenter the main plot on that point.

Figure 1.10. Overview chart in use



4.5. Undo

Debrief contains an unending list of undo-able operations, only cleared when the current session is closed. Most changes to the Debrief plot are undo-able - whether they be creating a new feature, changing an attribute of a feature (such as color), or changing the current perspective. If something doesn't undo, then the changes are we've missed it. Please report the oversight using the procedure in Section 5.1, “Reporting Bugs” [80]and we'll correct it.

4.6. Formatting the plot

The background color for the Debrief plot can be changed to suit different projection/printing mediums, or to suit the analysts personal choice. Additionally, the width at which lines of tactical objects are drawn can also be changed. Either of these options are accessed by double-clicking on the plot background. A properties panel will appear, use the drop-down list to change the background colour, or move the slider to change the line width itself.

Chapter 2. Manipulating Track Data

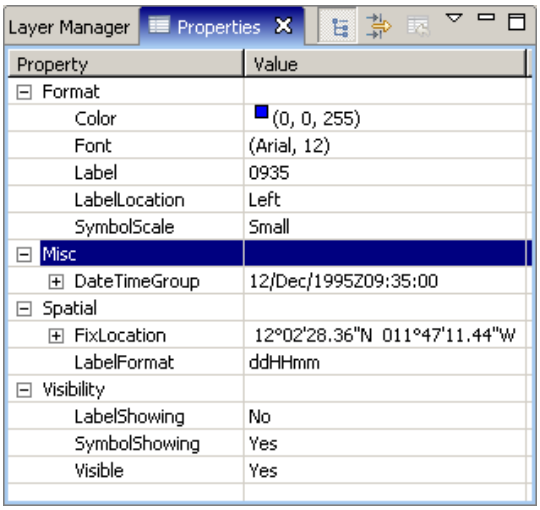
Before you are able to analyse or export plots from your data you need to know how to format what you see to improve its legibility. Learn more about formatting tracks in the *Editing your data* cheat-sheet.

Note

Apologies in advance to the British users of Debrief for the American-spelling of colour. The software environment within which Debrief is developed is American, and in this case superimposing the UK spelling upon it is not a battle worth fighting.

1. Property editing

Figure 2.1. Properties View



1.1. Edit Track

Track editing is performed on the Properties Window, and is initiated by selecting the track either from the plot or the Layer Manager, as described in the *Editing your data* cheat-sheet.

1.2. Set label and symbol display intervals

In addition to switching the symbols and labels on and off for individual positions, you can instruct Debrief to show symbols or labels at one of a series of frequencies - allowing you to quickly add 15 minute symbols and hourly time labels to a track. The timings of the particular positions that get highlighted do not have their origin at the start of the track - but from 0000 hrs - thus the 15 minute positions will be marked for 00, 15, 30, 45 minutes past the hour.

1.3. Track Shifting

The track-shifting function has been incorporated to allow users to move a track and its associated sensor data. The principle requirement for this functionality is to enable the user to move the track to a specific geographic reference point (i.e. GPS fix) or to lock the relative position of one track to another using sensor data. This positioning relative to another track is sometimes termed plot-lock or tie-point .

Note

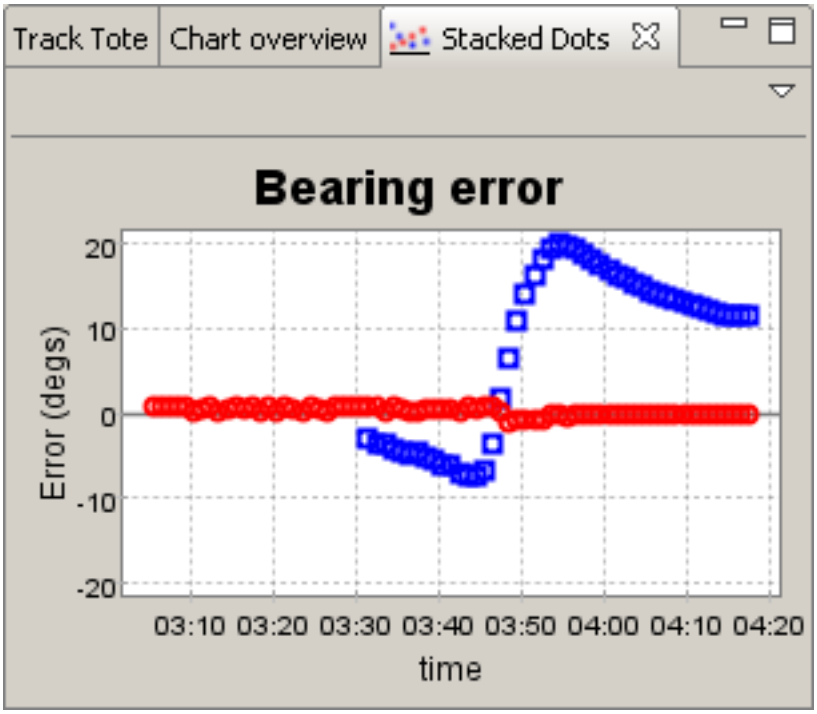
The Track-shifting implementation was originally provided by Mr David Gong at NUWC [x].

Debrief NG provides two modes of track shifting - moving individual points in a track to correct a potentially erroneous data point, or moving whole tracks to overcome navigational system inaccuracies - such as during a plot-lock or tie-in. See the two modes above in Section 4.2, “Mouse mode buttons” [11].

A Stacked Dots view is used to show bearing-errors when the track-shift is being conducted to line up one or two sets of sensor data. For each sensor bearing on a track, Debrief identifies the point on the opposing track nearest to that time. Each time the track is shifted (dragged) Debrief calculates the error between the sensor bearing and the current measured bearing, showing these in a plot (bearing error against time).

Debrief displays a symbol for each visible sensor data, with the symbol being plotted the same colour as the sensor data. When track-shifting a long exercise serial, the time-variable plot may become difficult to use due to the y-axis always auto-scaling to accommodate all available data. Overcome this by instruction Debrief to filter-to-time-period (from the time controller, Section 2.1, “The Time Controller” [29]) for the plot-lock period, thus viewing a much smaller data-set in the time-variable plot. Additionally, you can zoom in on a particular area (by dragging an area downward and to the right) of the time-variable plot for detailed analysis, dragging the mouse up and to the left to clear the zoom.

Figure 2.2. Stacked Dots View



For the track-shifting to work, your data must be configured as follows:

1. You must have a Debrief plot open (duh)

2. You must have a vessel track as the primary track [27] (Section 1.1, “Tote area” [27]) on the Tote (not an annotation or other time-related object)

3. You must have a secondary track on the Tote, but only one secondary track. This must also be a vehicle track.

4. The primary track must have sensor data.

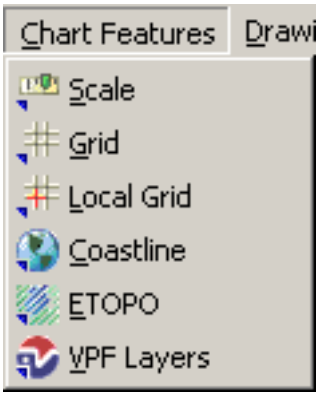
2. Adding chart features

2.1. Chart features toolbar

Debrief also allows you to add items to the *plot*.

These items are contained in two menus; Chart Features and Drawing. In Debrief, hover the mouse over them to see what type of item they create.

Figure 2.3. Chart features menu



Important

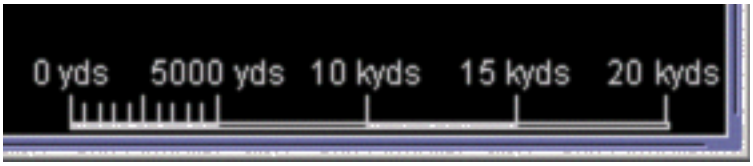
It is important to note that each time you click on an item from the toolbar, a new instance of it is created, it does not re-open an existing item.

2.2. Scale

The Scale button provides a scale, indicating to the viewer the current area of coverage of the plot. Once created the scale values can be set automatically or manually, as described below:

Auto Mode	In auto-mode Debrief assesses the current screen size and area of data covered, and attempts to set the most appropriate range of values and step size for the scale. A good working practice is to switch to auto-mode to allow Debrief to estimate the optimal values, then switch out of auto-mode to fine-tune the ScaleMax and ScaleStep values provided.
Color	The colour used to draw the scale.
Location	The corner of the plot where the scale is placed.
ScaleMax	The maximum value of the scale (in yards)
ScaleStep	The size of the steps used to break up the scale (again in yards)
Visible	You can clear the visibility flag to temporarily hide a scale, allowing you to switch between scales, for example.
Units	Use this list to select the units displayed in the scale

Figure 2.4. Sample scale



2.3. Grid

Next, try with a new grid:  .

Auto Mode	In auto-mode Debrief assesses the current screen size and area of data covered, and attempts to set the most appropriate range of values and step size for the scale. A good working practice is to switch to auto-mode to allow Debrief to estimate the optimal values, then switch out of auto-mode to fine-tune the ScaleMax and ScaleStep values provided.
Color	The colour used to draw the scale.
Delta	The interval between plotted lines
PlotLabels	Whether to label the grid. See tip below for details regarding how the labels are formatted
Visible	Whether you can hide the grid, of course.

Tip

Two methods are used to produce grid lines:

- *Lat/Long Grid*. Where angular delta units are selected (degrees, minutes), vertical and horizontal grid lines are calculated relative to the latitude of the plot (thus a 1 degree grid requested at 60 degrees North will have grid lines of 60nm separation in latitude, but with lines of longitude at 30nm separation).
- *Square Grid*. Where distance related delta units are selected (m, yd, km, nm, etc), the vertical and horizontal grid lines are constructed using the same delta distance (thus a selected delta of 1 kyd will have lines of 1 kyd separation in the horizontal and vertical).

2.4. Local Grid

The Local Grid () is a modified grid for which the origin has been over-ridden. Change the Origin attribute to move the grid origin. The PlotOrigin attribute has been provided to draw a small point at the origin of the grid - useful when initially designing/recording the grid.

2.5. Coastline

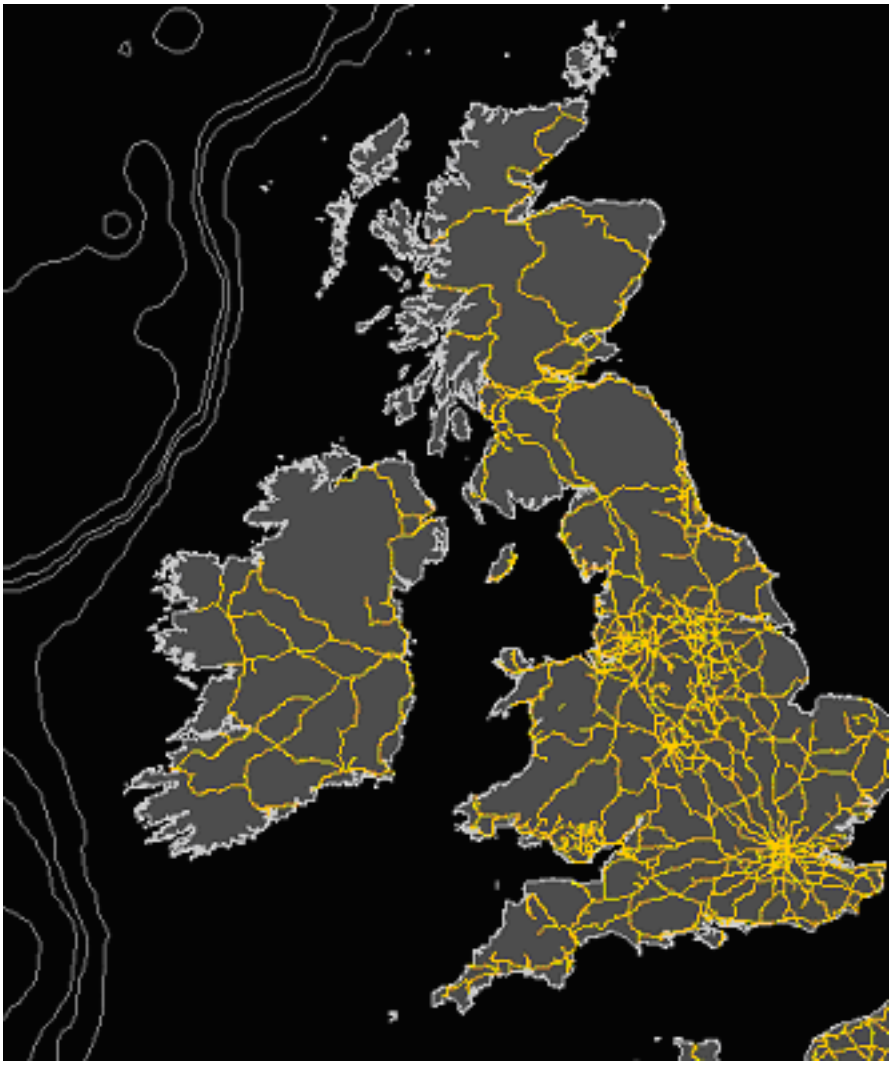
The Debrief installation includes a low-resolution coastline datafile. Whilst it does cover the whole globe, it does so at a low resolution, so is only useful for an overview. The vectored chart data discussed later provides a much lower resolution of data.

Figure 2.5. Sample of default coastline data.

2.6. Vectored data

The addition of vectored chart data is covered later in this document, in Section 2, “Viewing VPF Data” [53]. The image below provides a sample of the level of detail supplied.

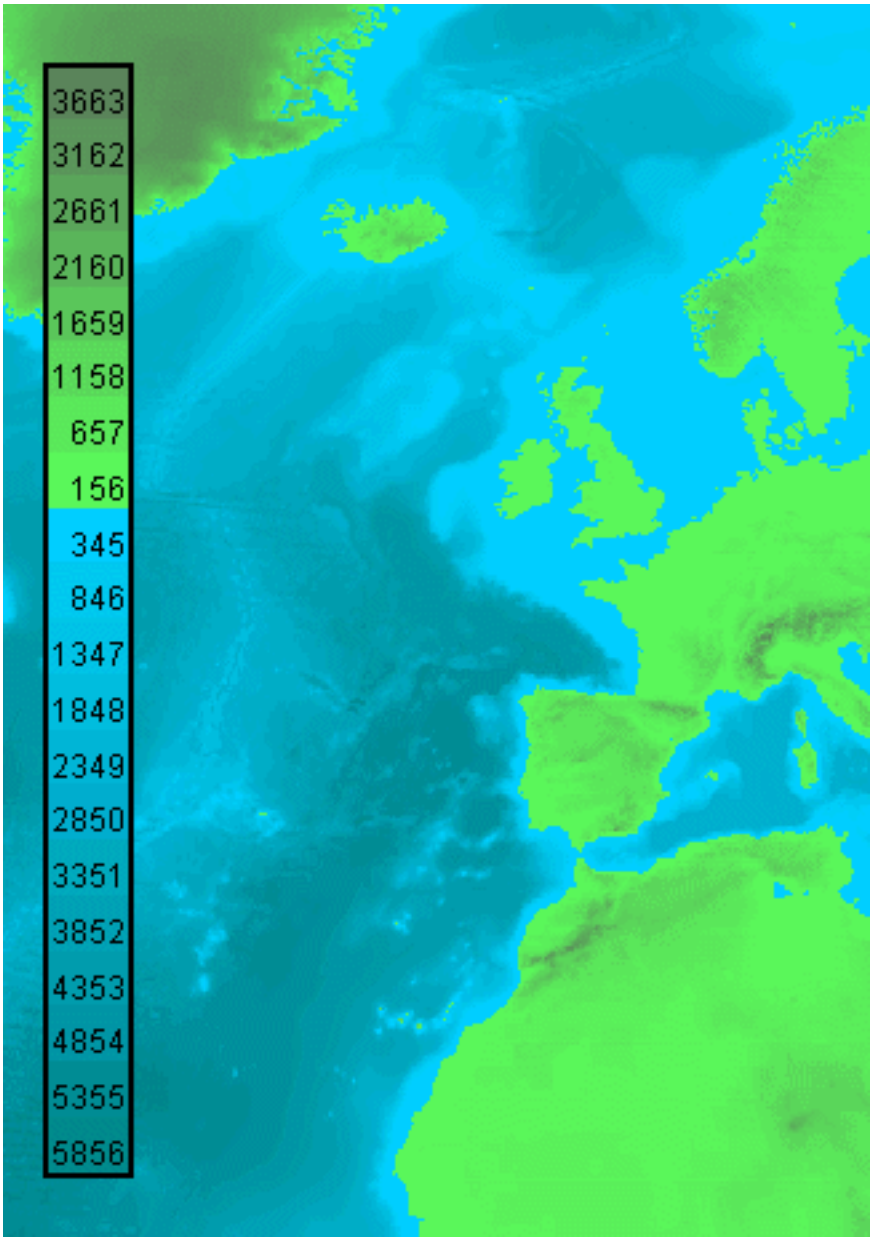
Figure 2.6. Sample of vectored coastline data.



2.7. ETOPO gridded bathy

Whilst the VPF dataset provides a contoured bathymetry within broad depth steps, the ETOPO dataset provides a gridded bathymetry in 5' or 2' steps. When you ask Debrief to plot an ETOPO background, Debrief will try to load an ETOPO-2 dataset first, followed by an ETOPO-5 dataset if that is unavailable. The image below provides a sample of the level of detail supplied.

Figure 2.7. Sample of ETOPO gridded bathymetry.



3. Adding drawing features

The Drawing toolbar and menu are used to place geographic features on the plot; features which are geographically fixed. In general, when you create them, their corners must be specified by copying a location from the plot and posting it into the property of the relevant corner.

Figure 2.8. Drawing toolbar

Note

By default, new drawing features are placed in the `Misc` layer, from where they can be moved to other layers as required. If you're creating lots of drawing features, and want to place them directly into target layers, select on Manually select target layer

from the top of the Drawing menu. When this setting is ticked, Debrief will prompt you to select a target layer for each new drawing feature added to the plot. Clear the ticked setting to return the target layer to the `Misc` layer

3.1. Label

The Label drawing item allows you to place a labelled symbol on the plot. Typically this may be used to annotate events on the plot, or to add an extra feature which did not warrant its own Debrief REP entry. Like all annotations Labels have time start and end properties. Because of this they can be placed on the Tote and used in analysis, as you will learn later.

Tip

A custom editor is supplied (see below) for editing locations of data items, it is used frequently in Debrief. Clicking on the expand button will let you edit individual lat/long fields (shown in figure Figure 2.10, “Location Editor (expanded view)” [21]). Alternatively, you may set the location to a specific point on the plot. Right-click on the plot and select Copy cursor location. Then click once on the location editor to reveal the Paste button (shown in Figure 2.11, “Location Editor (paste view)” [21]), and click it to paste the cursor location.

Figure 2.9. Location Editor (initial view)

⊟ Spatial	
⊞ Corner_TopLeft	12°08'54.15"N 011°41'53.9...

Figure 2.10. Location Editor (expanded view)

⊟ Corner_TopLeft	12°08'54.15"N 011°41' Paste
1. Lat Degrees	12
2. Lat Minutes	8
3. Lat Seconds	54.1529
4. Lat Hemisphere	N
5. Long Degrees	11
6. Long Minutes	41
7. Long Seconds	53.9131
8. Long Hemisphere	W
9. Depth	0.0 ft

Figure 2.11. Location Editor (paste view)

Corner_TopLeft	12°08'54.15"N 011°41' Paste

Note

Note that the text for the label itself can be a multi-line piece of text. When in the text-editor box for the label, just press the return key on your keyboard to move to the next line. The multi-line piece of text will be centre formatted on screen according to the Label Location property. The multi-line label is also available for all labels on the Debrief plot. Note that when the label is stored to disk in the Debrief plot-file format, the '\n' character used internally to represent the new line is converted to a '\\n' string to allow its easy storage.

3.2. Ellipse

The Ellipse shape works in much the same way as a Label, except an ellipse is drawn on the plot instead of the labelled symbol. The size of the ellipse is dictated by the Maxima and Minima values which specify the lengths of its semi-major and semi-minor axes, expressed in user-selectable distance units. These values are the distances from the centre of the ellipse to the furthest and closest points on it's perimeter, respectively. The direction of the ellipse is specified by the orientation, expressed in degrees. Debrief does not check that the maxima is larger that the minima, it merely plots an ellipse oriented about the semi-major axis.

3.3. Polygon

The Polygon drawing feature allows more varied shapes to be plotted within Debrief. A series of points (called a Path) are added to a Shape which are the connected up to create a Polygon. The points in the path may be typed in, double-clicked, or dragged to produce the correct polygon.

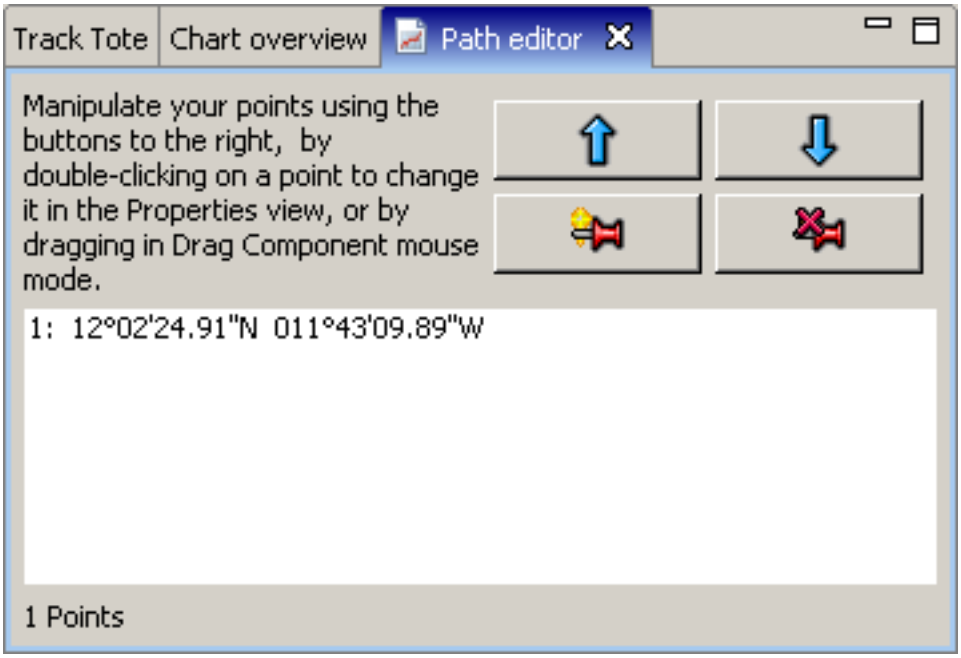
When a new polygon is created, its editor panel will open as shown below:

Figure 2.12. Initial view of polygon

Property	Value
⊟ Format	
Color	□(250, 250, 250)
Font	(Arial, 12)
Label	New polygon
LabelColor	□(250, 250, 250)
LabelLocation	Left
⊟ Misc	
Filled	No
Points	1 Points
⊟ Time-Related	
⊞ Time_Start	unset
⊞ TimeEnd	unset
⊟ Visibility	
LabelVisible	Yes
Visible	Yes

The properties shown are similar to those shown for most other shapes, with the exception of the actual location of the polygon. Instead of being able to directly edit the points in the polygon, an indication of the number of points is provided, together with a button which will take you to the Path editor. The Path editor is shown below:

Figure 2.13. Path editor



When it opens, the Path editor contains a single point (incidentally, this point it is placed a the centre of the current dataset). The four buttons at the top-right of the panel allow you to move the points up and down, create a new point, and delete a point.

To edit a point, first single-click on it, it will then be shown in the lower panel. You may then click on the Edit or Select Point buttons to either type-in the location of the point or to double-click the new location for the point. Double-click on a point to edit it's individual attributes in the Properties View, or right-click on a point to paste the location currently on the clipboard.

3.4. Rectangle, Circle, Line, Arc

The Rectangle, Circle and Line shapes work in the same way as the others described here, the Location Editor described above being used to set the corners, centre, or ends as required.

Note

Some of the more basic drawing items have a single DTG parameter instead of TimeStart and Time_End properties. Where there is a single time, the time-analytical features of Debrief (such as Snail mode) treat the feature as *alive* for three minutes either side of the time value.

Tip

To draw a line created from an origin with values of range and bearing, first create the line, and put the start point of the line at the origin. Now switch the mouse mode to Range/Bearing (). Now click on the button labelled Select Point for LineStart, and drag the mouse out from the origin point. The current range and bearing will be displayed at the bottom-left of the screen. When the mouse is at the desired range and bearing right-click and copy the position onto the clipboard. Next, paste that location into the LineEnd value.

3.5. Buoyfield

Debrief provides a *builder* dialog to assist in the creation of buoyfields. The creation of *buoyfields* is covered later in Handling Buoyfields.

3.6. General

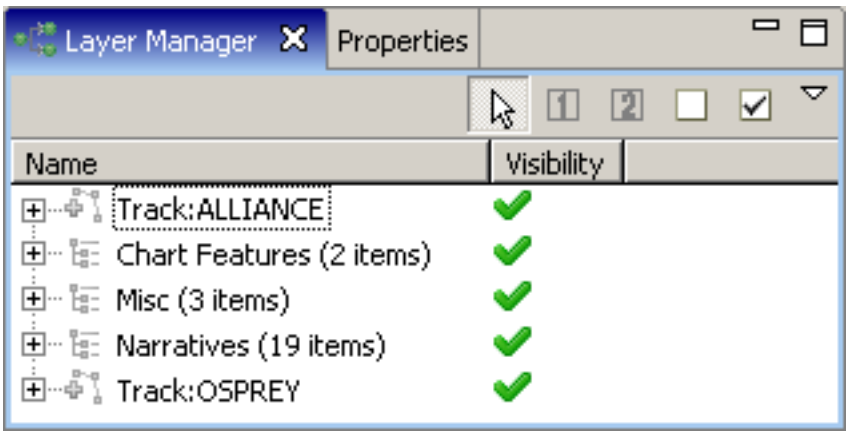
To edit any existing annotation, double on it on the *plot* or the Layer Manager to select it - it's attributes will then be available from the Properties View. Alternatively, right-click on an item on the Plot or Layer Manager, and editable attributes are available from the object's drop-down list. Only boolean (true/false) and list-related (top/bottom/left/right etc) are editable from the right-click menu.

4. Layer management

4.1. Layer Manager

The Layer Manager provides a tree-oriented view of all of the editable data within a plot, organised into layers. New items added to the plot are placed into a *Misc* layer. From here they can be reorganised into suitably themed layers via cut/paste.

Figure 2.14. Layer Manager View



Reveal the Layer Manager by selecting it from the Window/Show View... drop-down list. Once open you will see the data shown as a series of layers - each with a tick indicating if it's visible or not. A toolbar provides command buttons to make items visible or not visible, and buttons are enabled to make an entity the primary or secondary track when applicable. Note that not just tracks may be denoted primary or secondary: and graphic item with a position and date-time can be selected: thus Debrief can be configured to show a running indication of bearing and range for a series of vessels from a single label datum (representing a sonar-buoy or mine).

The drop-down menu provides further commands, significantly including the ability to add a new layer; Create Layer. Further commands are available when right-clicking on one or more Layer Manager elements: in particular the ability to cut/copy/paste elements between layers and plots, together with the command to view a time-variable graph of the selected elements (see Section 5, “Show time-related variables” [40]).

4.2. Cut/Copy Paste

Items may be cut, copied and pasted between Layers using the commands contained in the menus obtained by right-clicking on the item. The cut command may be used on it's own to delete an item. Copying items is a quick way of making duplicates of a correctly formatted annotation or screen item such as a rectangle or ellipse.

In addition to copying/moving items between layers, items and layers themselves can be copied or moved between sessions within Debrief.

To paste a layer (or track) into the top level of the layer manager just click on the white space below the displayed layers and select Paste Item form the popup menu.

Note

All top-level layers (that it screen items which appear at the top level in Layer Manager, such as Tracks, ETOPO data and Layers themselves) can have a line width assigned to them. This setting is observed when plotting all items in that layer. The smallest line width supported is *hairwidth*, which plots the finest line the particular output device (screen or printer) can produce. The line-width parameter can be accessed by right-clicking on the screen item directly in addition to via the Layer Manager.

5. Saving & re-opening plots

5.1. Save

You may (or my not) know what the *Replay* file format used to load data into Debrief looked like. Whilst it is compact and easy to use, unfortunately the Replay file format does not allow us to store all of the formatting we have applied to the Debrief *plot*, nor the coordinates of what you are currently viewing, nor how you have arranged to store your data in *layers*. This problem is overcome by the use of the Debrief *Plot-File* format, an application defined using XML. Have a quick read about *XML* in the Reference Guide or Glossary if you're not aware of it and are sufficiently interested, otherwise, here's an overview:

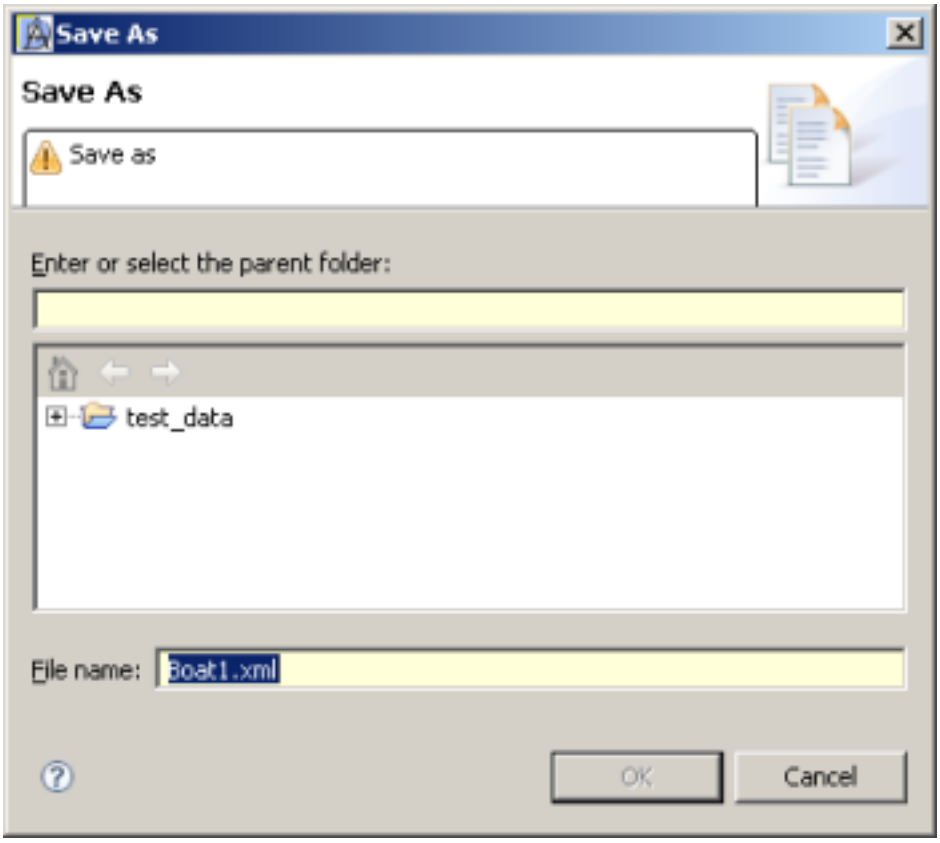
- Debrief stores plots using XML
- XML stands for eXstensible Markup Language, which allows structured data to be stored in text format, and is a world-wide standard promoted by http://www.w3c.org.
- XML files can be edited outside Debrief, allowing you to cut and paste between *plot-files* to build up a new *Plot-File*, all in a text editor

Whilst a plot is typically created from an REP file, it may not be saved back into that REP file - it must be saved as an XML Plot-File. To save your work as an *XML* file, click on the Save button:  from the Workbench toolbar or the File menu. A file dialog will open (see below), allowing you to save the current session as a Debrief *Plot-File* (`xml`).

Note

Debrief will only let you save the new plot into one of your project folders, so you must both select a folder and a provide a filename for the new file. Usefully, Debrief guesses that you probably want to name the plot the same as the REP file first loaded. Hey, it's only a guess.

Figure 2.15. Save-file dialog



Save-as functionality is provided through the Save As button, next to the Save button in the File menu.

Debrief plot files can grow very large, and on occasion the memory required for the save operation can reach the upper limit provided to the application by default. When Debrief fails during the Save operation due to reaching the upper memory limit it stops trying to save to file and shows a message dialog advising you to increase the limit:“Ran out of memory whilst saving plot, try adding -Xmx256m to the command line”. This extra parameter passed when Debrief is starting instructs your PC to provide Debrief with 256 Megabytes of memory.
If you continue to receive the error message try increasing the memory allocation to 512 Megabytes.

The save operation itself is conducted in a two-stage process. If the operation entails saving over an existing file (when you just do a plain Save, or Save As over-writing an older version) Debrief first writes the plot to a subtlely different filename in the same directory (`plot.xml` would get saved to `~plot.tmp`). If the save operation completes successfully then the older file is deleted and the temporary file renamed to the originally requested filename. Thus, your existing file only gets overwritten on successful completion of the save.

5.2. Adding more data to a plot

With a plot already open, you are able to drag a Debrief `XML` or `REP` file into it's chart, adding the content to the existing session. Dragging multiple files onto the plot area adds them all to the existing session. To create a composite plot like this, you can start by using Debrief's *New Plot Wizard* (see relevant Cheat Sheet)

Chapter 3. Analysing Data

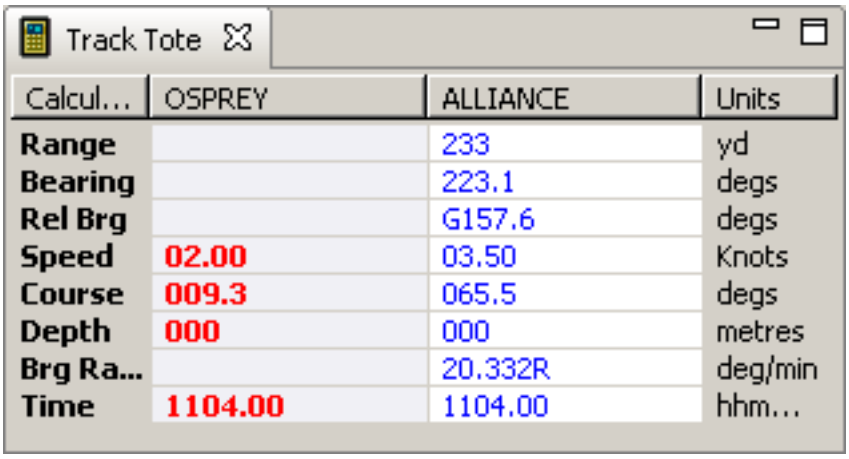
Analysis is the core function of Debrief. In this section (at last) you are going to gain some familiarity with how Debrief can be used to analyse maritime exercises.

1. Assigning tracks as primary and secondary

1.1. Tote area

We'll have a brief explanation of the Tote at this point, using the image below.

Figure 3.1. The Debrief Tote



The *Tote* area is used to show the current and relative status between two or more platform tracks. The information on the *Tote* is dynamic, showing vessel information at the time indicated in the Stepper Control.

Note

The primary and secondary tracks may not contain values exactly at the time in the time stepper control. For both the tote display and the plot-highlights, Debrief uses the data values recorded on or immediately after the indicated stepper time (see the Time [28] for more detail).

One *track* currently loaded is assigned as the Primary track (the blue track in this instance), and then one or more tracks are assigned as Secondary tracks. The primary track always displays absolute data such as current course, speed and depth, as illustrated above. The secondary tracks display this data together with relative data such as Range and Bearing to the Primary track.

Not only tracks can be placed on the Tote. Buoyfields and most annotations (labels, circles, etc) can also be set as primary or secondary track. If no time is available for an annotation it is deemed to always be valid, and calculations are shown although the time field is displayed as "n/a". However, if the annotation has start and end times the tote displays "n/a" when outside that period and calculated data when inside it.

Note

It may be useful to remember that the Primary Track is usually assigned to the Target, thereby allowing a constant display of target bearing and range.

There can be any number of secondary tracks. Debrief displays calculated results of the relationship between each one and the Primary Track. To see the relationship between a pair of secondary tracks, one of them must be set to the Primary track.

The following calculated data is presented, where the current point on each track is used for the calculation:

Range	The range between the current point on the secondary track and the current point on the primary track using the *earth model* as described in the Glossary, displayed in the units stored in the Debrief preferences window. Debrief NG introduced the option of showing *slant range*, accessible from Debrief's Preferences window. If you have indicated that you want to view slant range, then the range shown is a function of horizontal range and vertical separation.
Bearing	The bearing between the points.
Rel Bearing	The relative bearing between the heading of the secondary track, and the bearing to the primary track. There are two formats used for relative bearing, depending on the setting of *Relative bearing format* in the Window/Preferences dialog. If the format is specified as UK, the R and G characters at the start of the result are short for Red and Green, which indicate that the contact is to the Port or Starboard of the secondary track. If the format is specified as US, the value uses 0 as directly ahead and continues clockwise through 180 (astern) and back round to 359.9.
Speed	The current speed of the indicated track (no actual calculation is performed here; the value from the data-file is displayed directly).
Course	The current course of the indicated track (no actual calculation is performed here; the value from the data-file is displayed directly).
Depth	The current depth of the indicated track (no actual calculation is performed here; the value from the data-file is displayed directly).
Brg Rate	The instantaneous bearing rate of the primary track as observed from the secondary track. This value is explained in the Glossary.
Time	Where track data is not recorded at regular steps, and tracks do not have data at the same time steps, there is a likelihood that the value displayed on the tote will not have been recorded at the current tote time. The time field shows the actual time at which the data value was recorded for that track.

Note

When more than two tracks are loaded, the value n/a is shown in Tote calculations for the primary track which rely on other track data (range, bearing, rel bearing, brg rate). This is because it is unclear which inter-track relationship is being calculated. Where only two tracks are loaded (one primary and one secondary), the tote is able to show relative calculated data for both.

1.2. Assigning Tracks

Assigning primary and secondary tracks is covered in the respective *Cheat Sheet*.

Tip

If you have more than two tracks, it can be quicker to assign the primary track manually and then press Auto Populate to assign the remaining tracks as secondary.

1.3. Notes

Warning

A Circle currently only has a single "Centre" DTG value - so it will only be highlighted for 3 minutes either side of this point. Better results are obtained by using

a Label (which has a picture of a text label on it), since Labels have start and finish DTGs.

Note

It is not just tracks that can be added to the Tote, experiment with right-clicking on features on the plot and see if they have the "Set Primary Track" or "Set Secondary Track" commands available (although read the note below about these). In this way you can make a timed data point (represented by a Circle with a very small radius) the primary track, then add a number of vessel tracks as secondary tracks, and then as you move through the data you can constantly see the vessel range and bearings from this data point. This is particularly useful for seeing vessel ranges and bearings from a *sensor* such as a sonar buoy. Remember to set the DTG data for the data point to time(s) near those of the track - or else in your example Debrief will assume the "sonar buoy" is not yet active.

Note

The limits on the time period are the outer time limits of the visible data in the primary and secondary tracks currently displayed on the Tote, so following a filter operation (introduced later) the start/end times will be changed to reflect the time limits specified.

Note

The keyboard can also be used for moving backwards and forwards, although Debrief has to know that it's the Stepper that you want the keys to control. So, after clicking on the time-slider control you can then switch to keyboard control of the Tote as follows:

- Page Up/Down keys: these control small step backward/forward resp.
- Page Up/Down arrows: these control large step backward/forward resp.
- Home/End keys: these control goto First/Last resp.

Tip

In Autumn 2005 the Interpolate Points property was added to tracks. When ticked, Debrief interpolates positions between actual data points. The highlight cursor changes appearance when on an interpolated data point.

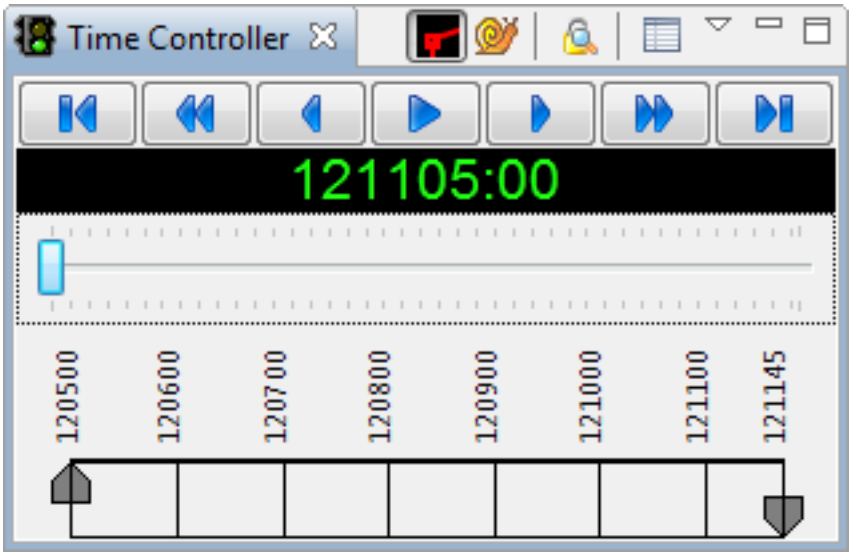
Figure 3.2. Display of an interpolated point

2. Controlling Time

2.1. The Time Controller

Time is managed within Debrief through the Time Controller. This view displays the current serial time, allows control of that time, and provides access to a series of time-related functions in Debrief.

Figure 3.3. Time Controller View

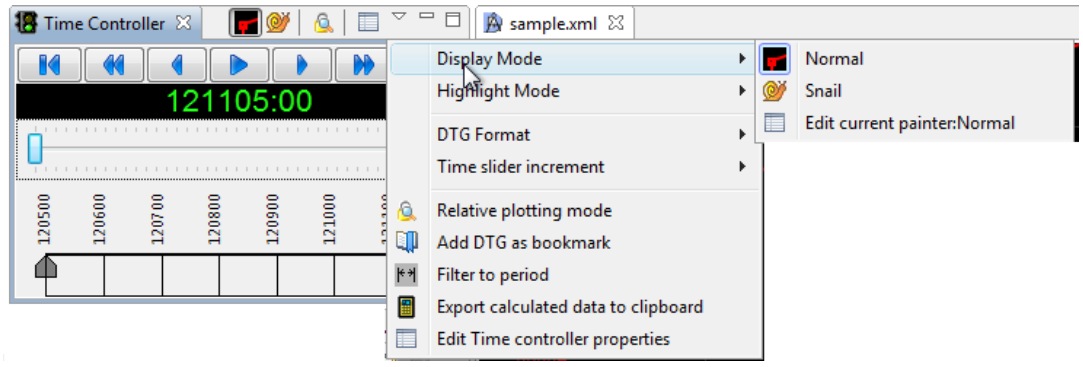


Your temporal (time-related) view of track data is dependent on three settings:

Display Mode	How the track is displayed
Highlight Mode	How the current position is displayed
Absolute/Relative mode	Whether the data is plotted North-oriented, or using the current heading of the primary track

You access these modes using the buttons on the Time Controller toolbar and the Time Controller drop-down menu.

Figure 3.4. Time Controller menu



As you can see, the first two items on the menu allow you to select the current Display/Highlight modes. In each sub-menu is a command to edit the currently mode. Later menu options allow you to format how information is displayed on the menu, and perform other time-related activities.

2.1.1. Display modes

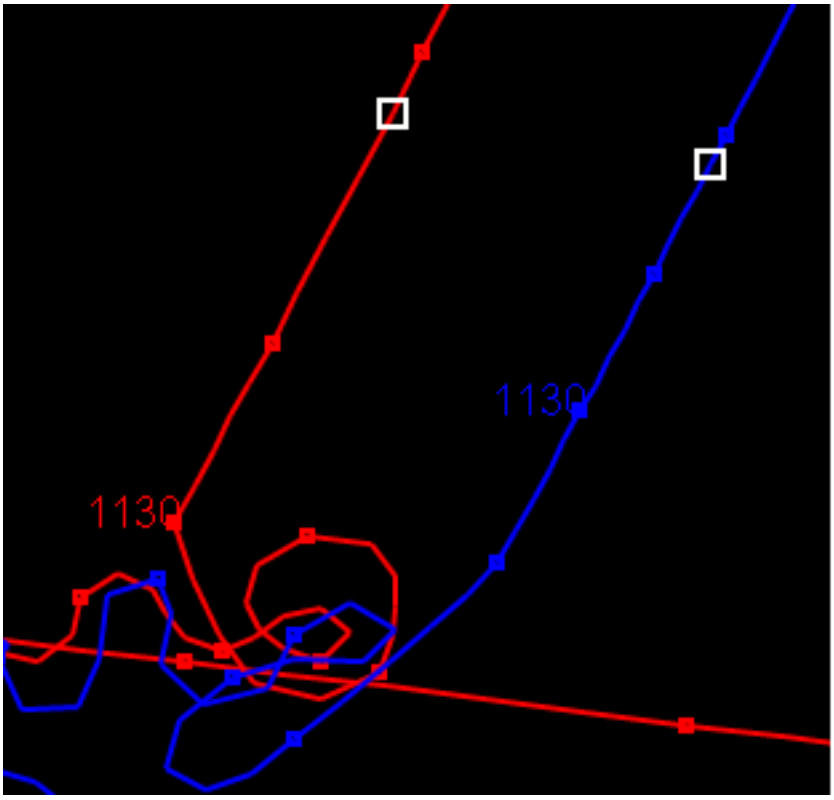
The first three icons on the Time Controller toolbar allow you to choose two combinations of plotting modes. The first two control how data is displayed: in Normal Mode, all exercise data is displayed, whereas in Snail mode, only the current position and a series of recent points are displayed (similar to a Snail with trail following behind it).

As you'd imagine, Normal Mode is the mode that is used for most analysis tasks. It's North oriented and shows all relevant data. It's also quite simple, only having two properties, both of which control the presentation of the highlight:

Color Yes, the Color of the highlight

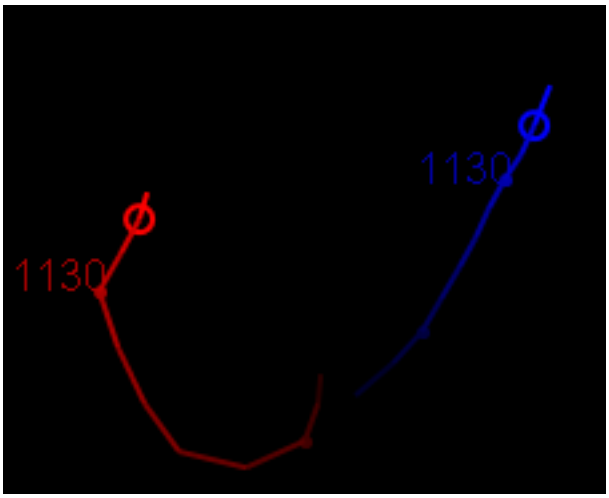
Size I know, I know, it's the size of the rectangle used to plot the highlight (measured in pixels)

Figure 3.5. Example of a normal trail



The Snail Trail is used for specific analysis tasks when you need to concentrate on the specific activities around a certain time, without the clutter of the remaining track data.

Figure 3.6. Example of a snail trail



The circle represents the current position, the stalk direction represents the current course, and its length gives a relative idea of the vessel speed (when compared to the length of the other vessel's

stalk, *boys will be boys*). The dots trailing back from the current position are a *snail trail* of points going back in time. If you move forward and backward with the stepper control you will see these trails moving. The following properties are editable for a snail trail:

Snail trail properties

Fade Points	this will cause the points in the trail to fade away to the background colour
Link positions	this will plot a line between the points in the trail
Plot Track Name	this will plot the track name alongside the current position
Point Size	this will change the size of the points together with the thickness of the lines drawn on the plot,
Trail Length	this will change the time period covered by the trail
Vector Stretch	this will change the *amplification* applied to the speed when drawing the speed vector; very fast vessels (or weapons) will need the this stretch reduced to allow stalks of sensible length.

2.1.2. Highlight Modes

Three highlight modes are provided:

Default Highlight	Shows a rectangle at the current position. From the default highlight properties you are able to select the colour and size of the rectangle to plot
Symbol Highlight	Shows the symbol (Section 1.1, “Introduction” [49]) currently assigned in the properties for each track. From the symbol highlight properties you are able to select the size at which the symbols should be plotted.
Range Ring Highlight	This mode shows a series of rings around the current location. The editable properties are listed below.

For range rings you are able to edit the following properties:

Arcs	The angle of arc that the spokes extend out to. The angle is measured either side of current heading.
Color	The colour to plot the range rings
Num Rings	How many range rings to plot, uniformly distributed from the center to the outer radius
Radius	The radius of the outermost range ring
Spoke separation	The angular separation of the spokes plotted within the range rings, centered on current heading.

2.1.3. Relative/Absolute Modes

Relative Mode affects the orientation of the plot. The plot is always oriented along the heading of the primary track, with the current primary track location placed at the centre of the plot.

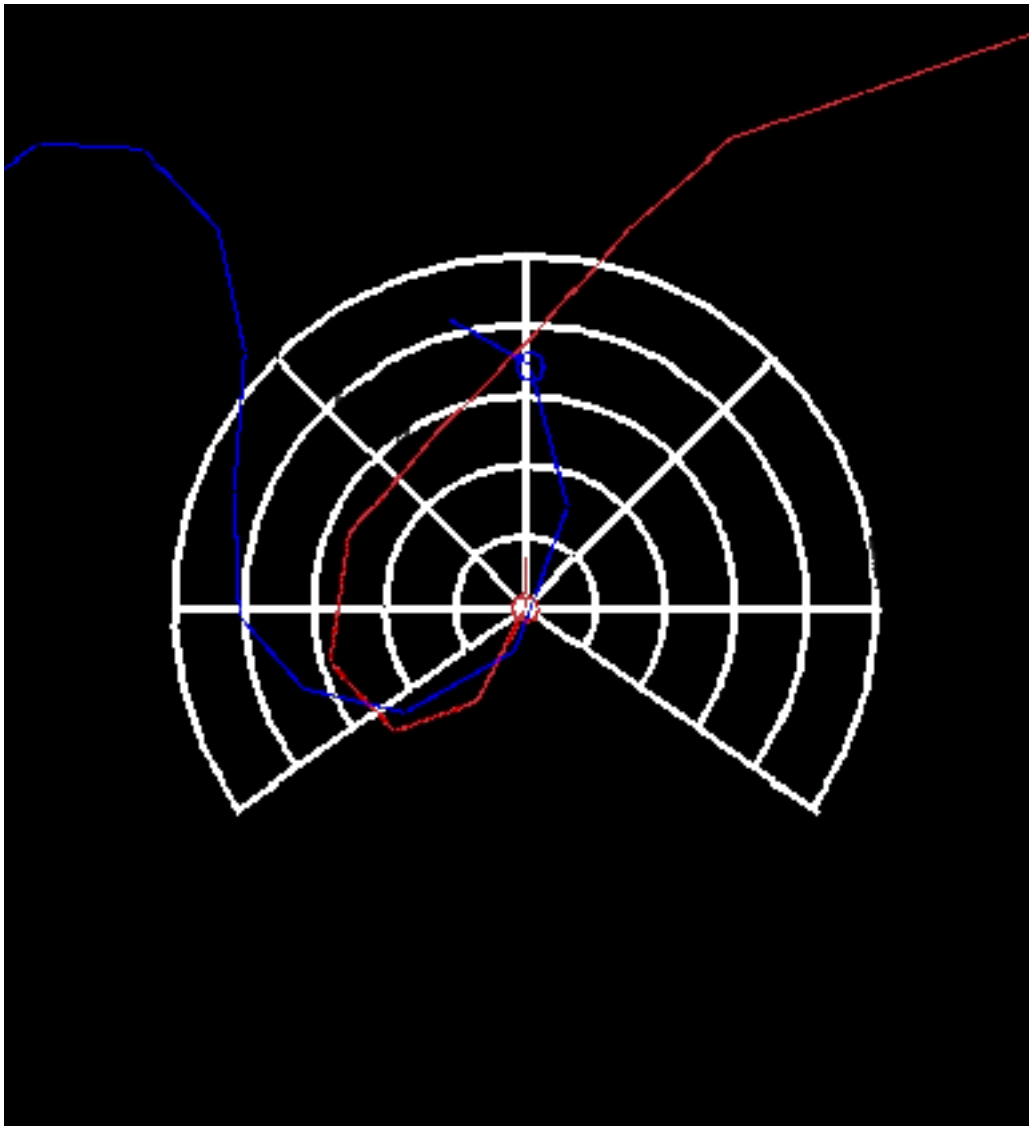
Tip

Relative Display Mode is particularly useful for analysing one vessel trailing another. Make the trailing vessel the primary track, and the vessel being trailed the secondary

track. As you step forward through the serial you will clearly be able to see the relative bearing of the contact as held by the trailing vessel.

The sample shown below gives a demonstration of the use of this relative mode. You can quickly see that the blue vessel is directly ahead of the red trailing vessel, and your use of the Range Ring Highlighter gives us a quick indication of range.

Figure 3.7. Sample of relative display mode



2.1.4. Time display

The green digits of the time display are tied closely to the slider-bar immediately beneath them. Dragging the slider controls the current display time together with how the current data is displayed. Other Debrief views (such as the Narrative Viewer and the Time-Variable graphs) update in response to time changes from the time slider.

A range of display formats are provided to make the displayed time more consistent with that in a supporting document, or of sufficient fidelity to support the current analysis.

2.1.5. Time slider

Beneath the time-display is the time-slider, used to quickly move through a time-period. By default the slider is of infinite resolution, stopping exactly on the second/millisecond proportionate to the

position of the slider. Debrief can be configured such that the slider stops on higher resolutions by selecting the relevant increment from the Time slider increment list in the Time Controller's drop-down menu.

2.1.6. Bookmarking

Debrief NG introduces the concept of *Bookmarks*. These represent the combination of a DTG, a remark, and the name of a plot-file, and are displayed in the Bookmarks view. With the view open you can quickly move between significant events across a number of files. Bookmarks are added by clicking on the  Add DTG as bookmark command from the Time Controller's drop-down menu. The bookmarks view will not automatically open, but the bookmarks themselves will be present when it is. The current DTG is used as a default remark - but you'll get most mileage our of the bookmarks by describing the event that you're bookmarking.

2.1.7. Manipulating according to time period

The pair of sliders at the foot of the Time Controller view allow you to set start and finish times. These times are not set in support of a single Debrief operation, but are used across a range of operations. When dragging the sliders, hold down the shift-key to move in whole segments (hours, days, as appropriate). Drag the shaded section to retain the period length but change its origin (again using the shift-key if appropriate).

2.1.8. Filter to time period

When the  Filter to time period radio button is depressed, changing the time of the start or end time-sliders will automatically filter the displayed data to the indicated period. In this mode, you can select a 6-hour period (for example), and drag it through the full serial time with shift-key depressed to view a moving "window" of data. In addition to filtering the visible data to the indicated period, the period covered by the time-slider is also reduced. Drag out the start/end time-sliders to return to the original time period.

2.1.9. Copy to clipboard

Another operation that relies on the selected time period is exporting calculated data to the clipboard. This operation is available form the  Time Controller's drop-down menu, and it performs a series of calculations for each data-point in the indicated time period. These calculations are then placed on the system clipboard in Comma-Separated-Variable format for reuse in other applications, Microsoft Excel, for instance.[1]

[1] A header line is written first, indicating the contents of each column:

- Track Time(hhmmss)
- Depth(metres)
- Speed(Knots)
- Course(degs)
- Range(yards)
- Bearing(degs)
- Rel Bearing(degs - using `Relative bearing format` specified in the Window/Preferencesdialog)
- Brg Rate(deg/min)
- Color (for this track point)
- Name
- PrimaryName

The results from the primary track are listed first, which (as in the *Tote*) do not show results of calculated operations:

```
CARPET 12/Dec/95 05:00:00 000 02.00 269.7 n/a n/a n/a n/a 0500 0500
```

Then the secondary tracks are listed:

```
 BUNKUM 12/Dec/95 05:50:00 000 00.00 000.0 12381 311.0 R49.0 R0.264 F5 0550
```

Tip

The Colour parameter shows the colour of the track point used in that calculation. On occasion analysts colour a track according to whether that participant is in contact or not. Exporting the colour flag to Excel allows the post-analysis data to be filtered according to periods in contact - or any other time-dependent aspect specified by the analyst. The application of a particular colour to sections of track is performed within the Layer Manager.

Note

The last two columns output give identifier information: the first of these is the name of the current item in this *track*, and the second is the point in the primary *track* nearest to the current time stamp: that-is the point in the primary track which has been used for the calculated results.

2.1.10. VCR controls

The VCR controls allow you to move forwards and backwards in time through the plot. Looking at the order of buttons in the Time Controller screenshot above, the commands allow you to move to the beginning, move a large step backwards, move a small step backwards and repeat the last time step continuously (small or large step, backwards or forwards). The remaining buttons repeat these operations in the "forward" direction. The size of small and large steps is controlled by the Time Controller properties window, accessed by selecting Properties/Time Controller. Also available from this set of properties is the Step Interval; the time interval that Debrief waits before automatically moving forwards.

2.1.11. Other time operations

Beyond the operations available from the Time Controller, the time-period is used to support other Debrief operations. The most significant of these operations is when producing time variable plots (see Section 5, “Show time-related variables” [40]). The current time period settings dictate the extent of what information is calculated for these plots.

3. Measuring range and bearing

3.1. Range & Bearing

It's worth reminding you at this point about the Range Bearing measuring tool  which is frequently useful in analysis. The calculated range and bearing is displayed at the mid-point of the line and at the foot of the DebriefNG screen (where it remains until you make another measurement).

The default units for the range displayed are configured using the CMAP section of the Preferences dialog in the Window menu. The preferences page can also be accessed by double-clicking on the range/bearing slave display at the foot of the screen.

3.2. Earth Model

The Range Bearing calculation is performed using the algorithms in the current Earth Model, as described later in Section 1, “Range/Bearing calculations within Debrief” [94].

The Earth Model used by the application is modular and interchangeable. In the initial release of the application the calculations use the Rhumb-Line Formulae for Short-distance sailing, as described in “Admiralty Manual of Navigation, Volume 2, 1973”. Short-distance sailing is defined as “the following of a rhumb-line track for a distance not greater than 600' ”.

4. Viewing tracks in 3-dimensions

4.1. 3D View

Lastly for the "viewing of plots", Debrief provides support for viewing track files in 3D. To obtain a 3-d view, the tracks currently loaded must contain Depth information, and the correct libraries (Java3D) must be installed on the machine.

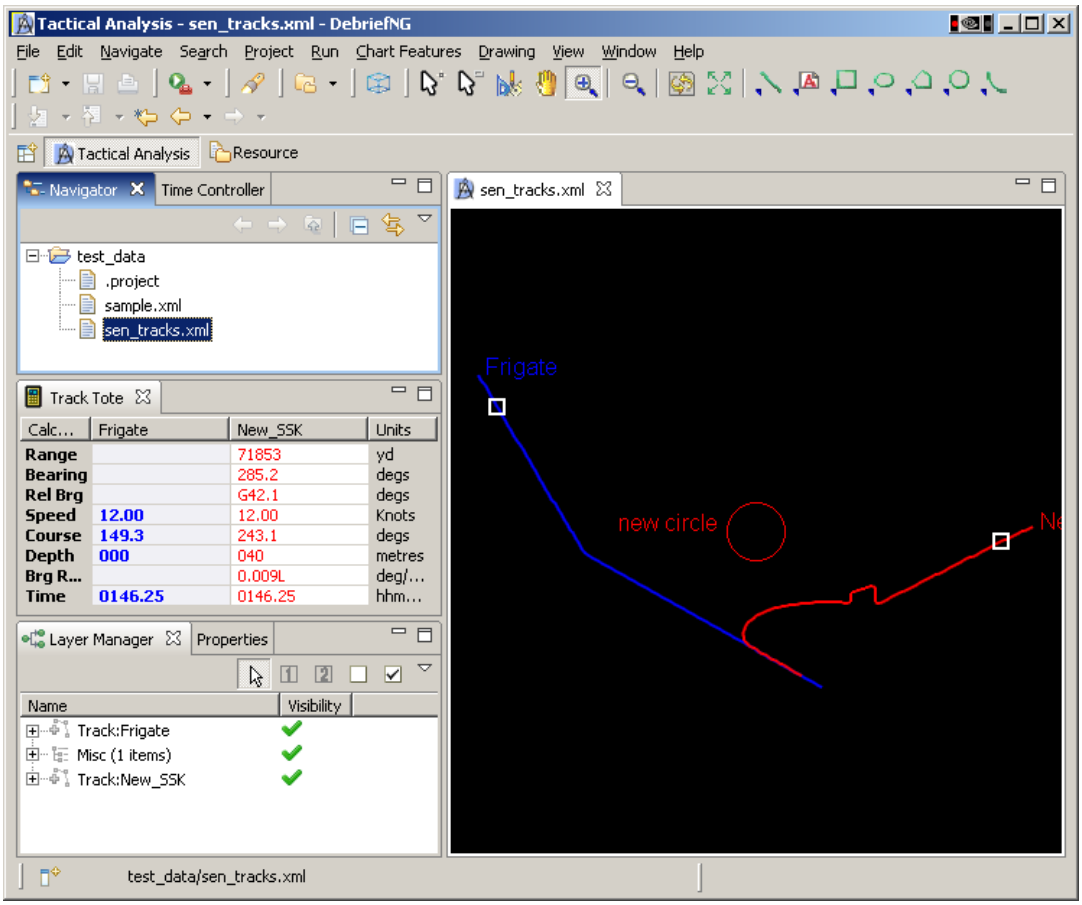
Note

Debrief contains a set of 3-d models for vessels most frequently analysed at the *MWC*. If you require a model not currently available, contact the Debrief Project Manager, and it will be added to the queue. If you already hold a copy of the model (as a `.wrl` file in VRML format), e-mail it in, which will speed up its inclusion. If you select a symbol-type for which there isn't currently a model, Debrief uses a correctly-oriented cone shape to represent the current location and course.

Unfortunately there is not any 3-d data in the sample plots we've been using so far, but another sample file does. Close the current session, open a new, blank, one, and load the data-file named `sen_tracks.xml`. This is a datafile produced using *ASSET*, a simulator under development at PlanetMayo.

Once the tracks are loaded, you will see the view as below:

Figure 3.8. View with tracks loaded



If you don't have primary and secondary tracks on the tote, initialise them now (or do it automatically, if you know how). Debrief now knows the time period covered by the tracks. The

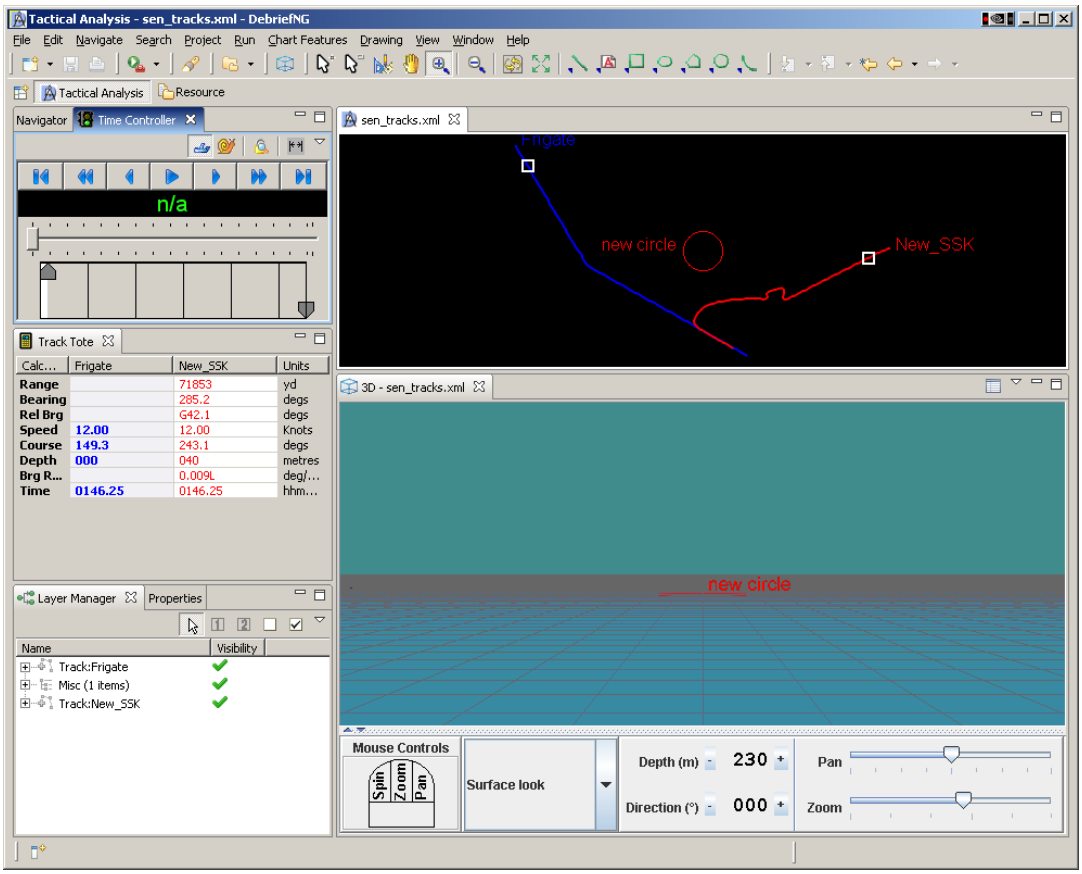
updated 3D viewer in Debrief now shows 3-dimensional models of the exercise participants. The model used is taken from the SymbolType property of each track (although the default type is Submarine). So before opening the 3-d view, change the SymbolType of the *Frigate* track to Frigate. To do this, right-click on the track and select Frigate from the SymbolType drop-down list.

Next press the "View in 3D" button  to open the 3-d view window.

4.2. Initial View

A new window will open, containing a 3D view of the current track data, as shown below (after a little screen-tidying):

Figure 3.9. Initial 3D View



Tip

Not all objects viewable on the normal 2-d Debrief plot are applicable in 3-dimensions, but most are. If one is absent which you think would usefully be shown in 3-d then please approach the Debrief Project Manager. Additionally it is not always clear whether a 2-d item (such as a labelled annotation) should be visible in 3-d. Such an instance is a labelled annotation added to the 2-d plot to help the readability of a plot in a report (for example a label placed on a sea-mount). This labelled annotation may just clutter the 3-d plot, so Debrief only shows labelled annotations when their symbol is visible; change the symbol of a labelled annotation to not visible, and it will disappear from the 3-d plot.

4.3. Controlling the view

The view is controlled with the mouse using the following buttons:

Table 3.1. Mouse controls used in 3-D View

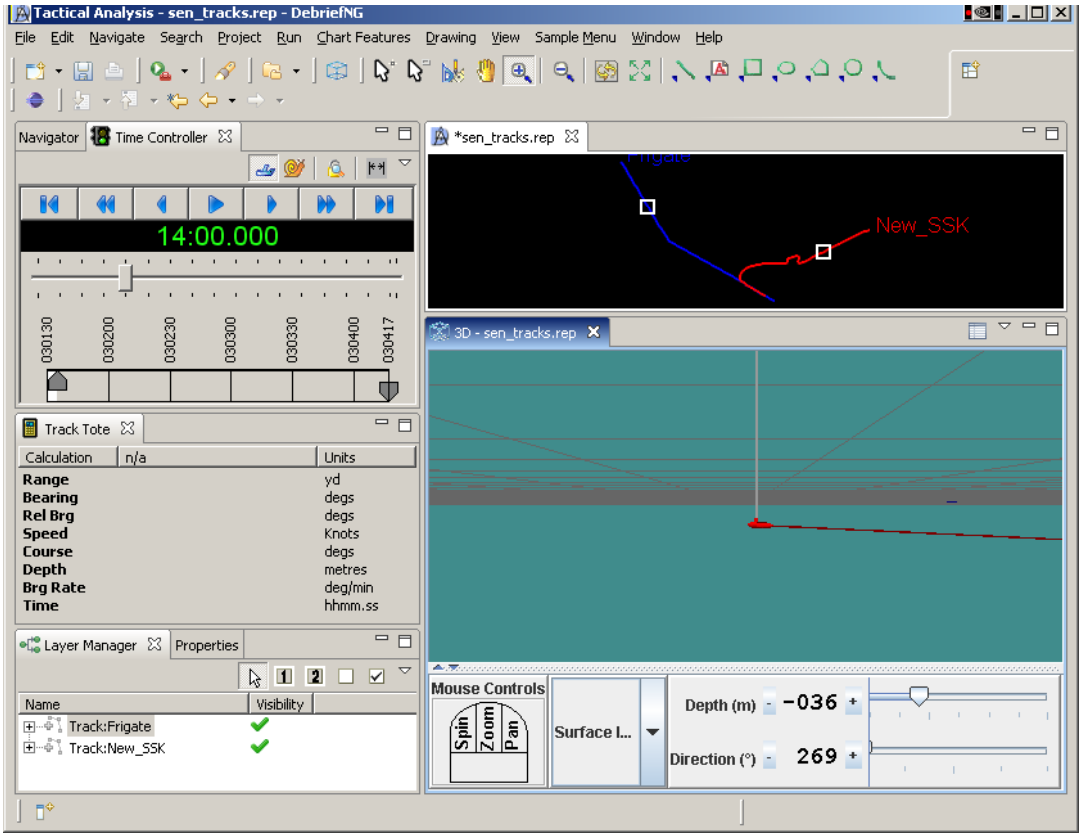
Button	Operation
Left	Rotate plot, effectively changing the current heading
Middle (or Alt--Left)	Zoom in and out of the view
Right	Pan the plot around the view

In addition to mouse-movement, controls have been added to the view to adjust the current view depth, heading, pan and zoom. Experiment with moving around the plot using these controls aswell as the mouse.

The drop-down list provided in the centre of the window provides a list of preset viewpoints. The Surface Look and Underwater Look viewpoints zoom out to the south of the centre of the plot, positioning the viewer either slightly above, or slightly below the water surface. The Top-Down viewpoint provides a god's-eye view of the data. Lastly are a series of viewpoints which each provide a "chase-plane" view of a particular participant. Thus it is possible to position yourself immediately above and behind a participant as it moves forward through an exercise serial.

Now, to return to the original view, select Surface Look from the drop-down list. Next rotate the heading to 270 degrees, take the depth down to -010, and zoom in to view the exercise from the stern of the submarine.

Figure 3.10. Adjusted 3-d view



A set of properties are also provided from the 3d view toolbar to support the 3-d editing. The following properties are provided:

Parameter	Description
Model Stretch	A scaling factor applied to the models, typically used for presenting broad overviews of a serial

Parameter	Description
Grid Delta	The spacing to use for the grid plotted on the surface
Show Coastline	Whether to plot a coastline on the sea surface (not yet implemented)
Show Bathy Lines	Whether to show a wireframe grid representing the surface of the ocean/land
Show Bathy Surface	Whether to show a shaded surface representation of the ocean bottom/land surface. Points above water are shaded green.
Show Drop Bars	Whether to show drop bars which stretch from a current vessel position vertically up or down to the sea surface, to assist in the spatial separation of vessels.
Show Foggy Ocean	Whether to shade the ocean surface
Show Sea Surface	Whether to plot a grid on the sea surface
Show Snail Trail	Whether to plot a snail-trail behind each vessel
Show Vessel Status	Whether to show a textual status string for each vessel
Snail Trail Length	The length (in time) of snail trail to plot for each vessel.

4.4. Step through the serial

Now re-arrange the 3-d plot and the Debrief plot so you have a clear view of both, and move the time-stepper forward a single step. You will see each 3-d vessel representation move. As you move through the track using the button on the Time Stepper, note the highlights on the 3-D track moving. Now push the "Step Automatically" button, to make the plot step forward on its own, leaving us to play.

Whilst the plot is stepping forward, switch your attention back to the 3-d view. Experiment with using the 3 mouse buttons (or two plus the alt-key) and zoom in on the current point on the red-track. Learning to coordinate these three operations may seem difficult, but you'll soon pick it up.

The data-file we're using isn't all that interesting in this view, since neither participant reacts in response to a depth-change from the other. But, 3-d views of real exercise results can be particularly insightful, clearly illustrating a weapon losing contact of its target due to changing depth layer, or one target gaining detections of another as it moves into the same layer.

4.5. Multi-monitor setups

The current Direct3D libraries from Microsoft only allow Debrief to show 3d plots on the primary monitor. If you only have one monitor, there isn't a problem. But, if you have a multiple monitor setup, the 3d plot will only appear if you have Debrief running on the primary monitor (that's the monitor that contains the windows task bar).

4.6. Memory Leaks

The current version of the 3D libraries used in this viewer contains memory leaks. In general memory leaks are tidied up in each revision of Java-3d, so it is in your interest to try to keep up to date with the newest releases. [2]

[2]Memory leaks occur when an application allocates memory to store data but fails to return the memory after the operation is complete. This results in the application's memory allocation growing. This may be overcome by exiting and re-starting Debrief, during which all allocated

4.7. Plot management

The objects in the 3-d plot reflect the structure in the Layer Manager. So, to hide/reveal a set of objects in 3-d just switch their layer off in the Layer Manager.

Tip

The objects in the 3-d view are a "cloned" set of the objects contained in the Layer Manager at the point at which Show 3d View was selected. Thus is it likely that moving objects between layers once the 3-d plot is open may not be reflected in the 3-d view. If extensive restructuring has been conducted in the Layer Manager and you wish the 3-d view to reflect this, close and re-open the view.

Note

The 3-d view now includes bathymetric data , though this is only shown if the bathy data is already shown in the 2-d plot. Obviously the 3-d view runs more smoothly with less data, so hide the 2-d bathymetry to remove it from the 3-d view.

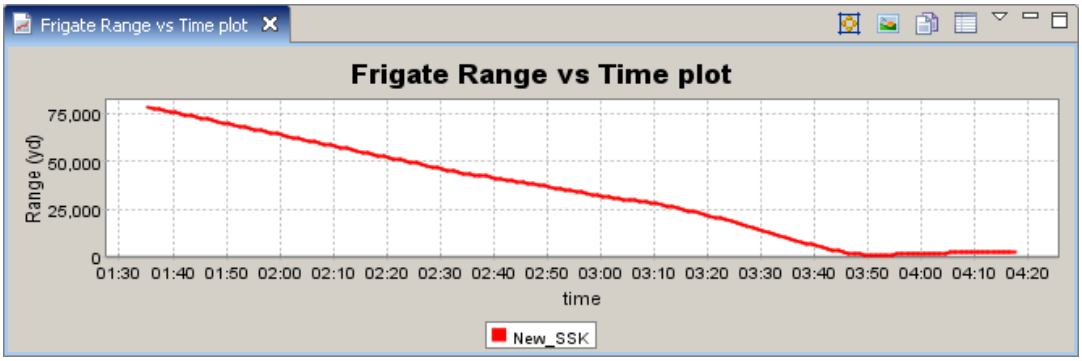
Tip

The 3-d view contains text labels that automatically orientate themselves towards the viewer. This orientation is computationally expensive (it is not a collection of 3-d objects being mathematically re-oriented, but a 2-d image being rescaled in 3-dimensions). The plot will be significantly slowed down if many text labels are set to visible. A good way of overcoming this problem is to set the parent layer to not-visible, then work through the items on that layer switching the label to not visible. Finally switch the layer back to visible. This method works quickly because time is not spent

5. Show time-related variables

Time-variable plots can be opened via the Layer Manager. When one or more items that are candidate subjects [3] for a time-variable plot are selected in the Layer Manager, it adds the Show XY Plot operation to its right-click menu. After requesting the plot, the user is invited to indicate which calculation is to be plotted, and which track is to be used as the primary *track* (where relevant).The example below shows the results of a Range calculation between two tracks. Note that dragging the mouse downwards in a rectangle inside the plot zooms in on the data, dragging the mouse upwards into a rectangle zooms out on the data, and that the Fit to Window button zooms out to show all of the current data. Where data is not present, a gap is inserted into the data-line.

Figure 3.11. View of time-variable plot



memory is cleared, sometimes resulting in large memory usage by Debrief after viewing several plots. This may be overcome by closing and restarting Debrief if a noticeable slow-down is observed. New libraries will be adopted as soon as a fix is available

[3] Items are suitable for inclusion in a time-variable plot when they have both temporal and spatial attributes, so this includes tracks, individual locations and annotations, but does not include grids, scales, or background datasets.

Note

Debrief performs special processing depending on whether the selected items contain valid time data or not. In general, when an annotation (such as a label) does not contain DTG data Debrief assumes that it is valid throughout the selected time period. Debrief produces the time-variable plot according to the following tables:

Table 3.2. Relative Calculations (range, bearing, etc)

	Primary contains DTG	**Primary does not contain DTG**
Secondary contains DTG	For each point on secondary, find nearest primary point (in time), use these for calculation	For each point on secondary, use single primary point for calculation
Secondary does not contain DTG	For each point in primary, use single secondary point for calculation.	Produce single calculations at start & end of time period

Table 3.3. Absolute Calculations (range, bearing, etc)

	Calculation
Data series contains DTG	For each point on series, calculate result
Data series does not contain DTG	Produce single calculations at start & end of time period

The Export to WMF  button writes the current graph to a Windows Metafile in the current WMF_Directory (as configured in the Debrief properties file, described in Section Section 3, “Debrief Properties ” [84], and the Export to Clipboard ( places a copy of the plot on the Windows clipboard for onward insertion into MS Word.

The Configure Plot() button opens a property window allowing you to control the presentation of the time-variable graph using the following properties:

Parameter	**Description**
DataLineWidth	The width to plot the data-lines on the graph
DateTickUnits	The interval (and format) to use on the date axis
RelativeTimes	Whether to plot absolute or relative times (used when analysing time-zero data). See Section 5.2, “Analysing time-zero data in time-variable plot” [ ] .
ShowSymbols	Whether to show datum symbols (useful to indicate different data recording rates)
Title	The Title of the graph
X_AxisTitle	The x-axis label on the graph
Y_AxisTitle	The y-axis label on the graph
xxx Font	The font to use for the respective label

5.1. Time-Variable Plot Tracker-Bar

If there are tracks on the Debrief *Tote*, a vertical bar is drawn through the plot at the correct time. If the current time on the Tote occurs before the earliest time on the Time-Variable plot then the

bar is drawn at the minimum value, and drawn over the maximum value if the current time on the Tote occurs after the latest time value. The bar redraws itself automatically as the time in the Tote changes.

5.2. Analysing time-zero data in time-variable plot

By default the time-variable plot shows absolute times (e.g. `12:34.00`). Some forms of analysis conducted with Debrief rely on the use of relative times, such as a weapon firing where T-Zero is the time of fire, as described in Section 6.1, “T-Zero times” [42]. To show these relative times on the lower axis of the time-variable plot, select RelativeTimes from the time-variable plot property window. Once this mode is selected, times will be shown relative to the last time-zero value set. By default, the times are shown in a normal `HH:mm:ss` format, but the DateTickUnits drop-down list contains a number of display formats which specifically support presentation of elapsed times.

6. Track & Time toolbox

6.1. T-Zero times

T-Zero times are still to be implemented, but the mode of operation will be very similar to that shown below, June 2006.

The Set time zero command reformats the labels on the plot to follow the standard presentation for weapon-analysis plots; where absolute times are replaced by times relative to a specific event (*T-Zero*)- such as Weapon Splash.

When the Set time zero command is started, the following sequence of events occur:

1. You are invited to enter the frequency in seconds at which symbols are drawn on the indicated tracks.

2. You are invited to enter the frequency in seconds at which location labels are displayed for the indicated tracks. In the labels at this indicated frequency the text is set to the elapsed time in seconds since the time-zero time.

These steps are applied to all of the track points between the Start time (which is used as T-Zero) and the Finish time.

Note

Note also that when the date format in the Time Stepper is set to *T+ SSS* the current time of the time stepper is no longer displayed as absolute time, but as elapsed seconds since the time currently displayed in the Start Time on this panel, or elapsed minutes and seconds (*T+ MM:SS*).

Note

Following analysis using t-zero data labels, it is possible to return a set of track labels to their default (as automatically assigned when imported from REP). Reset the track labels by selecting Reset DTG Labels from the edit track popup menu or property window.

Chapter 4. Exporting Data

1. Exporting images

1.1. Export Images

In addition to conducting tactical analysis, Debrief allows the analyst user to create plots for insertion into Microsoft Windows applications; particularly Word.

To do this load the data into Debrief and format the plot, adding scales, grids, coastlines, and annotations as desired.

To export this image as a Windows Metafile (WMF), select the Export WMF () operation from the File menu. Alternatively to copy the image to the clipboard (still as a WMF) select Copy from the Edit menu whilst the Debrief plot is active.

This will place the image in a time-stamped file named `d3_minute_second.WMF`, located in the directory indicated in the Debrief properties file. If the location is not specified in the properties file, the `WMF` image will be created in the Debrief installation directory.

Also, don't forget that time-variable graphs produced by the Track-Time toolbox can be exported to WMF aswell, as described in Section Section 5, “Show time-related variables” [40], titled Analysing time-zero data in time-variable plot [40] .

2. Virtual reality files

2.1. History

Important

Now that Debrief includes its own 3-D viewer, the export to 3-D functionality has been removed. Should you have any significant desire for this functionality to be replaced, please contact the administrator.

Chapter 5. Handling Buoyfields

1. Introduction to buoyfields

Note

In the first release of Debrief NG, Buoyfields have not been implemented. It remains an intention to provide this facility.

1.1. Buoyfields

Debrief provides functionality to ease the creation of Sonar Buoyfields, that is a pattern of sonar buoys which are laid out in a particular pattern. In addition to the geospatial characteristics of the pattern to be created (such as radius, or row spacing) Debrief also accepts temporal (such as time of the first buoy and buoyfield lifetime) and default formatting characteristics (such as colour and symbol to use). After field creation, the properties used for building the field are largely unavailable, but the buoys can be repositioned or time-stamped directly.

Although it may not be clearly visible in the following diagrams, all of the buoyfields are located with reference to a kingpin. This kingpin is located relative to a Jig Point (allowing for small, quick adjustments in the location of the field prior to dropping), as specified by a range and bearing.

1.1.1. Barrier

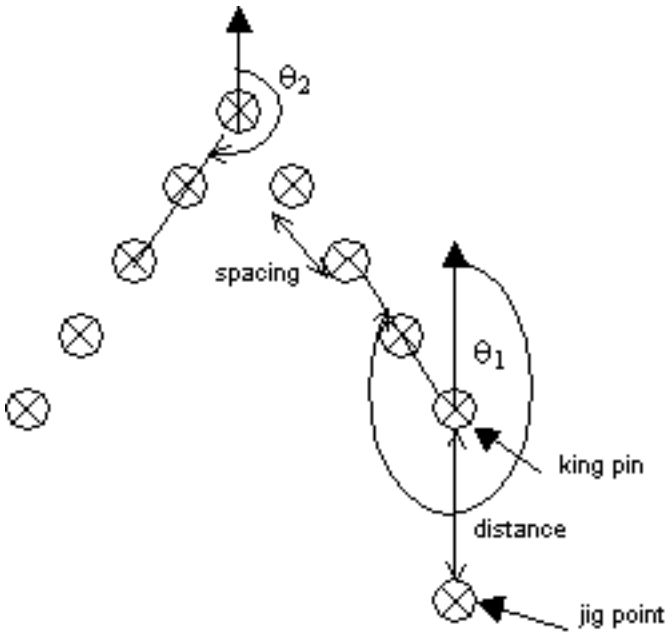
A Barrier is a straight line of buoys

Figure 5.1. Example of a barrier buoy-pattern

1.1.2. Wedge

A Wedge is a bent-line barrier of buoys, specified by buoy spacing and two true bearings. The first bearing is from the kingpin to the centre of the wedge and the second bearing is from the centre point to the last buoy. In Debrief, theta-one is termed the Right Hand Orientation (Orientation1) and theta-two is termed the Left Hand Orientation (Orientation2). Debrief allows even or odd numbers of buoys, and will ensure that the sides of the wedge remain symmetric.

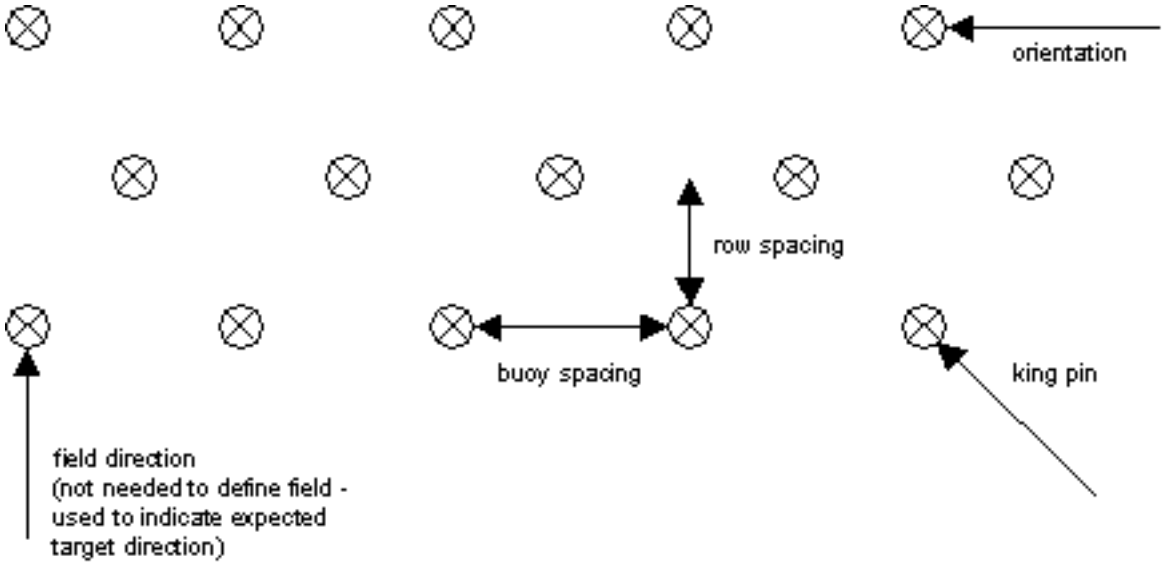
Figure 5.2. Example of a Wedge buoy-pattern



1.1.3. Field

A Field is a fixed pattern which contains more than two rows of buoys. The pattern is specified by buoy and row spacing, a true bearing, and an indication of whether the successive rows fall to the left, right, or centre of the kingpin. Note that whilst the diagram shows and the Naval specification order includes the Field Direction parameter, this is not required for buoyfield construction and is not taken from the user. Although the Field Direction is contained in the buoyfield creation order, it is included as an indication of expected target direction of travel, and has no graphic meaning.

Figure 5.3. Example of a Field buoy-pattern



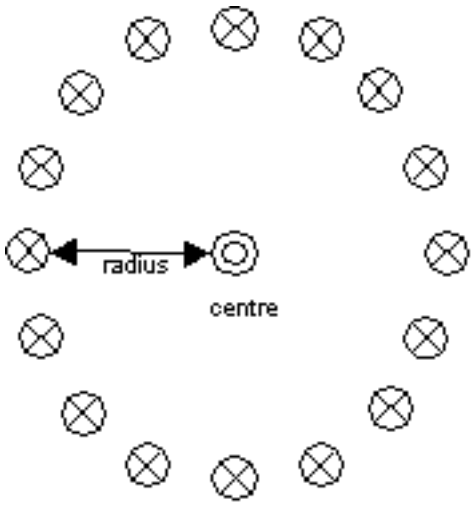
The Left, Right, or Centre offset value is interpreted in the following way:

Left: is interpreted as the "alternate" rows starting offset a ½ buoy width further in the field orientation
Right: is interpreted as the "alternate" rows being offset a ½ buoy width backwards along the field orientation
Centre: in interpreted as there being no offset in successive rows.

1.1.4. Circle

The Circle pattern describes a pattern of buoys arranged in a circle, specified by kingpin, radius, and number of buoys. An orientation parameter is also used, which indicates the bearing of the first buoy from the kingpin (centre), since it may not necessarily be to the North. A flag indicates if the circle is being laid clockwise or not.

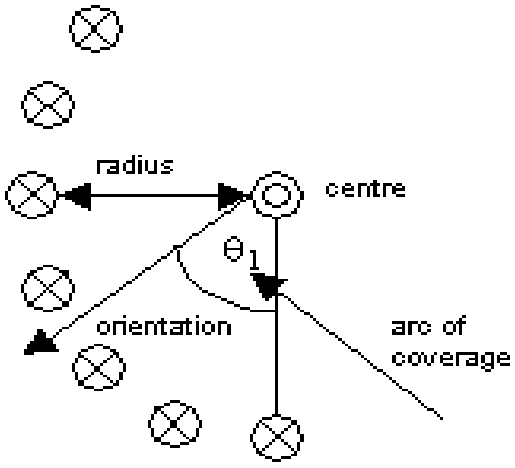
Figure 5.4. Example of a Circle buoy-pattern



1.1.5. Arc

The Arc pattern describes a pattern of buoys arranged in a segment of a circle. In addition to the kingpin and radius, an orientation parameter indicates the direction of the mid-point of the arc from the kingpin, and an arc value indicates the arc of coverage each side of the orientation. A flag indicates if the circle is being laid clockwise or not.

Figure 5.5. Example of an Arc buoy-pattern



2. Creating buoyfields

Note, at the time of writing (June 2006) Debrief NG does not contain the Buoyfields wizard. Debrief 2003 may be used to create a plot-file containing buoyfields, which can then be re-opened in Debrief NG.

2.1. Toolbar

A single toolbar button is used to create all types of buoyfield: 

After clicking on this button, a dialog pops up which invites you to select one of the buoyfield patterns listed on the previous page: using a drop-down list.

After selecting one of the patterns a set of editable properties are displayed in the *property editor* window.

2.2. Properties

A number of the properties edited from this window are common to all buoyfield types:

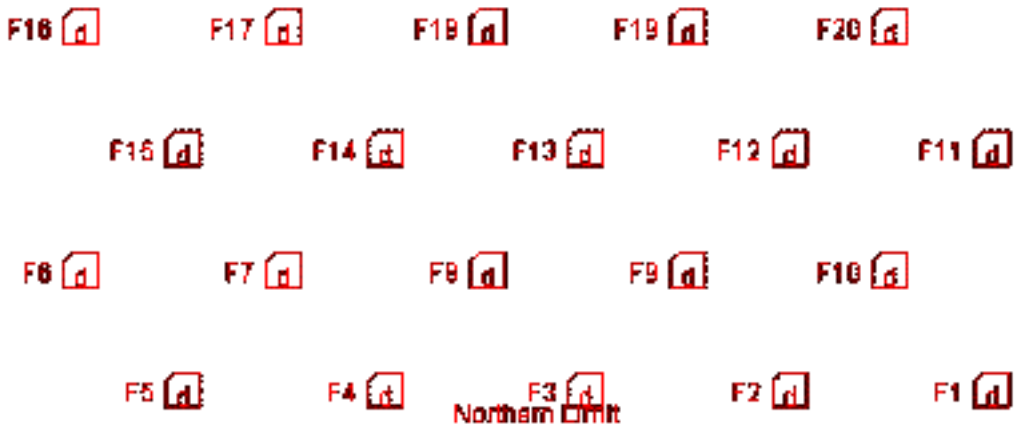
BuoyLifeTime	the time for which the buoys remain active (represented as the time between the start and finish DTG)
Color	the colour to plot the buoys and the buoyfield label
JigPoint	the location from which this buoyfield is referenced. As in any location, the jig point can be edited by double-clicking on the plot or using an edit panel. Note that the depth assigned for the jig point is used as the depth for all of the buoys. If the jig point is assigned by double-clicking, the depth value can then be set by clicking on the "Edit" button.
KingpinBearing/Range	the range and bearing of the kingpin for this pattern from the jig point
LabelVisible	whether to display the labels for buoys in this buoyfield
Name	the name of this buoyfield
Number	the total number of buoys in this field
StartDateTimeGroup	the DTG of the first buoy laid
SymbolScale	the scale of the symbols drawn to represent buoys.
SymbolType	the type of symbol to be used to represent buoys
TimeDelta	the time between each buoy dropped (each buoy will have the DTG of the previous buoy plus this TimeDelta, except the first buoy which takes the value of StartDateTimeGroup)

Once you have completed editing these parameters, press the Build button, which will create a buoyfield using the defaults indicated. The properties for the buoyfield as a whole will open up in the properties window. From these properties you make any necessary corrections to the name of the buoyfield, or the location of the name relative to the buoys themselves.

The buoyfield which is produced will look something similar to that shown below.

2.3. Example Buoyfield

Figure 5.6. Example of a buoy-pattern



The buoys themselves are represented using the Label type described elsewhere in this documentation. As Label objects, the user is able to right-click on any buoy and interactively edit the label colour/location, or double-click on the symbol to open the label up in the editor to access the full set of parameters.

3. Using buoyfields

3.1. Tote

Buoyfields have some similar properties to vessel tracks when placed on the *Tote*.

Once a buoyfield is made a Primary or Secondary track, a highlight is drawn around the buoy at or immediately after the current time. Additionally, if the buoyfield is designated as a secondary track the Tote continuously displays it's range and bearing from the current location on the primary track.

3.2. Analysis

Since buoyfields are essentially handled in the same way as Tracks, the full functionality of the Track & Time Toolbox (accessed from the Tote) is available, so graphic plots can be produced and calculated data copied to the clipboard using Buoy data.

Chapter 6. Symbol Sets

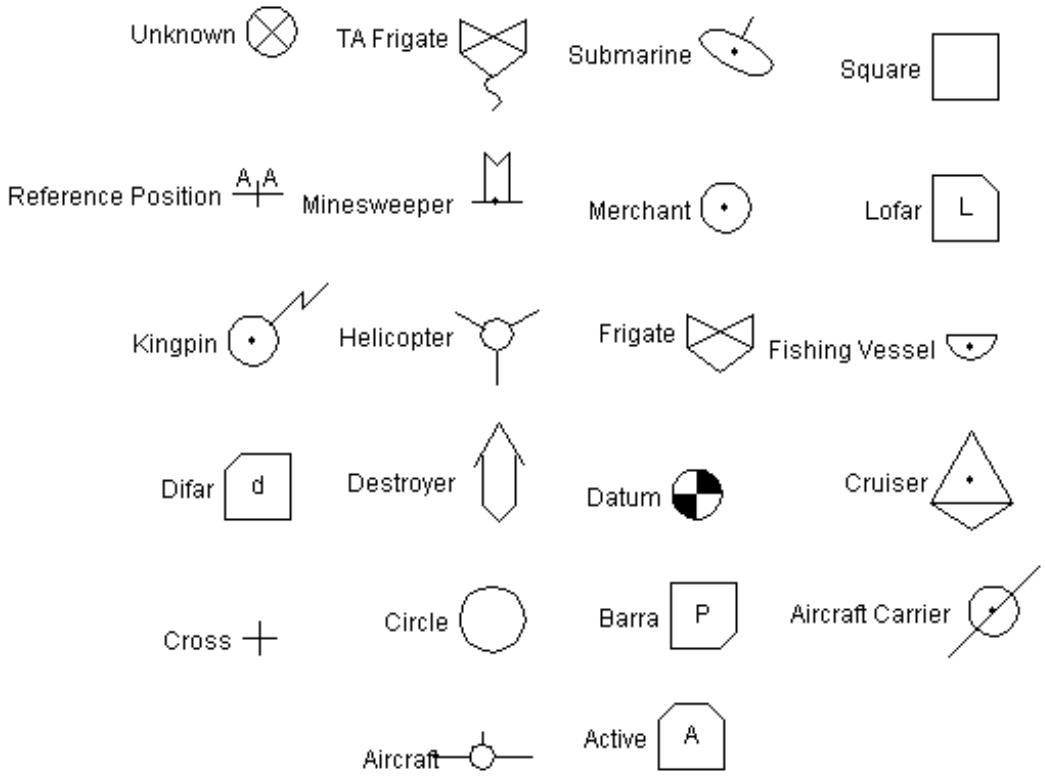
1. Debrief Symbol Sets

1.1. Introduction

A set of symbols is provided within Debrief. The symbols can either be attached to a text label, or used to highlight a current vessel location when stepping through tracks.

Twenty-two Symbols are provided in 3 sizes. When the 2 smaller sizes are shown, any internal characters (as used in the sonar buoys) are not plotted.

Figure 6.1. Symbols provided by Debrief



1.2. Use in buoy patterns

Symbols are assigned to buoys in a buoy pattern at the Pattern Builder stage. After selecting the type of pattern to be created the user is able to select which type of symbol is used for the buoys (at this stage only the buoy-related symbols are provided). For patterns which include a kingpin (Field, Barrier, Wedge), a kingpin symbol is plotted for the first buoy.

1.3. Use in labels

Any label placed on the plot has a symbol type property (although the symbol itself may not be visible). The symbol can be changed through the properties window, or by right-clicking on the label on the plot.

1.4. Use in tracks

The final area where symbols are used is when stepping through tracks. Each vessel track has a symbol type property which is shown when the plot has SymbolHighlighter selected as the current highlighter.

To access this property, select the Properties button on the *Tote* (first button on bottom row), and change the current *highlighter* to "SymbolHighlighter". Now the current data point on each track will be shown using the symbol for that track. 

Tip

By switching to Snail mode and reducing the reducing the Trail Length to zero, a GOP-type plot can be shown.

Chapter 7. External Datasets

1. Installing VPF Data

1.1. Introduction

The *VPF* [1] standard provides Debrief users with the ability to display the contents of a number of different vectored databases.

One of the databases, the Vector Map (VMap) Level O is an unclassified global database of many features, including coastlines, depth and elevation contours. This is the favoured VPF database for use in Debrief. But note that Debrief makes no assumptions of or optimisations for the VMap data - any VPF data source can be used. Debrief can also import Digital Nautical Chart (DNC) databases. The Digital Nautical Chart is produced by the National Imagery and Mapping Agency (NIMA) and is an unclassified, vector-based digital database containing maritime significant features essential for safe marine navigation.

Here is an introduction to VPF from the US National Imagery and Mapping Agency's web site (http://www.nima.mil):

> The Vector Product Format (*VPF*) is a standard format, structure, and organization for large geographic databases that are based on a georelational data model and are intended for direct use. VPF is designed to be compatible with a wide variety of applications and products. VPF allows application software to read data directly from computer-readable media without prior conversion to an intermediate form. VPF uses tables and indexes that permit direct access by spatial location and thematic content and is designed to be used with any digital geographic data in vector format that can be represented using nodes, edges, and faces.
>
> 
>
> VPF defines the format of data objects, and the georelational data model provides a data organization within which software can manipulate the VPF data objects. A Product Specification corresponding to a specific database product determines the precise contents of feature tables and their relationships in the database. In this context, each separate product or application is defined by a Product Specification implemented by using VPF structures.
>
> VPF data is stored in a structure described in the Military Standard, Vector Product Format, MIL-STD-2407 . The Standard specifies the structure for directories, tables, table columns, table join relationships, and media exchange conventions for all VPF data. The data structure itself can be thought of as a template or skeleton within which the geospatial features and metadata are stored. While the Standard describes the structure, it does not describe the contents of a set of VPF data; this is the role of "VPF Product Specifications."

[1] VPF(TM) is a Registered Trademark of US National Imagery and Mapping Agency

1.2. Introduction to VMAP Level 0

Vector Map (VMap) Level 0 is an updated and improved version of the National Imagery and Mapping Agency's (NIMA) Digital Chart of the World (DCW).

The primary source for the database is the 1:1,000,000 scale Operational Navigation Chart (ONC) series co-produced by the military mapping authorities of Australia, Canada, United Kingdom, and the United States. The complete database is available on a set of four CD-ROM's and contains more than 1,800 megabytes of vector data organized into 10 thematic layers. VMap Level 0 includes major road and rail networks, hydrologic drainage systems, utility networks (cross-country pipelines and communication lines), major airports, elevation contours, coastlines, international boundaries and populated places. VMap Level 0 includes an index of geographic names to aid in locating areas of interest. VMap Level 0 is accessible directly from the CD-ROM or can be transferred to a hard drive and used in many geographic information system (GIS) applications.

1.3. Obtaining VPF data

VPF data can be obtained from a number of different sources, although the main resource is the NIMA [http://www.nima.mil] web-site itself.

In the UK, the Director General of Military Survey (DG Mil Survey) were able to provide the VMap database (charged to a Royal Navy UIN).

If/when you find other sources please provide feedback and this page will be updated.

Searching the Internet will undoubtedly provide other sources for the information - it's probably worth shopping around. Since the data is in the public domain agencies only have to charge a "handling fee".

1.4. Storing VPF data

Debrief can read VPF data directly from CD-Rom but copying it to your hard disk provides the following advantages:

- It runs many times (>10) quicker
- It allows you to view the contents of multiple CD-Roms simultaneously (the VMap level 0 data comes on 4 CDs so unless you have 4 CD-readers in your machine this is the only way to get global coverage)

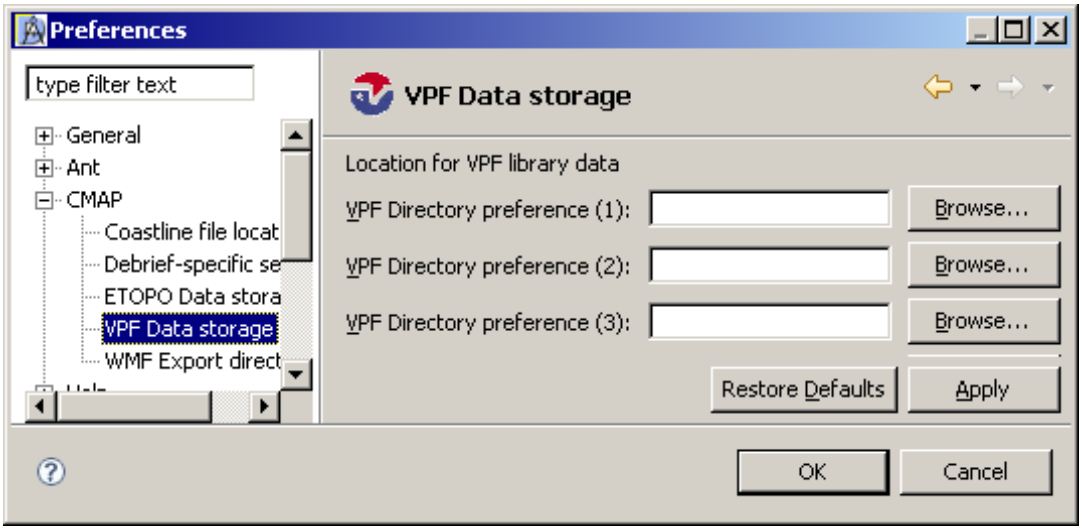
So, on the assumption that you do want to copy the data to your hard disk, here is a strategy for keeping the data tidy:

1. Create a folder in the top level of your hard disk called `VPF`
2. Now insert the first VPF CD-Rom and view it through your file manager (e.g. Windows Explorer). You need to find which directory contains the actual data, so have a look in each of the top level directories for a directory which contains the files `DHT` and `LAT`. Note, for VMap data this is a directory named `VMAPLV0`. When you've found this directory copy it across to the `VPF` directory you created on your hard disk.
3. So for VMap data, your hard disk you should have a directory named `VPF`, containing your first set of VMap data in a directory named `VMAPLV0`.
4. You will be copying in a number of data sets with this directory name, so rename `VMAPLV0` to the name of the current dataset (such as `NOAMER, SASAUS, EURNASIA or SOAMAFR`).
5. Next, swap your CD for the next CD in the series, copy the directory of interest across to the `VPF` directory on you hard disk, and rename it.
6. Once you've repeated this process for all of the CDs for the current database it's time to configure Debrief to load the data.

1.5. Configuring Debrief to read VPF data

Debrief determines where to find the *VPF* data using the VPF Data storage tab of the Preferences dialog from the Window menu. Indicate the four data-file locations using the file-browser buttons.

Figure 7.1. VPF preferences



1.6. Thank heavens for Open Source

As many of you are probably aware, Debrief is an Open Source application, which means that anybody is free to copy, change, and re-use the Source Code for Debrief, provided they meet the terms of the Debrief license.

The VPF plotting libraries we're using in Debrief are taken from another Open Source application, OpenMap from BBN Technologies.

The OpenMap application can be found at http://www.openmap.net. Debrief makes no modifications to the OpenMap application code.

2. Viewing VPF Data

2.1. Introduction

Debrief does not pay any attention to *VPF* data until it loads a plot which requires the data, or until the user requests that VPF layers be added.

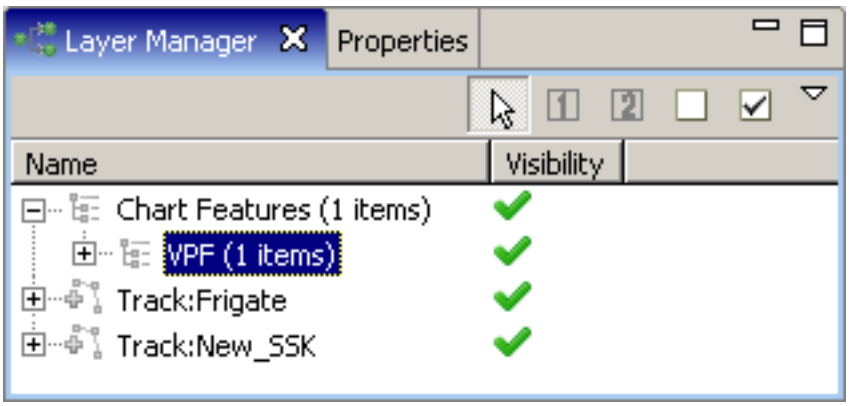
VPF layers are loaded into a plot using the "Create VPF Layers" button: 

Yes, it's a shrunken version of NIMA's VPF logo. If VPF layers have been loaded into Debrief, their details are stored in the plot file (i.e., whether they are switched on or off, and their colour). Note that the data itself is not stored in the plot file, just the names of the layers you're using (so there's no significant increase in file size).

2.2. Creating VPF Layers

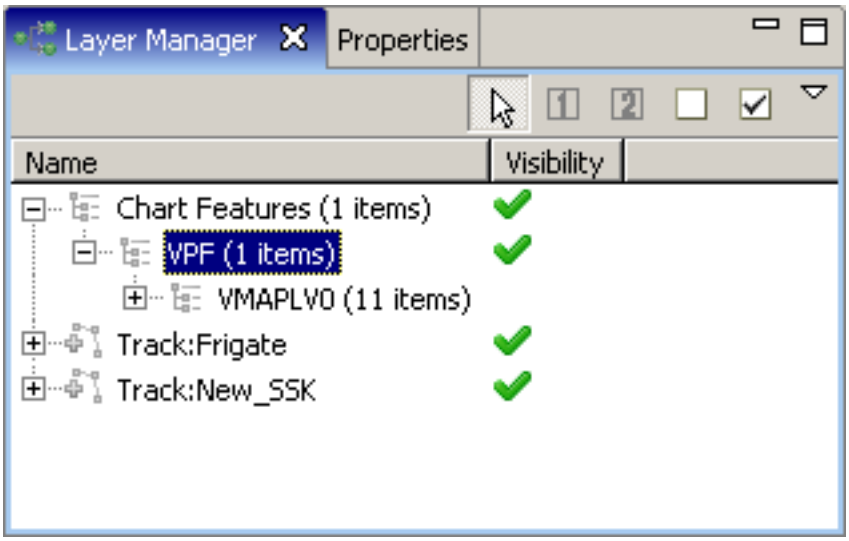
Once loaded into a plot configure the VPF data by switching to the Layer Manager. In there, open the Chart Features layer, to see the new VPF layer.

Figure 7.2. Layer Manager



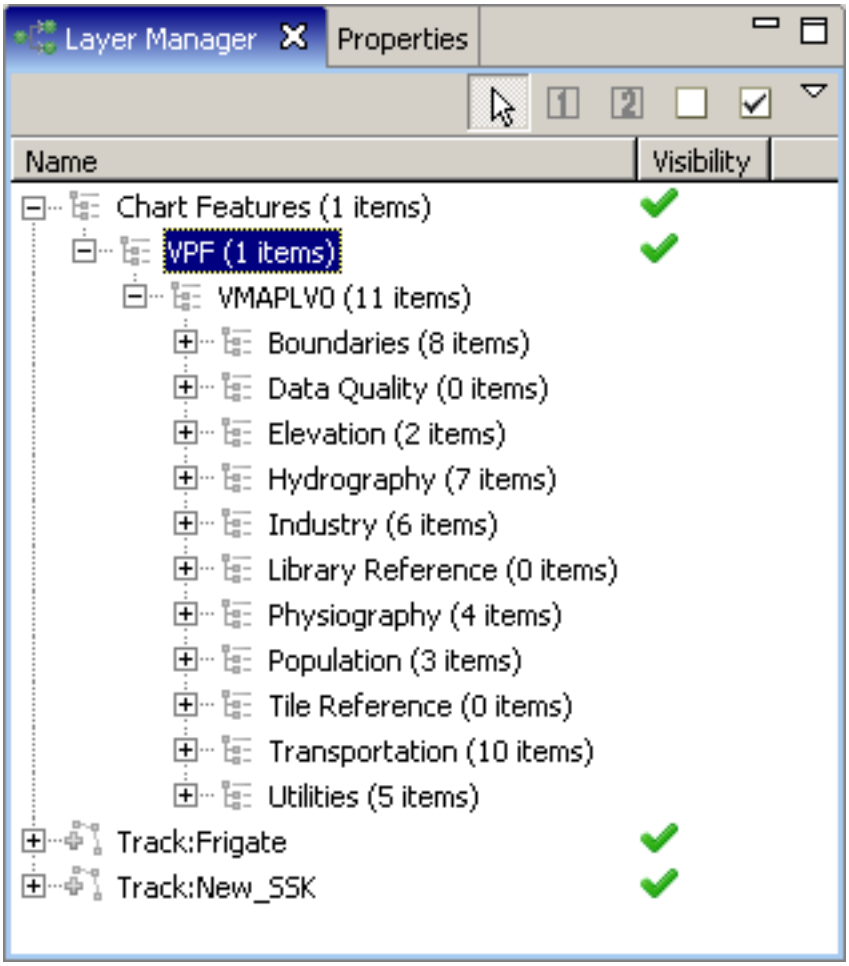
Click on the VPF layer to open it up, showing which VPF databases are currently loaded. In this example you can see that I have the VMap Level 0 (vmaplv0) data together with some Digital Nautical Chart (DNC) data. Each layer shows how many items are on that layer, and the empty check box shows that each layer is not currently visible.

Figure 7.3. Layers within Chart Features



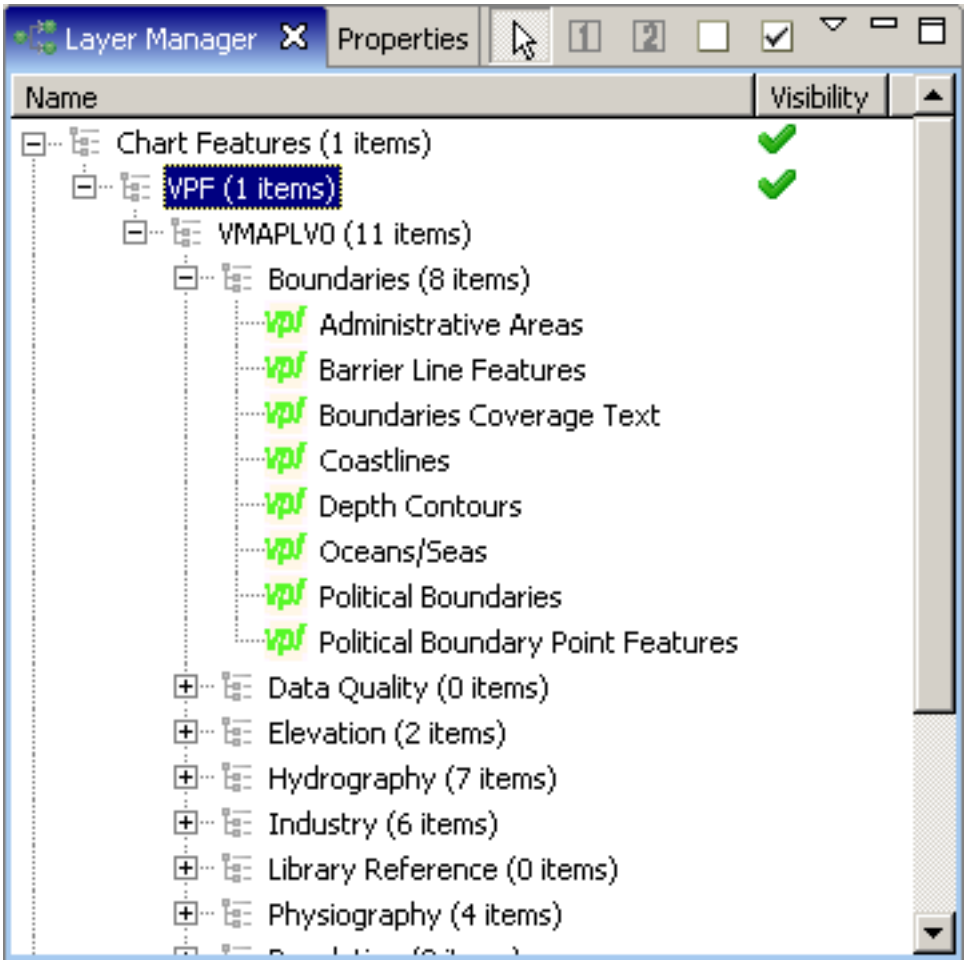
You are going to switch on coastline data first, so click open up the vmaplv0 database by clicking on the expand folder symbol. You will see the list of VMap themes listed. Later on, feel free to open them up and view the data they contain, but for now you will concentrate on the coastlines. The order of the themes may be different on your machine, this is of no concern.

Figure 7.4. List of VPF layers



Next, open up the Boundaries theme to see which Boundaries can be plotted by VMap.

Figure 7.5. Boundaries theme



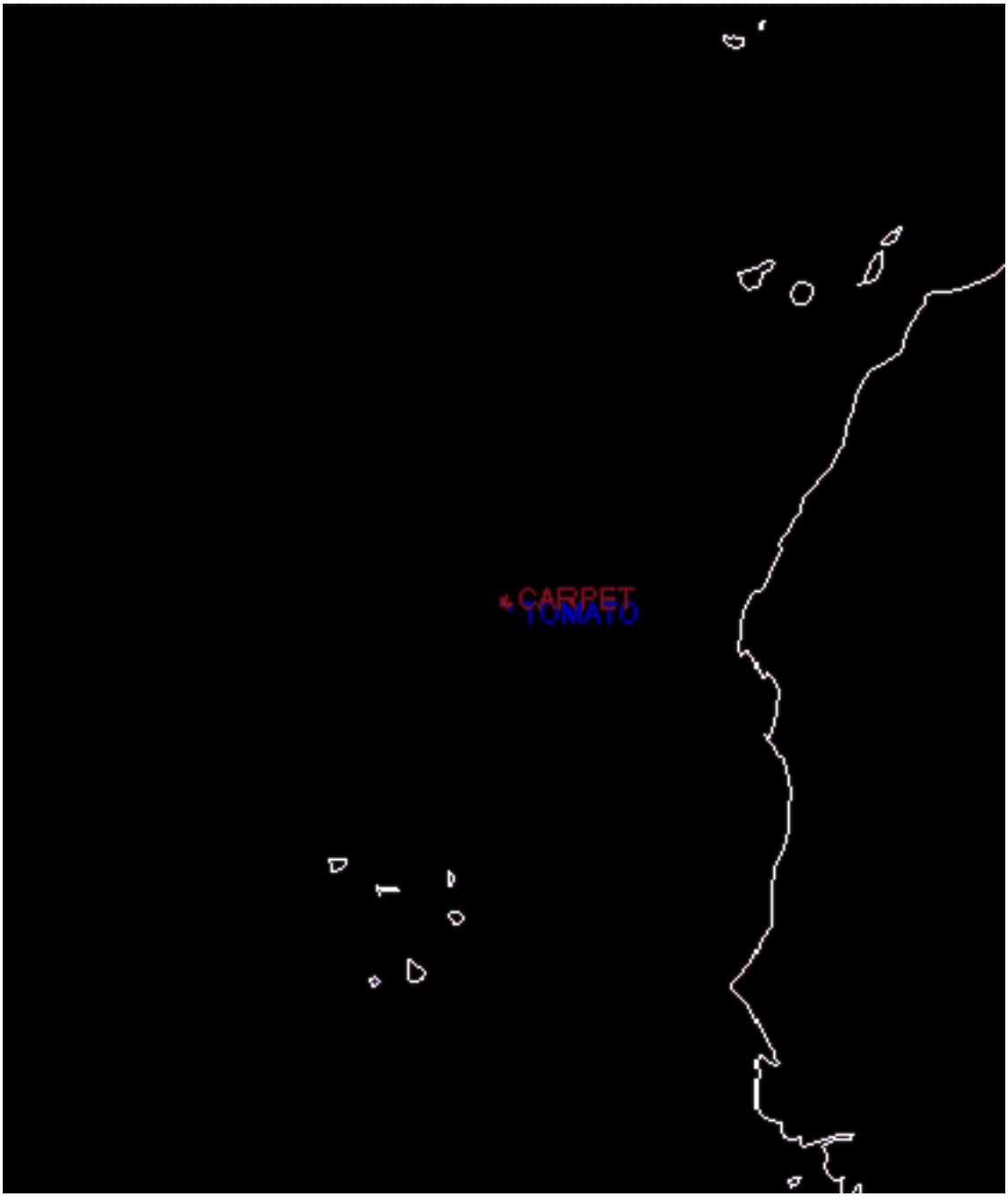
Now you are going to configure the plotting of coastlines. Right-click on the Coastlines entry, and from the Coastlines drop-down list, select Color, and then Light Grey. Note that White is the default colour for all VPF features.

Next, right-click on the coastlines entry again, and make the coastline Visible.

Whilst we've made the Coastlines visible, the layers above it aren't, so now you make its parents visible. Right-click on the Boundaries layer and make it visible. Next right-click on the vmaplv0 layer to make that too visible.

Nothing has yet appeared on the plot, but that is because you are not current over any coastline. Now zoom out 6 times until the plot looks like that below:

Figure 7.6. First coastline



Now you can see your new coastline of Africa.

Select the Pan control from the Plot toolbar: 

Now drag the plot southwards to move the view up towards Europe. Keep dragging until you have a clear view of a more familiar coastline, that of the British Isles. From this view you will add depth contours.

Figure 7.7. Coastline of British Isles

Right-click on the Depth Contours entry in the Boundaries layer and switch it's colour to Grey. Also switch the Contours entry to Visible.

Depth contours will now appear:

Figure 7.8. Coastline of British Isles

Note

Note, the VMap CD-ROM gives the following description with relation to depth contours:

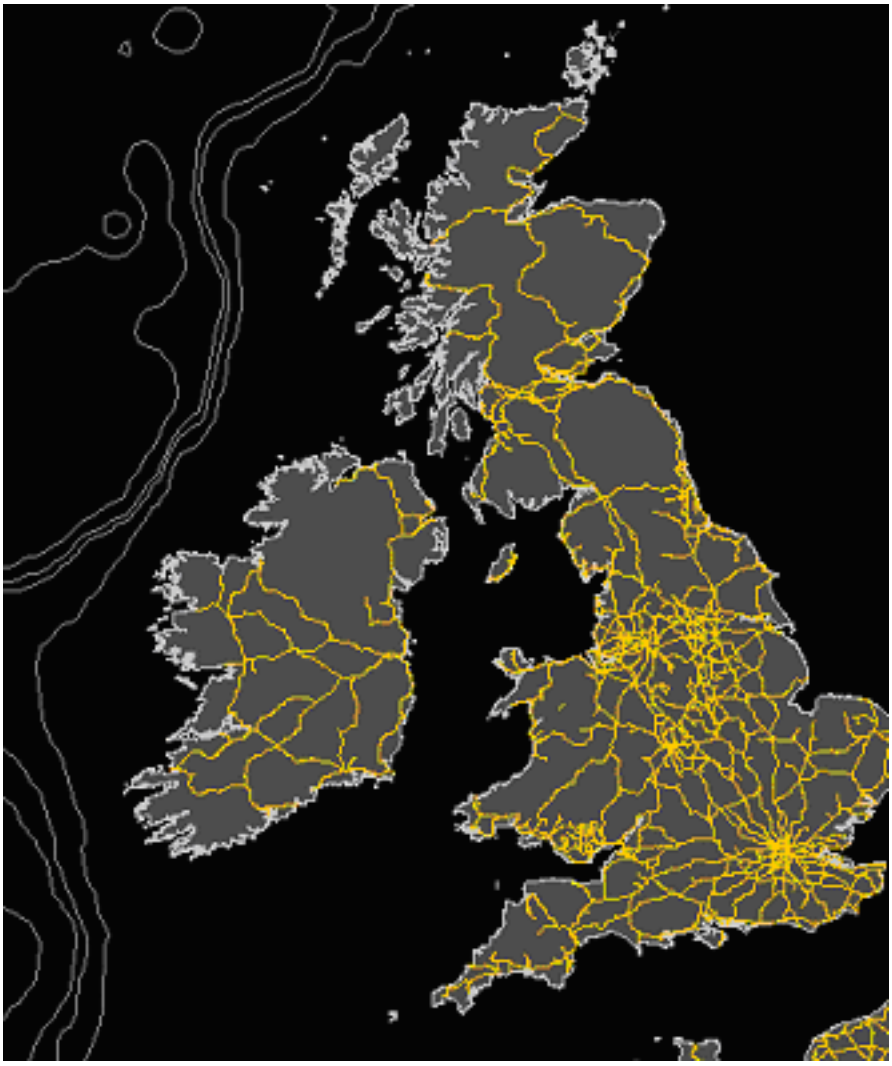
> Depth contours (BE015) were derived from digital bathymetric data provided by Defense Mapping Agency. Depths are expressed in the following intervals: 200, 600, 1000. 2000, 4000, 6000, and 8000 meters.

Feel free to zoom in on the data to see just how detailed the data gets (around Cowes on the Isle of Wight sections of coastline of only 200 yards length are clearly visible).

Also zoom out to view the whole of the British Isles and experiment with switching on features in other layers.

The plot below shows UK railways (from the Transportation Layer), and UK is shaded in by making the Administrative Areas (from the Boundaries Layer) visible.

Figure 7.9. Railways of British Isles



Alternatively zoom and pan across to your home, see the accuracy of your part of the coastline.

2.3. Storing VPF settings

As explained earlier, when you save the *VPF* plot your VPF settings get saved with it. So have a go at saving your current view, exiting/re-starting Debrief then re-loading it. You should find yourself seeing the same view which you saved.

2.4. VPF Best Practice

There's no doubting the volume of data available through VPF, and the VMap Level 0 database in particular. The next level down from VMap Level 0 is VMap Level 1; which contains approximately 10 times more detailed information.

How much of it is of use is more subjective though, so it's best to lead your strategy for its use by your analysis requirements.

3. Configuring VPF Defaults

3.1. The Problem

Yes, VPF and VMap in particular provide a great volume of information which may help with analysis and will certainly improve the quality of "overview" images in reports.

With this great volume comes a management overhead however. You could have constrained Debrief to only use VMap data, and then only load a sub-set of it (coastlines & depth contours). Instead, however, it loads the full set.

How can you make it easier to use?

Read on.

Warning

You'll need to know how to use a text editor and have a rough familiarity with *XML* or HTML text.

3.2. The solution

How does it Debrief handle VPF data?

- When you save a plot with VPF data loaded, Debrief stores the names of all of the layers loaded, together with the colours and visibility of any features on them.
- When you re-load an existing session with VPF data, Debrief reads what layers the users wants and only loads those layers.

Accordingly, you could create a datafile containing only the layers you want, and drop this file into Debrief sessions. This should give us a simplified set of VPF layers.

When you drag/drop an *XML* file into an existing session, the Layers in the *XML* file are copied into the existing session, although the projection, Tote, and GUI parameters are ignored.

3.3. How to do it - 1

So what you do is open a new Debrief session and load the smallest REP file you have into it - (just to give your data an "origin").

Then add your VPF layers using the Create VPF Layers button on the Chart Features toolbar. Customise these layers so that the layers/features of interest are made visible and set to your desired colours.

Finally save the view to a plot file in an easily accessible location - call it `default_layers.xml`

3.4. How to do it - 2

You are now going to edit this file to remove all unnecessary details except for your *layer*s.

Open a text editor (such as Notepad in MS Windows), and load your new xml file into it.

The contents of the file should be something like that shown below:

```
<?xml version="1.0" encoding="UTF-8"?>

<plot Created="Thu Jul 19 15:04:41 GMT+01:00 2001" Name="Debrie
<session>
<layers>
<layer Name="Chart Features" Visible="TRUE">
<vpf_database Visible="TRUE">
<vpf_library Visible="TRUE" Name="vmaplv0">
<vpf_coverage Visible="FALSE" Type="ind" Description="Industry"
```

```
<vpf_feature Visible="FALSE" Type="extracta" Description="Extra
<colour Value="WHITE" />
</vpf_feature>
<vpf_feature Visible="FALSE" Type="storagep" Description="Stora
<colour Value="WHITE" />
</vpf_feature>
<vpf_feature Visible="FALSE" Type="indtxt" Description="Industr
<colour Value="WHITE" />
</vpf_feature>
```

..... and so on, until

```
<layer Name="Annotations" Visible="TRUE">
<rectangle Label=" trial" LabelLocation="Left">
<colour Value="YELLOW" />
<fontcolour Value="YELLOW" />
<font Family="Sans Serif" Size="12" Bold="FALSE" Italic="FALSE"
<tl>
<shortLocation Lat="-9.2166417" Long="156.2783694" Depth="0.000
</tl>
</rectangle>
</layer>
</layers>
<projection Type="Flat" Border="1.050" Relative="FALSE">
<tl>
<shortLocation Lat="60.6482349" Long="-15.0669609" Depth="0.000
</tl>
<br>
<shortLocation Lat="47.9097909" Long="3.0063981" Depth="0.000"
</br>
</projection>
<gui>
<tote />
<component Type="Stepper">
<property Name="AutoStep" Value="1000" />
<property Name="Highlighter" Value="Default Highlight" />
<property Name="StepLarge" Value="600000.000" />
<property Name="CurrentTime" Value="691231 235959.999" />
</component>
</gui>
</session>
<details Text="Saved with Debrief version dated 19 Jul 01 12:44
</plot>
```

3.5. How to do it - 3

As you can see above, the file starts with an entry beginning with the characters <?xml. This line indicates that you are handling *XML* data.

Next you have a series of lines of data, which start with <plot> and end with </plot>. This format is similar to that found in HTML, and indicates that this file contains details of plot. Inside the <plot> you can see a <session>, which in turn contains a <layers> object, containing a series of <layer> items, followed by a <projection>, and a <gui>.

What you are going to do is strip out everything except the Layers object, and then thin this out.

So, move the cursor to the <layers> line. Delete everything before this (except for the <?xml version="1.0" encoding="UTF-8"?> line.

Now move down to the </layers> line, and delete everything from it to the end of the file.

As you look at the layers, you can see that each layer has a name (Chart Features is the first one in the example above). It is the Chart Features layer which you want to keep, so navigate down to the next Layer (which may have a track name), and delete that and all others (down to, but not including the </layers> line which marks the end of the data).

Within the Chart Features layer you can see an entry named <vpf_database> - this is the vpf data. Inside the vpf_database are <vpf_library> entries, one for each library loaded (probably just VMaplv0 in your instance).

Inside the <vpf_library> are a series of <vpf_coverage> entries. It is these which you will thin out. Work down through them deleting any you don't want. In your instance you only want to keep the boundaries data, so you will delete all other coverages. So, select blocks of text beginning with <vpf_coverage> and ending with </vpf_coverage> and delete those you don't want.

Finally, delete any <vpf_features> you don't want.

I've deleted all those I don't want, leaving the text below:

```
<?xml version="1.0" encoding="UTF-8"?>
<layers>
<layer Name="Chart Features" Visible="TRUE">
<vpf_database Visible="TRUE">
<vpf_library Visible="TRUE" Name="vmaplv0">
<vpf_coverage Visible="TRUE" Type="bnd" Description="Boundaries
<vpf_feature Visible="TRUE" Type="oceansea" Description="Oceans
<colour Value="BLUE" />
</vpf_feature>
<vpf_feature Visible="FALSE" Type="polbndl" Description="Politi
<colour Value="WHITE" />
</vpf_feature>
<vpf_feature Visible="TRUE" Type="polbnda" Description="Adminis
<colour Value="WHITE" />
</vpf_feature>
<vpf_feature Visible="FALSE" Type="depthl" Description="Depth C
<colour Value="WHITE" />
</vpf_feature>
<vpf_feature Visible="TRUE" Type="coastl" Description="Coastlin
<colour Value="CYAN" />
</vpf_feature>
</vpf_coverage>
</vpf_library>
</vpf_database>
</layer>
</layers>
```

3.6. At last!

Now, you can drag and drop this file into any Debrief session to instantly give you your "favourite" set of layers.

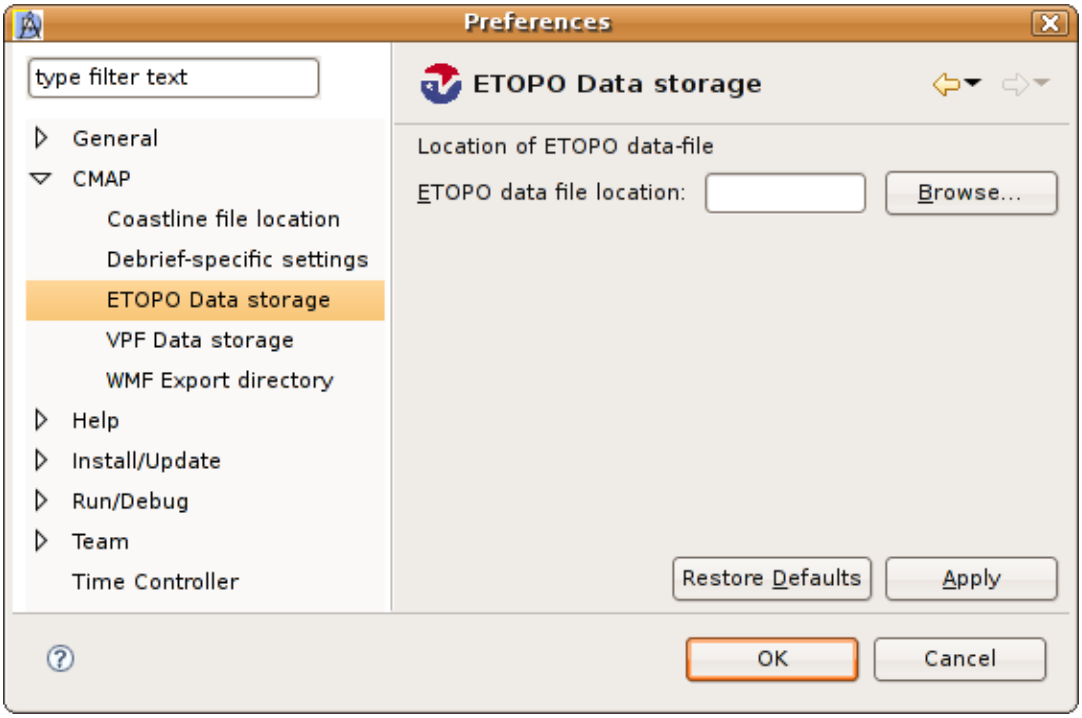
To test it, follow the steps described above (or copy the text I've just given you into a blank text file and save it as `default_layers.xml`). Then create a new Debrief session, load one of your demo tracks (`boat1.rep`), then drag in `default_layers.xml`. Zoom out, and have a look at the lovely, pre-formatted data.

4. ETOPO Data

4.1. Configuring Debrief to read VPF data

Debrief determines where to find the *ETOPO* data using the ETOPO Data storage tab of the Preferences dialog from the Window menu. Indicate the location of your ETOPO.RAW datafile using the file-browser button.

Figure 7.10. VPF preferences



4.2. Background to ETOPO data

Debrief 2002 added the capability to view gridded bathymetric data, provided through the ETOPO dataset. The ETOPO dataset and its use is explained in more detail in ETOPO gridded bathy

Note

The 'ETOPO-5' data set is originally from the U.S. National Geophysical Data Center (NGDC) in Boulder, Colorado (USA), and represents the "best" available digital terrain values as integrated from existing five and ten-minute digital sources. The data set has elevation values spaced at every five-minute latitude/longitude crossing on the global grid (approx. nine km.-sq. spatial resolution, or 12 x 12 pixels/degree), and a one-meter contour interval. Bathymetric values are included in this data set, starting at approximately 10,000 meters below sea level, while the elevation values extend up to heights of approximately 8,000 meters above sea level. Some original sources of the data used include the U. S. Defense Mapping Agency for the conterminous USA, Japan and Western Europe; the Australian Bureau of Mineral Resources, and the New Zealand Department of Scientific and Industrial Research.

GRID has reformatted the original NGDC data file to place the origin at 180 degrees West longitude, instead of at 0 degrees Greenwich Meridian. The 'ETOPO-5' data file has 2160 records of data with a length of 8640 bytes each: the size of the data array is 2160 lines by 4320 elements, but this is a l6-bit or two bytes per element data file. The origin of the data file is at 90 degrees North latitude and 180 West longitude, and it extends to 90 degrees South latitude and 180 degrees East longitude. The data file comprises 18.66 Megabytes. The version of this data file at GRID has been discovered to contain two records (lines) of flawed data values; that is, portions of lines 2055 and 2056, beginning at the Weddell Sea north of Antarctica and continuing eastward. GRID is currently waiting for a response from the data supplier (NGDC) before attempting any replacement of what appear to be anomalous data values.

There are two useful references for the 'ETOPO-5' data set. These are: "Edwards, Margaret Helen, 1986. Digital Image Processing of Local and Global Bathymetric Data. Master's Thesis. Department of Earth and Planetary Sciences, Washington Univ., St. Louis, Missouri, USA, 106 p." and "Haxby, W. F. et al., 1983. Digital Images of Combined Oceanic and Continental Data Sets and their Use in Tectonic Studies. EOS Trans- actions of the American Physical Union, vol. 64, no. 52, pp. 995-1004."

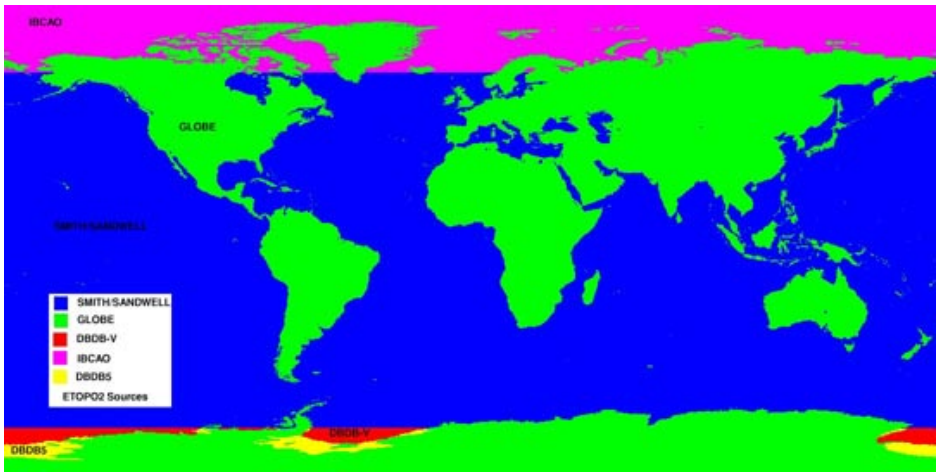
Note

In November 2002 the capability to read the US NOAA 'ETOPO-2' data set was been added to Debrief. The ETOPO-2 dataset may be purchased from the NOAA [http://www.ngdc.noaa.gov/mgg/gdas/gx_announce.Html], and is delivered on CD-ROM. The ETOPO-2 CD-ROM itself contains several datasets. The Dataset supported by Debrief is the `ETOPO2.RAW` file containing data in the *big-endian* format. This file must be copied into an `plugins \org.mwc.cmap.static_resources_1.0.0\data` sub-directory of the Debrief installation. This 2-minute dataset offers more than 6 times the detail of the ETOPO-5 dataset. This 2-minute data is derived from the following sources:

- The seafloor data between latitides 64 North and 72 South is from the work of *Smith and Sandwell, 1997*. These data were obtained from satellite altimetry observations combined with shipboard echo-sounding measurements.
- Seafloor data southward of 72 South are from the US Naval Oceanographic (NAVOCEANO) DBDBV version 4.1 at 5 minute spacing.
- Seafloor data northward from 64 North are from the new International Bathymetric Chart of the Arctic Ocean (IBCAO) Version 1.
- Land data is from the GLOBE Project, an internationally designed, and independently peer-reviewed global digital elevation model (DEM).

These sources are summarised below:

Figure 7.11. ETOPO2 Data Sources



This information has been taken from the NOAA Web-site [http://www.ngdc.noaa.gov/mgg/fliers/01mgg04.html]

The following options are provided for plotting ETOPO data:

Color	The color to plot the key itself.
Key location	This option determines where (and whether) to show the key for the depth data. Note that the ETOPO data will always appear behind other data, so it may be necessary to experiment with the key location.
Show land	Whether to plot land as land, or as very shallow water. Plotting the land as very shallow water is the favoured way of combining ETOPO data with VPF data. The VPF coastline data is of much higher resolution, differences being highlighted when ETOPO land is shown.
Visible	Whether to plot the ETOPO data

The ETOPO-2 dataset provides a few more customizable attributes, which allow you to choose between enhanced performance over appearance in the plot:

Bathy Res	This is the size of each bathy square to be plotted in screen pixels. A resolution (Res) of 1 pixel provides the most detailed bathymetric plot, though it takes the longest time. Frequently it is possible to increase the bathy-res without any visible degradation in the data displayed - whilst providing performance benefits
Bathy Visible	This flag indicates whether the bathymetric plot should be shown or not (sometimes the set of contours are sufficient).
Contour Depths	Into this box provide a comma-separated list of depth contours (in metres) to plot. The values should expressed in ascending order as whole numbers. If Debrief encounters a problem whilst reading the values it will return to the last valid set of depths
Contour Grid Interval	The contour plotting algorithm is very processor-intensive, calculating the contours when zoomed out over a large area can take many minutes. The contour plotting algorithm inspects every depth datum in the data area whilst finding contours. The speed of calculation can be improved by directing the algorithm to skip a number of depth values before performing the next

	calculation. This may make the contours themselves very slightly more blocky, though still useable.
Contour Optimise Grid	Setting this switch instructs Debrief to automatically increase the Contour Grid Interval as you zoom from the plot. This prevents the exponential increase in time take to conduct the contouring, ensuring that no more than 10000 calculations are performed, and when zooming in it ensures that where applicable at least 2000 calculations are performed.
Contours Visible	This flag indicates whether the contours should be plotted or not - not showing contours substantially speeds up the redraw time.

Tip

The ETOPO-5 dataset uses a significant amount of memory on your PC, typically 30Mb, though this only gets loaded once per Debrief session, however many plots are loaded. Writing a WMF file with ETOPO data visible requires even more memory, and can cause Debrief to hang or crash. This problem can be overcome by following the advice described in Section 1.3, “Starting the program” [83]

The ETOPO-2 dataset is many times larger than ETOPO-5, thus is not read into memory but accessed on the fly. For this reason it does not consume as much memory, but does require a fast PC to produce acceptable screen updates.

Chapter 8. Viewing Narratives

1. Introduction to narrative data

1.1. Introduction

The term Narrative Data is used within Debrief to refer to time-stamped textual data recorded during an exercise.

Typically this data would be narrative data recorded to provide an overview of events within an exercise *serial*, but an equally valid use of the facility would be for a textual record of information exported from a recording device. An example of this could be control messages returned from a weapon, or readings taken from an onboard sensor.

1.2. Preparing the data

Narrative data is loaded into Debrief using the ;NARRATIVE and ;NARRATIVE2 replay file format entries as described at Section 1.3, “Annotation Data” [88].

The narrative data can be located in a `.REP` file of its own, or together with other Debrief track and annotation data.

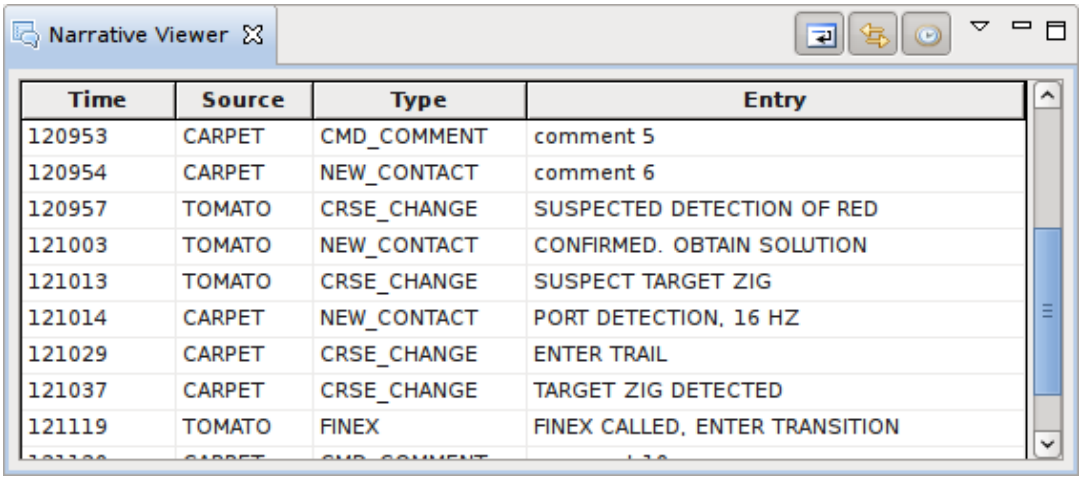
1.3. Loading the data

Narrative data is loaded into Debrief in the same way as other data, by dragging and dropping, or using the Import Data button.

1.4. Viewing the data

Once the data is loaded into Debrief, it is displayed in a list window - with one entry per line. When stepping through the data the "current" entry is highlighted, but the user is also able to double-click on an entry to move the Debrief step time to the time this entry was recorded.

Figure 8.1. Viewing a narrative



Time	Source	Type	Entry
120953	CARPET	CMD_COMMENT	comment 5
120954	CARPET	NEW_CONTACT	comment 6
120957	TOMATO	CRSE_CHANGE	SUSPECTED DETECTION OF RED
121003	TOMATO	NEW_CONTACT	CONFIRMED. OBTAIN SOLUTION
121013	TOMATO	CRSE_CHANGE	SUSPECT TARGET ZIG
121014	CARPET	NEW_CONTACT	PORT DETECTION, 16 HZ
121029	CARPET	CRSE_CHANGE	ENTER TRAIL
121037	CARPET	CRSE_CHANGE	TARGET ZIG DETECTED
121119	TOMATO	FINEX	FINEX CALLED, ENTER TRANSITION

Three settings affect how the Narrative Viewer integrates with the current plot time (as indicated on the Time Controller (see Section 2.1, “The Time Controller” [29]).

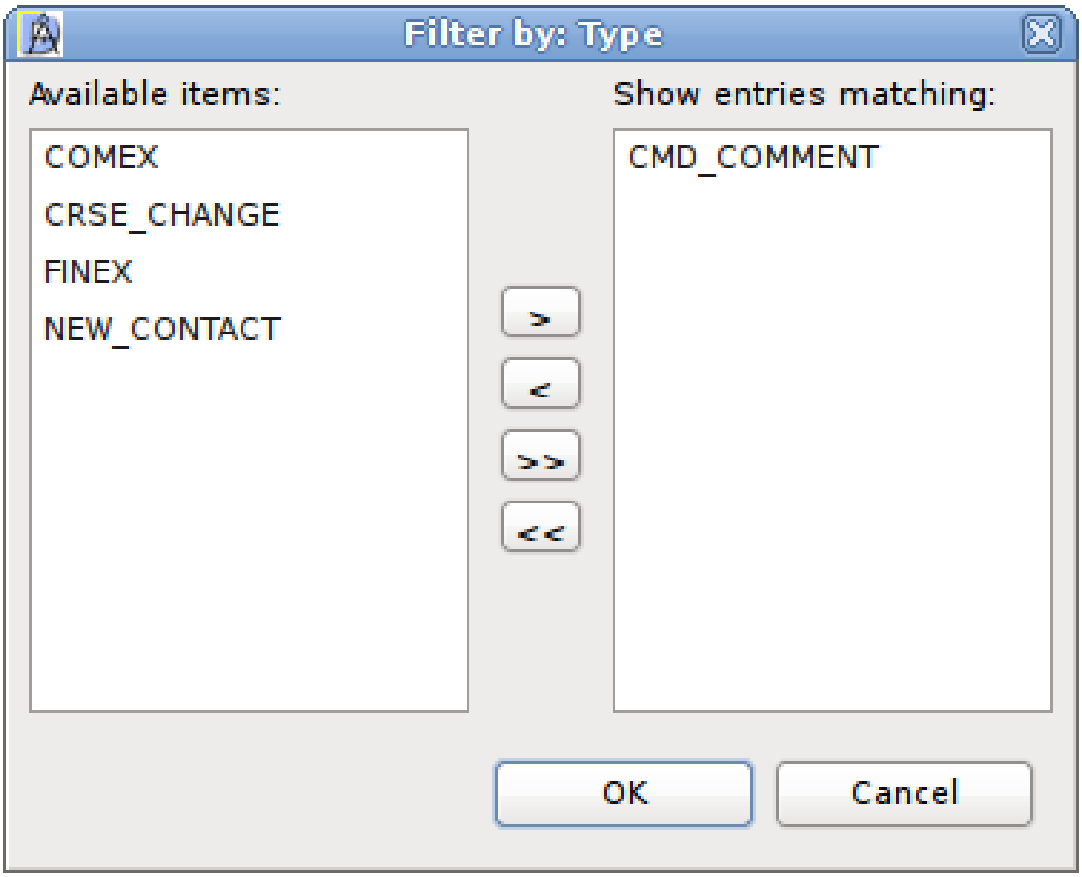
Trim the narrative entries to the visible screen space

Highlight the narrative entry nearest to the current plot time.

Change the plot-time to that of a narrative entry when the user double-clicks on it.

Also, the narrative viewer is able to filter your data for you. Double-click on the Source or Type headings and a filter dialog will open.

Figure 8.2. Filtering a narrative



In the example shown, select which types of data you wish to see in the filtered narrative by either double-clicking on them or selecting one then clicking the right arrow. When you've selected which data you want to see, click OK

The drop-down menu for the narrative viewer also allows you to select how you want the date column formatted - helpful for very frequent or infrequent narrative entries.

Chapter 9. Debrief video recording

1. Recording plots to video

1.1. Introduction

As its name suggests, one of the initially intended uses for Debrief was to conduct post-exercise debriefs.

Nowadays Debrief gets a lot more use for exercise analysis, and has evolved into an exercise analysis tool, but there are still aspirations to use it in debriefs. Unfortunately, its use in a debrief normally requires an expert operator; who is not always available. Recording a Debrief session to file, however, enables it to be inserted into a PowerPoint presentation for automated replay.

Debrief 2003 included experimental "Record to video" functionality. This feature was no match for the commercial equivalents. Applications such as HyperCam (http://www.hyperionics.com) allow more intuitive video recording, with much smaller file sizes - all for a very reasonable price. Until better performing video recording functionality becomes available in Java/Eclipse, purchase of off-the-shelf screen-recording video is the recommended route.

Chapter 10. Analysing sensor data

1. Sensor data

1.1. Introduction

As useful as it would appear it is only in the update to Debrief 2001 that Debrief has been able to properly show *sensor* data.

A rudimentary way of showing bearing lines has been present for a number of years, but Debrief has not performed any useful processing on this data.

The following terms are used:

Sensor	The *sensor* which recorded the data. Sensor data is grouped according to its sensor. This characteristic may be exploited by giving a unique sensor name to each track being recorded on a sensor - allowing tracks to be independently switched on and off.
Sensor Contact	This is an individual contact recorded on a sensor, a single bearing line reaching from the sensor location (origin) along the contact bearing to the contact range.

The sticky issue of whether to represent sensor data in absolute coordinates (where each line has it's own origin) or whether to represent the data in relative coordinates (where the sensor origin is assumed to be the current ownship position) is managed through the ability to enter NULL fields for the sensor location.

Support for relative coordinates is provided for two reasons:

- Some *sensor* data sources may genuinely not contain positional data - allowing relative coordinates will ease the workload in these instances.
- In the submarine plot-lock process it is quite common to experiment with a number of track-shifts until the hull-mounted sensor bearing fans tie up with the other vessel. By keeping the sensor data separate to the vessel data, the existing sensor data-file can be dropped into Debrief with the updated vessel track-file, allowing the user to perform a visual qualitative check on the shift applied.

1.2. Preparing Sensor Data

Sensor data is loaded into Debrief in REP files, just like any other Debrief data. The line format is:

```
                ;SENSOR: YYMMDD HHMMSS.SSS AAAAAA @@ DD MM SS.SS H DDD MM SS.SS H
 ;; date, ownship name, symbology, sensor lat/long (or the single word NULL), b
```

As you can see, unlike other Debrief line formats, this format allows for NULL fields. Where the sensor latitude and longitude values are replaced by the single word NULL, Debrief plots this sensor contact using a relative origin.

1.3. Relative Data

When (as described above) Debrief plots data using a relative origin, it follows the following procedure:

- The first time the sensor contact line is plotted, it examines its parent track to find the vessel position nearest to (or greater than) the sensor contact DTG.

- The sensor contact then calculates the position of its far end relative to this origin

1.4. Analysing sensor data

When first loaded, *sensor* data is not made visible, since with any reasonable volume of sensor data the plot quickly becomes illegible. Sensor data is switched on and off individually by accessing the sensor, via it's Track, from the Layer Manager.

It is once in *snail trail* mode that sensor data is most easily analysed. When in snail mode the Snail display mode performs the following processing:

- For each Track being plotted, the display mode looks to see if it contains any Sensor data.
- It then examines each list of Sensor data to see if it's visible If it is visible, it plots the current sensor contact (nearest to the *Tote* time), followed by the sensor contacts as disappearing contacts running back through the length indicated in the TrailLength parameter in the properties window.

Chapter 11. TMA solution data

1. Introduction to TMA data

1.1. Introduction

TMA algorithms are used to produce an estimate of range by analysing a sequences of sensor contacts which only contain bearing (and optionally range). The output of the algorithm is typically a series of estimated target locations optional estimates for course, speed and depth. Uncertainty in bearing and range may be indicated by representing the target location as an ellipse.

TMA contact data is always related to one of the currently loaded tracks, and may be represented either as an absolute location (at the centre of the ellipse) or as a range and bearing from the nearest point on that loaded track. The strategy for use of absolute versus relative data is described earlier in Section 1.2, “Preparing Sensor Data” [71]

1.2. Preparing TMA Data

Sensor data is loaded into Debrief in REP files, just like any other Debrief data. The line format is:

```
;TMA_POS: YYMMDD HHMMSS.SSS AAAAAA @@ DD MM SS.SS H DDD MM SS.SS H
TT...TT OOO.O  XXXX YYYY  CCC SSS DDD xx.xx
;; date, time, ownship name, symbology, tma lat, tma long, track name,
ellipse orientation (deg from north),  maxima (yds), minima (yds),
course, speed, depth (m), label string

;TMA_RB: YYMMDD HHMMSS.SSS AAAAAA @@ BBB.B RRR.R TT...TT OOO.O
XXXX YYYY  CCC SSS DDD xx.xx
;; date, time, ownship name, symbology, bearing (deg), range (yds),
track name, ellipse orientation (deg from north),  maxima (yds),
 minima (yds), course, speed, depth (m), label string
```

Note

There are two annotation format to represent TMA solutions (TMA_POS and TMA_RB). TMA_POS is used to define a TMA solution at a particular location, and TMA_RB is used to define a TMA solution at a specific range and bearing from the current ownship location at that specific DTG. Where a solution ellipse is not known the orientation, maxima and minima values can be represented by a single NULL value.

1.3. Relative Data

When (as described above) Debrief plots data using a range and bearing, it follows the following procedure:

- The first time the TMA solution is plotted, it examines its parent track to find the vessel position nearest to (or greater than) the TMA solution DTG.
- The TMA solution then calculates the position of its centre relative to this origin

1.4. Analysing sensor data

When first loaded, TMA data is not made visible, since with any reasonable volume of TMA data the plot quickly becomes illegible. TMA data is switched on and off individually by accessing the whole TMA track, via it's Track, from the Layer Manager.

It is once in *snail trail* mode that TMA data is most easily analysed. When in snail mode the Snail display mode performs the following processing:

- For each Track being plotted, the display mode looks to see if it contains any TMA data.
- It then examines each list of TMA data to see if it's visible If it is visible, it plots the current TMA solution (nearest to the *Tote* time), followed by the TMA solution as disappearing solutions running back through the length indicated in the TrailLength parameter in the properties window.

Part 2. Maintainer's Guide

This section will give you all you need to look after a collection of Debrief installations, from fault-finding through to optimising the installations used.

Chapter 12. Participating in Debrief development

1. Debrief online

Whilst the use of Debrief originated at the *MWC*, it now has a very distributed community of users. Communication across this distributed user-group is enabled via the Internet through three mechanisms: the Debrief home page, the SourceForge project management page, and the Debrief news list. These mechanisms are described in this chapter.

The Debrief web-site, found at http://www.debrief.info provides an introduction to Debrief, together with access to this tutorial and a *way-in* to getting started with Debrief.

Figure 12.1. Debrief home page

The SourceForge.net page, used to support the project management of Debrief, is described later in Providing Feedback, and the Debrief news list is described in the next section.

2. Keeping up to date

News lists (more commonly known as mailing lists) are one of the SourceForge.net services which are employed by Debrief. The lists provide an e-mail based way of communicating Debrief news and announcements amongst a list of interested users. Once an e-mail is sent to the e-mail address

of the newsgroup (in your case hosted at *SourceForge.net*), it is automatically forwarded to all of registered news-list members. A separate e-mail address (the news-list title with a *request* suffix) provides management support to the newsgroup.

As of summer 2001 there are around a dozen users registered on the news list, with the Debrief Project Manager publishing news each time a new version of Debrief is made available for download.

There is currently just one mailing list related to Debrief, this is named debrief-news. This is quite a low volume mailing list, with at the most two messages a month; so there's no need to worry about getting snowed-under with news items.

2.1. Joining

To join to the Debrief news list, send an e-mail to `<debrief-news-request@lists.sourceforge.net>` with the single word *subscribe* in the subject. You will receive a reply asking you to confirm that you wish to subscribe. This e-mail is to validate your e-mail address, and should be retained since it contains the password you must use to unsubscribe.

2.2. Submitting

Once you have joined the list, you are then entitled to submit news, should you wish to. So, if you have identified or fixed a significant bug, or have contributed significant content to the Debrief web-site, please feel free to e-mail the news list directly. Just send an e-mail to `<debrief-news@lists.sourceforge.net>`

2.3. Leaving

If you wish to leave the news list, just send an e-mail to `<debrief-news@lists.sourceforge.net>` with the single word *unsubscribe* in the subject line followed by the password you were provided with at your initial subscription. If you have lost your password please make a request to the Project Manager to be unsubscribed.

2.4. Web-based news-list management

In addition to these e-mail based methods of interacting with the Debrief news list, there is a web-page based interface at: http://lists.sourceforge.net/lists/listinfo/debrief-news. From this page you can subscribe, unsubscribe, and configure options for how the news messages are sent to you.

3. Debrief at the Maritime Warfare Centre

This section provides guidance on maintenance of Debrief installations for users on the Maritime Warfare Centre network.

3.1. Installing Debrief

The IT Support department should be approached to conduct fresh Debrief installations. The most up-to-date Debrief installer is located at `//Needles/Operational Analysis/collaborative/application/debrief/InstallDebrief.exe`.

3.2. Keeping your Debrief installation up to date

The development of the Debrief application is ongoing throughout the year. Occasionally particular users have new requirements, or bugs are found. Normally, only users working in related areas are informed about new features or minor bugs that only occur when performing that particular type of analysis. Where new features relate to users across MWC, or where a significant bug is fixed, e-mails are sent out to the Debrief user mailing list.

Debrief Users mailing list. On the MWC network mailing lists have been set up to allow e-mails to quickly be addressed to a group of users. One such list is the *Debrief Users* list. When you start using Debrief it is worth contacting IT Support to have your name added to the list - then you know you will be informed of any significant changes to Debrief.

Don't worry about always having the newest version of Debrief. Most software changes only apply to a small portion of the software and don't justify the upgrade effort. If, however, a Debrief fix applies specifically to your work, you should follow these steps to upgrade your Debrief installation:

Searching for software updates

1. Select Help/Software Updates/Find and install.
2. Select Search for updates of the currently installed features, then Next.
3. Follow the instructions provided to view what updates are available, select any relevant ones (though you can't really go wrong in retrieving all of them), and install them. Easy.

Now, Debrief may now have found any updates. Hey, maybe there weren't any. But maybe Debrief's not looking in the right place. If no updates are found, follow these steps to ensure the MWC download server is configured:

1. Select Help/Software Updates/Find and install.
2. Select Search for new features to install
3. You will see a list of download sites. If there isn't one with a name you associate with MWC, select New Local Site....
4. A file dialog will open. From it, enter `Network Neighbourhood` , and navigate to `//Needles/Operational Analysis/collaborative/applications/Debrief/Updates`. Then press OK and name the local site as **`MWC Updates`**. Accept this. Now, from the list of update sites, ensure **`MWC Updates`** is ticked, but not the others (since they rely on an internet connection).
5. Now you should be able to perform the search for software updates again.

4. Debrief across the Internet

This section provides guidance on maintenance of Debrief installations for users with world wide web (Internet) access.

4.1. Installing Debrief

Debrief can be downloaded via the downloads page at the Debrief site (http://sf.net/projects/debrief [http://sf,net/projects/debrief]). Be warned, because the full install contains the Etopo data set it is well over 150 Mb in size. If you aren't able to perform such a download, or you need a formal copy for your IS/IT department please send an e-mail justification to the project manager, and a CD-based copy will be forwarded to you via snail mail.

4.2. Keeping your Debrief installation up to date

The development of the Debrief application is ongoing throughout the year. Occasionally particular users have new requirements, or bugs are found. Normally, only users working in related areas are informed about new features or minor bugs that only occur when performing that particular type of analysis. Where new features relate to a wide body of users , or where a significant bug is fixed, e-mails are sent out to the Debrief user mailing list.

Don't worry about always having the newest version of Debrief. Most software changes only apply to a small portion of the software and don't justify the upgrade effort. If, however, a Debrief fix applies specifically to your work, you should follow these steps to upgrade your Debrief installation:

Searching for software updates

1. Select Help/Software Updates/Find and install.

2. Select Search for updates of the currently installed features, then Next.

3. Follow the instructions provided to view what updates are available, select any relevant ones (though you can't really go wrong in retrieving all of them), and install them. Easy.

Now, Debrief may now have found any updates. Hey, maybe there weren't any. But maybe Debrief's not looking in the right place. If no updates are found, follow these steps to ensure the Debrief update server is configured:

1. Select Help/Software Updates/Find and install.

2. Select Search for new features to install

3. You will see a list of download sites. If there isn't one with a name you associate with Debrief, select New Remote Site....

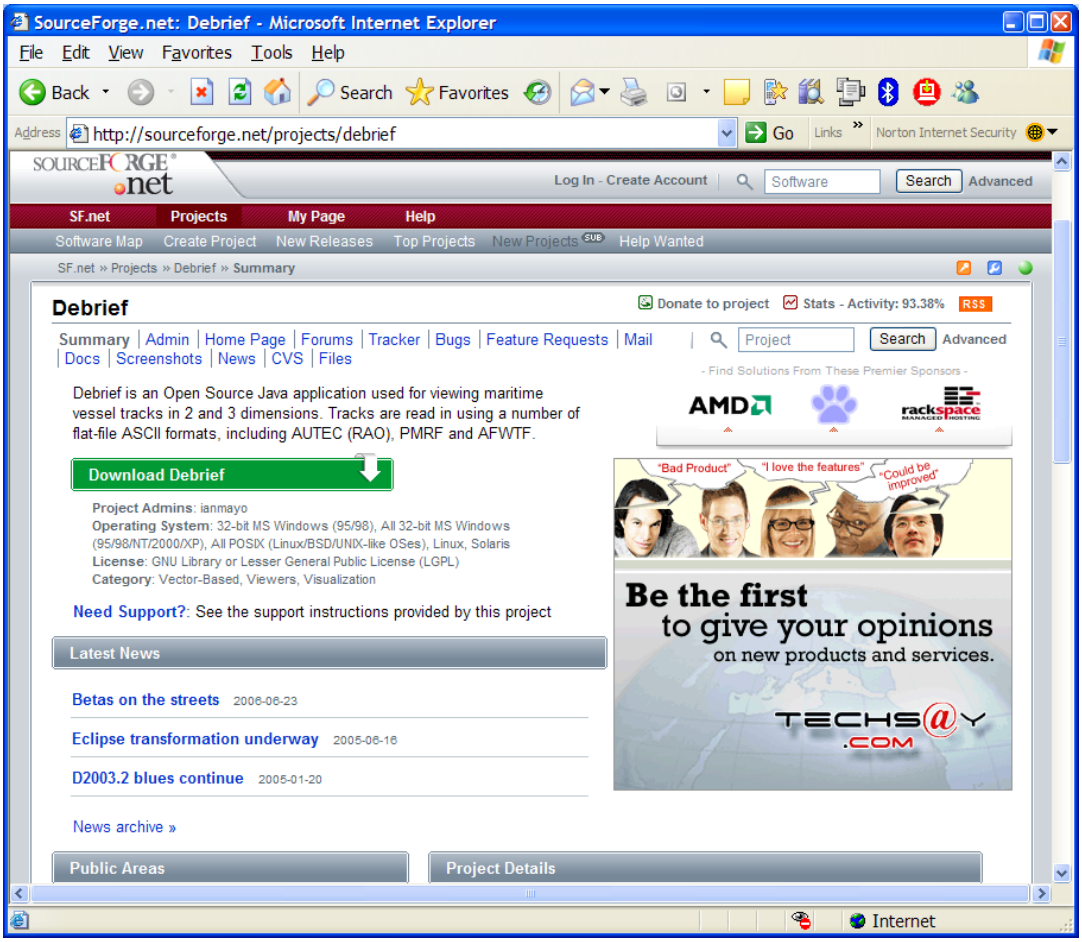
4. A file dialog will open. In it, enter `Debrief update site` for the Name, and `http://debrief.sourceforge.net/eclipse/` for the URL. Now, from the list of update sites, ensure **`Debrief update site`** is ticked, but not the others (since they rely on an internet connection).

5. Now you should be able to perform the search for software updates again.

5. Providing feedback

In addition to the Debrief web-site, the Debrief project makes use of SourceForge, an online development web-site which provides us with bug-tracking, file download support, and news groups.

Figure 12.2. SourceForge.net home page



From the screenshot of the SourceForge.net home page for Debrief you can see a list of the services it provides, ranging from discussion forums, through bug and feature-request trackers, to news lists and file downloads. All of these services are provided for free, funded by VA Linux Systems [http://valinux.com/].

SourceForge.net 's mission is to enrich the Open Source community by providing a centralized place for Open Source Developers to control and manage Open Source Software Development.

SourceForge is a free hosting service for Open Source developers which offers, among other things, a CVS repository, mailing lists, bug tracking, message forums, task management software, web site hosting, permanent file archival, full backups, and total web-based administration.

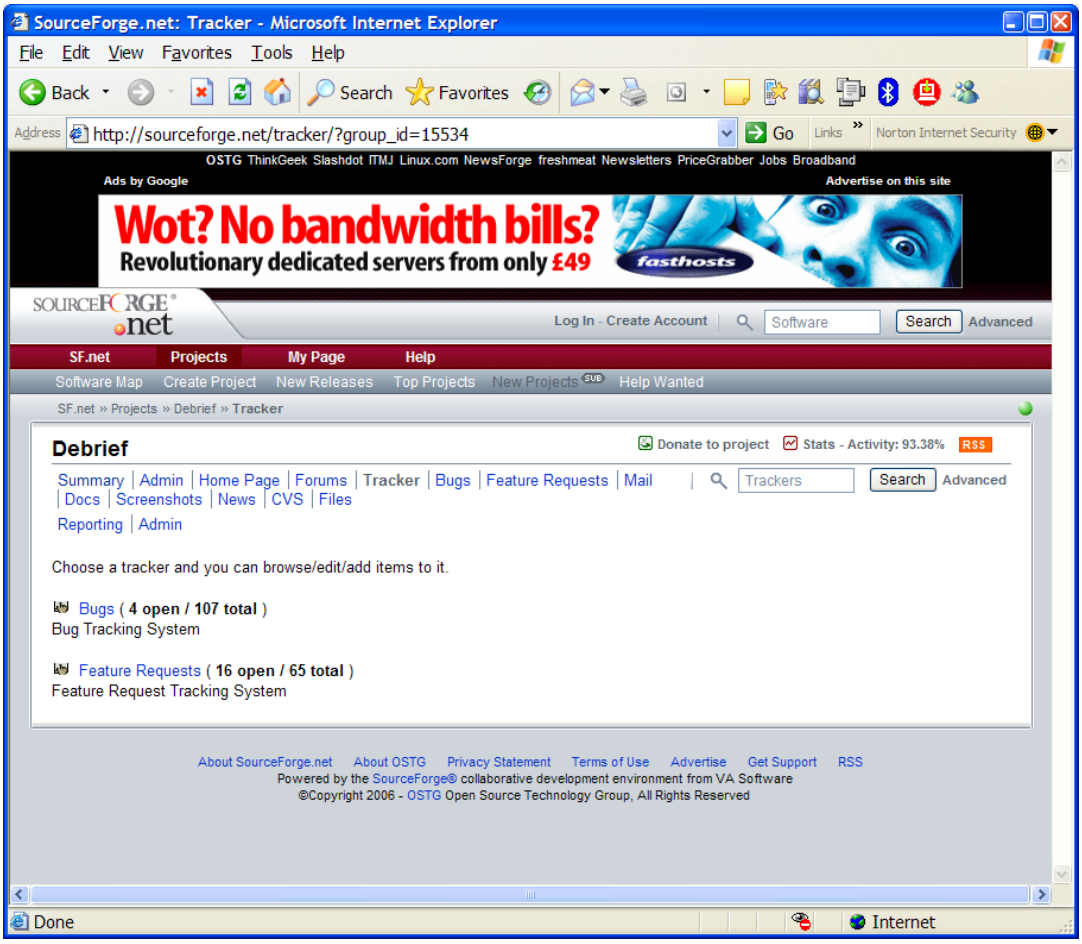
5.1. Reporting Bugs

To report a bug, or a feature which you believe is behaving incorrectly, press the Tracker link from the SourceForge.net homepage for Debrief. These *Trackers* allow registered and anonymous

users to submit items to the lists. The items are then tracked, and provided an e-mail address was provided, the submitter is kept informed of the progress of the submission.

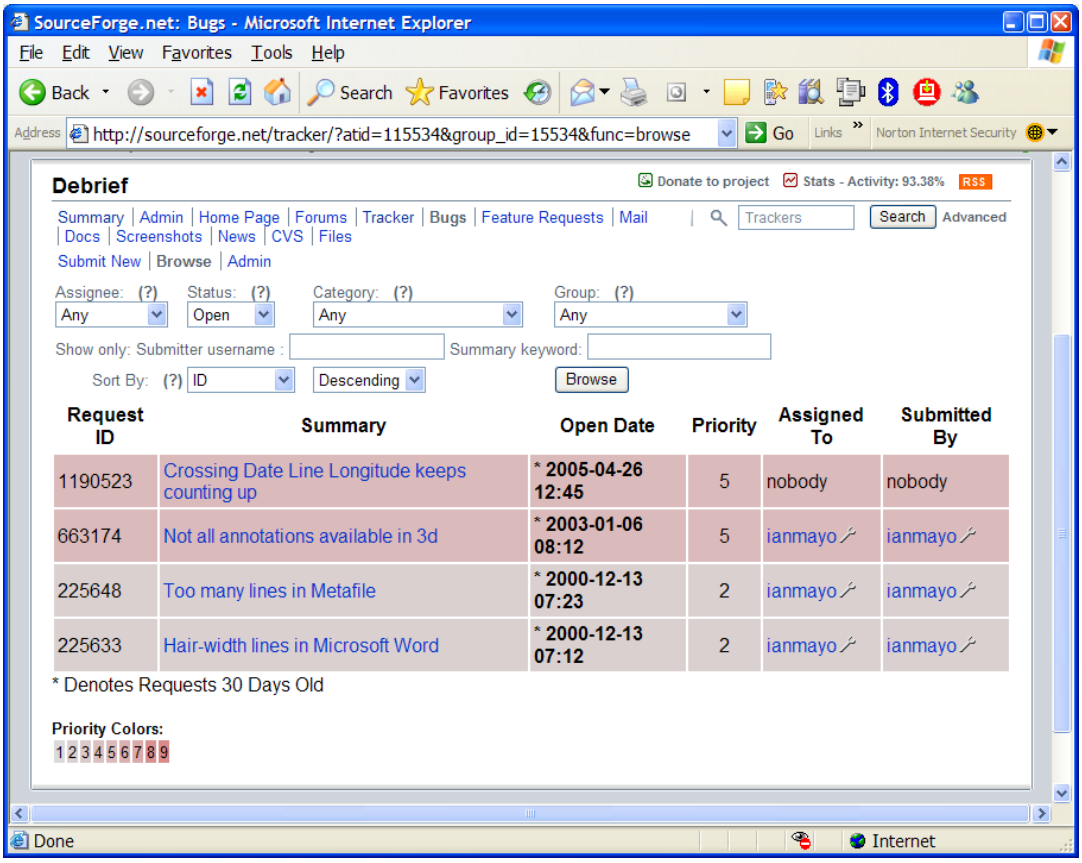
The Trackers page provides a list of types of data which may be tracked.

Figure 12.3. SourceForge.net Trackers page



To report a new bug, click on the Bugs link from this page. You will be taken to a page (see below) which allows you to both browse existing bugs and add new ones. To add a new bug, press the Submit New button, which will take you to the form used to submit new bug reports. Select the correct categories for the new bug using the categories provided, then a summary (title) and description of the bug. The form also requests your e-mail address, this is most useful since it allows the Project Manager to approach you with any supplementary questions.

Figure 12.4. SourceForge.net Bug Reporting page



Once you have submitted the report, you should receive an e-mail confirmation of the bug-report, followed by an e-mail from the Project Manager, normally containing a time-frame for resolution of the bug.

Tip

If you find yourself frequently submitting bugs via SourceForge.net (hopefully not) then it may prove worthwhile to register with SourceForge.net. Registering gives you your own ID within SourceForge.net, and gives you a dedicated personal page which allows you track the progress of all bug-reports and feature requests submitted.

5.2. Requesting new features

The procedure for requesting new features is much the same as for Reporting Bugs, described above. The time-frame for resolution of feature-requests is normally larger than for bugs, but if maintenance effort is being performed in a related area of Debrief, and the Project Sponsor is supportive, then the new feature should get added quite quickly.

Chapter 13. Debrief maintainer's guide

1. Installation Guidance

1.1. System Requirements

Table 13.1. Requirements for Debrief

Requirement	Minimum	Recommended
Processor	800 MHz	1400 MHz
Memory	256 Mb	1024 Mb
Java Virtual Machine	JVM 1.4	As new as possible. JVM 1.5 includes significant performance improvements, particularly in the list processing used extensively within Debrief
Java 3D Libraries (required for viewing Debrief plots in 3D [a])	Java 3D 1.3	Java 3D 1.3
XML Libraries	SAX and DOM support (through the enclosed jaxp.jar and parser.jar libraries)	
OpenMap libraries (used for plotting NIMA's Vector Map data)	4.2.1 (through the enclosed openmap.jar library)	

[a] To interactively view Debrief plots in 3 dimensions the Java 3d runtime files and libraries must be installed on the machine. These libraries are a free download from the Java website.

1.2. Directory Structure

Debrief installs itself into a `Debrief NG` folder within your `Program Files`. Within the Debrief NG folder you will find the DebriefNG executable in the top-level, together with some additional `dll` files used for optimised graphics (`gdiplus.dll`), and copy to clipboard (`JavaClipboard.dll`).

The `jre` folder contains a Java Runtime Environment that's been modified by the inclusion of a current set of Java3d libraries. In the absence of this `jre` folder Debrief (Eclipse) would use the system's default Java installation, if there was one. Including it overrides any other configured library.

The `configuration` folder contains details of Debrief's initial settings together with the locations of update download sites.

The `features` and `plugins` folders include details of what Eclipse plugins are present (where each feature is actually a group of plugins), together with the plugins themselves. The `workspace` folder is a default workspace provided for new users. It contains current work projects together with a large volume of metadata representing change history, screen layout and user preferences.

1.3. Starting the program

The shortcut placed on the Start|Program Files|Debrief NG menu actually calls the `DebriefNG.exe` file found in the startup directory. This application triggers the startup of

Debrief, passing to it the parameters contained in the `DebriefNG.ini` file. This file contains options to select the initial perspective, the memory allocation, and additional files to include on the classpath (3d files in our case).

2. Fault-Diagnosis Instructions

2.1. How-to

With the adoption of the Eclipse framework, Debrief enjoys the benefits of the Eclipse configuration and logging engine. Debrief uses this for information, warning, and error reporting.

Details of an individual Debrief NG installation can be obtained by selecting About Debrief NG from the Help menu. From the dialog that opens, select Configuration Details. This will open a text dialog providing the following information:

- Date stamp
- System properties (about the Java and OS environments)
- Plug-in registry (plug-ins and fragments - ids, versions, and names)
- Update manager log
- Current error log contents

In addition to these configuration details is a button to View Error Log. The Error Log is a continuous stream of comments recorded by Debrief - typically comments that are not of value to a typical user under normal usage. The log itself is called `.log` and is located in the `workspace/metadata` folder within your Workspace (initially, this is within your Debrief NG intallation directory).

Th error log may be examined for hints to the problem occurring. The error trace may be viewed in a text editor, and may be forwarded back to the Project Manager via post/e-mail/fax as applicable. The file does not contain any details of the data being edited, but the file may still be inspected before transmission

If these error messages do not provide any insight to the problem, you are left with the normal diagnosis steps as follows:

- Try to run another Java application on the machine, to check that Java is not corrupted
- If Java is ok then,
  - Check that Debrief can open and process the "sample" files included in the installation: boat1.rep and boat2.rep.
  - If Debrief can process these files, but not the current ones then
    - Try to re-install the Debrief application from 'save' disks to check that the sources are not corrupted
    - If at this point the error is still occurring then you will have to resort to reading as much of this Help guidance as practical, and then contacting the Project Manager

3. Debrief Properties

3.1. Introduction

Whilst Debrief used to use a dedicated settings file, the settings have now been incorporated into the general framework of the application - with context sensitive help available where applicable.

Part 3. Reference Guide

If the tutorial hasn't provided you with all of the answers you need, just have a look at the pages here in the Reference Guide.

Chapter 14. Debrief File Formats

1. Replay File Format

1.1. Track Data

Debrief accepts files in the Replay datafile format. This format uses an 80 character string containing data/time, vessel name, position, heading, depth and speed. More than one vessel track can be stored in each file. The gaps between the data fields can be any whitespace characters, that is any number of spaces or tabs, but there must be a value for each data field.

Note

Whilst file formats recorded here are shown against an 80 character index marker - the fields can be of any width - the data is broken down using delimiters, not fixed field widths.

The file-naming convention for files in the *Replay* format is to have a `.REP` suffix.

Through an extension to the format, annotation data can also be accepted. The annotation data format is described below. Whilst annotation and track data may be stored in single file, it is recommended that they are stored in separate files, to ease reading and plotting track-data only.

The standard data format is as follows

```
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
YYMMDD HHMMSS.SSS XXXXXX SY DD MM SS.SS H DDD MM SS.SS H CCC.C SS.S DDD
```

or since Autumn 2004 multi-word track names can be entered by enclosing them in double-quotation marks (") and 4-figure year values are supported:

```
123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
YYYYMMDD HHMMSS.SSS "XYX XYX XYX" SY DD MM SS.SS H DDD MM SS.SS H CCC.C SS.S DDD
```

The field descriptions are:

Table 14.1. Fields in Debrief position entry

Field number	Description
1	Date, either 2 of 4 figure date, followed by month then date
2	Time
3	Vessel Name - either as single, unquoted word, or as a multi-word phrase enclosed in quotation marks.
4	Symbology (see below)
5	Latitude Degrees
6	Latitude Minutes
7	Latitude Seconds
8	Latitude Hemisphere
9	Longitude Degrees
10	Longitude Minutes
11	Longitude Seconds
12	Longitude Hemisphere
13	Heading (0..359.9 degrees)
14	Speed (knots)
13	Depth (metres) [a]

[a]Where depth data is not available, the IEE symbol NaN (Not a Number) may be used. Debrief reflects the absence of this data where applicable.

1.2. Symbology attributes

The symbology data field describes the representation required for this vessel track, specifying the colour of track to use and symbol-type to represent that vehicle. The Symbol style read in is used when tracks are to be shown by using symbol (in 2D or 3D), as described in Section 1.1, “Introduction” [49] .

Table 14.2. Debrief symbology data fields

Symbol Style		Colour	
A	Aircraft	@	White
C	Carrier	A	Blue
D	Destroyer	B	Green
F	Frigate	C	Red
H	Helicopter	D	Yellow
M	Merchant	E	Purple
N	Minesweeper	F	Orange
P	Torpedo	G	Brown
R	Troop Carrier	H	Cyan
S	Submarine	I	Light Green
T	TA Frigate	J	Gold
U	Cruiser	K	Pink
V	Fishing Vessel		
@	Unknown		

Note

Note that historically only the first color style read in from file for a particular track was implemented, this has been changed in order that the colour value is used for each position read in.

1.3. Annotation Data

In addition to the positional vessel track data, annotations can also be added. Each annotation is placed on a single line in a replay file, each line beginning with the comment marker; a semi-colon ';'.To put a comment on a line, begin with two semi-colons.

The format for the different types of annotations is:

```
1234567890123456789012345678901234567890123456789012345678901234567890123456789
012345678901234567890123456789

;LINE: @@ DD MM SS H DDD MM SS.S H DD MM SS.S H DDD MM SS H XX.XXX
;; symb, start lat & long, end lat & long, text label (optional)

;TEXT: @@ DD MM SS H DDD MM SS H XX..XX
;; symb, lat & long, text

;NARRATIVE:  YYMMDD HHMMSS  TTT.TTT  XX..XX
;; dtg, track name, narrative entry

;TIMETEXT: @@ YYMMDD HHMMSS DD MM SS.S H DDD MM SS.S H XX..XX
;; symb, date, lat & long, text

;PERIODTEXT: @@ YYMMDD HHMMSS YYMMDD HHMMSS DD MM SS.S H DDD MM SS.S H
DDD XX..XX
;; symb, start date, end date, lat & long, Depth(optional) text

;GRID: @@ DD MM SS.S H DDD MM SS.S H MM.MM MM.MM XX..XX
;; symb, centre lat & long (N13) lat increment, long increment, text

;RECT: @@ DD MM SS.S H DDD MM SS.S H DDMMSS H DDDMMSS H
;; symb, tl corner lat & long, br corner lat & long

;SGSAGEOG: @@ YYMMDD HHMMSS YYMMDD HHMMSS DD MM SS.S H DDD MM SS.S H
DD MM SS.S H DDD MM SS.S H XX..XX
;; symb, start date, end date, tl lat & long, br lat & long,
text label (1 word)

;SGSAGRID: @@ YYMMDD HHMMSS YYMMDD HHMMSS AA NN AA NN XX..XX
;; symb, start date, end date, tl corner, br corner, text label (1 word)

;WHEEL: @@ YYMMDD HHMMSS DD MM SS.S H DDD MM SS.S H YYY YYY XX.XX
;; symb, centre date, centre lat, centre long, inner radius (yards), outer radi

;CIRCLE: @@ DD MM SS.S H DDD MM SS.S H YYY XX.XX
;; symb, centre lat, centre long, radius (yards), label

;ELLIPSE: BD YYMMDD HHMMSS DD MM SS.SS H DD MM SS.SS H CCC XXXX YYYY xx.xx
;; symb, date, time, lat, long, orientation, maxima (yards),
minima (yards), label
```

```
;BRG: BD YYMMDD HHMMSS DD MM SS.SS H DD MM SS.SS H CCC XXXX xx.xx
;; symb, date, time, lat, long, orientation, length (yards), label (one word)

;SENSOR: YYMMDD HHMMSS.SSS AAAAAA @@ DD MM SS.SS H DDD MM SS.SS H
BBB.B RRRR yy..yy xx..xx
;; date, ownship name, symbology, sensor lat/long (or the single word NULL),
bearing (degs), range(yds), sensor name, label (to end of line)

;TMA_POS: YYMMDD HHMMSS.SSS AAAAAA @@ DD MM SS.SS H DDD MM SS.SS H
TT...TT OOO.O  XXXX YYYY  CCC SSS DDD xx.xx
;; date, time, ownship name, symbology, tma lat, tma long, track name,
ellipse orientation (deg from north),  maxima (yds), minima (yds), course,
speed, depth (m), label string

;TMA_RB: YYMMDD HHMMSS.SSS AAAAAA @@ RRR.R BBB.B TT...TT OOO.O
XXXX YYYY  CCC SSS DDD xx.xx
;; date, time, ownship name, symbology, bearing (deg), range (yds),
track name, elipse orientation (deg from north),  maxima (yds), minima (yds),
course, speed, depth (m), label string

;NARRATIVE:  YYMMDD HHMMSS  TTT.TTT  XX..XX
 ;; dtg, track name, narrative entry. Note the track name can be multi-word if

;NARRATIVE2:  YYMMDD HHMMSS  TTT.TTT AA..AA XX..XX
 ;; dtg, track name, narrative entry-type, narrative entry.  Note the track nam
```

Note

Note, when null positions are entered for *Sensor* data, the position is taken from the track named in "ownship name". Debrief finds the nearest track location equal to or greater than the indicated time, and uses this as the sensor lat and long.

Note

As with normal positional data, annotation entries which use a track/ownship name may now contain multi-word phrases if they are enclosed in quotation marks ("), and the year can be expressed as 2 or 4 figures.

Annotation positions are specified in degrees as in the standard replay file format, and symbology representations are as in the above tables.

2. Debrief File Format

2.1. Introduction

XML is the Extensible Markup Language. Like its predecessor SGML, *XML* is a meta-language used to define other languages. The article in the Reference Guide taken from MSDN gives some background to *XML*.

2.2. Adoption of XML

An acknowledged shortcoming of the former file format (`dpl`) used in Debrief was the frequency with which Debrief modifications rendered previous `DPL` files unreadable. This use of Java

serialization offered low cost in development terms and was seen as essential to store the more complex data used in Debrief 2000 (such as projections, and formatting data).

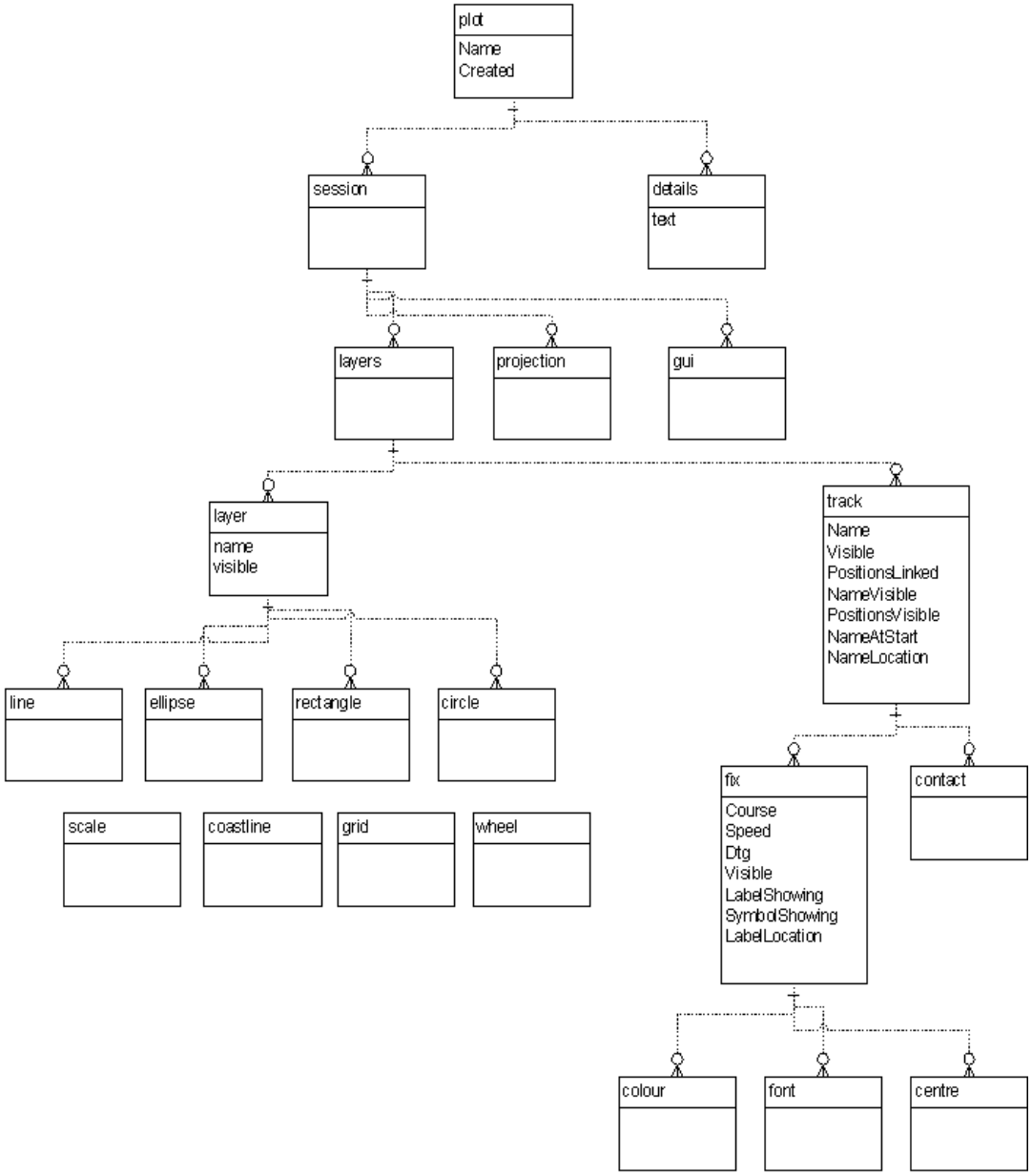
The only way to divorce the Debrief version from the file format has been to adopt an independent file format. This independent format must be structured in order to store Debrief data tidily, and the development to support it must not be too "expensive". The adoption of XML meets both of these criteria = it is quite capable of storing Debrief data, and there are a number of libraries available which support it.

XML provides additional benefits beyond those described above:

- It is human-readable, offering the following benefits:
  - Apparently corrupt data files can be examined in a text editor and repaired
  - Data files can be edited outside Debrief
- It is a world-wide standard independent of Debrief and Java
  - Applications are freely available which support it: Internet Explorer will open XML data files for inspection
  - When used in conjunction with the Debrief DTD, the IBM Xeena editor provides easy editing/ creation of Debrief plots
  - As more organisations/industries adopt XML, there is increased likelihood that import/export filters will not be required, since XML documents are easily translated between each other.

The XML structure adopted within Debrief is as follows:

Figure 14.1. XML File structure in Debrief



3. Introduction to XML

Extensible Markup Language (*XML*) is a meta-markup language that provides a format for describing structured data. This facilitates more precise declarations of content and more meaningful search results across multiple platforms. In addition, XML will enable a new generation of Web-based data viewing and manipulation applications.

3.1. Extensible

In XML you can define an unlimited set of tags. While HTML tags can be used to display a word in bold or italic, XML provides a framework for tagging structured data. An XML element can declare its associated data to be a retail price, a sales tax, a book title, the amount of precipitation, or any other desired data element. As XML tags are adopted throughout an organization's intranet, and by others across the Internet, there will be a corresponding ability to search for and manipulate data regardless of the applications within which it is found. Once data has been located, it can be delivered over the wire and presented in a browser such as Internet Explorer 5 in any number of ways, or it can be handed off to other applications for further processing and viewing.

3.2. Structural representation of data

XML provides a structural representation of data that has proved broadly implementable and easy to deploy. Industrial implementations in the Standard Generalized Markup Language (SGML) community and elsewhere demonstrate the intrinsic quality and industrial strength of XML's tree-structured data format.

XML is a subset of SGML that is optimized for delivery over the Web; it is defined by the World Wide Web Consortium, ensuring that structured data will be uniform and independent of applications or vendors. This resulting interoperability is kick-starting a new generation of business and electronic-commerce Web applications.

XML, which provides a data standard that can encode the content, semantics, and schemata for a wide variety of cases ranging from simple to complex, can be used to mark up the following:

- An ordinary document.
- A structured record, such as an appointment record or purchase order.
- An object with data and methods, such as the persistent form of a Java object or ActiveX control.
- A data record, such as the result set of a query.
- Meta-content about a Web site, such as Channel Definition Format (CDF).
- Graphical presentation, such as an application's user interface.
- Standard schema entities and types.
- All links between information and people on the Web.

Once the data is on the client desktop, it can be manipulated, edited, and presented in multiple views, without return trips to the server. Servers can now become more scalable, due to lower computational and bandwidth loads. Also, since data is exchanged in the XML format, it can be easily merged from different sources.

XML is valuable to the Internet, as well as to large corporate intranet environments because it provides interoperability using a flexible, open, standards-based format, with new ways of accessing legacy databases and delivering data to Web clients. Applications can be built more quickly, are easier to maintain, and can easily provide multiple views on the structured data.

3.3. Data is separated from the presentation and the process

The power and beauty of XML is that it maintains the separation of the user interface from the structured data. Hypertext Markup Language (HTML) specifies how to display data in a browser, XML defines the content. For example, in HTML you use tags to tell the browser to display data as bold or italic; in XML you only use tags to describe data, such as city name, temperature, and barometric pressure. In XML, you use stylesheets such as Extensible Style Language (XSL) and Cascading Style Sheets (CSS) to present the data in a browser. XML separates the data from the presentation and the process, enabling you to display and process the data as you wish by applying different style sheets and applications.

This separation of data from presentation enables the seamless integration of data from diverse sources. Customer Information, purchase orders, research results, bill payments, medical records, catalog data, and other information can be converted to XML on the middle tier, allowing data to be exchanged online as easily as HTML pages display data today. Data encoded in XML can then be delivered over the Web to the desktop. No retrofitting is necessary for legacy information stored in mainframe databases or documents, and because HTTP is used to deliver XML over the wire, no changes are required for this function.

XML documents are easy to create; if you are familiar with HTML, you can quickly learn to create one. In this example, XML is used to describe a weather report:

```
<weather-report>
<date>March 25, 1998</date>
<time>08:00</time>
<area>
<city>Seattle</city>
<state>WA</state>
<region>West Coast</region>
<country>USA</country>
</area>
<measurements>
<skies>partly cloudy</skies>
<temperature>46</temperature>
<wind>
<direction>SW</direction>
<windspeed>6</windspeed>
</wind>
<h-index>51</h-index>
<humidity>87</humidity>
<visibility>10</visibility>
<uv-index>1</uv-index>
</measurements>
</weather-report>
```

You can display this data in many different ways, or hand it off to other applications for further processing. You can also apply a style sheet to display this information in a browser.

Chapter 15. Debrief Algorithms

1. Range/Bearing calculations within Debrief

Ranges and bearings within Debrief are calculated according to the Rhumb-Line formulae, taken from the Admiralty Manual of Navigation, Volume 2, BR 45(2), Revised 1973, pages 5-11.

The Rhumb-line calculation is used both for the display of measured range-bearing and for the projection of data onto the monitor. Briefly, the to convert from real-world to screen coordinates the following procedure is followed:

1. Determine the area represented by the current viewed data in world coordinates
2. Determine the centre of this data area - this is used as the origin of the data
3. For each data point to be plotted, find its range and bearing from the origin
4. Convert this range and bearing into a delta-x and delta-y in screen coordinates
5. Produce a data point in screen coordinates by adding these deltas to the centre of the screen

Note

The result of this projection algorithm (as with most others) is that whilst the information plotted at the centre horizontal section (mid-latitude) of the screen is an accurate representation, travelling further north and south from it degrades the accuracy. This is negligible in data areas near the equator or where only a small area is covered (less than a couple of hundred nautical miles). With greater data areas, and nearer the poles, the effects are more noticeable, however.

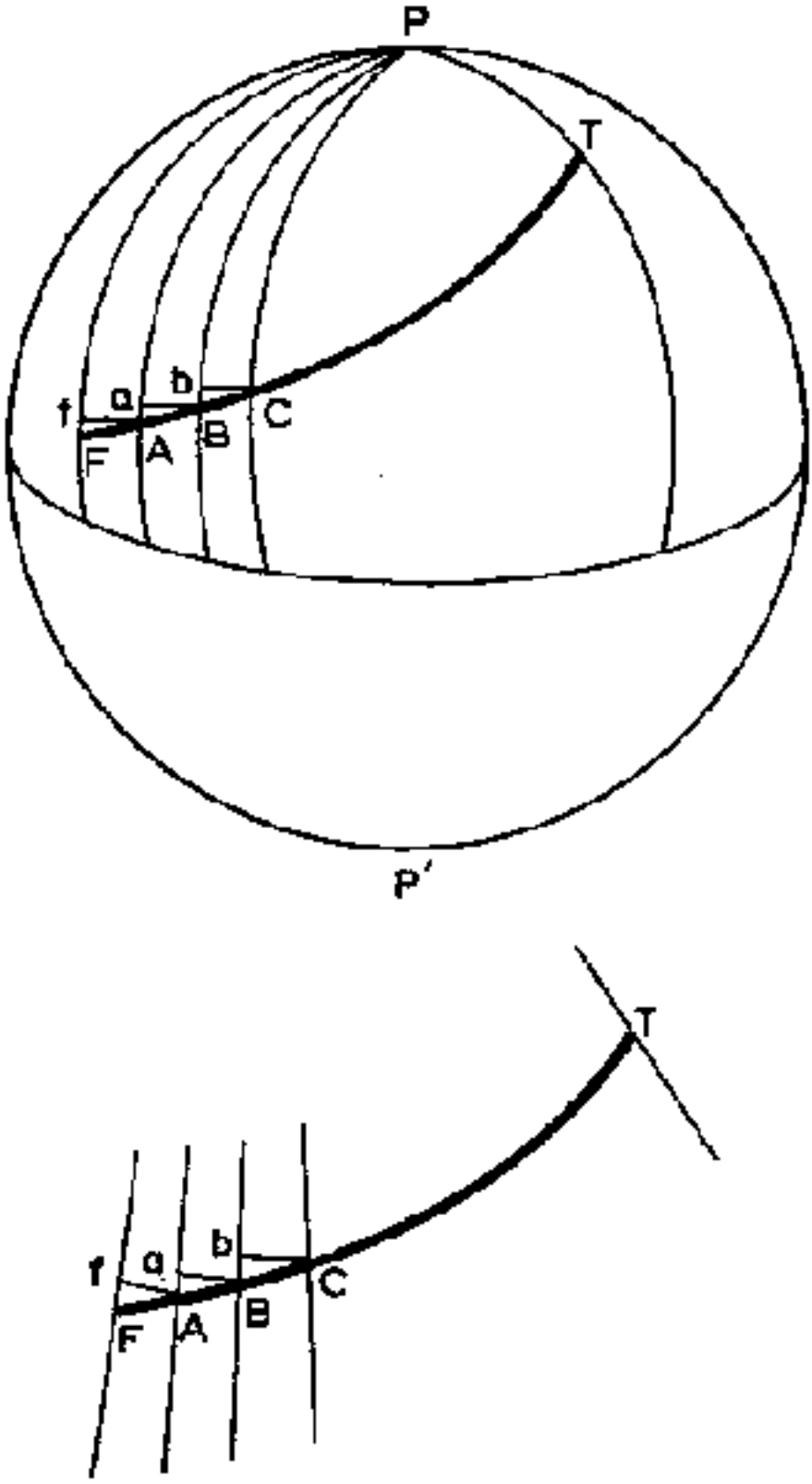
For those without access to the Admiralty Manual of Navigation, here is an abbreviated snippet from it:

1.1. The Rhumb-Line Formulae

A line on the earth's surface whihc cuts all meridians at the same angle is called a rhumb line.

With his knowledge of the distance along a parallel of latitude and the departure between two places, the navigator can find the course he must steer in order to follow the rhumb line joining the two places, and also the distance he will travel along whilst doing so. In Figure 15.1, “Rhumb line parts” [95] FT is the rhumb-line divided into a large number of equal parts FA, AB, BC...

Figure 15.1. Rhumb line parts



Af, Ba, Cb... are the arcs of parallels drawn through A, B, C ... and the angles at f, a, b ..., are therefore at right-angles. If the divisions of FT are made sufficiently small, the triangles FaF, ABa, BbB ... are themselves small enough to be treated as plane triangles. Also, since the course angle at F, A, B, C, ... remains constant by the definition of a rhumb-line, these smal triangles are equal.

1.2. Short-distance sailing

By the term *short-distance sailing* is meant the following of a rhumb-line track for a distance not greater than 600'. Within this limiting distance, the navigator can obtain all he wants to know about the track from the three formulae:

```
              departure = d.long cos (mean latitude)   (1)
departure = distance sin (course)     (2)
d.lat = distance cos (course)         (3)

The course is given by (2) divided by (3). Thus:
departure / d.lat = tan (course)
```

2. Contouring Algorithm

The contouring algorithm employed within Debrief was originally produced by Paul Bourke back in 1987, for submission to the Byte magazine. Here's the article pretty much verbatim:

2.1. Introduction

This article introduces a straightforward method of contouring some surface represented as a regular triangular mesh.

Contouring aids in visualizing three dimensional surfaces on a two dimensional medium (on paper or in this case a computer graphics screen).

Two most common applications are displaying topological features of an area on a map or the air pressure on a weather map. In all cases some parameter is plotted as a function of two variables, the longitude and latitude or x and y axis. One problem with computer contouring is the process is usually CPU intensive and the algorithms often use advanced mathematical techniques making them susceptible to error.

2.2. CONREC

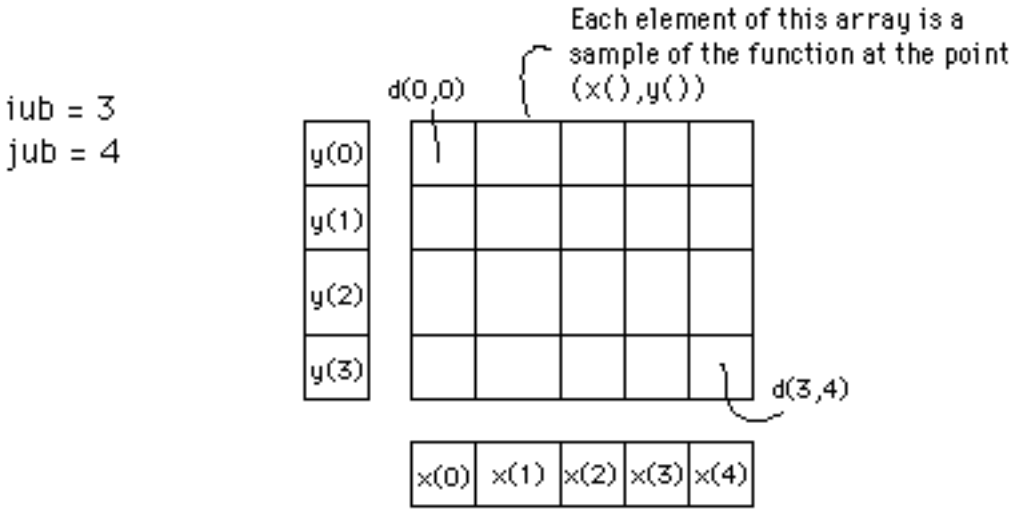
To do contouring in software you need to describe the data surface and the contour levels you want to have drawn. The software given this information must call the algorithm that calculates the line segments that make up a contour curve and then plot these line segments on whatever graphics device is available.

CONREC satisfies the above description, it is relatively simple to implement, very reliable, and does not require sophisticated programming techniques or a high level of mathematics to understand how it works.

The input parameters to the CONREC subroutine are as follows :

- The number of horizontal and vertical data points designated iub and jub.
- The number of contouring levels, nc.
- A one dimensional array z(0:nc-1) that saves as a list of the contour levels in increasing order. (The order of course can be relaxed if the program will sort the levels)
- A two dimensional array d(0:iub,0:jub) that contains the description of the data array to be contoured. Each element of the array is a sample of the surface being studied at a point (x,y)
- Two, one dimensional arrays x(0:iub) and y(0:jub) which contain the horizontal and vertical coordinates of each sample point. This allows for a rectangular grid of samples.

Figure 15.2. Some of the CONREC input parameters.



The contouring subroutine CONREC does not assume anything about the device that will be used to plot the contours. It instead expects a user written subroutine called VECOUT. CONREC calls VECOUT with the horizontal and vertical coordinates of the start and end coordinates of a line segment along with the contour level for that line segment. In the simplest case this is very similar the the usual LINE (x1,y1)-(x2,y2) command in BASIC. See the source code listing below.

2.3. Algorithm

As already mentioned the samples of the three dimensional surface are stored in a two dimensional real array. This rectangular grid is considered four points at a time, namely the rectangle d(i,j), d(i +1,j), d(i,j+1), and d(i+1,j+1). The centre of each rectangle is assigned a value corresponding to the average values of each of the four vertices. Each rectangle is in turn divided into four triangular regions by cutting along the diagonals. Each of these triangular planes may be bisected by horizontal contour plane. The intersection of these two planes is a straight line segment, part of the the contour curve at that contour height.

Depending on the value of a contour level with respect to the height at the vertices of a triangle, certain types of contour lines are drawn. The 10 possible cases which may occur are summarised below

a. All the vertices lie below the contour level.

b. Two vertices lie below and one on the contour level.

c. Two vertices lie below and one above the contour level.

d. One vertex lies below and two on the contour level.

e. One vertex lies below, one on and one above the contour level.

f. One vertex lies below and two above the contour level.

g. Three vertices lie on the contour level.

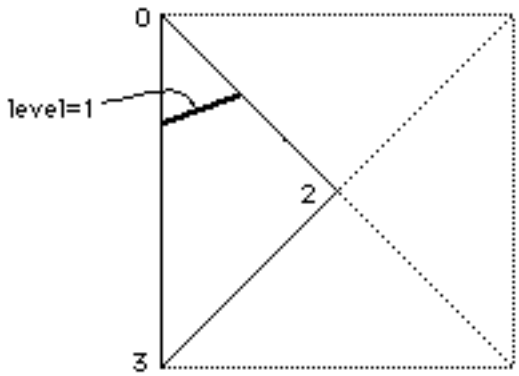
h. Two vertices lie on and one above the contour level.

i. One vertex lies on and two above the contour level.

j. All the vertices lie above the contour level.

In cases a, b, i and j the two planes do not intersect, ie: no line need be drawn. For cases d and h the two planes intersect along an edge of the triangle and therefore line is drawn between the two vertices that lie on the contour level. Case e requires that a line be drawn from the vertex on the contour level to a point on the opposite edge. This point is determined by the intersection of the contour level with the straight line between the other two vertices. Cases c and f are the most common situations where the line is drawn from one edge to another edge of the triangle. The last possibility or case g above has no really satisfactory solution and fortunately will occur rarely with real arithmetic.

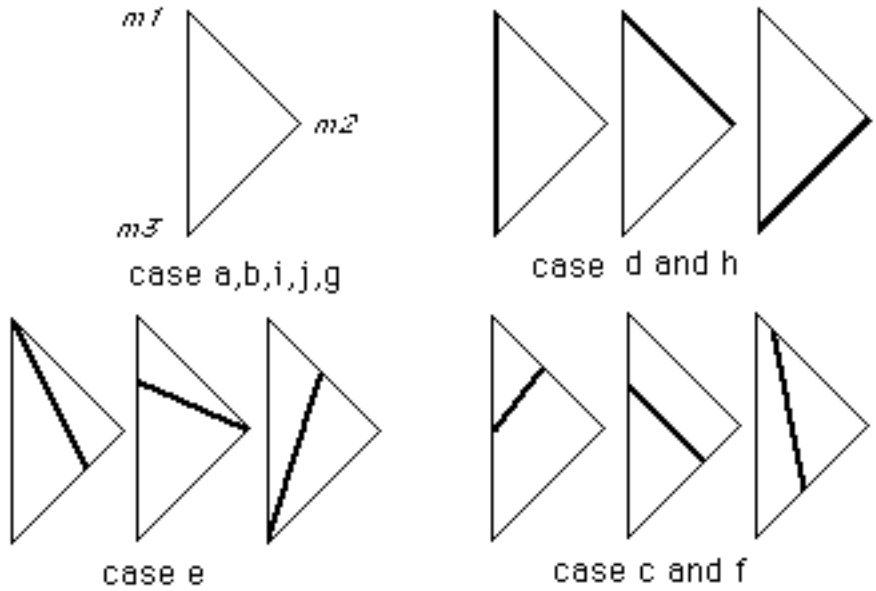
Figure 15.3. Possible line orientations



2.4. Example

As a simple example consider one triangle with vertices labelled m1,m2 and m3 with heights 0, 2 and 3 respectively

Figure 15.4. Line permutations within contouring algorithm.

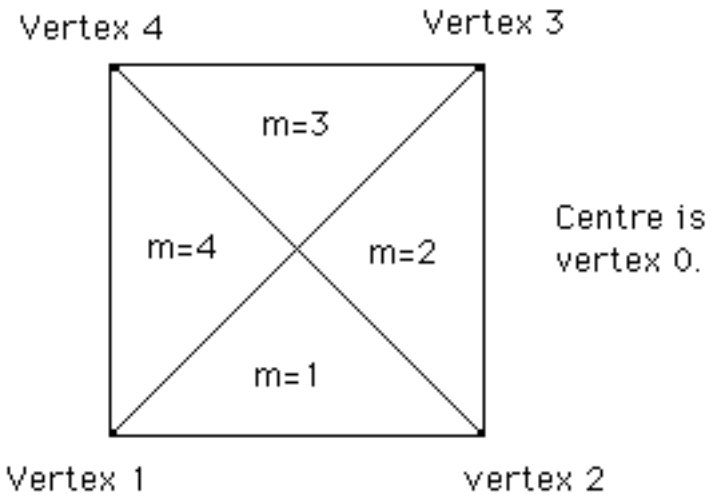


To calculate where a contour line at a height of 1 should be drawn, it can be seen that this is case f described earlier. Level 1 intersects line segment m1-m2 half the way along and it intersects line segment m1-m3 one third of the way along. A line segment is drawn between these two points. Each rectangular mesh cell is treated this way.

2.5. Subroutine

An attempt is made at optimization by checking first to see if there are any contour levels within the present rectangle and second that there are some contour levels within the present triangle. The indices i and j are used to step through each rectangle in turn, k refers to each contour level and m to the four triangles in each rectangle.

Figure 15.5. Some of the notation used for identifying the rectangles and triangles in the subroutine



Note that for large arrays the whole data array need not be stored in memory . Since the algorithm is a local one only requiring 4 points at a time, the data for each rectangle could be read from disk as required.

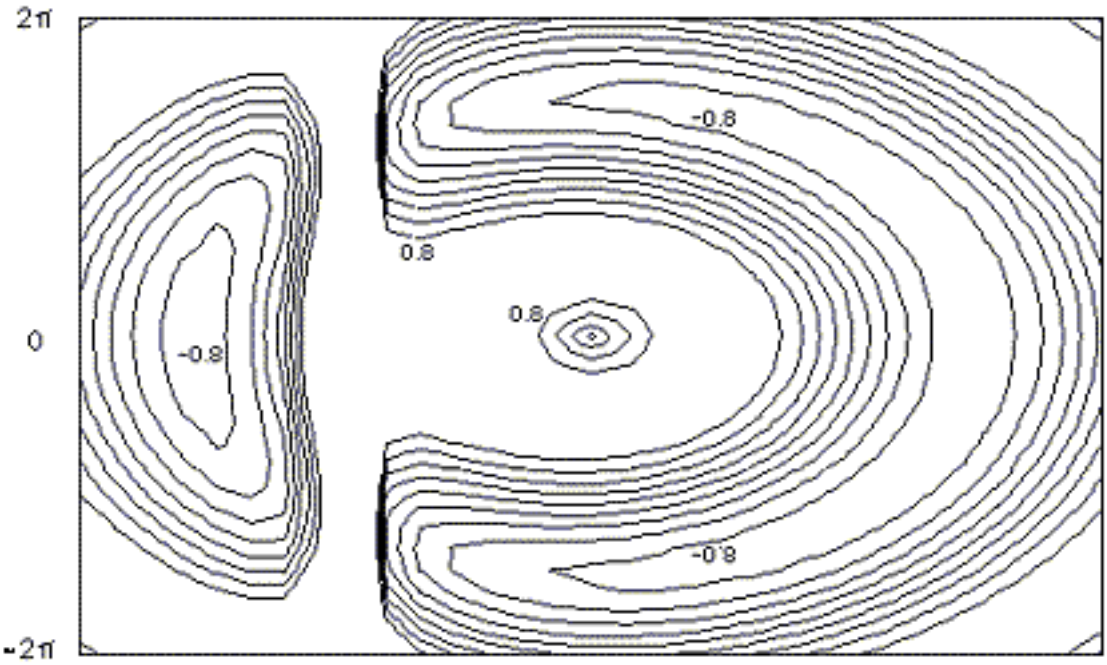
2.6. Example

Contour map and the following function

Figure 15.6. Function to be contoured

$$f(x,y) = \sin\left((x^2+y^2)^{1/2}\right) + \frac{1}{((x-c)^2+y^2)^{1/2}}$$

Figure 15.7. Sample contour plot using CONREC algorithm



The sample contour plot itself

2.7. Note

On occasion users have reported gaps in their contour lines, this should of course never happen. There is however a pathological case that all local contouring algorithms suffer from (local meaning that they only use information in the immediate vicinity to determine the contour lines). The problem arises when all four vertices of a grid cell have the same value as the contour level under consideration. There are a number of strategies that can be employed to overcome this special event, the correct way is to consider a larger region in order to join up the contours on either side of the problem cell. CONREC doesn't do this and just leaves the cell without any contour lines thus resulting in a gap. This special case essentially never happens for real values data, it is most commonly associated with integer height datasets. The simplest solution is to offset the contour levels being drawn by a very small amount.

Appendix 1. GNU Free Documentation License

Version 1.1, March 2000

Copyright ©) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA. Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of

the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special

permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

Appendix 2. Debrief Glossary

1. Glossary

This section contains explanations of terms used within Debrief

Glossary

AUTEC

Atlantic Underwater Test Facility. Data files from this facility which are to be imported into Debrief should be suffixed with "RAO". The origin of AUTEC is:

```
23º 26' 37.6280" N
77º 38' 6.8250" W
```

AFWTF

Atlantic Fleet Weapons Training Facility. Data files from this facility which are to be imported into Debrief should be suffixed with "PRN". The origin of AFWTF is :

```
17º 38.1577' N
065º 4.2065' W
```

Annotation

An annotation is the generic term used to describe the graphic elements added to a plot which do not represent vehicle positions, or bearings recorded on vehicle-mounted sensors. Examples of annotations are rectangles, ellipses and lines.

ASSET

The Advanced Scenario Simulator for the Evaluation of Tactics, a modular simulation suite intended for high-level simulation of maritime tactical scenarios by a relatively inexperienced user (read: uniformed). ASSET was actually the pre-decessor to Debrief, with the initial Debrief software being created to analyse ASSET simulation results. Quickly it was recognised that Debrief could also be usefully employed in the analysis of real exercise tracks. Aaah, how close we came to not having Debrief at all...

Bearing Rate

Bearing rate within the application is calculated using the following formula:

```
Bdot = ((Tspd * sin(Tcrse) - Ospd * sin(Ocrse)) *
cos(brg) - (Tspd * cos(Tcrse) - Ospd * cos(Ocrse)) *
sin(brg)) / range * 60
```

```
Rdot = Range Rate (yds/min)
Bdot = Bearing Rate (degs/min)
Tspd, Tcrse = Target course and speed (degs & yps)
```

```
Ospd, Ocrse = Ownship course and speed (degs & yps)
brg = Bearing to target from ownship (degs)
range = Range to target from ownship (yds)
```

Positive and negative bearing rates are named Right and Left according to naval convention, abreviated to R and L in the tote.

Bookmarks

The combination of a DTG, a remark and the name of a plot-file. Debrief NG presents the series of bookmarks allowing you to quickly move through events of interest across a series of plot-files.

Buoyfields

A series of sonar buoys which are laid in a particular pattern during Anti-Submarine Warfare.

Build Date

Each copy of Debrief is aware of the date is was built. Find this out by selecting About from the Help menu.

CMAP

Core Maritime Analysis Platform a framework of components intended to be reused across a range of maritime analysis applications. The two initial CMAP applications are Debrief and *ASSET*.

Coastline File

Debrief expects to find a coastline file (named World.dat) in its installation directory. Debrief loads this file in the background as soon as it opens; regardless of whether the user has requested to add a coastline to the current plot. Once the coastline is loaded (for the standard 1.2Mb file this takes around 8 seconds) there is no further performance penalty within the application.

The coastline file should be formatted in the following way:

- The coastline consists of a series of coastline segments. Each segment is drawn as a continuous polygon by the application.
- Each segment begins with the # -b separator on a line of its own
- Then there are a series of lines each containing a point in latitude and longitude expressed in decimal degrees (to 6 decimal places in the standard file).

```
# -b
-7.491098    4.257159
-7.523953    4.245425
-9.112761    5.008146
-9.464786    5.339050
-9.807424    5.681688
# -b
-9.807424    5.681688
-10.004558    5.845966
-11.152161    6.606341
-11.131039    6.639197
-11.163895    6.672052
-11.307052    6.761232
-11.351642    6.803475
```

Display Mode	Each of the display modes used on the Debrief plot represents a different method of stepping through the plot. When in the normal mode each track is shown in full, with a highlight drawn over the current point (as indicated in the Tote). When in snail mode only the current point plus an optional back-track is plotted.
Earth Model	The Earth Model used by the application is modular and interchangeable. In the initial release of the application the calculations use the Rhumb-Line Formulae, as described in 'Admiralty Manual of Navigation, Volume 2, 1973'. Short-distance sailing is defined as "the following of a rhumb-line track for a distance not greater than 600"
DTG	Acronym representing Date Time Group
ETOPO	Standard for digital topographic data, predominantly distributed by the NOAA
Cheat Sheets	Cheat sheets guide users through tasks. The task is broken down into steps and presented to the user one step at a time, and the user checks off the steps as he/she completes them.
Highlighter	The highlighter is the graphic used to indicate the current point on a track. Use of different highlighters allows range rings or a vessel-specific symbol to be plotted at the current point
Layer	A collection of objected plotted on the Debrief Plot. Each layer can be switched on and off individually using the Layer Manager. When written to a plot-file, each layer is stored separately - making it quite easy to copy individual layers out of a plot-file using a text-editor and collating them into a new file. (An example of this would be drawing exercise areas into one session, then moving them all into one layer, save the file to disk, copy this layer to a file of its own, then dropping this file of exercise areas into new files - an example of this is in the VPF best-practice.)
MWC	[Extracts taken from MWC 2000 Flyer] Under CinCFleet, the Maritime Warfare Centre (MWC) is a 'one-stop shop' for the evolution and dissemination of maritime/joint doctrine and concepts through teaching, tactical development, operational analysis, force development and wargaming. It also provides the focus for the development and practice of operational level warfighting, planning and decision making. The MWC was formed on 1 October 1995 merging the activities of the Maritime Warfare Development Centre at HMS Dolphin and the Maritime Tactical School at HMS Dryad to create a focal point for doctrine and tactical development.
Narrative	A series of time-related text messages. Typically these may represent the narrative recorded in a control room during an exercise, but alternatively they may contain a series of status messages retrieved from a sensor or weapon. All that is required is that the message have a DTG attached and that it may be represented in text form.

Overview	A zoomed out plot showing the full dataset currently loaded. Double clicking on this plot forces the main plot to re-centre on the selected point, and dragging an area on this plot forces the main plot to zoom in on the selected area. The formatting on the overview chart is identical to (and unchangeable from) that on the main plot, with the exception that text is not plotted - to reduce clutter.
Perspective	Each Workbench window contains one or more perspectives. A perspective defines the initial set and layout of views in the Workbench window. Within the window, each perspective shares the same set of editors. Each perspective provides a set of functionality aimed at accomplishing a specific type of task or works with specific types of resources. For example, the Tactical Analysis perspective combines views that you would commonly use while editing analysing tactical files, while the Contact Ork perspective would help aliens (typically named Mork) in contacting their mother planet - together with the VOIP software link direct to Orson. As you work in the Workbench, you will probably switch perspectives frequently, especially if you're having Mindy trouble. Perspectives control what appears in certain menus and toolbars. They define visible action sets, which you can change to customize a perspective. You can save a perspective that you build in this manner, making your own custom perspective that you can open again later. You can use the General > Perspectives preference page to open perspectives in the same window or in a new window.

PFWTF	Pacific Fleet Weapons Training Facility (see AFWTF). The origin of PFWTF is :
	
	`22º 7.16646' N`
	`159º 55.17' W`

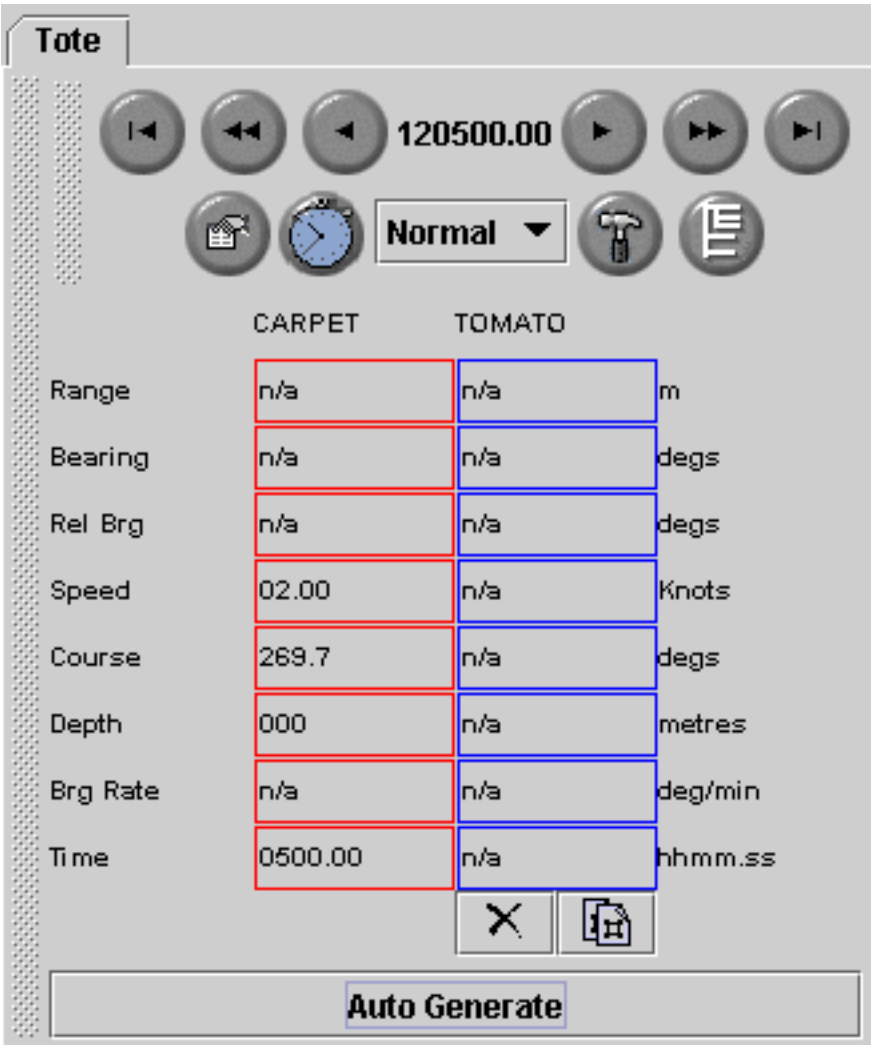
Plot	A graphic *God's-eye* representation of the current dataset. By default the plot has a black background but this may be altered by the user.
Plot-File	A file containing the following: • The data originally loaded from the Replay file • Any formatting applied to the data originally loaded • The details of any features added to the plot from the toolbars • The coordinates of the current view of the data • The settings of any controls used in Debrief (time on the Tote, primary/secondary tracks, etc) Plot-Files end with an XML suffix and may be viewed in Internet Explorer or edited using Notepad.

Properties Window	A view containing a list of all of the editable properties for an object within Debrief. Where applicable, custom editors are supplied (Color, Location, DTG, etc)

Range Rate	Range rate within the application is calculated using the following formula:
	
	`Rdot = (Tspd * cos (Tcrse - brg) - Ospd * cos(Ocrse - b`
	
	`Rdot = Range Rate (yds/min)`
	`Tspd, Tcrse = Target course and speed (degs & yps)`
	`Ospd, Ocrse = Ownship course and speed (degs & yps)`
	`brg = Bearing to target from ownship (degs)`

Replay	Replay is the name of the Unix application used for viewing tracks back in the early 90's at MWC
Serial	A "block" of exercise time, typically a sub-section of an exercise. An analyst will normally analyse and exercise one serial at a time, and it is usual for the exercise data to be broken down into serials.
Sensor	A sensor is defined as source of bearing-related information. As such, it could clearly be an acoustic sensor such as a sonar, but could also be a periscope or radar. Debrief makes no specific assumptions regarding what type of sensor is being represented.
Sensor Contact	This is an individual contact recorded on a sensor, a single bearing line reaching from the sensor location (origin) along the contact bearing to the contact range.
Session	The layers, projection details, and settings of any GUI-elements for the current view
Slant Range	The line of sight distance between two points, not at the same level relative to a specific datum. Normally in Debrief range is just calculated in two-dimensions (at the surface), but slant ranges can be requested via the Preferences window. Slant ranges are particularly useful in Debrief when analysing the proximity of two entities that are very close when measured at the surface but who possess a significant depth separation.
Snail Trail	A mode within Debrief where only the current vessel position plus a short back-track of previous positions is shown.
Stepper Control	The Stepper Control is the collection of controls at the top of the Tote panel. The Stepper Control provides controls to move the current time backwards and forwards, controls to edit the stepper itself (edit properties and change display mode), and convenience buttons such as View Layer Manager and View Track & Time Toolbox.

Figure 2.1. The Debrief Stepper Control



Symbology

This pair of text characters contained in an REP file indicate the formatting to be applied to this particular track/fix/annotation, although they can be over-ridden once the data is open in Debrief.

T-Zero

A specific time of interest, particularly the start of a particular event. Contexts typically have their own convention of t-zero. Lightweight torpedo launches use weapon-splash time.

TMA Solution

Warships (submarines in particular) use Target Motion Analysis (TMA) to produce an estimate of target range. course and speed when the target is held on a bearing/frequency only sensor. TMA solutions frequently represent uncertainty over target location by representing the location as an ellipse - given by a centre-point, an orientation and dimensions for the maximum and minimum axis (as diameter, not radius).

Track

A series of positions recorded for a particular vehicle (ship, submarine, helo, etc). A track has its own characteristics such as colour, label and symbol frequency, and the symbol used to represent it when the symbol highlighter is in use.

Tote	A GUI panel located by default at the lower-left hand side of the Debrief window. The Tote contains the Time Stepper, beneath which are shown the primary and secondary tracks, when assigned. When stepping forward through a *serial* the Tote contains data calculated from the current vessel positions.
VPF	The Vector Product Format is the format of vectored data which may be viewed by Debrief. The main type of VPF data is the Vector Map Level 0, an unclassified global database which includes coastlines, national borders and depth contours. Its supplier (WWW.NIMA.MIL [http://www.nima.mil]) describes it as: “The Vector Product Format (VPF) is a standard format, structure, and organization for large geographic databases that are based on a georelational data model and are intended for direct use. ”
XML	The eXtensible Markup Language, as recommend by the World Wide Web consortium. See the XML description of the Debrief reference guide

Appendix 3. Index

Index

Symbols

A

B

C

D

E

F

W

X

Colophon

Debrief has been authored in DocBook XML version 4.2 using XMetal version 3. Version 6.5.2 of Saxon used version 1.61.2 of the Docbook XSL stylesheets and version 0.20.5 of fop to publish the document to PDF and HTML.

The majority of the text was originally imported from previous Debrief help-files, but the reviews conducted in the transfer to DocBook have resulted in many corrections and clarifications.