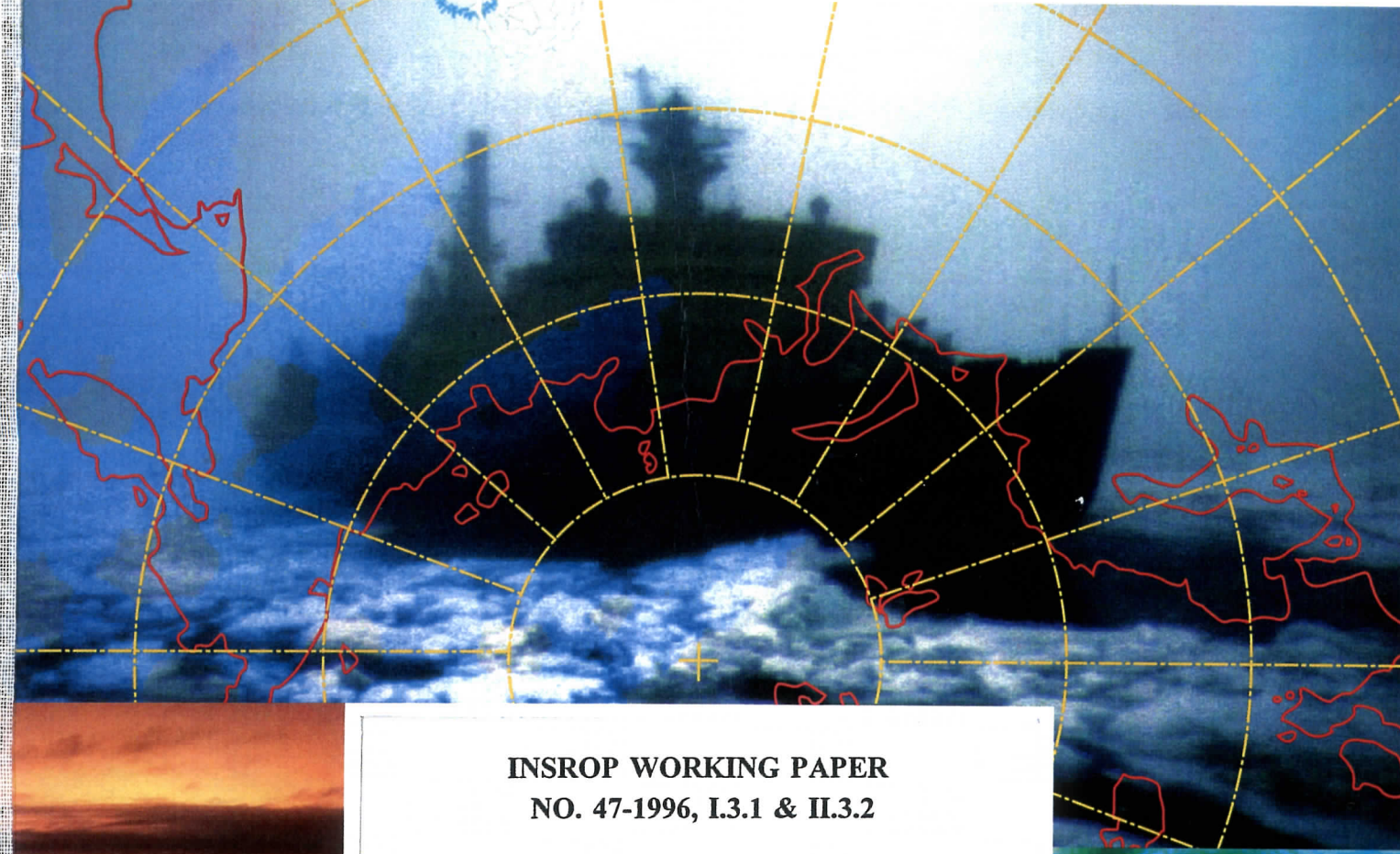
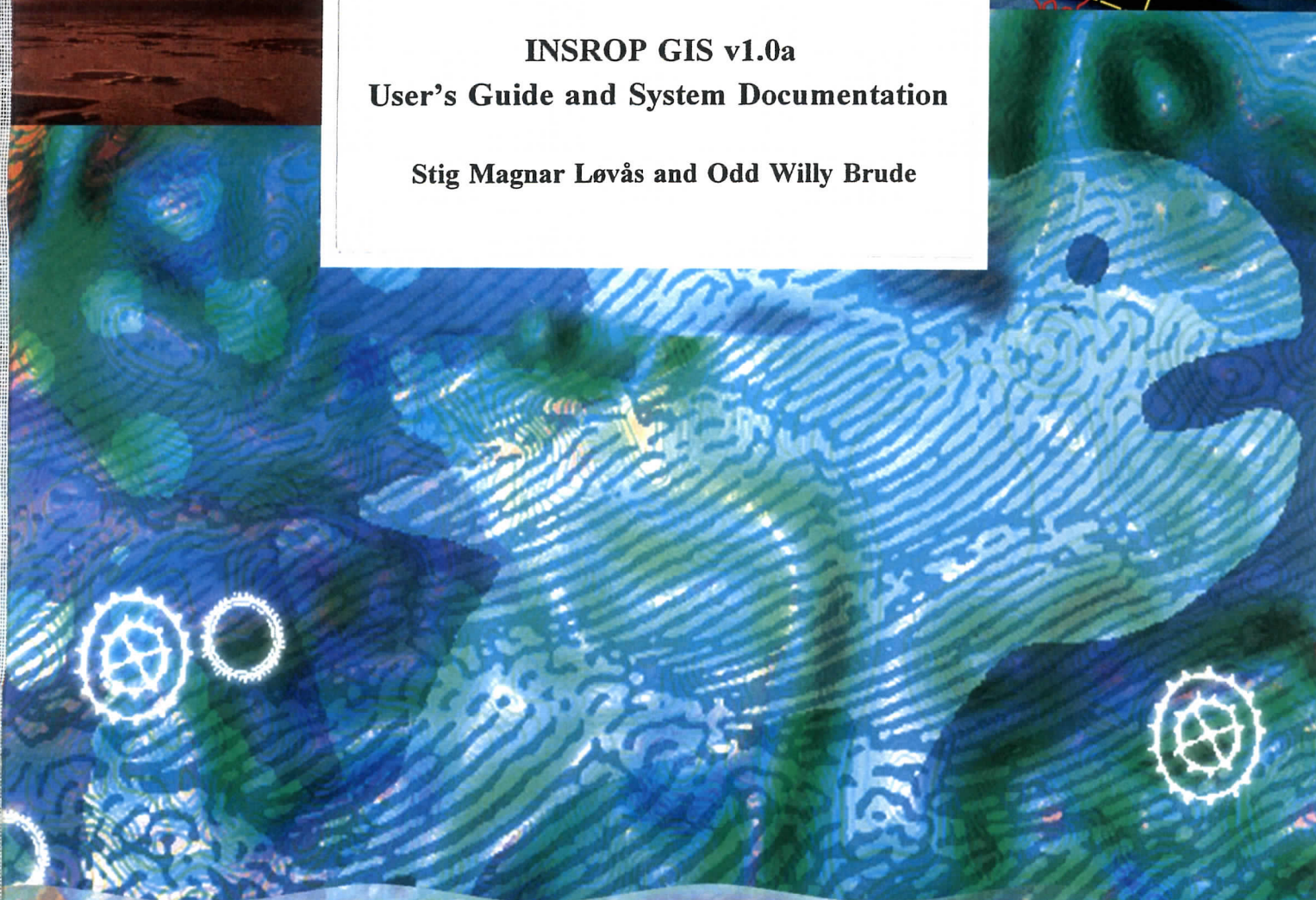


INSROP WORKING PAPER
NO. 47-1996, I.3.1 & II.3.2

INSROP GIS v1.0a
User's Guide and System Documentation

Stig Magnar Løvås and Odd Willy Brude



INSROP International Northern Sea Route Programme

Central Marine Research & Design Institute, Russia

The Fridtjof Nansen Institute, Norway

Ship and Ocean Foundation, Japan

International Northern Sea Route Programme (INSROP)

Central Marine Research & Design Institute, Russia

The Fridtjof Nansen Institute, Norway

Ship & Ocean Foundation, Japan

INSROP WORKING PAPER NO. 47-1996

Sub-programme I: Natural Conditions and Ice Navigation.

Project I.3.1. Design and Development of Information System and Project II.3.2: GIS - Implementation of Data Base.

Title: User's Guide and System Documentation.

By:
Stig Magnar Løvås (supervisor of project I.3.1)
SINTEF Civil and Environmental Engineering
Klæbuveien 153, 7034 Trondheim, Norway.
Tel: +47-73592300, Fax: +47-73592376

Odd Willy Brude, Norwegian Polar Institute
P.O. Box 5072 Majorstua, 0301 Oslo, Norway.
Tel: +47-22959500, Fax: +47-22959501.

Date: 9 May 1996.

Reviewed by: Ph.D. Richard Luxmore, World Conservation Monitoring Centre, Cambridge, England.

What is an INSROP Working Paper and how to handle it:

FOREWORD - INSROP WORKING PAPER

INSROP is a five-year multidisciplinary and multilateral research programme, the main phase of which commenced in June 1993. The three principal cooperating partners are **Central Marine Research & Design Institute (CNIIMF)**, St. Petersburg, Russia; **Ship and Ocean Foundation (SOF)**, Tokyo, Japan; and **Fridtjof Nansen Institute (FNI)**, Lysaker, Norway. The INSROP Secretariat is shared between CNIIMF and FNI and is located at FNI.

INSROP is split into four main projects: 1) Natural Conditions and Ice Navigation; 2) Environmental Factors; 3) Trade and Commercial Shipping Aspects of the NSR; and 4) Political, Legal and Strategic Factors. The aim of INSROP is to build up a knowledge base adequate to provide a foundation for long-term planning and decision-making by state agencies as well as private companies etc., for purposes of promoting rational decisionmaking concerning the use of the Northern Sea Route for transit and regional development.

INSROP is a direct result of the normalization of the international situation and the Murmansk initiatives of the former Soviet Union in 1987, when the readiness of the USSR to open the NSR for international shipping was officially declared. The Murmansk Initiatives enabled the continuation, expansion and intensification of traditional collaboration between the states in the Arctic, including safety and efficincy of shipping. Russia, beeing the successor state to the USSR, supports the Murmansk Initiatives. The initiatives stimulated contact and cooperation between CNIIMF and FNI in 1988 and resulted in a pilot study of the NSR in 1991. In 1992 SOF entered INSROP as a third partner on an equal basis with CNIIMF and FNI.

The complete series of publications may be obtained from the Fridtjof Nansen Institute.

SPONSORS FOR INSROP

- Nippon Foundation/Ship & Ocean Foundation, Japan
- The government of the Russian Federation
- The Norwegian Research Council
- The Norwegian Ministry of Foreign Affairs
- The Norwegian Ministry of Industry and Energy
- The Norwegian Ministry of the Environment
- State Industry and Regional Development Fund, Norway
- Norsk Hydro
- Norwegian Federation of Shipowners
- Fridtjof Nansen Institute
- Kværner a.s.

PROFESSIONAL ORGANISATIONS PERMANENTLY ATTACHED TO INSROP

- Ship & Ocean Foundation, Japan
- Central Marine Research & Design Institute, Russia
- Fridtjof Nansen Institute, Norway
- National Institute of Polar Research, Japan
- Ship Research Institute, Japan
- Murmansk Shipping Company, Russia
- Northern Sea Route Administration, Russia
- Arctic & Antarctic Research Institute, Russia
- ARTEC, Norway
- Norwegian Polar Research Institute
- Norwegian School of Economics and Business Administration
- SINTEF NHL (Foundation for Scientific and Industrial Research - Norwegian Hydrotechnical Laboratory), Norway.

PROGRAMME COORDINATORS

- **Yuri Ivanov, CNIIMF**
  Kavalergardskaya Str.6
  St. Petersburg 193015, Russia
  Tel: 7 812 271 5633
  Fax: 7 812 274 3864
  Telex: 12 14 58 CNIMF SU
- **Willy Østreng, FNI**
  P.O. Box 326
  N-1324 Lysaker, Norway
  Tel: 47 67 11 19 00
  Fax: 47 67 11 19 10
  E-mail: sentralbord@fni.no
- **Masaru Sakuma, SOF**
  Senpaku Shinko Building
  15-16 Toranomon 1-chome
  Minato-ku, Tokyo 105, Japan
  Tel: 81 3 3502 2371
  Fax: 81 3 3502 2033
  Telex: J 23704

SINTEF

SINTEF Civil and Environmental Engineering
Coastal and Ocean Engineering

Address: N-7034 Trondheim, NORWAY
Location: Klæbuveien 153

Telephone: +47 73 59 23 00
Fax: +47 73 59 23 76
Telex: 55 620 sintf n

Enterprise No.: 948007029

SINTEF REPORT

TITLE

INSROP GIS v1.0a - User's Guide and System Documentation

AUTHOR(S)

Stig Magnar Løvås, SINTEF NHL
Odd Willy Brude, Norwegian Polar Institute

CLIENT(S)

Fridtjof Nansen Institute

REPORT NO.	CLASSIFICATION	CLIENT'S REF.	
STF60 F95102	Restricted	Henning Simonsen	
CLASS. FRONT PAGE	ISBN	PROJECT NO.	NO. OF PAGES/APPENDICES
Open		605675	75 / 1
ELECTRONIC FILE CODE		DISCIPLINARY RESPONSIBILITY Stig Magnar Løvås Stig Magnar Løvås	
FILE CODE	DATE	RESPONSIBLE SIGNATURE	
	1996-05-09	Torbjørn Sotberg	

ABSTRACT

The multi-disciplinary aspect of INSROP requires a flow of information between INSROP projects. INSROP sub-programmes focusing on the physical and biological environment already in the planning of INSROP realized the advantages of co-operating on the design and development of a GIS to serve as a tool for storage, retrieval and analysis of information from the two sub-programmes. By co-operating, duplication of work was minimized, and data of common interest could be shared without having to reformat the data. Once organized into a GIS, the information is also more readily available for further processing and analysis. During the course of the INSROP GIS projects, new software developments enabled developing INSROP GIS as a hyper media information system based on GIS technology.

The development of INSROP GIS is confined to the needs of the INSROP programme and within the financial and temporal frames of the GIS development projects, and is intended to serve two correlated purposes: 1) during INSROP, to serve as a tool for organization and storage of INSROP data and for project-related analysis work; and 2), to grow into a computerized up-to-date realization of the INSROP knowledge base.

This document is primarily intended to document INSROP GIS in its present state, and provide guidance for use of INSROP GIS. Although some basic ArcView information is also included, the INSROP GIS documentation is not intended to replace the ArcView documention. Hence, any user of INSROP GIS should first get familiar with ArcView and the ArcView user documentation before learning to make use of the additional INSROP GIS functionality.

KEYWORDS	ENGLISH	NORWEGIAN
GROUP 1	Information Technology	Informasjonsteknologi
GROUP 2	Arctic	Arktis
SELECTED BY AUTHOR(S)	Geographic Information System (GIS)	Geografisk informasjonssystem (GIS)
	Northern Sea Route	Nordøstpassasjen
	ArcView	ArcView

PREFACE

In order to facilitate storage, retrieval and analysis of information obtained within the INSROP programme, an INSROP Information System has been developed using GIS technology. The system has been jointly developed by Sub-programmes I and II and is hence primarily designed to serve as a tool for these sub-programmes. The system is however designed in a modular way to enable inclusion of items from other sub-programmes later on.

The design and development of the information system is the responsibility of two projects: 'Project I.3.1 - Design and development of information system' and 'Projects II.3.1/II.3.2 - Geographical Information System'. Project manager for Project I.3.1 is Stig Magnar Løvås from SINTEF Civil and Environmental Engineering, while Kjell Are Moe from Det Norske Veritas (DNV) and Vidar Bakken from the Norwegian Polar Research Institute (NP) are project managers for Projects II.3.1/II.3.2. The person responsible for the project work at NP is Odd Willy Brude. In addition, GRID-Arendal (GA) has been sub-contracted by both projects. The person responsible for the project work at GA was Christopher Smith. The project work is carried out as a joint venture and the deliverables are common to both projects. The system design was presented in INSROP Working Paper No.4 (Løvås *et al*, 1994).

During the 1994/95 project work, ArcView 2.0, and later 2.1, became available to the INSROP GIS development team. This software product provided new capabilities that made us modify the original design. The main changes are that INSROP GIS is now primarily an ArcView 2.x-application available for PCs running Microsoft Windows 3.x. ARC/INFO is used to prepare data sets for use by ArcView, and to carry out analyses beyond the capabilities of ArcView.

The authors would like to thank the appointed project reviewer, Dr. Richard Luxmoore, World Conservation Monitoring Service, for his comments, which have been incorporated in the Working Paper.

Trondheim, March 20,1996,

Stig Magnar Løvås

Stig Magnar Løvås,
Supervisor, Project I.3.1

INSROP GIS REQUIREMENTS

ArcView on PC requires minimum an 80386 Intel-based microprocessor, but we strongly recommend 486/66 as a minimum. The basic ArcView software requires 13 MB harddisk space, and INSROP GIS requires an additional 2MB (without any data sets). The PC must have at least 8 MB RAM, but we recommend at least 16 MB. In addition you must set swap space, so that you have at least 17 MB of virtual memory available.

The INSROP GIS data sets range in size from a few KB to several MB. Especially the base cartographic data sets require 5-20 MB each (scale and content dependent), while some of the climatic source data sets may require several 100 MB. However, in INSROP GIS we will focus on derived (analysed) data sets, and these will require less storage space. For more information about size of data sets, consult the INSROP data sets catalogue (Brude *et al.*, 1996).

TABLE OF CONTENTS

1 INTRODUCTION

The aim of INSROP is to build up a scientific knowledge base on possibilities for, and consequences of, commercial navigation along the Northern Sea Route. The purpose of this knowledge base is to provide a foundation for long-term planning and rational decision-making by Russian and non-Russian public authorities and private interests regarding the use of the Northern Sea Route (NSR) for transit and regional development.

The NSR is a vast area, and INSROP encompasses a variety of topics. In the world of today, decision makers are often literally 'drowned' by information, and it may therefore be tough always to make decisions based on a sound evaluation of all relevant issues of a case. To improve the quality of decision-making from a variety of information, the information must be organized and communicated effectively. Maps and photos (e.g. from remotely sensed data) communicate information far more rapidly than any text-based description.

The multi-disciplinary aspect of INSROP also requires a flow of information between INSROP projects. INSROP sub-programmes focusing on the physical and biological environment already in the planning of INSROP realized the advantages of co-operating on the design and development of a GIS. This GIS was intended to serve as a tool for storage, retrieval and analysis of information from the two sub-programmes. By co-operating, duplication of work was minimized, and data of common interest could be shared without having to reformat the data.

Current Geographic Information Systems (GIS) technology enables decision-makers access to all organized information about an issue as hyper media. This means that textual documentation, tables, charts and maps are integrated and can be retrieved and studied depending on the problem at hand. Once organized into a GIS, the information is also more readily available for further processing and analysis. During the course of the INSROP GIS projects, new software developments have enabled us to develop INSROP GIS as a hyper media information system based on GIS technology. As such it has the potential of serving as base or part of an NSR information system for an operational NSR situation, but the current version of INSROP GIS is limited to the needs of the INSROP programme and within the financial and temporal frames of the GIS development projects.

The development of INSROP GIS is therefore intended to serve two correlated purposes: 1) during INSROP, to serve as a tool for organization and storage of INSROP data and for project-related analysis work; and 2), to grow into a computerized up-to-date realization of the INSROP knowledge base.

This documentation is primarily intended to document INSROP GIS in its present state, and provide guidance for use of INSROP GIS. Although some basic ArcView information is also included, the INSROP GIS documentation is not intended to replace the ArcView documention. Hence, any user of INSROP GIS should first get familiar with ArcView and the ArcView user documentation before learning to make use of the additional INSROP GIS functionality.

This Paper comprises the outcome of the system development project work. Although the GIS project has focused on system development, a number of INSROP GIS data sets have been prepared. These are documented in an INSROP data set catalogue (Brude *et al.*, 1995). In 1995, Project I.3.1 also included parts of Project I.5.1 (terminated after 1994). This work will be documented in a separate Working Paper.

The paper is organized into several chapters, where Chapters 2 and 3 provide background and introductory information about the INSROP GIS concept. Chapter 4 deals with distribution and installation issues, while Chapter 5 comprises brief information about the INSROP GIS software components. Chapter 6 includes a more detailed system

documentation, including the user interface and definition files for automating some frequently used system tasks. Chapters 7 - 13 include information about use of the system and especially additions to the basic ArcView capabilities. Chapter 14 provides information about the INSROP World Wide Web service, while Chapter 15 includes examples of ASCII import files.

2 BACKGROUND

2.1 What is INSROP GIS?

INSROP GIS is an ArcView 2.1 application. This means that all ArcView 2.1 capabilities are available in INSROP GIS. In addition INSROP GIS provides a customized user interface including access to the INSROP GIS hypertext documentation, a set of special options for INSROP purposes, and INSROP GIS layout templates to ease creation of INSROP hardcopy output.

INSROP GIS is developed to provide INSROP projects with the power to visualize, explore, query and analyze geographic INSROP data, and also to enable decision makers to gain easy access to organised INSROP information. Use of INSROP GIS requires basic knowledge on how to use ArcView 2.1. Information about ArcView 2.1 can be found in the "Introducing ArcView"-booklet and the ArcView on-line help system.

Note: INSROP GIS is a software concept developed primarily for purposes within the frameworks of the INSROP. How GIS technology is to be introduced, organised and utilized within an institution involved in INSROP is however beyond the scope of the GIS development projects. As INSROP GIS is not a commercial product; any support depends on the financial and temporal frameworks of the INSROP GIS development projects. The INSROP GIS distribution policy is decided by the INSROP partners and is available from the INSROP Secretariat after it has been ratified by the INSROP partners.

2.2 What is ArcView?

ArcView is a commercial software product made by Environmental Systems Research Institute (ESRI), Redlands, California, USA, the makers of ARC/INFO. It is a powerful, easy-to-use tool that enables you to visualize, explore, query and analyze geographic data spatially.

ArcView 2.1 is not provided as a part of INSROP GIS, but needs to be purchased separately from national vendors of ESRI products. National vendors of importance to INSROP include:

Japan:
PASCO Corporation
Phone: +81-3-3715-1601
Fax: +81-3-3715-1607

Norway:
Geodata AS
Phone: +47-22901800
Fax: +47-22901890

Russia:
Data+
Phone: +7-095-238-9111
Fax: +7-095-230-2090

2.3 The design and development process

The process of designing an INSROP information system using GIS technology has been a challenging task. The system should, in principle, address the critical factors affecting navigation along the Northern Sea Route (NSR), including the possible consequences of such navigation. Additionally, the system applications should be developed to serve project needs during the course of the program. Since the INSROP data acquisition and analysis projects are progressing in parallel with the system design, there is limited information on data availability, amount, and format. Therefore, the design team identified the following initial guidelines:

- The system structure should be flexible to permit modifications without restructuring the overall design. This applies to the database schema, the analysis methods, the output products, and the system infrastructure.

- The system should be useful both for applications within a single project and for executing applications that require data from several projects. Consequently, the system should be useful to users with minimum requirements, but at the same time serve the needs of users performing complex analyses involving large amounts of data.

- The hardware and software components for the system should be internationally available and provided by vendors that offer technical support and routine product upgrades. Additionally, the GIS software component should support data exchange (e.g., import/export of ASCII files) in formats (e.g., comma-delimited) common to popular software applications, such as word processing, spreadsheet and database programs.

To communicate the progress of the system design and to provide opportunities for feedback, the design team prepared three progress reports which were distributed to selected managers responsible for data acquisition and analysis projects. Following the second progress report a request for queries was also distributed to selected project managers to identify user needs. The answers to this request should help guide the priority for the types of data to be stored within the system and the analysis applications to be developed.

The realization of the INSROP GIS concept includes three main phases. The first phase identified the specifications for the information system. These specifications address a broad range of topics, such as, database contents, a metadata standard, information products, analysis tools, database infrastructure, and choice of technology. This initial phase also included design considerations for the system modules, including the functional structure of the information system, the database infrastructure and data flow. The outcome of the first phase was documented in Løvås *et al.* (1994). The second phase includes the system development, testing, and preparation of system documentation, while the third phase focuses on the use of the system, including training of users in other INSROP projects. As the system development depends on analysis specifications and data provided by data acquisition and analysis projects, and also since new analysis requests arise during use of the system, the latter two phases are partly overlapping, and both require close interaction with system users.

Due to new software releases and results from other projects during the course of INSROP, the system design in Løvås *et al.* (1994) has been modified somewhat, but the basic design principles are maintained. The major change from Løvås *et al.* (1994) is that ArcView, rather than ARC/INFO, has become the major software platform for INSROP GIS. One of the reasons for this is improved user accessibility, both in terms of hardware and software requirements and in ease of use.

2.4 The INSROP GIS development strategy

Most software products today claim that they are user-friendly. This is generally true, but even the simplest product requires some knowledge by the user. And in many cases there is a conflict between ease-of-use and user-flexibility. One may develop software that solves a specific problem, even a complex one, by a limited amount of user-interaction, but such programs can only solve the problem it was created for and has hence a low level of user-flexibility. In INSROP there are users with different needs for problem solving, different levels of computer availability and knowledge, and different needs for result presentation. Within the restricted frameworks of a scientific research programme it is unrealistic to solve all INSROP needs by one product. Hence, the INSROP GIS concept includes a range of software products targeted for different user groups.

As the purpose of INSROP is to build up a knowledgebase for decision-making, a critical aspect of the success of INSROP is the communication of the knowledge to decision-makers. It is widely accepted that visualized knowledge is more easily communicated than pure text-based knowledge. Maps and charts therefore represent a high level of information communication efficiency. Hence, the ability to prepare easy-to-understand maps and charts is a clear must from INSROP projects. Common spreadsheets are powerful for creation of standard charts, but have no link to maps. Map-printing software is more limited, but most graphical presentation software may create maps using a local coordinate system. Such maps are however not geo-located, that is, the map elements are just graphical objects and the software does not know where in the world the map is located. To achieve this one needs to use GIS software. And since ArcView also has a (although limited compared to powerful spreadsheets) chart-creation capability, most of the visualized results from INSROP can be prepared by an ArcView application.

Hardcopy prints are good and easy communication media, but their drawback is that they cannot be modified. To show various situations or conditions, several maps/charts are needed. Current computer technology allows putting software and data onto a powerful portable computer and present maps and charts directly from the computer. One may start by showing the decision-maker a set of pre-made maps/charts, but the software will also allow modification, e.g. zooming, of the map upon requests or to visualize changes. This level of communication requires more in terms of computer resources and user knowledge, but also give a higher level of user flexibility. Also here we feel that ArcView provides a good compromise between user-friendliness and user-flexibility.

In order to communicate visualized information, one must also be able to prepare source data suitable for visualization. A basic requirement here is that the source data are digital, but in addition the following steps are required:

1. Structure the data in a format the software can import
2. Import the data and store them in format used by the software
3. Prepare the data for display

The first step can either be done manually, by using/developing conversion software, or if the data originates from another software package, the data may be exported in a suitable format. The second step affects how easy it is to import data and make them useable by the software, while the last step may involve a range of sub-steps from simple color assignment (e.g via classification) to actual manipulation of the source data. This step may also include processing of data through selections and analyses. As we want to produce results with a minimum of required computer and software-specific knowledge, and still allow a high

degree of flexibility, we have chosen to develop INSROP GIS in a way that may accommodate different user categories, including decision-makers.

First of all, with the release of ArcView 2.0, we felt that many of the purposes of developing an INSROP information system could be realised on a (powerful) PC. This provides a lower threshold for users in terms of purchase costs and use, as software interfaces adhering to the Microsoft Windows user interface provides a high degree of familiarity to most PC users. In addition, the object-oriented programming language associated with ArcView, AVENUE, enables development of customized ArcView interfaces and functionality. Hence, while maintaining the flexibility of the standard ArcView version, we could develop special scripts for importing source data (from special ASCII formats) and guide the user through the importing process (Step 2). We have also developed scripts for automating the display process, and we have developed special query and analysis scripts. This is one step in simplifying the preparation of visualized information, while still maintaining a high degree of flexibility. The process may also be simplified further, as ArcView use projects (*.apr-files) to store information about how to display a prepared sets, or sub-sets, of data. The project files do not comprise any data, but links to data, hence the user may have a range of project files combining and displaying source data in different ways depending on the purpose. Hence, a decision-maker (information receiver) having a copy of a project file and the relevant source data sets, may study the information just as it was prepared by the source user. In a way this is similiar to a hardcopy map, but the decision-maker will also be able to process the information further, depending on his own wishes, and he may also add his own data.

The ArcView application, including the AVENUE scripts, is the main part of the INSROP GIS concept, and the part generally referred to when using the term INSROP GIS. However, not all GIS requests can be solved in ArcView. For more sophisticated GIS analyses and especially for manipulation of spatial source data, ARC/INFO is required (chosen for its capabilities, widespread use and because ArcView can use ARC/INFO data directly). To also take advantage of more powerful computers, the UNIX-version of ARC/INFO is used. There are also some analyses, especially time series analyses of gridded data, that are inefficient to analyses in ARC/INFO. Here we have chosen developed tailormade FORTRAN or C programs, and in a few cases also combining FORTRAN and C programs. The FORTRAN/C-programs are either run separately or in combination with ARC/INFO. This part of the INSROP GIS project work requires more in terms of computer resources and programming knowledge, and the programs developed for general use, but with the purpose of preparing analysed data for use in ArcView. These programs are therefore not included in the INSROP GIS distribution versions.

To conclude this chapter, we, the developers of INSROP GIS, feel that developing INSROP GIS as an ArcView application enables the development of a powerful, flexible and user-friendly software product for visualizing a variety of INSROP results. We also feel that the power and flexibility of ArcView justifies the purchase costs compared to other software capable of producing run-time versions for free distribution. The option of developing all new software from scratch to enable free distribution, while still retaining equal flexibility, was soon ruled out as unrealistic within the existing financial and temporal frameworks, and a waste of money compared to purchasing a commercial product. Self-developed general-purpose software also tends to become rapidly dated compared to the development pace in large commercial software companies. Self-developed software is however valuable for extending the functionality of the more general-purpose software, especially for solving user-specific problems and automating common user-specific processes.

3 THE INSROP GIS DESIGN

3.1 Principal System Infrastructure

The overall objectives of INSROP require comprehensive inter-disciplinary analysis results. The organisation of INSROP is more discipline-oriented. Hence, it is considered sensible to use an information system organised with levels as illustrated in Figure 1. In this figure, the Data Sources are the institutions/projects responsible for supplying data in a specified format with accompanying metadata (i.e. information about data) and data quality specifications. The institutions/projects performing Thematic Integration are responsible for integrating the data from their respective data sources and transferring the data to the institution(s)/project(s) responsible for the Overall Integration and Distribution. The Data Users will then have access to the most recent accepted data sets from different sources within a common framework (INSROP GIS). The expression *Core Database* does not necessarily mean that all data are stored in one location, but rather that the available data are stored and documented in a common format suitable for use by INSROP GIS.

3.2 Basic Hardware and Software Components

The hardware and software components of the INSROP information were selected as a cost effective solution to implement state-of-the-art computing technology. The software components of the information system include the most recent versions of ARC/INFO® and ArcView®. These products, developed by ESRI (Environmental Systems Research Institute, Redlands, California, USA) are widely used by the GIS community and provide the necessary tools required to develop a comprehensive GIS application. INSROP GIS is developed as an ArcView-application for use on PCs (486 or better) running Microsoft Windows. However, as part of the INSROP GIS concept, ARC/INFO, running on UNIX workstations, is used to prepare the data sets for use by ArcView and to run analysis tasks beyond the capabilities of ArcView on a PC. Other useful, although not required, system components include access to Internet or a CD-ROM reader to facilitate the exchange of large volumes of digital data. A more detailed review of the hardware and software components required for INSROP GIS is included in Section 6.2.

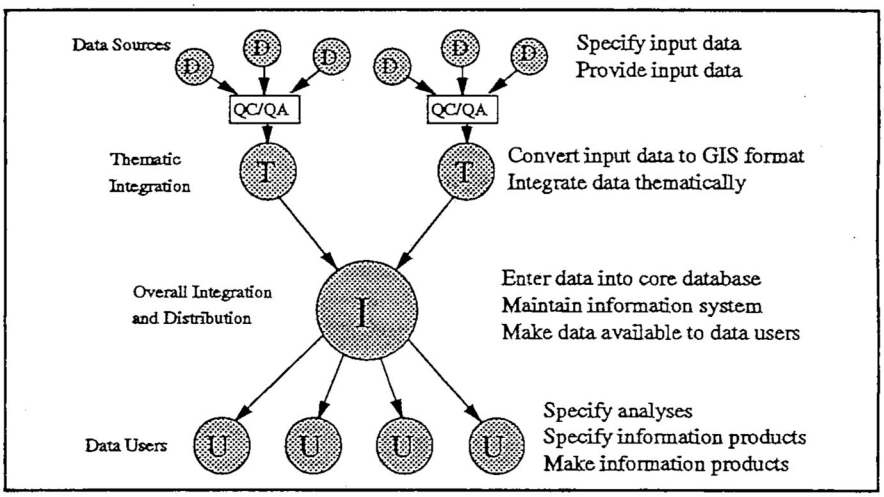


Figure 1 Information system organisation levels

3.3 Components of INSROP GIS

To achieve the required flexibility the INSROP GIS concept consists of several components: the installation component, the INSROP GIS software component, the INSROP GIS database, the INSROP GIS project library, and the ARC/INFO applications. The installation component comprises software to install INSROP GIS on a PC as a Microsoft Windows application. The INSROP GIS. software component includes the customized ArcView user interface and hardcopy layout templates, as well as the INSROP hypertext documentation and analysis software. The INSROP GIS database consists of all data sets accessible in ArcView, while the INSROP GIS project library comprises task-oriented ArcView project files serving special INSROP project needs, or to be used as a basis for further analysis work. The ARC/INFO applications comprise the AML's (ARC Macro Language) developed for INSROP-related ARC/INFO work. This component is not included in the basic INSROP GIS distribution package.

4 DISTRIBUTION AND IMPLEMENTATION OF INSROP GIS DATA AND SOFTWARE

4.1 General content of the installation package

The basic INSROP GIS distribution package includes the INSROP GIS software component and the basic parts of the INSROP GIS database and INSROP GIS project library. These basic parts include base cartographic data sets and associated project files. However, as the base cartographic data sets are generally large in size, special INSROP GIS packages can be distributed to users upon request. Such special packages would typically be distributed to users who already have installed the necessary base cartographic data, and will include additional or updated data sets, associated project files, or other updates. All INSROP GIS distribution packages are distributed as a set of diskettes or transferred via Internet.

To enable easy installation of INSROP GIS, the distribution packages consist of an executable SETUP program to be run from the MS Windows Program manager. This SETUP program is developed using InstallSHIELD™ 2.0 (© Stirling Technologies, Inc., Schaumburg, Illinois, USA) and provides the user with options regarding which INSROP GIS components to install and in which drive/directory to install INSROP GIS. For PC network users, the INSROP GIS database, which requires the most storage space, may be stored on a network drive, while the other components may be stored on the user's local drive. The actual INSROP GIS distribution version consists of compressed files which may be split on several diskettes. If installing from diskettes, the user is prompted for entering the proper diskettes depending on the chosen installation options. During the installation process, relevant components of INSROP GIS are made available as Program Items in a special INSROP GIS MS Windows Program Group. Existing Program Items are updated.

To ease preparation of the distribution packages, a program similar to SETUP is developed. This program allows the distributor to choose which data sets or other INSROP GIS components to include in each distribution package. However, both the SETUP program and distribution preparation program need to be updated when new data sets are to be included in an INSROP GIS distribution package.

4.2 INSROP GIS distribution versions

INSROP GIS represents software developed within the INSROP, but also with considerable additional resources from the main developing institutions (SINTEF and the Norwegian Polar Institute). It may not be in the interest of INSROP to enter the source code into the public domain (at least not yet). However, as the INSROP policy for distribution of software and data is not yet finalised, we cannot specify which users within, as well as outside, INSROP will get access to what versions of software and data. The main choices are no access, limited access and full access. Users with no access will not get any version of INSROP GIS software, while users with limited access may get a limited version, and users with full access will get the complete version. This is not just a question of policy, but also of technical possibilities. The current version of ArcView includes possibilities for *encrypting* the source code of an ArcView application (this yields both the system scripts and the specially developed application scripts). However, as this option for limiting accessibility requires more work for preparing distribution updates, we recommend that the complete version of INSROP GIS is to be released to INSROP users. This means that restrictions in

the distribution of INSROP GIS is limited to restricting who shall get a copy of INSROP GIS and the terms for how users may or may not distribute a modified/unmodified version of INSROP GIS to other users.

As the complete unencrypted version of INSROP GIS can be further developed and customized by any user, this version is termed *Development version*, while the encrypted version may be termed *Basic version*, as this is the version intended for most widespread distribution after INSROP is completed. As this version also will have full functionality, it will not represent any limitation to most users, but it enables INSROP (and/or the INSROP GIS developing institutions) to maintain the rights of the developed source code.

4.3 Modifying the standard ArcView startup file

When ArcView (or an ArcView application) is initiated, a file named **startup**, located in the *etc* sub-directory below the main ArcView directory (default ArcView location: c:\win32app\arcview). The startup file comprises an AVENUE script specifying the **HOME** system environment variable. If HOME was not specified (e.g. in the AUTOEXEC.BAT file), the script will search for the existence of **TEMP**, **AVHOME** or **windir** system environment variables. When developing INSROP GIS we did not consider this an optimum solution. Hence, we modified the script so that when the HOME variable is not defined, the Working directory (as defined in the MS Windows Program Item Properties) is used as the HOME directory. The importance of the HOME directory is e.g. that any customized interface (stored in a file called default.apr) and layout templates (stored in a file called template.def) are read from this directory. If there are no such customized files in the HOME directory, the basic ArcView interface and layout templates are used. When installing the INSROP GIS software for the first time you must replace the existing startup file with the INSROP GIS startup file included in the INSROP GIS installation files. Upon installation this file is copied to the main INSROP GIS directory (as specified by the user. Default is C:\INSROP). The file is named **startup.nsr**. Remember to make a backup copy of the old startup file before replacing the file.

5 THE INSROP GIS SOFTWARE

5.1 ArcView Basics

INSROP GIS is a customized GIS application based on ArcView® 2.1 software [2]. ArcView works with *views*, *tables*, *charts*, *layouts*, and *scripts*, stored in one file called a *project*. A project file stores information about the contents and status of each project component, including storage locations of the data sets used in the project. The actual data are not stored in the project file. Hence, if any data set is updated, the updates are also reflected in the ArcView project. However, this also means that data sets cannot be moved to another location without updating the project file. Project files are stored with an '.apr' extension.

A view is an interactive map that enable display, exploration, queries and analysis of geographic data. The view defines the geographic data and how to display them, but does not contain the actual data themselves; only references to these data. The spatial part of a database stores information about where the data are located, while the tabular part of the database provides descriptive information about the elements contained in the data base. This information can be used to choose how (colour, symbology, text etc.) the data are to be displayed. The data sets to be included in a view are included as themes, and must be prepared in ARC/INFO format, as a raster image, or created as ArcView shapefiles from X-Y coordinates. Each theme can be thought of as a geographic information layer. The options for manipulating images are limited, and images are primarily intended to be used as background. For analysis of cell-based data (raster data) the ARC/INFO GRID module is employed.

Tables and charts work in a fairly similar manner as a simple spreadsheet in the sense that the tables comprise data records with several columns, and that charts can be created from the values in data columns. Data values in a column can also be calculated from values in other columns. However, pure spreadsheet tasks are better performed in state-of-the-art spreadsheets. The strength of the ArcView tables and charts is the connection to the geographic data in the view. Spatial data sources such as ARC/INFO coverages have attribute tables comprising descriptive information about the geographic features they contain. Any selection of spatial features in a view also means that the tabular records of these spatial features are selected. Any statistical analysis of data columns in the attribute table or displayed charts will now only include the values from the selected features. This is particularly useful to analyse features satisfying certain spatial criteria, e.g. being within a limited area or within a limited distance from another feature, e.g. a ship track. Also, any selections made in an attribute table mean that the corresponding spatial features are selected. This is particularly useful for seeing where features satisfying certain descriptive criteria are located. In addition to the attribute tables, other tables (in dBase, INFO, or delimited ASCII format) can be joined (one-to-one relation) or linked (one-to-many relation) to an attribute table, provided they both include a suitable data column. Linked tables can be linked to other tables and thereby enabling many-to-many relations.

A layout is the framework for preparing graphical output, such as maps, charts, and table records, for hardcopy printing or to be saved on files for import into word processors or publishing applications. The elements of a layout can be dynamically linked to the view, table and/or chart windows, so changes in any of these windows are reflected in the layout. In addition the layout may comprise various graphical elements such as graphical files, text, frames, north arrow, scale bar and legend box. The scale bar and legend box are as default linked to the view to reflect map scale and legends of the selected themes in the view. The layout may also comprise several views, charts etc.

A script is the component of an ArcView project that contains AVENUE code. AVENUE is an object-oriented programming language from which ArcView is developed, and AVENUE can be used to customize ArcView or to program special analysis tasks not available from the default ArcView interface. INSROP GIS is developed using AVENUE scripts, but is not required to use INSROP GIS.

Each of the ArcView components are run in separate windows. The ArcView window is the main window and serves as the background for all other ArcView operations. The project window allows management of the individual components of the project.

The project files also include information on the display status, screen location, size etc. of the various component windows. The ArcView user's interface also includes menu, button and tool bars at the top of the ArcView window. These vary for each ArcView component, and the menu, button and tool bars shown in the ArcView window are always the ones for the active window. The ArcView window also includes a status bar at the bottom of the window. The ArcView interface is stored in the file 'default.apr' in the 'etc' sub-directory below the ArcView installation directory.

5.2 The INSROP GIS Interface

The INSROP GIS interface is stored in the file 'default.apr' in the INSROP GIS working directory (i.e. C:\INSROP). This file includes the basic ArcView interface, all INSROP GIS modifications to the menu, button and tool bars, and all AVENUE scripts providing extended INSROP GIS functionality.

The INSROP GIS AVENUE scripts are triggered from menu choices, or from clicking on buttons or tools. Several of the scripts are developed for simplifying manual operations or to guide the user through steps in the information processing/analysis process. In addition there are scripts enabling access to the INSROP hypertext documentation from the INSROP GIS interface. The INSROP GIS Interface is described in more detail in Section 6.3.

5.3 INSROP GIS Layout Templates

The INSROP GIS layout templates are stored in the file 'template.def' in the INSROP GIS working directory (i.e. C:\INSROP). If a user prepares and stores new or modified layout templates, this file is updated. The INSROP GIS layout templates are modified from the default ArcView layouts and include the INSROP logo. As new layout templates are easy to create by modifying existing ones, we have not put priority into providing a range of layout templates. The present version of INSROP GIS includes INSROP GIS versions of the basic ArcView Landscape and Portrait layout templates.

5.4 INSROP Hypertext Documentation

The INSROP hypertext documentation comprises basic information about INSROP, such as programme organization (Sub-programmes, projects and committees) and publishing procedures, and executive information about INSROP projects and institutions involved in INSROP. In addition, information about INSROP GIS and INSROP GIS data sets and projects are included.

The use of hypertext means that the information is searchable (through predefined

keywords) and organized as information topics with links to related topics. The hypertext documentation is prepared as MS Windows hypertext files using RoboHELP® (©Blue Sky Software Corporation, La Jolla, California, USA). This means that ArcView is not required to access the hypertext documentation, as, in addition to being accessible from the menus in the INSROP GIS interface, the information is accessed by double-clicking the proper icons in the INSROP GIS Program Group. The INSROP GIS hypertext documentation is further described in Chapter 13.

6 SYSTEM INFRASTRUCTURE

6.1 Directory structure

The default main INSROP GIS directory is **C:\INSROP**. If using the SETUP program to install INSROP GIS, this directory name can be specified by the user during the installation process. However, it is recommended to use the default value if possible. The name of the main INSROP GIS directory is allocated to a system environment variable, NSR_PATH, which is added to the user's AUTOEXEC.BAT file.

Below the main INSROP GIS directory there are four basic INSROP GIS sub-directories:

\INSROP
- \DATABASE
- \IMAGES
- \PROJECTS
- \WINHELP

The **C:\INSROP\DATABASE** directory is the main database directory. Similar to the main INSROP GIS directory, this directory can be specified by the user during the installation process (of INSROP data sets). The name of the main INSROP GIS database directory is also allocated to a system environment variable, NSR_DATA, which is added to the user's AUTOEXEC.BAT file (if not existing from before). This possibility means that the database can be stored on a network server, while the INSROP GIS software is stored on the local disk. The INSROP GIS datasets are organized into information topics and stored in a sub-directory structure below the main database directory. See Section 6.4 for further documentation on the database directory structure and organization of datasets.

The **C:\INSROP\IMAGES** directory comprises graphical files such as the INSROP logo (used by the INSROP layout templates) and may also be used for files intended for e.g. *hotlinking*.

The **C:\INSROP\PROJECTS** directory comprises the INSROP GIS project library. It is recommended that all INSROP GIS projects (*.apr) are stored in this directory. The project files may be organized into sub-directories.

The **C:\INSROP\WINHELP** directory comprises the INSROP GIS hypertext documentation. This documentation consists of a set of pre-made Microsoft Windows hypertext files.

6.2 INSROP GIS requirements

ArcView on PC requires mimimum an 80386 Intel-based microprocessor, but we strongly recommend 486/66 as a minimum. The basic ArcView software requires 13 MB harddisk space, and INSROP GIS requires an additional 2MB (without any data sets). The PC must have at least 8 MB RAM, but we recommend at least 16 MB. In addition you must set swap space, so that you have at least 17 MB of virtual memory available.

The INSROP GIS data sets range in size from a few KB to several MB. Especially the base cartographic data sets require 5-20 MB each (scale and content dependent), while some of the climatic source data sets may require several 100 MB. However, in INSROP GIS we will focus on derived (analysed) data sets, and these will require less storage space.

For more information about size of data sets, consult the INSROP data sets catalogue (Brude *et al.*, 1995).

6.3 INSROP GIS Interface

In INSROP GIS, similar to ArcView, you work with views, tables, charts (and scripts), stored in one file called a project. All project files are stored with an .apr extension. You work with one project at a time. Projects enable keeping together all components required for a task or application and available from a common window interface. The main window is called the ArcView window, and serves as the framework for all ArcView/ INSROP GIS operations. Normally you will not see this window without at least the Projects window open. When starting INSROP GIS (or ArcView), that is the standard startup interface (see Figure 6.1).

The project window lists all components of a project. When you open one of the components of the project, it is displayed inside its own window. You can have any number of windows open in ArcView, but there is only one active window at any time.

The ArcView window comprises a Menu bar, a Button bar, a Tool bar and a Status bar. The menu, button and tool bar comprises controls that are specific for each type of window, and only the controls of the active window are visible at any time. The status bar displays various messages or the progress of certain operations. The following sections show the ArcView window for the various active window types and describes the added or modified controls which extend the functionality of INSROP GIS compared to the basic ArcView configuration.

6.3.1 Projects window

The menu bar comprises two new pulldown menus: **Convert data** and **INSROP GIS Help**. In addition, the *File* menu comprises one additional item: **Open INSROP Project**. The button bar comprises three additional buttons (see Figure 6.1). There are no tools. Table 6.1 documents the modified menu bar, while the added buttons are described below:

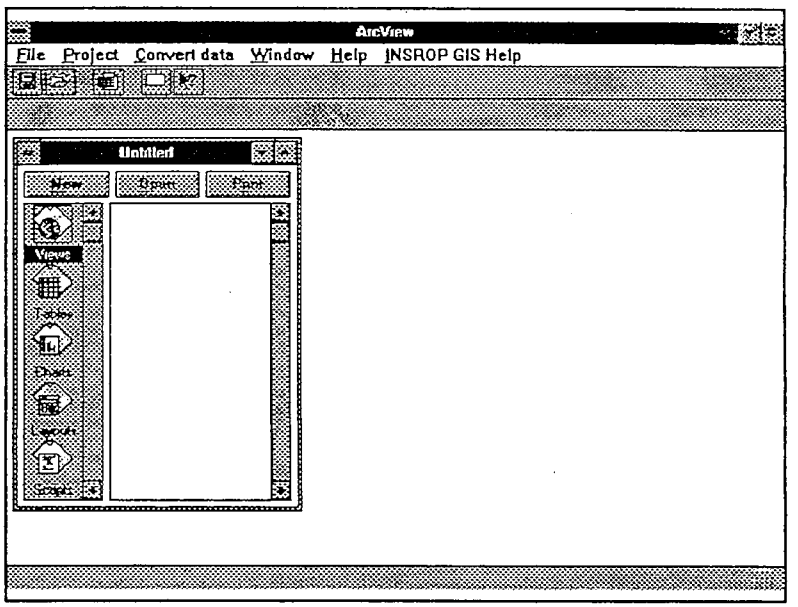


Figure 6.1 Projects window interface

Table 6.1 Menu bar modifications

Menu	Menu item label	Function	Triggered AVENUE script
File	Open INSROP Project	Select and open a project from the INSROP GIS project library	INSROP.ProjectOpen
Project	Encrypt all scripts	Encrypt all scripts in the project and remove them from the script editors.	INSROP.EncryptScripts
Convert data	Several items	Convert raw data in a special format to a suitable INSROP GIS data set	Various script depending on the item chosen. (Optional menu)
INSROP GIS Help	Contents	Displays the initial INSROP GIS (hypertext) Help contents topic	INSROP.HelpContents
	INSROP GIS Projects	Displays the INSROP GIS Projects topic	INSROP.Doc_Proj
	Search for INSROP GIS Help on...	Opens the search dialog box for an INSROP GIS keyword to search on	INSROP.HelpSearch
	Obtaining Technical support	Provides information about INSROP GIS support	INSROP.HelpSupport
	About INSROP GIS	Provides basic information about this version of INSROP GIS	INSROP.About

Button bar modifications:

Icon: 

Function: Select and open a project from the INSROP GIS project library
Triggered AVENUE script: INSROP.ProjectOpen

Icon: 

Function: Resize the ArcView background window to the maximum fit within the screen.
Triggered AVENUE script: INSROP.avMaxWin

6.3.2 Views window

The menu bar comprises three new pulldown menus: **Query**, **Analyze** and **INSROP GIS Help**. In addition, the *View* and *Theme* menus comprise addtional items. The button bar comprises four additional buttons (See Figure 6.2). There is one additional tool. Table 6.2

documents the modified menu bar, while the added buttons are described below:

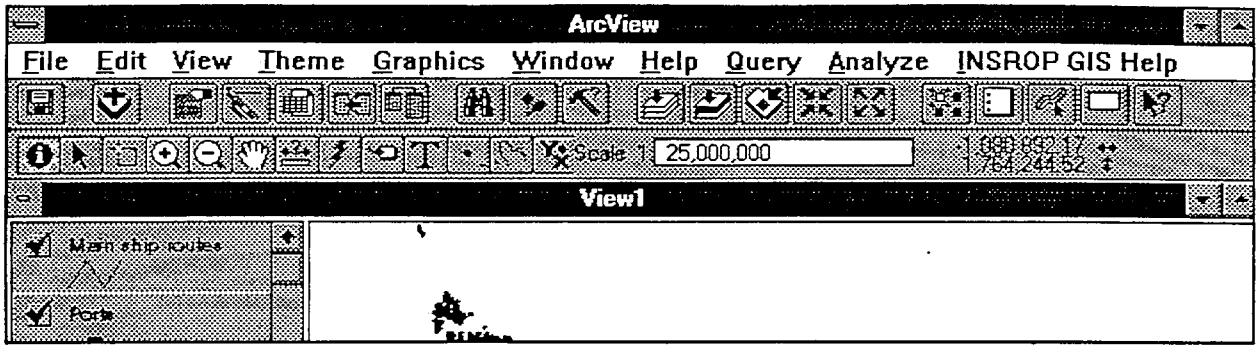


Figure 6.2 Top part of the View window interface

Table 6.2 Menu bar modifications

Menu	Menu item label	Function	Triggered AVENUE script
View	Use INSROP Projection	Set the projection to the default INSROP projection (see Section 9.2.1). All data must be stored in decimal degrees to use this function.	INSROP.DefaultProjection
	Add Graticule	Specify and add a graticule to the view	INSROP.Graticule
	Remove Graticule	Removes an existing graticule	INSROP.Graticule
	Add INSROP Theme	Select and add data set(s) from the INSROP GIS data base into the view	INSROP.AddTheme
	Create New Feature Theme	Create new features as a shape file theme	INSROP.CreateFeatures
	Create New Route Theme	Create a new route consisting of a line theme and an associated point theme with attributes	INSROP.RouteFromFileNew
Theme	Display attribute	Select a theme attribute to classify and display with a pre-defined legend	INSROP.DefaultLegend
	Display 'linked' attribute	Display an attribute in an associated table. Enables display of attributes in a one-to-many related table.	INSROP.DisplayLinkedAttribute
	Display Vector Arrows	Display arrows based on vector specifications in a point theme attribute table	INSROP.VectorArrows
	Remove Vector Arrows	Remove created vector arrows	INSROP.VectorArrows

	Project Features	Project the line segments making up polyline and polygon features	INSROP.ProjectFeatures
	Update Feature Sizes	Update or add projected Area/Perimeter or Length measures to the theme attribute table	INSROP.AreaPerimeterLength
	Modify Selected Features	Modify selected features in an existing theme	
	Add New Features	Add new features to an existing theme	
Query	Various query topics	Select a query on the chosen query topic	INSROP.QueryMain
Analyze	Quality control	Make a quality report of the attributes of the active themes	INSROP.QaQc_Theme
	Create Buffer Theme	Create a buffer theme around the selected features in the active themes	INSROP.Buffer
INSROP GIS Help	Contents	Displays the initial INSROP GIS (hypertext) Help contents topic	INSROP.HelpContents
	Search for INSROP GIS Help on...	Opens the search dialog box for an INSROP GIS keyword to search on	INSROP.HelpSearch
	Obtaining Technical support	Provides information about INSROP GIS support	INSROP.HelpSupport
	About INSROP GIS	Provides basic information about this version of INSROP GIS	INSROP.About

Button bar modifications:

Icon: 

Function: Specify tables and fields and join them
Triggered AVENUE script: INSROP.JoinTables

Icon: 

Function: Specify tables and fields and link them
Triggered AVENUE script: INSROP.LinkTables

Icon:
Function: Create a buffer theme around the selected features
Triggered AVENUE script: INSROP.Buffer

Icon: 

Function: Resize the ArcView background window to the maximum fit within the screen.
Triggered AVENUE script: INSROP.avMaxWin

Tool bar modifications:

Icon: 

Function: Digitize new features directly on the screen (editable themes only)
Triggered AVENUE script: INSROP.CreateFeaturesFromPoints

Tool bar modifications:

Icon: 

Function: Display the unprojected Y,X (e.g. Latitude/Longitude) coordinates of the mouse location (when clicked)
Triggered AVENUE script: INSROP.ShowMouseXY

6.3.3 Tables window

The menu bar comprises two new pulldown menus: **Analyze** and **INSROP GIS Help**. The button bar comprises one additional button (See Figure 6.3). There are no additional tools. Table 6.3 documents the modified menu bar, while the added button is described below:

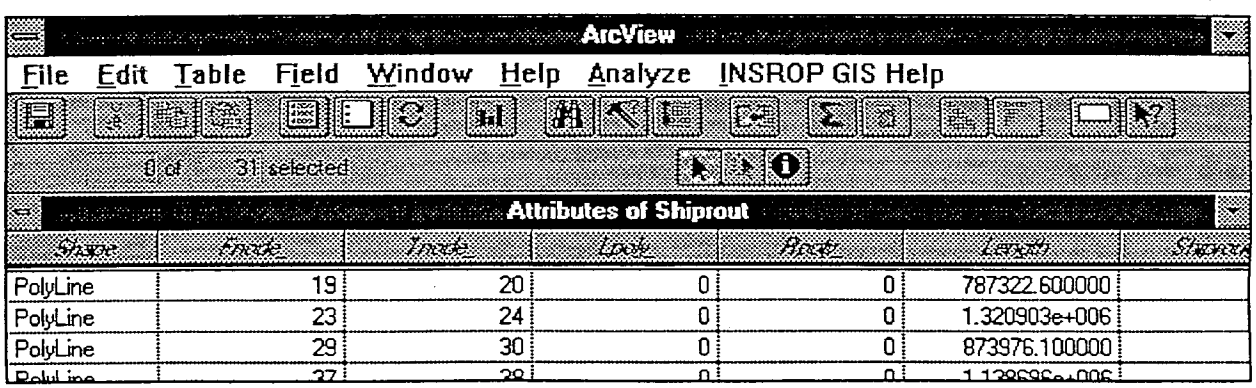


Figure 6.3 Top part of Table window interface

Table 6.3 Menu bar modifications

Menu	Menu item label	Function	Triggered AVENUE script
Analyze	Quality control	Make a quality report of the attributes of the active tables	INSROP.QaQc_Table
INSROP GIS Help	Contents	Displays the initial INSROP GIS (hypertext) Help contents topic	INSROP.HelpContents
	Search for INSROP GIS Help on...	Opens the search dialog box for an INSROP GIS keyword to search on	INSROP.HelpSearch
	Obtaining Technical support	Provides information about INSROP GIS support	INSROP.HelpSupport
	About INSROP GIS	Provides basic information about this version of INSROP GIS	INSROP.About

Button bar modifications:

Icon: 

Function: Resize the ArcView background window to the maximum fit within the screen.

Triggered AVENUE script: INSROP.avMaxWin

6.3.4 Charts window

The menu bar comprises one new pulldown menu: **INSROP GIS Help**. The button bar comprises one additional button. There are no additional tools. Table 6.4 documents the modified menu bar, while the added button is described below:

Table 6.4 Menu bar modifications

Menu	Menu item label	Function	Triggered AVENUE script
INSROP GIS Help	Contents	Displays the initial INSROP GIS (hypertext) Help contents topic	INSROP.HelpContents
	Search for INSROP GIS Help on...	Opens the search dialog box for an INSROP GIS keyword to search on	INSROP.HelpSearch
	Obtaining Technical support	Provides information about INSROP GIS support	INSROP.HelpSupport
	About INSROP GIS	Provides basic information about this version of INSROP GIS	INSROP.About

Button bar modifications:

Icon: 

Function: Resize the ArcView background window to the maximum fit within the screen.

Triggered AVENUE script: INSROP.avMaxWin

6.3.5 Layouts window

The menu bar comprises one new pulldown menu: **INSROP GIS Help**. In addition, the *Layout* menu comprises two new menu items(See Figure 6.4). The button bar comprises one additional button. There are no additional tools. Table 6.5 documents the modified menu bar, while the added button is described below:

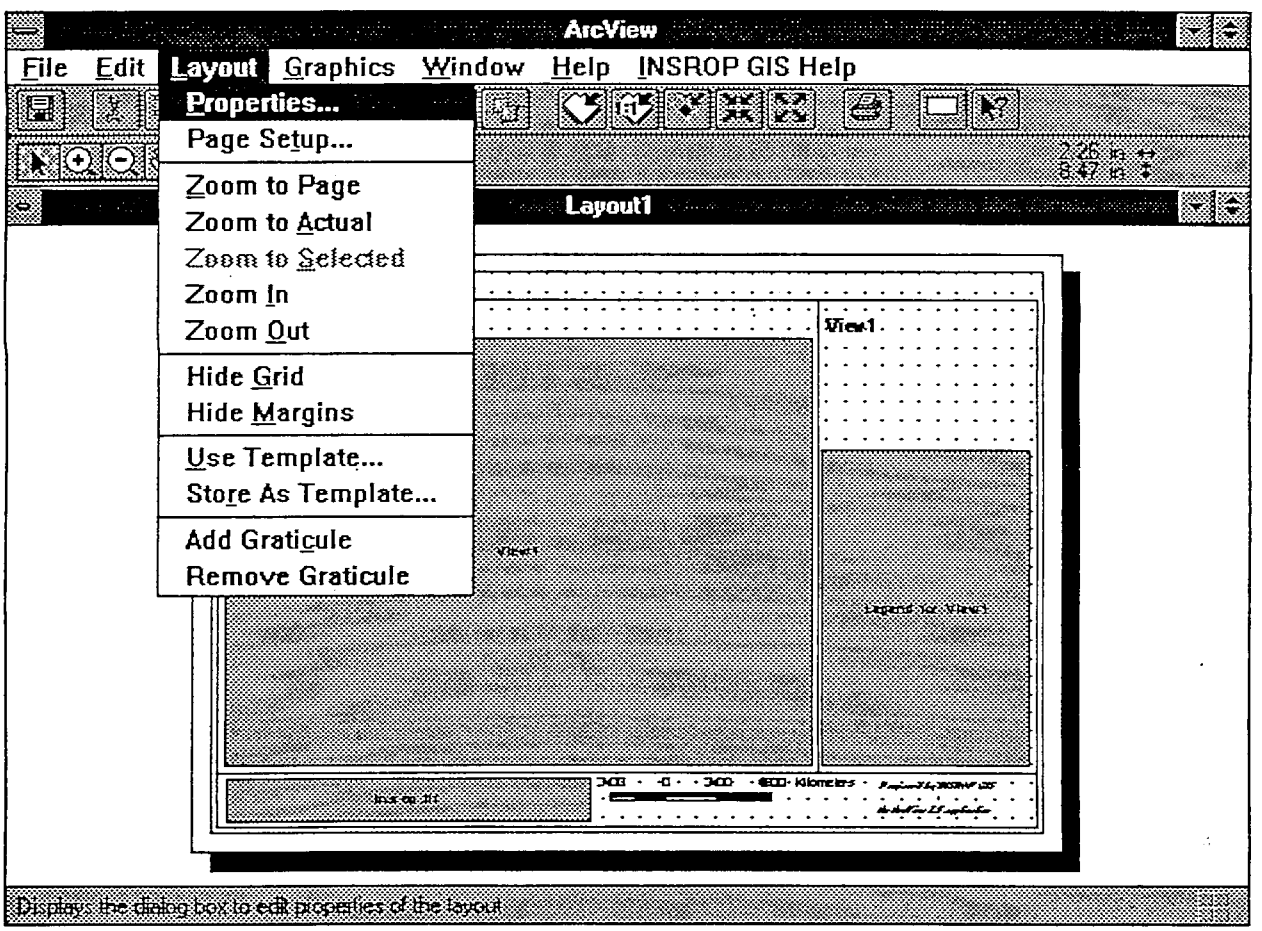


Figure 6.4 Layout window interface with the Layout menu open

Table 6.5 Menu bar modifications

Menu	Menu item label	Function	Triggered AVENUE script
Edit	Simplify LegendFrame	Simplify the legendframe by converting it into a graphical object, and thereby enabling editing (i.e. deleting) selected legend items	INSROP.SimplifyLegend
Layout	Add Graticule	Specify and add a graticule to the layouts view frame	INSROP.Graticule
	Remove Graticule	Removes an existing graticule	INSROP.Graticule
INSROP GIS Help	Contents	Displays the initial INSROP GIS (hypertext) Help contents topic	INSROP.HelpContents
	Search for INSROP GIS Help on...	Opens the search dialog box for an INSROP GIS keyword to search on	INSROP.HelpSearch
	Obtaining Technical support	Provides information about INSROP GIS support	INSROP.HelpSupport
	About INSROP GIS	Provides basic information about this version of INSROP GIS	INSROP.About

Button bar modifications:

Icon: 

Function: Resize the ArcView background window to the maximum fit within the screen.

Triggered AVENUE script: INSROP.avMaxWin

6.3.6 Scripts window

The menu bar comprises no new pulldown menus. However, the *Edit* menu comprises two new menu items. The button bar comprises one additional button. There are no tools. Table 6.6 documents the modified menu bar, while the added button is described below:

Table 6.6 Menu bar modifications

Menu	Menu item label	Function	Triggered AVENUE script
Edit	Find	Find a string in the script	INSROP.SedFindString
	Replace	Find and replace a string in the script	INSROP.SedFindAndReplace

Button bar modifications:

Icon: 

Function: Resize the ArcView background window to the maximum fit within the screen.

Triggered AVENUE script: INSROP.avMaxWin

6.4 INSROP GIS definition files

6.4.1 Database topics definition file

This file must exist. It is named **datasets.def** and is stored in the $NSR_DATA (i.e. C:\INSROP\DATABASE) directory. The purpose of the file is to define database information topics and the physical (storage) names of the database sub-directories comprising the data sets within each information topic. The file also defines whether there are additional sub directory levels. The file is used by the 'Add INSROP theme' menu in the Views window to allow the user to select datasets from information topic names rather than the more cryptic storage names with maximum 8-characters.

The file comprises three comma-separated (quoted) string columns, and the first two lines (= header lines) are mandatory and must not be changed. The first column comprises the physical name of each database information topic directory, while the second column comprise the associated descriptive name of each information topic. The third column have three legal values:

N or n	The data sets within this topic is located in the dataset topic directory
Y or y	The data sets within this topic is located in special dataset sub-directories below the dataset topic directory. THIS IS THE STANDARD DATABASE ORGANISATION.
F or f	This data sets topic have no data sets

If the column is empty, this is interpreted as N. The column specifies whether the data set shall be made available to the user. If the column comprises **F** or **f**, the data set topic at that line will not be available to the user. The reason for this option is that not all users may have the complete version of the INSROP GIS database, but it may still be useful to see which dataset topics exist in the complete version.The next lines comprise the actual information topic definitions. The following comprises the first three lines in the default datasets.def:

```
"Directory","Alias","Subsets"
"Overall dataset directory definition file",,
"basecart","Base Cartography","Y"
```

6.4.2 Data set definition files

These files must exist, and are stored in the database information topic sub-directories specified by the database topics definition file (datasets.def, ref. Section 6.4.1). The files are named **<subdirectory name>.def**, and has a similar structure and functionality as the datasets.def file. The following shows the first three lines in the basecart.def file:

```
"Directory","Alias","Subsets"
"Overall dataset directory definition file",,
"dcw_1","DCW 1:1 mill.","N"
```

The first two lines are common to all these files and must not be changed. The first column comprises the physical name of each data set directory, while the second column comprise the associated descriptive name of each data set. The third column have three legal values:

N or n	The data set files are located in the specified dataset directory.
Y or y	The data set files are located in a sub-directory of the specified dataset directory.
F or f	This data set is not available.

If the column is empty, this is interpreted as N. The "N" in the third column means that the actual datasets are stored in the $NSR_DATA\BASECART\DCW_1 directory and will show up in the 'Add INSROP Theme' dialogue window (if the DCW 1:1 mill base cartography datasets are selected).

6.4.3 Dataset member definition files

These files are stored in the sub-directories specified by the dataset definition file (**<subdirectory name>**.def, ref. Section 6.4.2). The files are named **<dataset directory name>.def**. The following shows the first four lines in the dcw_1.def file:

```
"Theme-ID","Themealias","Featureclass","Aliasfile","Coordsys"
"Digital Chart of the World (1:1 mill.) theme definition table",,,,
"dnnet","Drainage network","line","dnnetlin.fld","dd"
"dnnet","Drainage network","poly","dnnetpol.fld","dd"
```

The first line is common to all these files and must not be changed. The second line is a comment line identifying the data set. The four commas ending the line must be there. The third and fourth lines define two data set members (Two ArcView themes stored as one ARC/INFO coverage). The first column ("Theme-ID") comprises the storage (ARC/INFO coverage) (base)name of the data set members, dnnet, while the second column ("Themealias") specifies the theme alias name that will appear in the View's Table-of-Contents (TOC). The third column ("Featureclass") specifies the type of spatial feature (point, line, or polygon), while the fourth column ("Aliasfile") specifies the name of the respective dataset member attribute definition files. The featureclass specification is used to determine the format of the dataset member attribute definition file. The fifth column ("Coordsys") specifies the spatial coordinate system of the dataset member. "dd" means decimal degrees and is the recommended spatial coordinate system for INSROP GIS. The coordsys information is in some cases used by INSROP GIS to check that a theme in a projected View is stored as decimal degrees (However, it is the user's responsibility to only

include themes with similar coordinates systems in a View). Only the first column is mandatory, the other ones are optional.

These files are not required for all datasets, see Section 6.4.4. If you have many similar data sets in a directory, e.g. only separated in time, and you want to use a .def file for documentary purposes, you may use "All" in the *Theme-ID* column and "alias.fld" as *Aliasfile*.

6.4.4 Data set member attribute definition files

These files are stored in the same sub-directory as the dataset member definition file (**<dataset directory name>**.def, ref. Section 6.4.3). The filenames are also specified in this file, except in cases where there are e.g. many similar themes (for instance separated in time). In such cases a file named **alias.fld** may be used as valid for all themes in a data set directory, and the dataset member definition file (ref. Section 6.4.3) is not necessary. The following shows the contents of the dnnetlin.fld file:

"Fieldname","Alias","LUT-file"
"Field name aliases and corresponding LUT files for DCW DNNET line data",,
"Dnlntype","Drainage line type","dnlntype.lut"
"Dnlnstat","Drainage line status","dnlnstat.lut"

The first line is common to all these files and must not be changed. The second line is a comment line identifying the dataset member. The two commas ending the line must be there. The last two lines specifies the two descriptive attributes suitable for classification with a pre-defined default legend. The first column ("Fieldname") specifies the physical (storage) field names in the attribute table of the theme. The next column ("Alias") specifies a descriptive name for each attribute for use as choice options when selecting the Theme+'Display attribute' menu option in the Views interface. The third column ("LUT-file") specifies the name of the attribute values definition file (look-up-table) associated with each attribute (if one exists). The third column may be empty, in which case default symbols and random colours will be employed in the legend classification.

The main purpose of these files (and the ones described in Section 6.4.5) is to automate legend classification using the same symbology for all similar data themes, but they also serve as documentation of attributes and range of attribute values of each dataset member.

6.4.5 Data set member attribute value look-up-files

These files are stored in the same sub-directory as the attribute definition files (ref. Section 6.4.4). The filenames are also specified in this file. These LUT files differ in content depending on whether the theme has point, line or polygon data. The following shows the first two lines for each of the three LUT-file formats:

Point data:
"Value","Label","Symbol#","Color","Size"
"Look up table for *aTheme* point data: *anAttribute*",,,,

Line data:

"Value","Label","Symbol#","Color","Width"
"Look up table for *aTheme* line data: *anAttribute*",,,,

Polygon data:
"Value","Label","Symbol#","Color","OlWidth","OlColor","BgColor"
"Look up table for *aTheme* polygon data: *anAttribute*",,,,,,

The first line is common to all these LUT-files of each type and must not be changed. As seen, the first four columns are equal for all types, while the remaining column(s) differ, as they define the sixe and look of the chosen symbol. The second line is a comment line identifying the dataset member and attribute. The commas ending the line must be there. The next lines in the files will specifiy how each attribute value should be displayed. The first column ("Value") specify a legal value which may be included in the attribute field of this dataset member. The second column ("Label") specifies what this value represents. The label will also be used in the legend to identify the symbol for each attribute value.

The third column ("Symbol#") is the number of the symbol in the ArcView symbol palette (where 0 is the number of first symbol in the list). For point data the symbol# identifies a symbol in the *Marker* symbol palette, for line data the symbol# identifies a symbol in the *Pen* symbol palette, while for polygon data the symbol# identifies a symbol in the *Fill* symbol palette. The symbol# column may be empty (,,), in which case the default symbol will be used (filled circle, solid line, or solid fill polygon).

The fourth column ("Color") specifies the name of the color to be used for the symbol. This column may be empty (,,), in which case a random color will be used. The color names are similar to UNIX color name table, and comprises 738 colornames defined by their RGB values. Appendix A list the contents of the AVENUE script specifying the colornames. To get a transparent fill, the color must be specified as "Transparent".

The next columns are optional. The fifth column specifies a size property of the symbol. For point data the fifth column ("Size") specifies the marker size in pts, while for line data the fifth column ("Width") specifies the line width in pts. For polygon data the fifth column ("OlWidth") specifies the outline width in pts, while the sixth ("OlColor") and seventh ("BgColor") columns defines the outline and background color respectively. To achieve a polygon fill without an outline, the outline color must be specified as "Transparent".

As an example, the following shows the first four and the last lines of the dnlntype.lut file:

"Value","Label","Symbol#","Color","Width"
"Look up table for DCW DNNET line data: Dnlntype",,,,
1,"Stream, river, channelized river",0,"lightblue",
2,"Inland water body shoreline",0,"lightblue",
...
9,"Tile boundary or null arc",,"Transparent",

The main purpose of these files (and the ones described in Section 6.4.4) is to automate legend classification using the same symbology for all similar data themes, but they also serve as documentation of attributes and range of attribute values of each theme.

6.5 INSROP GIS Projects Library

During the course of the INSROP GIS development several task-oriented project files are prepared. Some of these include views with basic cartographic data to serve as a base for adding new data sets. Other project files are prepared for general or specific analysis tasks, generally prepared to serve the needs of one or several INSROP projects. Some of these project files may include a modified INSROP GIS interface and/or additional AVENUE scripts for special purposes. The INSROP GIS project files are stored in the PROJECTS sub-directory, i.e. C:\INSROP\PROJECTS, but an **Open INSROP Project**-button in the customized projects button bar enables the user to directly select a project from this catalogue.

While the INSROP GIS database provides organized data for use in INSROP GIS (ArcView), the project files provide a higher organization level, in the sense that the project files include an organized and classified subset of the INSROP GIS database. Project files may also comprise the end result of analyses, allowing the user to study the results and e.g. zoom in on special areas or get additional information about certain features.

6.6 INSROP GIS Database

The database module organises and stores INSROP data for retrieval and analysis. This involves both storage of structural data (i.e., actual data) and descriptive data (i.e., metadata). A prime objective for organising data from several sources into a database is to make it available to users other than the original data supplier. To support this objective, it is equally important to include information about the data set. The section provides basic information about the INSROP GIS database. A separate report (Brude *et al*, 1995) descibes the database in more detail. In addition, the INSROP GIS database metadata provide documentation on each INSROP GIS data set.

6.6.1 Core Database

Although INSROP GIS may utilize data sets from any location a disk, all INSROP GIS data sets are organized in sub-directories of the DATABASE directory, which again is a sub-directory of the main INSROP GIS directory (i.e. C:\INSROP). The DATABASE sub-directories include information themes, with 8-character abbreviated names, e.g. BASECART for Base Cartographic data, to comply with current DOS restrictions. The individual base cartographic data sets are stored as sub-directories below the BASECART directory. E.g., the special INSROP extract of the Digital Chart of the World in 1:1 mill. is found as ARC/INFO coverages in the C:\INSROP\DATABASE\ BASECART\DCW_1 directory. However, this storage structure is primarily only important for system development or updates. The installation software will store the data in the proper location, and access to the data are provided by selecting the requested information theme from a menu providing proper descriptive names (e.g. Base Cartography).

6.6.2 Metadata

Data stored within a GIS are inherently complex. Data sets may be defined by a variety of parameters from data structures (e.g. raster, vector, triangulated irregular networks) to geographic environments (e.g. spheroids, datums, projections, co-ordinate systems) to

sources and processing procedures (e.g. remotely sensed, rectified, classified, digitised). Consequently, there must be an organised method available to store, query, and retrieve information. Therefore, it is the role of metadata to provide the GIS user with documented information describing basic data characteristics and allow for the assessment of GIS data sets within the core database. For example, once a metadata database has been established the information can support user interface applications by providing documentation on data sets. Also, the metadata database serves a strategic role in supporting the GIS database by providing the capability to assess data quality, determine missing or suspect information, and survey the history of processing steps (i.e., lineage) associated with the GIS data.

Two types of metadata information are employed in connection with INSROP GIS [1]. The first type is the documentation of the data sets, which also comprises documentation about the automation process according to the international Directory Interchange Format (DIF) standard. The other type of metadata documents how data sets are used. In INSROP GIS, data sets are combined into INSROP GIS projects, which are physically stored as ArcView project files. Both metadata types are to be made available in the INSROP GIS online help system, and the DIF documentation is also available as ASCII text files.

The documentation of an INSROP GIS data set shall provide a user with enough information to enable a decision of whether the data set suits the users needs and satisfies the users requirements. Each data set should be completed with the requested information, including the DIF documentation. Except for the DIF documentation, the documentation standard chosen for INSROP GIS data sets is a modified version of the documentation layout used for the ArcView Sample Data (see Guide to ArcView Sample Data in the online ArcView Help documentation).

The documentation of an INSROP GIS project file shall provide information on where to find the project file, which data sets are included in the project, and any other information which may be important to a user. This documentation is not a part of the INSROP GIS data set documentation. However, the project file documentation provides links to the data sets used in the INSROP GIS project to provide the requested information. Similar to the data set documentation it is organized with one part showing the contents of the INSROP GIS project, and a second part with several elements documenting the various components of the project file.

For more information about how to document INSROP GIS data sets and projects, see Brude *et al.* 1995.

6.6.3 Directory Interchange Format (DIF)

The *Directory Interchange Format* (DIF), originally created by NASA and used to document data sets in the NASA Global Change Master Directory, is a well known and well documented standard used to store metadata information. The standard suggested for INSROP is a modified version of the NASA DIF and is also proposed to be used by a number international programs active in arctic data collection, such as GRID-Arendal's Arctic Database for Europe and Asia and the International Arctic Data Directory.

Modifications have been made to the NASA DIF to better describe information about data automation methods. Also, the modified DIF differs from the NASA format with respect to the keyword thesaurus used to describe data sets. The entries in this DIF are based on a United Nations Environment Programme's Infoterra Thesaurus. This thesaurus provides a broader selection of terms that are more appropriate for INSROP data sets. Another problem with a DIF is the lack of structures for documenting more specific details related to spatial data sets. A group called "Data set member" is added in order to support this.

Data sets that are completed using the DIF format will adhere to a developing international standard of metadata and have the potential to be incorporated into several international databases.

7 HOW TO GET STARTED

7.1 Starting INSROP GIS

INSROP GIS is started by double-clicking the INSROP GIS icon in the INSROP GIS MS windows program group. This activates ArcView with the customized INSROP GIS interface and associated AVENUE-scripts, as stored in the INSROP GIS *default.apr* file.

Note: It is required that INSROP GIS is installed properly (see Chapter 4), so that ArcView recognises the INSROP GIS working directory as the *HOME* directory.

7.2 Creating, opening or saving a project

When starting INSROP GIS, a new project (called Untitled) is "created". If you want to work on an existing project, you must select **Open Project** or **Open INSROP Project** from the *File* menu in the Project window, or press the **Open INSROP Project**-button in the Project button bar. The difference between these options is that the **Open INSROP Project** option, as default, enables choosing project files from <INSROP-dir>\PROJECTS directory, while the **Open Project** option, as default, enables choosing project files from the current working directory.

To save a project with its current name, you have three options (which are available in all ArcView windows):

i. Choose **Save Project** in the *File* menu.
ii. Type Ctrl+S as a shortcut for the **Save Project** menu item.
iii. Press the **Save Project** button (The leftmost button in each button bar)

To save a project with a new name, you must make the Project window active, and choose the **Save Project As...** menu item in the *File* menu.

7.3 Using projects from the INSROP GIS project library

The INSROP GIS project library includes a set of ArcView projects prepared for a special INSROP task (see also Section 6.4 and Brude *et al.*, 1995). Some of these projects, i.e. base cartographic projects, are intended to serve as building blocks when creating a new project. This is achieved by importing a project into a new project and thus avoid having to add these data sets separately each time they are wanted. To find out whether a project in the INSROP GIS project library may be useful for your purpose, please study the INSROP GIS project library documentation (The hypertext version should be the most updated one, see Section 7.4).

Note: As an ArcView project comprises links to data sets, and not the actual data, all linked data sets must be available for the project to open properly. When trying to open or import a ArcView project which includes links to deleted or moved data sets, ArcView has a project repair capability which will ask for the new data location. If you really do not wish to include a moved data set, this project repair may be cancelled, either for each missing data file at a time, or for all missing data sets.

7.4 Accessing metadata

The purpose of metadata is to provide information about *something*, and thereby enables a potential user to decide on the usefulness of *something* for any given purpose (see also Section 6.6.3). The INSROP GIS metadata consist of two main parts: INSROP GIS projects metadata and INSROP GIS database metadata. A version of INSROP GIS should include updated versions of these metadata as hypertext. However, as the database or projects library is likely to be updated between new versions of INSROP GIS, the metadata may also be accessible as a set of ASCII files.

The INSROP GIS hypertext metadata are accessed through the *INSROP GIS Help* menu (available from all INSROP GIS (ArcView) windows or directly by double-clicking the proper hypertext file in the INSROP GIS MS Windows project group. The INSROP GIS project library metadata are accessed from the hypertext program item labelled **INSROP GIS Projects**, while the database metadata are accessed from the hypertext program item labelled **INSROP GIS Data.**

The metadata ASCII files are accessed by opening them in a text editor.

Note: The database ASCII metadata files are located in the main sub-directory of each data set, and they should have **.dif** as filename extension. The project library ASCII metadata files are located in the <INSROP-dir>\PROJECTS sub-directory. The files are stored with <project name>**.inf** filename, making them easy to open in Notepad (MS Windows ASCII text editor) as *.inf* is the default Notepad filename extension.

8 MAKING DATA USEFUL IN INSROP GIS

GIS data consists of a spatial component and a tabular component. The spatial component comprises spatial coordinates (e.g. latitude and longitude) describing the location and shape of spatial features. The tabular component comprises attributes describing the characteristics of the spatial features. In ArcView, the spatial features are displayed in a view, while the attributes can be used to classify the spatial features and thereby defining how they should be displayed.

ArcView can handle both spatial and non-spatial data sets. With spatial data we mean GIS data where the spatial and tabular components are stored permanently in a unified set of files (i.e. ARC/INFO coverages or ArcView shapefiles). However, non-spatial tabular data can be joined or linked (through common IDs) to a spatial data set and will then serve as an extended attribute table.

In INSROP GIS we have also prepared a set of AVENUE scripts allowing creation of spatial data sets (ArcView shapefiles) by importing spatial coordinates and attribute data organised in an ASCII file. If you in addition store and document the data sets you create according to the INSROP GIS requirements, these data sets becomes part of your local INSROP GIS data base.

8.1 Using data from the INSROP GIS database

The INSROP GIS database (see also Section 6.6) includes data sets prepared for use in an ArcView application, i.e. INSROP GIS. Some of these projects, i.e. base cartographic projects, are intended to serve as building blocks when creating new project. This is achieved by importing a project into a new project and thus avoiding having to add these data sets separately each time they are wanted. To find out whether there are, or which, data in the INSROP GIS database that may be useful for your purpose, please study the INSROP GIS database documentation (The hypertext version should be the most updated one, see Section 7.4, or study Brude *et al*, 1995).

The INSROP GIS database includes mainly spatial data sets, but also some non-spatial data sets. Spatial data sets are for use in a view document, while non-spatial data sets are for use in a table document. Spatial data may consist of one (ArcView) theme or a set of themes, where a theme comprise a set of spatial features (with attributes) of one feature type (Point, Multipoint, Line, Polyline, Polygon) or an image. Non-spatial data are organised into tabular files in one of the following formats: INFO, dBase, delimited ASCII file.

8.1.1 Add an INSROP theme to a view document

To add a theme from the INSROP GIS database into a view, choose **Add INSROP Theme** in the *View* menu (See Figure 8.1). This will initiate the following actions:

1. Get the thematic list of INSROP GIS data sets
2. Select a topic from which to add an INSROP GIS data set
3. Get the list of INSROP GIS data sets within the selected topic
4. Select an INSROP GIS data set within this topic
5. Get the definition file associated with the selected data set
6. Select one or several data themes within the selected INSROP GIS data sets

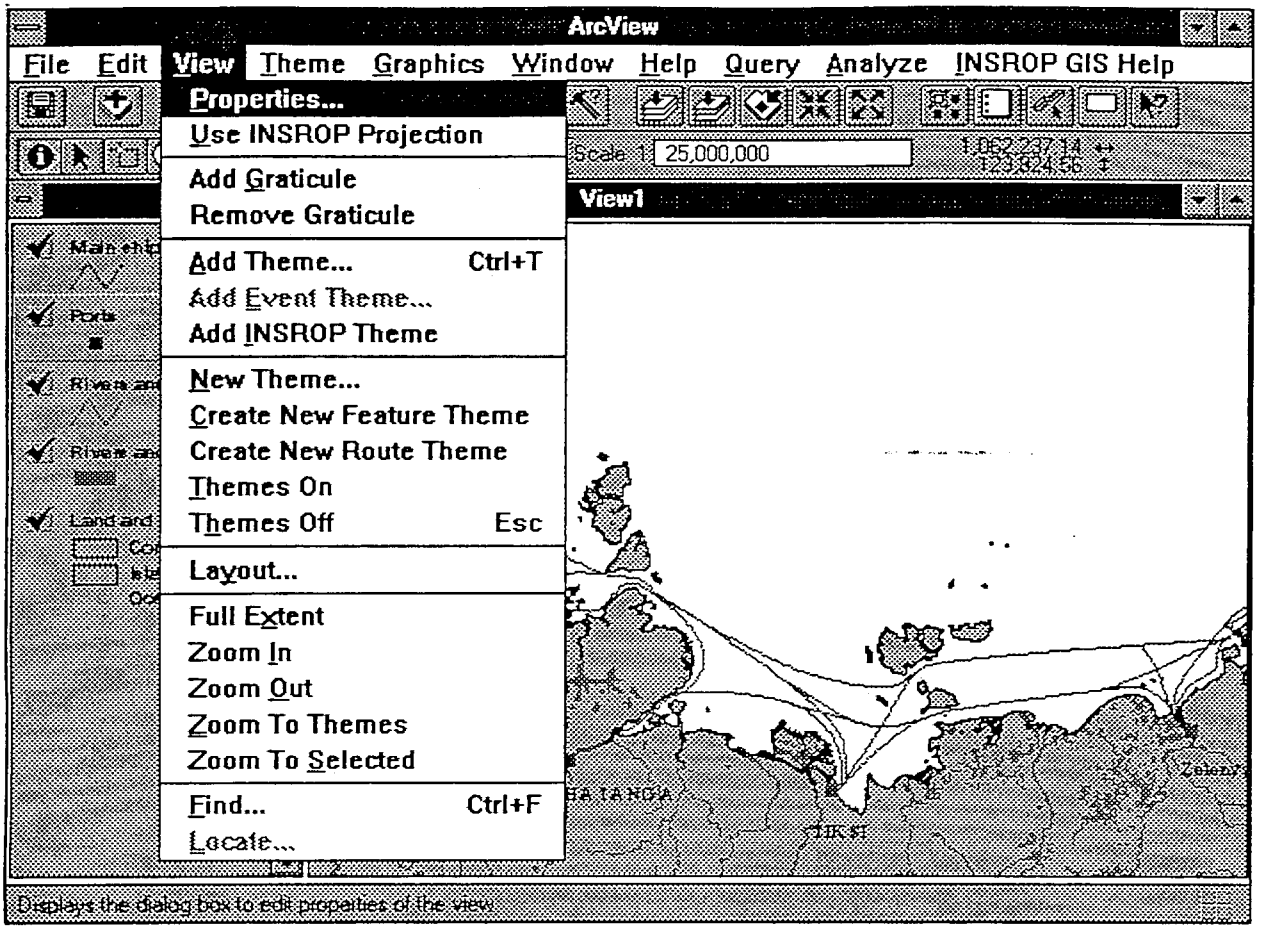


Figure 8.1 The view menu in the View window interface

These steps are described in more detail in the following:

Get the thematic list of INSROP GIS data sets

The thematic list of INSROP GIS data sets is read from the overall dataset definition file, *datasets.def*, see Section 6.4.1.

Select a topic from which to add an INSROP GIS data set

Based on the information acquired in Step 1, the user is presented a list of data set topics, from which one topic may be selected.

Get the list of INSROP GIS data sets within the selected topic

Based on the user's choice in Step 2, INSROP GIS goes to the proper database sub-directory. This sub-directory comprises a topic dataset definition file, see Section 6.4.2, which is read by INSROP GIS.

Select an INSROP GIS data set within this topic

Based on the information acquired in Step 3, the user is presented a list of data sets,

from which one data set may be selected.

Get the definition file associated with the selected data set

Based on the user's choice in Step 4, INSROP GIS goes to the proper data set directory. This directory comprises a dataset definition file, see Section 6.4.3, which is read by INSROP GIS.

Select one or several data themes within the selected INSROP GIS data sets

A data set may comprise one or several data set themes. INSROP GIS opens an "Add Theme"-dialogue window, where the user may select one or more themes to add to the view. This step is similar to the standard ArcView "Add Theme" menu item. However, in addition to adding the selected themes, INSROP GIS gives the themes the names defined in the data-set definition file read in Step 5.

8.1.2 Add a tabular file as a table document

As ArcView may import several table formats and the file extensions also may vary, we have **not** prepared a special **Add INSROP Table** menu item. Consult the INSROP GIS data set documentation to which tables exist and where they are stored, and use the standard ArcView **Add Table** menu item to open the table.

8.2 Creating a new feature theme from an ASCII file

8.2.1 General procedure for creating new feature themes by importing spatial ASCII data

INSROP GIS can create new feature themes by importing spatial data stored in delimited ASCII table files. The steps involved when importing spatial ASCII data are as follows:

1. Select type of spatial feature to create
2. Select input file
3. Select delimitor
4. Select spatial coordinate units
5. Specify output shapefile name
6. Specify the ID attribute and number of other attributes to include
7. Confirm whether the file columns are defined in headerline(s)
8. Specify which columns comprise the X and Y coordinates
9. Define attribute field names and specify the fields
10. Read the input file and create the features

These steps are further described in the following:

Select type of spatial feature to create

The spatial features may be points, multipoints, lines, multilines, polygons, multi-

polygons or "routes". The "routes" are not the same as ARC/INFO routes, see Section 8.2.8. The user is prompted to select one of these feature types.

Select input file

The user is presented a file selection dialogue window, and must navigate the directory tree to find and select the proper ASCII file. As default, the ASCII input file is expected to have **.pos** as filename extension, but any legal filename (pattern) can be specified by the user.

Select delimitor

The ASCII files specifying the coordinates and attributes of the spatial features comprise this information in delimited columns. As delimitor, comma, semi-colon or space may be used. Two delimitors in a row is interpreted as an empty column value. Hence, there must never be more than one delimitor between each column, unless the column value really should be empty. There can also not be more than one type of delimitors in each file. If a comma or semi-colon is used as delimitor, any spaces on either side of the delimitor are ignored.

Select spatial coordinate units

The input data may be stored as decimal degrees, degrees and minutes, or some projected coordinates. Input coordinates stored as degrees and minutes will be converted to decimal degrees.

Specify output shapefile name

The user is presented a file creation dialogue window, and must navigate the directory tree to find and select the proper storage location and specify the file name. An existing file may be selected (and overwritten) if the file is not in use.

Specify the ID attribute and number of other attributes to include

Although not strictly required by ArcView, the INSROP GIS development team has found it useful to require that an ID is assigned to each spatial feature. In most cases this ID is just a serial (integer) number, but it may also be a character string. In principle the ID may also be e.g. a location name, but we recommend that the ID attribute is kept short. For string IDs the table field is created as a variable-length character field (see Section 8.2.9), and the user is prompted for maximum number of characters in the string. For number IDs the table field is created as a short integer field with a maximum of 4 digits (-999 - 9999)

The specification of the ID attribute includes "ID-column label", "ID-column #" and "ID-type (String/Number)", and the user is prompted for these values. The default values are "ID", "1", "N", which means that the ID attribute field should be named ID, the IDs are stored in column #1 in the ASCII file and consist of Integer values. The ID column does not have to be the first column, but all columns to the left of the ID column are ignored.

In addition to the specification of the ID column, the user is prompted for the number of attributes to include in the created shapefile. The default value is the number of attributes INSROP GIS has identified in the ASCII file, but the user may specify a smaller number if some of the rightmost attribute columns should be ignored.

Confirm whether the file columns are defined in headerline(s)

The user is prompted to confirm whether the ASCII file column names are included in headerline (s).

Specify which columns comprise the X and Y coordinates

The spatial coordinates must be in two sequential columns, but it doesn't matter whether the X- or the Y-coordinate column is the first of the two. For geographic coordinates the longitude (+ = East) represents the X coordinate and the latitude (+ = North) represents the Y coordinate. If the ASCII file includes a headerline with N and E included in the two coordinate column names (i.e. North and East), INSROP GIS use this to determine which coordinate columsn comprise which coordinate. If not, the user is prompted to specify the column numbers for the two coordinate columns.

Define attribute field names and specify the fields

If the ASCII file includes a headerline, the attribute field names are read from the headerline, and the user is prompted for the type of each attribute. If not, the user is prompted for attribute table field names and specifications. For decimal and string attributes, the user is also prompted for some additional field specifications (see Section 8.2.9).

Read the input file and create the features

Based on the given user input, the ASCII file is read and the shapefile created. During this process the system checks that line features are defined by a minimum of two points and polygon features by a minimum of three points.

8.2.2 Importing ASCII point features

A point feature is characterized by a location and descriptive attributes. The ASCII file may have zero or one header line. A headerline must comprise the ID column name, coordinate column names and attribute field names. The following lines must comprise the actual ID, coordinates, and attribute values for each point feature.

To import point features from an ASCII file, choose the **Create New Feature Theme** menu item in the *View* menu in the view window.

8.2.3 Importing ASCII multipoint features

A MultiPoint feature is characterized by a set of points sharing the same descriptive attributes. The ASCII file may have zero or two header lines. The first headerline must comprise the ID column name and attribute field names, while the second headerline comprise the coordinate column names. The next lines define the multipoint features, where each feature is defined with an attribute line, one coordinate line for each point in the multipoint feature, and an 'end-of-feature' line. The attribute line comprises the feature ID and other attribute values for the multipoint feature. The 'end-of-feature' line just comprises the word END in the beginning of the line. After reading this line, INSROP GIS interprets the next line as the attribute line of the next multipoint feature.

To import multipoint features from an ASCII file, choose the **Create New Feature Theme** menu item in the *View* menu in the view window.

8.2.4 Importing ASCII line features

A line feature is characterised by a continuous line, whose location and shape is specified by a set of points, and descriptive attributes. The ASCII file may have zero or two header lines. The first headerline must comprise the ID column name and attribute field names, while the second headerline comprise the coordinate column names. The next lines defines the line features, where each feature is defined with an attribute line, one coordinate line for each point defining the line feature, and an 'end-of-feature' line. The attribute line comprises the feature ID and other attribute values for the line feature. The 'end-of-feature' line just comprises the word END in the beginning of the line. After reading this line, INSROP GIS interprets the next line as the attribute line of the next line feature.

To import line features from an ASCII file, choose the **Create New Feature Theme** menu item in the *View* menu in the view window.

8.2.5 Importing ASCII multiline features

A Multiline feature is characterised by a set of discontinuous lines, where the location and shape of each line is specified by a similar number of point sets, and sharing the same descriptive attributes. The ASCII file may have zero or two header lines. The first headerline must comprise the ID column name and attribute field names, while the second headerline comprise the coordinate column names. The next lines defines the multiline features, where each feature is defined with an attribute line, one coordinate line for each point defining the multiline feature, a number of 'newpart' lines, and an 'end-of-feature' line. The attribute line comprises the feature ID and other attribute values for the multiline feature. The 'newpart' line just comprises the word NEW in the beginning of the line. After reading this line, INSROP GIS interprets the next line as the first point coordinate in the next line part of the multiline feature.The 'end-of-feature' line just comprises the word END in the beginning of the line. After reading this line, INSROP GIS interpretes the next line as the attribute line of the next multiline feature.

To import multiline features from an ASCII file, choose the **Create New Feature Theme** menu item in the *View* menu in the view window.

8.2.6 Importing ASCII polygon features

A polygon feature is characterised by a closed line, whose location and shape is specified by a set of points, and descriptive attributes. The ASCII file may have zero or two header lines. The first headerline must comprise the ID column name and attribute field names, while the second headerline comprise the coordinate column names. The next lines define the polygon features, where each feature is defined with an attribute line, one coordinate line for each point defining the polygon feature, and an 'end-of-feature' line. The attribute line comprises the feature ID and other attribute values for the polygon feature. The 'end-of-feature' line just comprises the word END in the beginning of the line. After reading this line, INSROP GIS interprets the next line as the attribute line of the next polygon feature.

To import polygon features from an ASCII file, choose the **Create New Feature Theme** menu item in the *View* menu in the view window.

8.2.7 Importing ASCII multipolygon features

A MultiPolygon feature is characterised by a set of polygons, whose location and shape is specified by a similar number of point sets, and sharing the same descriptive attributes. The ASCII file may have zero or two header lines. The first headerline must comprise the ID column name and attribute field names, while the second headerline comprise the coordinate column names. The next lines define the multipolygon features, where each feature is defined with an attribute line, one coordinate line for each point defining the multipolygon feature, a number of 'newpart' lines, and an 'end-of-feature' line. The attribute line comprises the feature ID and other attribute values for the multipolygon feature. The 'newpart' line just comprises the word NEW in the beginning of the line. After reading this line, INSROP GIS interprets the next line as the first point coordinate in the next polygon of the multipolygon feature.The 'end-of-feature' line just comprises the word END in the beginning of the line. After reading this line, INSROP GIS interprets the next line as the attribute line of the next multipolygon feature.

To import multipolygon features from an ASCII file, choose the **Create New Feature Theme** menu item in the *View* menu in the view window.

Note: If a multipolygon includes overlapping polygons, ArcView 2.1 interprets the overlap zone as a hole. Unless this is the user's intention, multipolygon features should be created from non-overlapping polygons. For non-overlapping polygons, a multi-polygon feature is conceptually similar to an ARC/INFO Region feature.

8.2.8 Importing ASCII "route" features

A"route" data set is created from one ASCII file, but consists of two data set members: a point theme and a line theme. The point theme comprises the location of points along the "route", e.g. a ship route or historical voyage, and has attribute values describing some aspects of each point along a route. The line theme is created as lines through each set of point coordinates and with attributes describing some aspects valid for each route feature. The INSROP GIS "routes" could also be modelled as ARC/INFO routes, which in ArcView is accomplished to the use of Events. However, as the information in the point themes is useful for analyses not requiring the actual route lines, we have chosen to keep the two information themes separate.

The procedure for importing "route" features is fairly similar to the general procedure for importing spatial ASCII data (see Section 8.2.1), but as the type of spatial feature (points and line) is preset, Step 1 is skipped. Also, as INSROP GIS queries and analyses on routes require decimal degrees as source data, only decimal degrees (DD) and degrees and minutes (DM) are legal input coordinate systems.

Another difference is that when specifying the output shape file name in Step 5, you cannot use more than 6 characters (before the .shp extension) for the filename. The reason is that these 6 characters are used as basename, where the point theme is extended with **_p**, and the line theme with **_l**. In this way the two themes are stored in different shapefiles (which they must be), but it is easy to identify the two shapefiles making up a "route" feature.

After the "route" shapefiles are created they are added to the view as new themes.

The point theme is given a name consisting of the 6-character basename + "Waypoints", while the line theme is given a name consisting of the 6-character basename + "Route". Table 8.1 shows how the user's choice of shapefile name is utilized.

Table 8.1 Example of naming conventions for "route" features.

User's shapefile name	Feature type	Actual shapefile names	Theme names
sail95.shp	point	sail95_p.shp	sail95 Waypoints
	line	sail95_l.shp	sail95 Route

8.2.9 Attribute field specifications

An ArcView table field, e.g. an attribute field, is defined by its *Name*, *Type*, *Width* and *Precision*. The Name is the true name of the field on the disk, while the Type is the data type of the field. The Width is the number of characters the data in the field occupies, while the Precision is the decimal precision of the field. Note that a decimal point is included in the field width (e.g. 12.34567 has a field width of 8 and precision of 5). Table 8.2 shows a list attribute table field types (used by INSROP GIS) and associated field widths and decimal precisions. The field type column comprises the field type choices available to the user of INSROP GIS, while the FieldEnum is the associated field type created by ArcView. The Width and Precision columns show the values used in INSROP GIS to create the table fields. An asterisk (*) means that the user is prompted for this value. The field width must be large enough to include the largest/longest foreseen attribute value.

Table 8.2 ArcView attribute table field specifications

Field type	FieldEnum	Width	Precision
Integer - 1 byte	#FIELD_BYTE	3	0
Integer - 2 bytes	#FIELD_SHORT	5	0
Integer - 2 bytes	#FIELD_LONG	10	0
Real - 4 bytes	#FIELD_FLOAT	*	*
Real - 8 bytes	#FIELD_DOUBLE	*	*
String - fixed length	#FIELD_CHAR	*	0
String - variable length	#FIELD_VCHAR	*	0
Boolean - Y/N	#FIELD_LOGICAL	1	0
Date - 'yyyymmdd'	#FIELD_DATE	8	0
Date - yyyy-mm-dd'	#FIELD_ISODATE	10	0
Time - 'hh:mm:ss:dd.nnn'	#FIELD_ISOTIME	10	0
Date and Time - 'yyyy-mm-dd hh:mm:ss:dd.nnn	#FIELD_ISODATETIME	24	0

8.3 Creating a new feature theme interactively

8.3.1 General procedure for creating new feature themes interactively

INSROP GIS can create new feature themes interactively. However, both for speed and reduced risk of erroneous input, we recommend the procedure described in Section 8.2. Nevertheless, the steps involved are:

1. Select type of spatial feature to create
2. Select spatial coordinate units
3. Specify output shapefile name
4. Specify the ID attribute and number of other attributes to include
5. Specify attribute field names and specify the fields
6. Specify attribute values for each feature
7. Specify coordinates of the points defining each feature
8. Create the features

These steps are further described in the following:

Select type of spatial feature to create

The spatial features may be points, multipoints, lines or polygons. The user is prompted to select one of these feature types.

Select spatial coordinate units

The input data may be stored as decimal degrees, degrees and minutes, or some projected coordinates. Input coordinates stored as degrees and minutes will be converted to decimal degrees.

Specify output shapefile name

The user is presented a file creation dialogue window, and must navigate the directory tree to find and select the proper storage location and specify the file name. An existing file may be selected (and overwritten) if the file is not in use.

Specify the ID attribute and number of other attributes to include

Although not strictly required by ArcView, the INSROP GIS development team has found it useful to require that an ID is assigned to each spatial feature. In most cases this ID is just a serial (integer) number, but it may also be a character string. In principle the ID may also be e.g. a location name, but we recommend that the ID attribute is kept short. For string IDs the table field is created as a variable-length character field (see Section 8.2.9), and the user is prompted for maximum number of characters in the string. For number IDs the table field is created as a short integer field with a maximum of 4 digits (-999 - 9999)

The specification of the ID attribute includes "ID-column label" and "ID-type (String/Number)", and the user is prompted for these values. The default values are "ID" and "N", which means that the ID attribute field should be named ID and consist of Integer values. In addition, the user is prompted for the number of attributes to include in the created shapefile.

Specify attribute field names and specify the fields

The user is prompted for attribute table field names and specifications. For decimal and string attributes, the user is also prompted for some additional field specifications (see Section 8.2.9).

Specify attribute values for each feature

The user is prompted to specify the attribute values for each feature. All attributes must be given a value. If for some features, one or more attribute values are unknown, a special value representing unknown values must be assigned as attribute value.

Specify coordinates of the points defining each feature

For point features the coordinates were entered in the previous step, but for other feature types the user is prompted to enter the coordinates of the points defining a feature one by one. Enter the proper coordinates and choose *OK* for all points to be included in the feature, and choose *Cancel* after all points defining a feature are entered. Then the user is prompted for the ID and attributes of the next feature. When the data for all features to be defined are entered, Choose *Cancel* to finish this step.

Create the features

Based on the given user input, a shapefile is created. During this process the system checks that line features are defined by a minimum of two points and polygon features by a minimum of three points.

8.4 Importing tabular ASCII data

Delimited ASCII tables can be imported using the ArcView's *Add Table* menu item in the Project window. By default .txt filename extensions are expected, but the filename (pattern) is easily modified in the "Add Table"-dialogue window which appears.

8.5 Using and documenting your own data

You may extend your INSROP GIS database by adding your own data, either data prepared in ARC/INFO format or as ArcView shapefiles, by creating ArcView shapefiles using the procedure described in Sections 8.2 or 8.3, or by using one of the tailormade conversion scripts prepared for INSROP GIS. If you store and document the data sets you obtain or create according to INSROP GIS requirements, these data sets become part of your INSROP GIS database and are then available for use as any other data set in the INSROP GIS database. The INSROP GIS requirements for storage and documentation are described in the following sections.

8.5.1 Storage locations of INSROP GIS data

The INSROP GIS database is divided into a set of information topics, and all data

sets within each information topic are organised in sub-directories below the proper database information topic sub-directory (see Section 6.1). To include your own data into your INSROP GIS database you must decide into which information topics your data sets belong. If necessary you may establish a new information topic.

After location the proper information topic database directory, you must decide whether your data set is a new data set or if it may be included into another data set. The latter case is mainly appropriate if your data represents an update, extension or replacement to an existing data set. In most cases we foresee that your own data represents a new data set, in which case you must create a new dataset directory for your data.

Note: Data in ARC/INFO format includes an INFO directory in addition to the actual data directories. This INFO directory comprises information valid to the ARC/INFO workspace the data set was created from, and it is required to enable ArcView to use these data. You can add ArcView shapefiles to a directory with data in ARC/INFO format, but you cannot combine ARC/INFO datasets from different workspaces (with different INFO directories) into one directory. If you nevertheless want to combine such ARC/INFO data sets into one directory, you must either convert the data sets to ArcView shapefiles (e.g. by using the ARCSHAPE command), thereby skipping the need for the INFO directory, or you must organize the data sets into one ARC/INFO workspace.

8.5.2 Documentation of INSROP GIS data

The documentation of a new INSROP GIS data set involves the following steps:

1. Update the database topics definition file
2. Update the proper data set definition file
3. Create a data set member definition file
4. Create data set member attribute definition files
5. Create data set member attribute value look-up-files
6. Create DIF documentation

These steps are further described in the following:

Update the database topics definition file

If your data set is to be included in one of the existing database topics this step is not necessary. However, if you must create a new database information topic, you must add a new line to the database topics definition file (datasets.def) and fill in the proper contents of this file (see Section 6.4.1).

Update the proper data set definition file

If your data set represents a new data set, you must update the proper data set definition file. If you have created a new database topic, you may copy and modify an existing data set definition file. To define the new data set you must add a new line to the data set definition file and fill in the proper contents (see Section 6.4.2).

If your new data represents a modification to an existing data set you may skip this step.

Create a data set member definition file

Create a data set member definition file (see Section 6.4.3) for the data set. If a data set consists of several files with the same attributes, i.e. the files comprise the same information attributes but differ in geographic or temporal coverage, the combined set of files represents one data set member. As the files may have completely different filenames, a special file name, *Alias.fld*, should be used for the attribute definition file, and the Theme-ID should be set to *All*. However, in such cases this step may be skipped.

Create data set member attribute definition files

An attribute definition file must be created for each different data set member (see Section 6.4.4).

Create data set member attribute value look-up-files

An attribute value look-up-file must be created for each attribute to be displayed with a default legend (see Section 6.4.5).

Create DIF documentation

The DIF-documentation comprises the metadata part of the INSROP GIS data sets. This documentation is not necessary for using the data set in INSROP GIS, but is useful for other users who may want to utilize your data. It is recommended to complete the metadata information shortly after creating a data set, while the information to be included in the DIF documentation is "fresh in memory". For more information on the contents of the DIF documentation, see the INSROP GIS hypertext documentation and Brude *et al* (1996).

9 VIEWS: EDIT, DISPLAY, QUERY AND ANALYZE SPATIAL DATA

9.1 Creating or opening a View

A new view is created from the Project window by double-clicking the View icon, while an existing view is opened by double-clicking a view name in the list of view documents in the project. In addition, these options are also available from the **File** menu when the Project window is the active window.

9.2 INSROP GIS view options

The view options described in this section are included in the View menu in the Views menu bar (See Figure 8.1). In addition to the menu items for adding or creating INSROP themes (see Chapter 8), the View menu comprises the following menu items affecting the display of spatial data:

- Use INSROP Projection
- Add Graticule
- Remove Graticule

9.2.1 Use INSROP Projection

When the source data are stored in geographic coordinates (latitude and longitude), ArcView may display the data in a range of projections. For most INSROP purposes we recommend the Lambert Equal-Area Azimuthal projection. Hence we have chosen this projection, with 120 degrees East as Central Meridian and 71 degrees North as Reference Latitude, as the default INSROP projection. The default distance units are set to kilometers.

When you are projecting a view, make sure all data themes in the view are stored in geographic coordinates. As reminder, when choosing the **Use INSROP Projection** menu item, the user is asked whether this is true.

9.2.2 Creating a Latitude-Longitude graticule in a view

A latitude-longitude graticule can be added to a view by choosing the **View-Add Graticule** menu item. The user is asked to specify graticule resolution in degrees and minutes. The graticule is displayed as 0.3 mm wide black lines. If you want to change the graticule, just specify a new one. The old one is automatically removed. If you want to just remove an existing graticule, choose the **View-Remove Graticule** menu item.

Note: A graticule can at present only be added to a projected view. This again requires that the spatial data are stored as decimal degrees.

9.3 INSROP GIS theme display options

The theme display options described in this section are included in the Theme menu

in the Views menu bar (See Figure 9.1). In addition to the following menu items affecting the display of spatial data:

- Display attribute
- Display 'linked' attribute
- Display Vector arrows
- Remove Vector arrows

the menu also includes menu items for updating themes. The updates include updating feature sizes, modifying selected features in a theme, or adding new features to an existing theme. The following INSROP GIS menu items affect updating of themes:

- Project Features
- Update Feature Sizes
- Modify Selected Features
- Add New Features

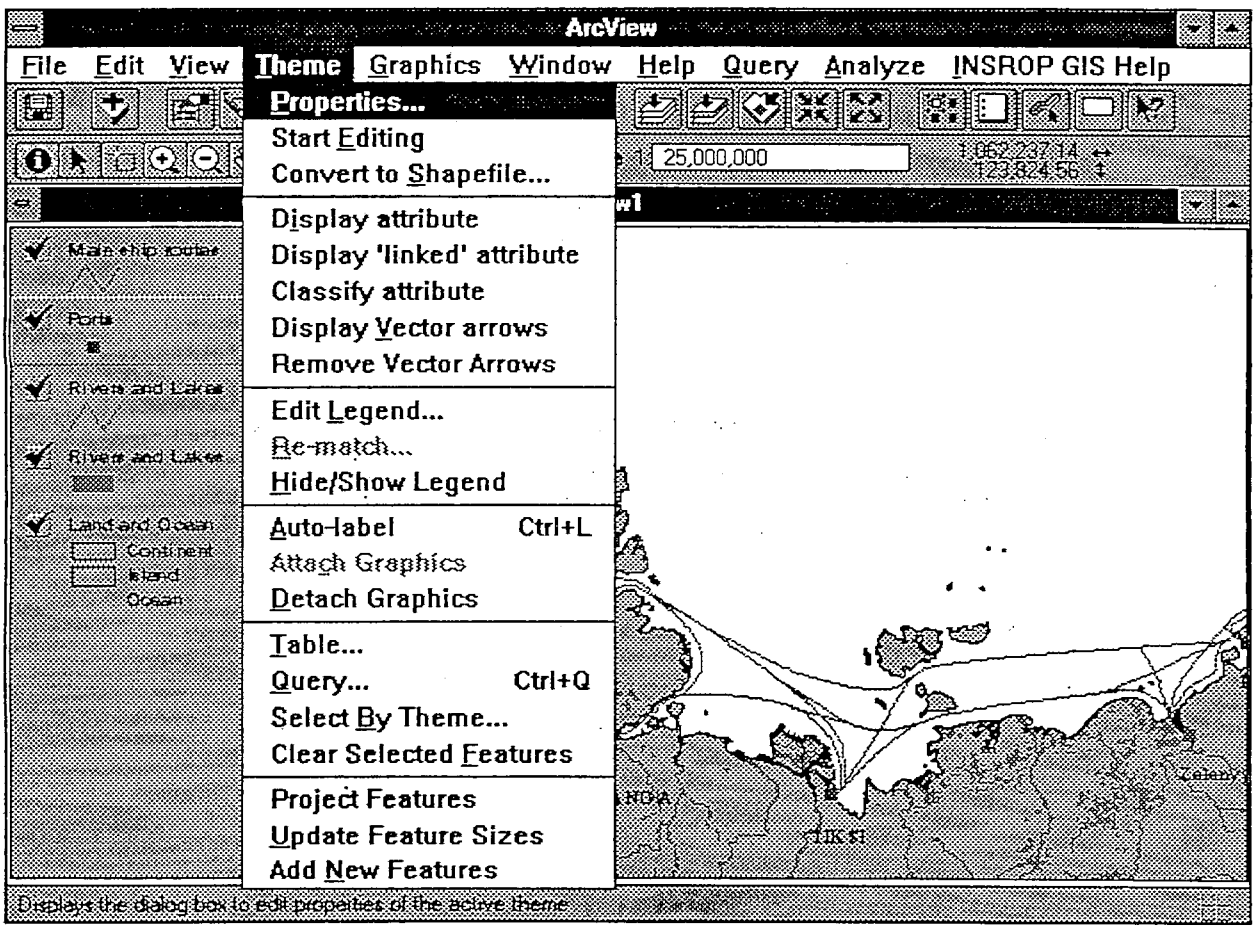


Figure 9.1 The Theme menu in the View window interface

9.3.1 Display attribute

The process of displaying an attribute with a legend as defined in the attribute value look-up-table (see Section 6.4.5) involves the following steps:

1. Get colorname definitions

2. Get a list of active themes
3. Get a list of all attributes in the active theme
4. Get the proper data set member definition file
5. Get the definitions of the active theme
6. Get the data set member attribute definition file of the active theme
7. Select the attribute to display
8. Get the proper attribute value look-up-file
9. Update the legend and theme display based on the contents of the attribute value look-up-table

These steps are described in more detail in the following:

Get colorname definitions

The colornames used by this INSROP GIS function are defined in an AVENUE script called INSROP.RGBColorNameDictionary (see Appendix A). These colorname definitions are similar to the UNIX colorname definitions. This step requires no user input.

Get a list of active themes

If there are active themes, they are put in a list and processed one at a time. If no themes are active the user is asked to make a theme active and try again. If there is more than one active theme, the following sequence of steps ise repeated for each active theme. This step requires no user input.

Get a list of all attributes in the active theme

This step is useful when there is more than one active theme. If subsequent active themes have the same attributes, Steps 4-8 are only carried out for the first of these themes, and the information obtained then is used for all the remaining themes in the sequence. This step requires no user input.

Get the proper data set member definition file

This step searches for a data set member definition file in the directory where the active theme is stored. If more than one is found, the user is prompted to select the proper data set member definition file. If none is found, INSROP GIS assumes that the data set member attribute definition file is called *alias.fld* and Step 5 is skipped.

Get the definitions of the active theme

A theme is a data set member, and the data set member definition file from Step 4 may comprise the theme definitions as described in Section 6.4.3. Only the *Theme-ID* is a mandatory definition parameter. If no *Theme-ID* satisfying the active theme is found, or the *Aliasfile* column value is empty, INSROP GIS assumes that the data set member attribute definition file is called *alias.fld.*

Get the data set member attribute definition file of the active theme

If not already existing in the project, this file is added to the project as a table

document. To determine whether the file already exists, INSROP GIS checks whether the table name and comments (see Table properties) are identical. The table comments are taken from the comment line (Line 2) in the data set member attribute definition file when the table is added to the project. If no suitable data set member attribute definition file is found, the user is prompted for one.

Select the attribute to display

The data set member attribute definition file should comprise definition parameters for the attributes the user is allowed to display with this INSROP GIS function. As the function is intended for unique classification, there should only be a limited set of possible attribute values. In ArcView 2.1 the default maximum number of different attribute values to allow unique legend classification is 65. Although INSROP GIS enables increasing this limit it is not recommended for practical purposes. If the number of different attribute values is large, we recommend using *Quantile* or *Equal interval* classification, or request a new INSROP GIS query (if none already exists).

The user is prompted to select one attribute to display.

Note: To use a specified default symbol/color (not depending on a particular attribute value), use "Default" in the attribute *Fieldname* column in the data set member attribute definition file and in the *Value* column in the attribute value look-up-file.

Get the proper attribute value look-up-file

The name of this file was defined in the data set member attribute definition table. If not already existing in the project, this file is added to the project as a table document. To determine whether the file already exists, INSROP GIS checks whether the table name and comments (see Table properties) are identical. The table comments are taken from the comment line (Line 2) in the attribute value look-up-file when the table is added to the project. If no suitable attribute value look-up-file is found, the user is prompted for one.

Update the legend and theme display based on the contents of the attribute value look-up-table

Based on the actual attribute values in the active theme and the attribute value look-up-table, the display of features in the theme and the associated legend is updated.

If there is more than one active theme, the process will now start over at Step 3 until all active themes are processed, or the user has *Cancelled* the process.

9.3.2 Display 'linked' attribute

The main purpose of this option is to enable display of one-to-many related attributes. Such attributes can be linked to a theme attribute table, but the theme legend cannot display symbols/colors based on an attribute in a linked table. One-to-one related attributes can be joined to the active theme and displayed directly, but one-to-many related attributes must be reduced to a one-to-one relation first. The process of displaying an attribute in an associated table therefore involves the following steps:

1. Choose the table with the ('linked') attribute to display

2. Query the table to enable one-to-one join of selected records
3. Export the selected records to a temporary table
4. Join the temporary table to the attribute table of the active theme
5. Select the attribute to display
6. Display the theme with a legend based on the 'linked' attribute

These steps are described in more detail in the following:

Choose the table with the ('linked') attribute to display

The first step creates a list of table documents that are not feature attribute table of themes. The user is then prompted to select the table comprising the attribute he wants to display.

Query the table to enable one-to-one join of selected records

This step opens the Query window and enables the user to specify a query to reduce a one-to-many related table to a one-to-one relation on selected records. E.g. if a table comprises 12 monthly values for each location, a query selection one particular month will enable display of this monthly value at each location.

Export the selected records to a temporary table

The user is first prompted to create a temporary table. Existing tables can be overwritten if they are not active. Then the selected records are exported to this temporary table.

Join the temporary table to the attribute table of the active theme

The user is prompted to select the attribute in the temporary (associated) table and the attribute table of the active theme that are common and therefore enables a join of the two tables.

Select the attribute to display

The user is prompted to select the associated ('linked') attribute to display.

Display the theme with a legend based on the 'linked' attribute

Now that the attribute is joined to the theme attribute table, the legend can be displayed based on the attribute values of the originally one-to-many related attribute. First a unique classification is attempted, but if there are too many unique attribute values, the user is prompted to select *Quantile* or *Equal interval* classification and number of intervals.

9.3.3 Display vector arrows

Vector arrows can be displayed for point themes where the attribute table comprises attributes defining either length and direction of a vector, or the vector components. For projected views (requires source data in decimal degrees), the direction is the compass direction

from each point, and the vector components are along the north and east direction from each point. If the source data are projected, the vector direction and components must be in the X-Y coordinate system of the projected source data. It is required that both the *Map units* and the *Distance units* are defined in the *View properties*, as the map units define the actual length of a vector, while the display units define the corresponding display length of the vector.

The display of vector arrows involves the following steps

1. Make the requested theme(s) active
2. Specify line width of the arrow
3. Select type of vector definition
4. Select the proper attribute fields
5. Specify arrow scale
6. Create arrows

These steps are described in more detail in the following:

Make the requested theme(s) active

The theme(s) for which to display vector arrows must be active, and they must all be point themes comprising attributes defining the vectors. Also, as the arrow colors are taken from the legend of each theme (see Step 6), each legend should be classified in the way the user wants the arrows to appear.

Specify line width of the arrow

Only one type of arrow is currently available, and this is a solid-line arrow, where the arrowhead is drawn at 45 degree angles and extending 0.25 of the size of the vector. The user is prompted to specify the line width, in millimeters, of the arrow.

Note: The user's choice of arrow line width will be used for all active themes. If the user wants to vary the line width for different themes, only the the themes with the same line width should be made active and then processed at a time.

Select type of vector definition

For each active theme, the user is prompted to select how the vectors are defined. The choices are:

- Length/Speed and direction
- X/Y (E/N) components

Select the proper attribute fields

Depending on the choice made in the previous step, the user is presented with a list of attribute fields wihtin the current active theme and prompted to select the required vector definition attribute fields (one at a time).

Specify arrow scale

To establish the display length of the arrows, the display scale of the arrows must be known. For the current active theme, the user is prompted to specify which vector length 1 centimeter in the display should represent. If the user specifies 10, this means that a vector with length 4 (of units used in the theme attribute table) will be displayed as a 4 mm long arrow.

Create arrows

To create an arrow, the location, length, direction, color and width of the arrow must be specified. The location is determined from the coordinates of the selected points in each active theme. If no points in a theme are selected, all points in this theme are used. The display length is taken from the vector attribute(s) and the user-specified arrow scale. The direction is determined from the vector attribute(s), and for project views the direction is in the X-Y coordinate system specified by the local North-East coordinate system in each point. The width of the arrow was specified by the user in the previous step.

The color of an arrow is taken from the legend of the theme. This may seem as an unnecessary approach, as it may appear easier to automatically update the legend based on the vector length. However, as different users may want to classify the arrows differently, we have not found it beneficial to use this latter approach. Taking the colors from the legend classification also means that the point theme can be classified by any attribute, not just the arrow length. This may be quite useful for some purposes. For instance, if a point theme comprise both ice concentration and ice drift vectors, the vectors may be displayed based on ice drift speed and with colors based on classified ice concentration values. Also, vector arrows can be used to show migration routes and with colors showing number of migrating species.

To display vector arrows it is not necessary that the active theme(s) are displayed in the view (switched on). If the themes are displayed, both the legend symbols and the vector arrows will be displayed. To only display arrows, switch the theme(s) off. The vector arrows will anyway be displayed in the viewframe in a layout, but to include the theme legend in the legendframe in a layout, the theme must be displayed in the view.

9.3.4 Remove vector arrows

To remove all vector arrows displayed from one or several point themes, make the theme(s) active and choose the **Remove Vector Arrows** menu item in the *Theme* menu.

9.3.5 Project Features

As a default, ArcView (similar to any other GIS software) displays line and polygon features as a set of straight lines connecting the point coordinates defining the feature. When the source data are stored as decimal degrees, ArcView can project the points, but the lines are still straight. This is not always acceptable, since especially administrative zones are often defined by latitudinal or longitudinal lines connecting point coordinates defining the zone vertices. In a map using the Mercator projection (common for sea charts) latitudinal and longitudinal lines form a cartesian coordinate system, and the default display using straight lines is correct. However, if one chooses to use more area- and/or distance-correct projection, the straight lines will deviate from the corresponding projected lines. To solve this problem, we have included a utility to project entire features, not just the vertice coordinates.

The utility will project all line segments of the selected features in the active themes. If no features are selected in an active theme, the user is prompted to verify whether all features in the theme should be projected. New shapefiles including all features (selected/ projected and unselected/unprojected) from the active themes are created. The user is prompted for the name of each new shapefile. The original themes are deleted from the view and the new ones added. The new projected themes use the similar legend as the corresponding source theme, and the name is the same as the name of the source theme + the string 'projected'.

Note 1: This utility will only work if the View has a projection.

Note 2: Do not delete the shapefile with the unprojected features. The projected features are only valid for the projection they were created for. The resolution of the projected features is also scale-dependent. Hence, if you zoom in to a larger scale or you change the projection, you should start over again from the original feature theme. Remember also that you cannot reuse a shapefile that is included in the active project. Hence, to reuse one or several shapefiles with projected features, you must delete the theme(s) from the view and save the project. You must also add the original themes again and set up the proper legends. We recommend that you prepare all unprojected feature themes in a source project file, and that you import this project file into the project where you need to project the features.

9.3.6 Update Feature Sizes

Data sets created by ARC/INFO include feature size attributes calculated when the topology was established. Line themes include the length of each line feature (ARC/INFO arc), while polygon features include the area and perimeter of each polygon. However, as these measures are based on source data coordinates (ArcView Map units), the size of features defined by decimal degree coordinates have no meaning. For data sets created as new shapefiles, e.g. by importing spatial ASCII data, no feature size attributes are calculated and included. Hence, for classification of features based on size, or to make calculation involving a feature size, e.g. areal density, attributes showing real feature sizes must be included.

To achieve this, make the theme(s) to update active and choose the **Update Feature Sizes** menu item in the *Theme* menu. This utility will check whether the active theme(s) has proper feature size attribute fields, and if not not it will create and add such attribute fields. Then the size of each feature will be calculated and entered into the proper attribute field records. For Line themes the feature size attribute field is named "Length", while for Polygon themes the feature size attribute fields are named "Area" and "Perimeter". The feature sizes are in *Map units* and are specified with 3 decimals.

9.3.7 Modify Selected Features

This utility is not available in this version of INSROP GIS. The purpose of this utility is to enable modifying the location/shape of selected features by deleting, moving or adding points.

A workaround for shapefiles created from spatial ASCII files is to update the ASCII

file and recreate the shapefile theme. To recreate an existing shapefile, the shapefile theme must be deleted from the project and the project saved, before the *Create INSROP Theme* utility is used.

9.3.8 Add New Features

This utility is not available in this version of INSROP GIS. The purpose of this utility is to add new features to an existing feature theme.

A workaround for shapefiles created from spatial ASCII files is to update the ASCII file and recreate the shapefile theme. To recreate an existing shapefile, the shapefile theme must be deleted from the project and the project saved, before the *Create INSROP Theme* utility is used.

9.4 INSROP GIS queries

This section describes the general INSROP GIS query process, while the actual queries are documented further in Brude *et al.* (1995). It is important to remember that the INSROP GIS queries represent a set of queries tailored to special data sets or data types. Basic queries of the type "What is where?" or "Where is what?" are meant to be answered by using the inherent ArcView thematic or spatial selection, classification and/or display capabilities.

INSROP GIS queries are initiated from the Query menu in the Views menu bar. However, most of these queries are for special purposes and are therefore included in project files in the INSROP GIS. The reason for this is that many INSROP GIS queries are customized for a special purpose and apply only to a limited set of data sets. If the AVENUE scripts carrying out these special queries were to be included in the customized INSROP GIS default.apr file, this would be unnecessary overhead to users not requiring these queries. Hence, we have chosen to organize the INSROP GIS queries thematically and include only query scripts of a more general character in the basic INSROP GIS software.

The main steps in the INSROP GIS query process are as follows:

1. Create a view with the necessary information included
2. Make the themes you want to query active
3. Choose an information topic
4. Select an information sub-topic
5. Select the type of data you are querying
6. Select a query
7. Run the query to provide the requested information

A more detailed description of these steps is included in the following:

Create a view with the necessary information included

In the current version of INSROP GIS it is the user's responsibility to ensure that the necessary information to run a query and provide the requested results is present in the project. The term 'necessary information' includes both the source data (ArcView themes) and the AVENUE scripts required to run the query.

Note: For some purposes (and for 'untrained' users) it would be beneficial if the required information were found and included when running the query. However, for other queries, especially when running a sequence of queries on the same data set(s), it is faster to assume that the source data are already present. As a compromise between these two choices, the INSROP GIS project library includes prepared ArcView projects comprising the necessary data and AVENUE query scripts to run the customized queries. These projects may be copied, and provided the necessary theme attributes are present, the information in the themes may be modified, e.g. by deleting or updating themes or features.

Make the themes you want to query active

INSROP GIS queries run on active themes. Some require that only one theme is active, while others work on all active themes. If the query topics in the Query menu are "greyed out", this means that no themes are active.

Choose an information topic

The Query menu lists a set of information topics. At present the following topics are implemented:

- Navigation
- Ice conditions
- Meteorology
- Environmental Atlas

When INSROP GIS queries within other topics are implemented, this list will be extended. Select one of the topics to initiate the query sequence.

Select an information sub-topic

Within each of the overall information topics, there may be a range of information sub-topics. The user is presented with a list of information sub-topics and may select one of them.

Select a data set

Although INSROP GIS at this stage knows which data theme(s) are active and the selected information topic, there is at present no capability of INSROP GIS to interpret which information the active theme(s) represent. If there is only one data set comprising the source data within the selected information sub-topic, or all the sub-topic source data sets are structured similarly, this step is skipped. However, for some information sub-topics, there exist different data sets with partly the same information. In such cases, INSROP GIS needs to know which data set to query and the user is presented with a list of available data sets within this information sub-topic and must select which data set the active theme(s) are part of.

Select a query

While the three previous steps (select information topic, sub-topic and data set) were

included in the basic INSROP GIS software (default.apr), this step may assume that the required AVENUE query scripts are included in the active project. However, INSROP GIS queries of a more generic character are included in the basic INSROP GIS software.

Run the query to provide the requested information

Now INSROP GIS will initiate the selected query. For some queries additional input is required, in which case the user is prompted for this information. The results of the queries may be displayed in a range of ways, e.g. with a classified legend, as a report in a message window, or as a Chart. This depends on the selected query. See Brude *et al.* (1995) for more details.

9.5 INSROP GIS analyses

The analyses described in this section are included in the **Analyze** menu in the Views menu bar.

9.5.1 Quality control

The quality control works on the contents of the attribute table of one or more themes in a view. The purpose is to get statistics on the contents of the attribute fields, as well as to check the range of the attribute values.The following steps are involved:

1. Select the theme(s) to control
2. Specify a table to store the outcome of the quality control
3. Specify the attribute table field(s) to be controlled
4. Specify legal value range of each attribute table field
5. Run the quality control
6. Report results

These steps are described in more detail in the following:

Select the theme(s) to control

Any active themes in the view will be quality controlled. If none are active, the user is asked to select one of the themes in the view.

Specify a table to store the outcome of the quality control

The user is prompted for a name of the file where the quality control results will be stored. The file will be created as a dBase table with the following fields:

- Field — The name of a controlled field
- Type — The type of the field
- #Records — Number of records in the table
- MinLim — The specified minimum limit of the legal value range
- MaxLim — The specified maximum limit of the legal value range
- Min. Value — The minimum data value in the field
- Max. Value — The maximum data value in the field
- #values < MinLim — Number of records with data value less than the specified minimum limit of the legal value range

- #values > MaxLim — Number of records with data value greater than the specified maximum limit of the legal value range
- #Nodata — The specified Nodata value
- #Empty records — The number of empty (without data value) records in the field

Specify the attribute table field(s) to be controlled

The user may choose to control all fields in the table, or one particular field. In the latter case the user is asked to select which field to control.

Specify legal value range of each attribute table field

For numeric fields, the user may specify the lower and upper limits of a legal value range, as well as a Nodata (Unknown/Undetermined) value. By default these values are *nil*, in which case any data value will be accepted as a legal value.

Run the quality control

In this step the value in each record in each field is checked and the various control parameters updated. For non-numeric fields only the Field, Type and #Records fields will be filled with information.

Report results

In this step, the table created in Step 2 is filled with the outcome of Step 6 and opened in a Table window so the results may be inspected.

9.5.2 Create a buffer theme

The main purpose of this utility is to enable analyses of features appearing (or not) within a certain distance from other features. A typical use is to select features within one theme that are within a certain distance from the selected features of another theme. The features selected by a buffer criterion can then be analyzed further. Hence, a buffer analysis generally requires two main steps:

- Create a buffer theme
- Use the buffer theme for some purpose

There is also another reason for developing this utility. The ArcView's *Theme-Select by Theme* option in the View window works on source data coordinates. This doesn't matter for users wanting to select features of a theme based on whether they intersect or completely contains the selected features of another theme, but for users wanting to select features of a theme based on whether they are within a distance of the selected features of another theme there is a critical difference. The reason is that despite ArcView requests a distance in Distance units, the distance measure is converted to source data units. Then the features within the distance in source data units are selected. This is correct for projected source data (unprojected view), but when the source data are in decimal degrees this is incorrect as the real length represented by a decimal degree unit is not the same along the latitude and longitude axis. As a workaround to this problem, a user may create a buffer theme covering a user-specified distance around the selected features of the active themes, and then select select features of a theme based on whether they intersect or completely contains the selected features of the created buffer theme.

The steps involved in creating a buffer theme are as follows:

1. Specify the buffer distance
2. Get possible existing buffer themes
3. Specify name of buffer theme to create
4. Create the buffer theme attribute table
5. Create the buffer features for each active theme
6. Merge overlapping buffer features into non-overlapping features
7. Update the buffer theme legend

These steps are desribed in more detail in the following:

Specify the buffer distance

The user is prompted to specify a buffer distance in the *Distance units* of the *View*. As the buffer distance is a non-directional measure it must be a positive number.

Get possible existing buffer themes

A buffer theme can be deleted and recreated as any other ArcView theme. However, as the buffer theme in many cases just acts as a temporary theme used to select features in other themes, and it may also be useful to vary the buffer distance to see the effect, we have introduced a possibility to just replace an existing buffer theme without deleting the theme first. To ArcView a buffer theme is just like any other feature theme. Hence, for INSROP GIS to recognize a theme as a buffer theme, the theme name must contain the character string **buf**. Themes named e.g. Buffer01.shp, BufTheme, RouteBuf will all be recognized by INSROP GIS as potential buffer theme.

Specify name of buffer theme to create

If there is one or more themes in the view comprising the character string **buf** (see the previous step), INSROP GIS will prompt the user for whether an existing buffer theme should be replaced. If the **Yes**-button is clicked, the user is prompted to select the buffer theme to replace. If the **No**-button is clicked, or INSROP GIS does not identify any existing theme as a buffer theme, the user is prompted to specify a shapefile to create.

Create the buffer theme attribute table

The buffer features are given three attributes: **Buffer-ID**, **Theme**, and **Buffer Distance**. The Buffer-ID includes an ID-number of a feature, the Theme comprises the name of the theme which selected feature this buffer feature was created from, and the Buffer Distance comprises the specified buffer distance used to create the buffer.

Create the buffer features for each active theme

This step processes one active theme at a time, and the creation of a buffer polygon differs depending on whether the active theme comprises point, line or polygon features. The common part is that for each of the selected features in the active theme, one buffer polygon is created. If there are no selected features in the active theme, the user is prompted for whether all features are to be buffered. The reason for having the user to confirm this is to avoid time-consuming buffering of large themes due to the user forgetting to select the fea-

tures of interest first.

For *point* features, a circular polygon centered on each point and with radius equal to the buffer distance is created. For points closer than two times the buffer distance, the buffer polygons will overlap each other.

For *line* features, the buffer polygon should cover a zone on both sides of the line, and where all points defining the polygon are located one buffer distance from the nearest point on the line. For a straight line defined by two points this is easy, as the buffer polygon is a rectangle defined by the two points located one buffer distance on either side of the startpoint of the line, and along an axis normal to the direction of the line, and by two similar points on either side of the endpoint of the line. Also straight lines defined by more than two points are handled properly, but for non-straight lines the creation of a buffer polygon becomes more complicated. The method implemented in the present verion of INSROP GIS is to create buffer rectangles around each straight line segment and buffer circles centered on each breakpoint along the line. This creates a set of overlapping polygons which needs to be merged in order to get the final buffer polygon(s). The method works ok in most cases, but there are still problems for *some* buffer rectangles crossing the 180°E longitude.

For *polygon* features, each buffer polygon should cover a zone including the polygon itself and extending one buffer distance outside the polygon perimeter. The problem is complicated to solve in the present version of ArcView/Avenue, and is postponed.

Merge overlapping buffer features into non-overlapping features

As the final buffer theme should only comprise non-overlapping polygons, overlapping buffer polygons should be joined into one buffer polygon. This is not a trivial task using the present AVENUE functions (The default function for merging polygons creates holes for overlap zones). As no INSROP projects have specified an immediate need for this utility, we have chosen to postpone this utility until the next version of INSROP GIS. By then we will also know more about what new capabilities are included in the next upgrade of ArcView (ArcView 3.0). However, we have implemented a polygon merging utility that merges polygons that only have one overlap zone (=no holes are created). This process is somewhat time-consuming.

Update the buffer theme legend

As default, the buffer theme features are displayed in yellow, and with the buffer distance as legend label. If necessary, you may use the Legend Editor to change this display.

9.6 INSROP GIS View Tools

INSROP GIS includes one additional tool in the *Tool Bar* of the *View* window. This tool is described in the following section.

9.6.1 Onscreen digitizing

In some cases, e.g. to create features for spatial selection of features in other themes, it is useful to be able to create simple features by digitizing on the screen. Therefore we have added a tool which enables creation of point, line or polygon features by clicking the mouse. To use this tool the active theme must be editable. After the tool is activated, the

location of the cursor when the mouse is clicked will be stored. For points themes, the points will be stored directly, while for line and polygon themes the points will be stored in memory until the Control-button is pressed while the mouse is clicked. Then a new feature is created from the points in memory and appended to the shapefile of the active theme. To remove the last point(s) from memory, hold the Shift-button down while the mouse is clicked.

When creating a new feature the user is prompted for attribute values. If the theme was created without any attributes, an attribute field named "ID#" is created and updated with the record number as attribute value.

The tool is activated like any other tool; by clicking the tool button. It is deactivated by activating another tool.

9.6.2 Getting the source data location of a point in a view

The *Map unit* coordinates of the location of the mouse cursor are constantly updated and displayed in the rightmost part of the *Tool Bar*. For unprojected views the map unit coordinates and the source data coordinates are identical. However, for projected views the tool bar displays the cursor coordinates in projected map units, while the source data are stored in decimal degrees. In many cases it is interesting to see the decimal degree coordinates of point in the view. For this purpose we have added a YX-tool in the Tool Bar of the View. When this tool is active, the location of the cursor when a mouse button is clicked is displayed in source data coordinates in the leftmost part of the Statusbar. For project views this means that the coordinates of the point are displayed as decimal degrees. For unprojected views, the advantage of this tool is that the coordinates of the point are only updated when a mouse button is clicked and remains unchanged in the Statusbar until a mouse button is clicked again, or another message appears in the Statusbar.

The tool is activated like any other tool; by clicking the tool button. It is deactivated by activating another tool.

9.6.3 Customized Hotlink tools

Some INSROP GIS projects include customized scripts utilizing the hotlink capability. These scripts are not included in the basic INSROP GIS software. The utilities are executed by activating the relevant theme and the hotlink tool and clicking on a feature in the theme. In most cases this will initiate a Chart display of a time series for the feature clicked at. For example, this utility may be used to display minimum, mean and maximum monthly air temperatures for the measurement station clicked at, of monthly time series of river flowrates at the measurement station clicked at.

The typical data model when such customized hotlink tools are useful, are when a feature theme comprises a set of spatial features and their attributes, incl.a feature ID, and the actual time series data are stored in a database file which also comprises the feature ID. The feature theme will then be stored with only one record per feature, while the database file will have a set of time series records for each feature ID.

If you instead of displaying time series for one location, want to show the values at a give point in time for all locations, you must use the **Theme - Display 'linked' attribute** menu item (See Section 9.3.2).

10 TABLES: ATTRIBUTE DATA

10.1 Creating or opening a Table

A table can be created in two ways; From the Project window, or from the View window. From the Project window a new table is created by double-clicking the Table icon or by choosing the **Project-Add Table** menu item. An existing table is opened by double-clicking a table name in the list of table documents in the project. From the view the attribute table of a theme is opened by making the theme active and choosing the **Open table** button in the views button bar. In addition, some View functions will add tables to a project, i.e. definition tables (ref. Sections 6.4.4 and 6.4.5).

10.2 INSROP GIS analyses

The analyses described in this section are included in the **Analyze** menu in the Tables menu bar. *There are currently no INSROP GIS Table analyses implemented.*

10.2.1 Quality control

The quality control of a table works similarly to the quality control of a theme in a view (see Section 9.5.1).

11 CHARTS: DISPLAY AND QUERY ATTRIBUTE DATA

11.1 Creating or opening a Chart

A Chart is primarily created from the Table window, and works pretty much like charts created from a spreadsheet, but an existing chart may be opened from the Project window by double-clicking a chart name in the list of chart documents in the project. INSROP GIS has no extended functionality in the Chart window, but in some cases, charts are created directly from a View (see Chapter 9).

12 LAYOUTS: PREPARING AND PRINTING PLOTS

12.1 Creating or opening a Layout

A layout can be created in two ways; from the Project window, or from a View. From the Project window a new layout is created by double-clicking the Layout icon. An existing layout is opened by double-clicking a layout name in the list of layout documents in the project. From the view a layout is created by choosing the **View - Layout...** menu item.

A layout can be created with a pre-defined layout template. When creating a layout from a view, the user is automatically requested to select a layout template. In the Layout window, a layout template can be selected by the **Layout-Use Template ...** menu item.

12.2 Components of a Layout

A layout may comprise ArcView elements (Views, Tables, Charts) and ancillary objects such as legends and scale bars i special *frames*. To get data into these frames, they must be linked to an ArcView element and will then depend on properties in the linked element (e.g. the scale of a view). In addition, created graphics (points, lines , polygons, polylines, rectangles and circles) and text can be added.

As the frames in a layout are linked to ArcView elements, and do not comprise any data, any updates in the linked elements are reflected in the Layout.

Note: The scale of a view is determined from scale of the View window when the window's *Maximize button* (▲) is visible (and not the *Restore* button) in the upper right corner of the window frame. Run the MS Windows tutorial if you are uncertain about the meaning of these two buttons.

Tip: To maximize a View window, press the Maximize button in the upper right corner of the window frame. This changes the Maximize button to the Restore button. By closing and reopening the window (from the project window), the Restore button will be changed to the Maximize button without resizing the window.

12.3 Editing the legend of a viewframe

When using one of the layout templates, each viewframe is associated with a legendframe. By default the legend in a legendframe comprises the active themes in the view linked to the viewframe. In some cases it might be preferable to delete some themes (or other elements) from the legend. To enable editing of elements in the legend, the legendframe must be simplified, which means that the link to the view is cut, and the legend is converted to a collection of simple graphical elements. These graphical elements can now be selected and edited similarly to any other basic graphical element in a layout.

Note: Once the legendframe has been simplified, the link to the view no longer exists, and any updates in the view will not be reflected in the legend. Hence, it is recommended that simplification of a legend is the final editing step before the map is printed. If you ever need to recreate the legend after simplification, delete all parts of the simplified legend and create a new legendframe linked to the view in the viewframe.

12.4 Creating a Latitude-Longitude Graticule

A latitude-longitude graticule can be added to a view frame by choosing the **Layout-Add Graticule** menu item. The user is asked to specify graticule resolution in degrees and minutes. The graticule is displayed as 0.3 mm wide black lines. If you want to change the graticule, just specify a new one. The old one is automatically removed. Also any existing graticule in the View linked to the view frame is also removed. If you want to just remove an existing graticule, choose the **Layout-Remove Graticule** menu item.

Note 1: A graticule can at present only be added to a projected view. This again requires that the spatial data are stored as decimal degrees.

Note 2: When adding a graticule to a view frame without an existing graticule, the size of the view frame will be reduced to allow printing graticule labels (degrees and minutes) without exceeding the original view frame area. When removing an existing graticule the original size of the view frame will be restored.

13 HOW TO GET HELP

13.1 On-line help

13.1.1 - ArcView help

All basic ArcView on-line help is available from the **Help** menu in the INSROP GIS interface. The ArcView context-sensitive help is also maintained.

13.1.2 - INSROP GIS help

On-line INSROP GIS Help is available from the **INSROP GIS Help** menu in the INSROP GIS interface. There is no context-sensitive help for special INSROP GIS interface components.

13.2 System support

13.2.1 - ArcView support

To get ArcView support you should contact your national vendor, or ESRI directly. See ArcView's help system for more information.

Note: ArcView technical support is generally fee-based, unless you have a support agreement. Anyway, be sure that the problem is ArcView-related and not INSROP GIS-related. If in doubt, you may try to call or e-mail the INSROP GIS development team first.

13.2.2 - INSROP GIS support

In principle, INSROP GIS is provided 'as is', and without any support agreement. However, in the interest of creating a useful product we will try and solve problems that arise during the use of INSROP GIS. Our possibility to help is nevertheless limited to the financial and temporal frameworks of the INSROP GIS development projects. Anyway, to increase our possibilities for solving problems, we will appreciate receiving an INSROP GIS Bug Report, as described in the INSROP GIS on-line documentation.

13.2.3 - INSROP GIS enhancements

The INSROP GIS development team welcomes any suggestions for enhancing INSROP GIS. We will try to include suggested enhancements in future versions of INSROP GIS. Send us an e-mail or fax describing what you want to achieve and what data you are working with, or whether you want to change some existing features in INSROP GIS. Enhancements aimed at improving the outcome of INSROP will be given high priority when planning future INSROP GIS versions.

14 INSROP World Wide Web Service

As part of Projects I.3.1/II.3.2 an INSROP World Wide Web (INSROP WWW) service has been established. The main INSROP WWW-server is located at the Norwegian Polar Institute (NP) in Norway and can be accessed at:

- URL: **http:/www.npolar.no/insrop/**

The establishment of the INSROP WWW used parts of the hypertext documentation prepared for INSROP GIS as a starting point. The INSROP WWW includes general information about INSROP, how INSROP is organised, which institutions are involved (including addresses), information about the INSROP Sub-programme and projects, and information about INSROP GIS. In addition a few sample images prepared by the INSROP GIS projects have been included as well as some news-related sections. A list of INSROP publications is also included. All INSROP Newsletters and Working Papers are listed, and the articles printed in each Newsletter are available through this publication list. The list of published Working Papers are linked to the Working Paper summaries printed in the Newsletters. Hence, this service provides Web-users with subject-related information about INSROP and from INSROP projects, and will hence e.g. improve the basis for decisions on whether to order specific Working Papers.

The creation of an INSROP WWW represents a change in priorities for Task 1 of the 1995-work of Project I.3.1. However, considering the variety of digital and analogue formats or the source material for printed Working papers and the rapid growth of WWW availibility and use, we consider INSROP WWW to be a good and realistic realization of the idea behind Project I.3.1 Task 1.

The next step in the development of the INSROP WWW is to include information on INSROP GIS data sets. This will be based on the INSROP GIS data set documentation. The INSROP software and data set distribution policy is not yet finalised, but the plan is to include public domain results from INSROP (including software and data) in the INSROP WWW. Similarly to the INSROP GIS concept, not all data sets may be located at the WWW-server at NP. Instead, the INSROP WWW at NP may include links to other WWW-servers at INSROP institutions who are documenting and maintaining specific INSROP data sets and/or other information..

INSROP WWW is currently maintained by Odd Willy Brude (co-author of this report), NP, and he may also be reached by e-mail **oddbr@npolar.no**. If you want to contribute to the development of the INSROP WWW, you should contact him and agree on how to prepare the information you want to include.

If possible, the information to be included should be prepared in HTML-format. A recommended way of preparing your own information in HTML format is to access the INSROP WWW, find a suitable page, and save it on your local computer in HTML-format. Then you may replace the original text with your own text and save it with a new filename, before you contact Odd Willy Brude to agree on how to tranfer the new file(s) to NP.

15 EXAMPLES

15.1 ASCII import files

The following sections show examples of ASCII import files. The header line(s) are recommended both as documentation and to speed up the import process, but they are not necessary as the content of the header lines may be given interactively.

15.1.1 Point data

```
ID;N;E;Date,Time,Speed,Dir
1;68.578;33.052;1991-9-4;21:15;5;0
2;69.19;33.33;1991-9-5;16:30;9.7;20
```

15.1.2 Multipoint data

```
ID;Range
N;E
1;13
68.578;33.052
69.19;33.33
70.03;36.45
END
2;20
77.19;43.33
79.39;48.53
78.13;53.21
END
```

15.1.3 Line data

```
ID;Depth
N;E
1;10
68.578;33.052
69.19;33.33
70.03;36.45
END
2;20
77.19;43.33
79.39;48.53
78.13;53.21
END
```

15.1.4 Multiline data

```
ID;Depth
N;E
1;10
68.578;33.052
69.19;33.33
70.03;36.45
NEW
70.578;37.052
71.19;38.33
73.03;39.45
END
2;20
77.19;43.33
79.39;48.53
78.13;53.21
END
```

15.1.5 Polygon data

```
ID;Species
N;E
1;Ringed seal
68.578;33.052
69.19;33.33
70.03;36.45
END
2;Bearded seal
77.19;43.33
79.39;48.53
78.13;53.21
END
```

15.1.6 Multipolygon data

```
ID;Species
N;E
1;Ringed seal
68.578;33.052
69.19;33.33
70.03;36.45
NEW
70.578;37.052
71.19;38.33
73.03;39.45
END
2;Bearded seal
77.19;43.33
79.39;48.53
78.13;53.21
END
```

16 REFERENCES

Brude, O.W., Løvås, S.M., and C. Smith (1996): "INSROP GIS Data Sets - Implementation and Documentation", INSROP Discussion Paper, Projects I.3.1/II.3.2, Norwegian Polar Institute, Norway (In preparation)

Løvås, S.M, Smith, C. and K.A. Moe (1994): "INSROP Information System - Specification and Design: Overall System Documentation", INSROP Working Paper No. 4, Projects I.3.1/II.3.1, SINTEF NHL, Trondheim, Norway, ISBN 82-7613-079-8.

Glossary

ARC/INFO

ARC/INFO is a geographic information system (GIS) software package made by Environmental Systems Research Institute (ESRI), the makers of ArcView.

ARC/INFO is used worldwide by thousands of different organizations for handling, managing and analyzing geographic information. ARC/INFO is designed for users who require a complete set of tools for processing and manipulating spatial data, including digitizing, editing, coordinate management, network analysis, surface modelling and grid cell based modelling.

ARC/INFO format spatial data includes vector coverages and raster grids. Attributes describing geographic features are stored as tabular files in INFO format (The database system built into ARC/INFO), dBase format, or other external database systems.

ArcView users can access almost any spatial data stored in ARC/INFO format for desktop mapping and analysis.

Appendix A: INSROP GIS colornames

Avenue-script: INSROP RGBColorNameDictionary

```
' INSROP.RGBColorNameDictionary

RGB_Dictionary = Dictionary.Make(738)
RGB_Dictionary.Add( "alice blue", { 240, 248, 255 } )
RGB_Dictionary.Add( "aliceblue", { 240, 248, 255 } )
RGB_Dictionary.Add( "antique white", { 250, 235, 215 } )
RGB_Dictionary.Add( "antiquewhite", { 250, 235, 215 } )
RGB_Dictionary.Add( "antiquewhite1", { 255, 239, 219 } )
RGB_Dictionary.Add( "antiquewhite2", { 238, 223, 204 } )
RGB_Dictionary.Add( "antiquewhite3", { 205, 192, 176 } )
RGB_Dictionary.Add( "antiquewhite4", { 139, 131, 120 } )
RGB_Dictionary.Add( "aquamarine", { 127, 255, 212 } )
RGB_Dictionary.Add( "aquamarine1", { 127, 255, 212 } )
RGB_Dictionary.Add( "aquamarine2", { 118, 238, 198 } )
RGB_Dictionary.Add( "aquamarine3", { 102, 205, 170 } )
RGB_Dictionary.Add( "aquamarine4", { 69, 139, 116 } )
RGB_Dictionary.Add( "azure", { 240, 255, 255 } )
RGB_Dictionary.Add( "azure1", { 240, 255, 255 } )
RGB_Dictionary.Add( "azure2", { 224, 238, 238 } )
RGB_Dictionary.Add( "azure3", { 193, 205, 205 } )
RGB_Dictionary.Add( "azure4", { 131, 139, 139 } )
RGB_Dictionary.Add( "beige", { 245, 245, 220 } )
RGB_Dictionary.Add( "bisque", { 255, 228, 196 } )
RGB_Dictionary.Add( "bisque1", { 255, 228, 196 } )
RGB_Dictionary.Add( "bisque2", { 238, 213, 183 } )
RGB_Dictionary.Add( "bisque3", { 205, 183, 158 } )
RGB_Dictionary.Add( "bisque4", { 139, 125, 107 } )
RGB_Dictionary.Add( "black", { 0, 0, 0 } )
RGB_Dictionary.Add( "blanched almond", { 255, 235, 205 } )
RGB_Dictionary.Add( "blanchedalmond", { 255, 235, 205 } )
RGB_Dictionary.Add( "blue violet", { 138, 43, 226 } )
RGB_Dictionary.Add( "blue", { 0, 0, 255 } )
RGB_Dictionary.Add( "blue1", { 0, 0, 255 } )
RGB_Dictionary.Add( "blue2", { 0, 0, 238 } )
RGB_Dictionary.Add( "blue3", { 0, 0, 205 } )
RGB_Dictionary.Add( "blue4", { 0, 0, 139 } )
RGB_Dictionary.Add( "blueviolet", { 138, 43, 226 } )
RGB_Dictionary.Add( "brown", { 165, 42, 42 } )
RGB_Dictionary.Add( "brown1", { 255, 64, 64 } )
RGB_Dictionary.Add( "brown2", { 238, 59, 59 } )
RGB_Dictionary.Add( "brown3", { 205, 51, 51 } )
RGB_Dictionary.Add( "brown4", { 139, 35, 35 } )
RGB_Dictionary.Add( "burlywood", { 222, 184, 135 } )
RGB_Dictionary.Add( "burlywood1", { 255, 211, 155 } )
RGB_Dictionary.Add( "burlywood2", { 238, 197, 145 } )
RGB_Dictionary.Add( "burlywood3", { 205, 170, 125 } )
RGB_Dictionary.Add( "burlywood4", { 139, 115, 85 } )
RGB_Dictionary.Add( "cadet blue", { 95, 158, 160 } )
RGB_Dictionary.Add( "cadetblue", { 95, 158, 160 } )
RGB_Dictionary.Add( "cadetblue1", { 152, 245, 255 } )
```

```
RGB_Dictionary.Add( "cadetblue2", { 142, 229, 238 } )
RGB_Dictionary.Add( "cadetblue3", { 122, 197, 205 } )
RGB_Dictionary.Add( "cadetblue4", { 83, 134, 139 } )
RGB_Dictionary.Add( "chartreuse", { 127, 255, 0 } )
RGB_Dictionary.Add( "chartreuse1", { 127, 255, 0 } )
RGB_Dictionary.Add( "chartreuse2", { 118, 238, 0 } )
RGB_Dictionary.Add( "chartreuse3", { 102, 205, 0 } )
RGB_Dictionary.Add( "chartreuse4", { 69, 139, 0 } )
RGB_Dictionary.Add( "chocolate", { 210, 105, 30 } )
RGB_Dictionary.Add( "chocolate1", { 255, 127, 36 } )
RGB_Dictionary.Add( "chocolate2", { 238, 118, 33 } )
RGB_Dictionary.Add( "chocolate3", { 205, 102, 29 } )
RGB_Dictionary.Add( "chocolate4", { 139, 69, 19 } )
RGB_Dictionary.Add( "coral", { 255, 127, 80 } )
RGB_Dictionary.Add( "coral1", { 255, 114, 86 } )
RGB_Dictionary.Add( "coral2", { 238, 106, 80 } )
RGB_Dictionary.Add( "coral3", { 205, 91, 69 } )
RGB_Dictionary.Add( "coral4", { 139, 62, 47 } )
RGB_Dictionary.Add( "cornflower blue", { 100, 149, 237 } )
RGB_Dictionary.Add( "cornflowerblue", { 100, 149, 237 } )
RGB_Dictionary.Add( "cornsilk", { 255, 248, 220 } )
RGB_Dictionary.Add( "cornsilk1", { 255, 248, 220 } )
RGB_Dictionary.Add( "cornsilk2", { 238, 232, 205 } )
RGB_Dictionary.Add( "cornsilk3", { 205, 200, 177 } )
RGB_Dictionary.Add( "cornsilk4", { 139, 136, 120 } )
RGB_Dictionary.Add( "cyan", { 0, 255, 255 } )
RGB_Dictionary.Add( "cyan1", { 0, 255, 255 } )
RGB_Dictionary.Add( "cyan2", { 0, 238, 238 } )
RGB_Dictionary.Add( "cyan3", { 0, 205, 205 } )
RGB_Dictionary.Add( "cyan4", { 0, 139, 139 } )
RGB_Dictionary.Add( "dark goldenrod", { 184, 134, 11 } )
RGB_Dictionary.Add( "dark green", { 0, 100, 0 } )
RGB_Dictionary.Add( "dark khaki", { 189, 183, 107 } )
RGB_Dictionary.Add( "dark olive green", { 85, 107, 47 } )
RGB_Dictionary.Add( "dark orange", { 255, 140, 0 } )
RGB_Dictionary.Add( "dark orchid", { 153, 50, 204 } )
RGB_Dictionary.Add( "dark salmon", { 233, 150, 122 } )
RGB_Dictionary.Add( "dark sea green", { 143, 188, 143 } )
RGB_Dictionary.Add( "dark slate blue", { 72, 61, 139 } )
RGB_Dictionary.Add( "dark slate gray", { 47, 79, 79 } )
RGB_Dictionary.Add( "dark slate grey", { 47, 79, 79 } )
RGB_Dictionary.Add( "dark turquoise", { 0, 206, 209 } )
RGB_Dictionary.Add( "dark violet", { 148, 0, 211 } )
RGB_Dictionary.Add( "darkgoldenrod", { 184, 134, 11 } )
RGB_Dictionary.Add( "darkgoldenrod1", { 255, 185, 15 } )
RGB_Dictionary.Add( "darkgoldenrod2", { 238, 173, 14 } )
RGB_Dictionary.Add( "darkgoldenrod3", { 205, 149, 12 } )
RGB_Dictionary.Add( "darkgoldenrod4", { 139, 101, 8 } )
RGB_Dictionary.Add( "darkgreen", { 0, 100, 0 } )
RGB_Dictionary.Add( "darkkhaki", { 189, 183, 107 } )
```

```
RGB_Dictionary.Add( "darkolivegreen", { 85, 107, 47 } )
RGB_Dictionary.Add( "darkolivegreen1", { 202, 255, 112 } )
RGB_Dictionary.Add( "darkolivegreen2", { 188, 238, 104 } )
RGB_Dictionary.Add( "darkolivegreen3", { 162, 205, 90 } )
RGB_Dictionary.Add( "darkolivegreen4", { 110, 139, 61 } )
RGB_Dictionary.Add( "darkorange", { 255, 140, 0 } )
RGB_Dictionary.Add( "darkorange1", { 255, 127, 0 } )
RGB_Dictionary.Add( "darkorange2", { 238, 118, 0 } )
RGB_Dictionary.Add( "darkorange3", { 205, 102, 0 } )
RGB_Dictionary.Add( "darkorange4", { 139, 69, 0 } )
RGB_Dictionary.Add( "darkorchid", { 153, 50, 204 } )
RGB_Dictionary.Add( "darkorchid1", { 191, 62, 255 } )
RGB_Dictionary.Add( "darkorchid2", { 178, 58, 238 } )
RGB_Dictionary.Add( "darkorchid3", { 154, 50, 205 } )
RGB_Dictionary.Add( "darkorchid4", { 104, 34, 139 } )
RGB_Dictionary.Add( "darksalmon", { 233, 150, 122 } )
RGB_Dictionary.Add( "darkseagreen", { 143, 188, 143 } )
RGB_Dictionary.Add( "darkseagreen1", { 193, 255, 193 } )
RGB_Dictionary.Add( "darkseagreen2", { 180, 238, 180 } )
RGB_Dictionary.Add( "darkseagreen3", { 155, 205, 155 } )
RGB_Dictionary.Add( "darkseagreen4", { 105, 139, 105 } )
RGB_Dictionary.Add( "darkslateblue", { 72, 61, 139 } )
RGB_Dictionary.Add( "darkslategray", { 47, 79, 79 } )
RGB_Dictionary.Add( "darkslategray1", { 151, 255, 255 } )
RGB_Dictionary.Add( "darkslategray2", { 141, 238, 238 } )
RGB_Dictionary.Add( "darkslategray3", { 121, 205, 205 } )
RGB_Dictionary.Add( "darkslategray4", { 82, 139, 139 } )
RGB_Dictionary.Add( "darkslategrey", { 47, 79, 79 } )
RGB_Dictionary.Add( "darkturquoise", { 0, 206, 209 } )
RGB_Dictionary.Add( "darkviolet", { 148, 0, 211 } )
RGB_Dictionary.Add( "deep pink", { 255, 20, 147 } )
RGB_Dictionary.Add( "deep sky blue", { 0, 191, 255 } )
RGB_Dictionary.Add( "deeppink", { 255, 20, 147 } )
RGB_Dictionary.Add( "deeppink1", { 255, 20, 147 } )
RGB_Dictionary.Add( "deeppink2", { 238, 18, 137 } )
RGB_Dictionary.Add( "deeppink3", { 205, 16, 118 } )
RGB_Dictionary.Add( "deeppink4", { 139, 10, 80 } )
RGB_Dictionary.Add( "deepskyblue", { 0, 191, 255 } )
RGB_Dictionary.Add( "deepskyblue1", { 0, 191, 255 } )
RGB_Dictionary.Add( "deepskyblue2", { 0, 178, 238 } )
RGB_Dictionary.Add( "deepskyblue3", { 0, 154, 205 } )
RGB_Dictionary.Add( "deepskyblue4", { 0, 104, 139 } )
RGB_Dictionary.Add( "dim gray", { 105, 105, 105 } )
RGB_Dictionary.Add( "dim grey", { 105, 105, 105 } )
RGB_Dictionary.Add( "dimgray", { 105, 105, 105 } )
RGB_Dictionary.Add( "dimgrey", { 105, 105, 105 } )
RGB_Dictionary.Add( "dodger blue", { 30, 144, 255 } )
RGB_Dictionary.Add( "dodgerblue", { 30, 144, 255 } )
RGB_Dictionary.Add( "dodgerblue1", { 30, 144, 255 } )
RGB_Dictionary.Add( "dodgerblue2", { 28, 134, 238 } )
```

```
RGB_Dictionary.Add( "dodgerblue3", { 24, 116, 205 } )
RGB_Dictionary.Add( "dodgerblue4", { 16, 78, 139 } )
RGB_Dictionary.Add( "firebrick", { 178, 34, 34 } )
RGB_Dictionary.Add( "firebrick1", { 255, 48, 48 } )
RGB_Dictionary.Add( "firebrick2", { 238, 44, 44 } )
RGB_Dictionary.Add( "firebrick3", { 205, 38, 38 } )
RGB_Dictionary.Add( "firebrick4", { 139, 26, 26 } )
RGB_Dictionary.Add( "floral white", { 255, 250, 240 } )
RGB_Dictionary.Add( "floralwhite", { 255, 250, 240 } )
RGB_Dictionary.Add( "forest green", { 34, 139, 34 } )
RGB_Dictionary.Add( "forestgreen", { 34, 139, 34 } )
RGB_Dictionary.Add( "gainsboro", { 220, 220, 220 } )
RGB_Dictionary.Add( "ghost white", { 248, 248, 255 } )
RGB_Dictionary.Add( "ghostwhite", { 248, 248, 255 } )
RGB_Dictionary.Add( "gold", { 255, 215, 0 } )
RGB_Dictionary.Add( "gold1", { 255, 215, 0 } )
RGB_Dictionary.Add( "gold2", { 238, 201, 0 } )
RGB_Dictionary.Add( "gold3", { 205, 173, 0 } )
RGB_Dictionary.Add( "gold4", { 139, 117, 0 } )
RGB_Dictionary.Add( "goldenrod", { 218, 165, 32 } )
RGB_Dictionary.Add( "goldenrod1", { 255, 193, 37 } )
RGB_Dictionary.Add( "goldenrod2", { 238, 180, 34 } )
RGB_Dictionary.Add( "goldenrod3", { 205, 155, 29 } )
RGB_Dictionary.Add( "goldenrod4", { 139, 105, 20 } )
RGB_Dictionary.Add( "gray", { 192, 192, 192 } )
RGB_Dictionary.Add( "gray0", { 0, 0, 0 } )
RGB_Dictionary.Add( "gray1", { 3, 3, 3 } )
RGB_Dictionary.Add( "gray2", { 5, 5, 5 } )
RGB_Dictionary.Add( "gray3", { 8, 8, 8 } )
RGB_Dictionary.Add( "gray4", { 10, 10, 10 } )
RGB_Dictionary.Add( "gray5", { 13, 13, 13 } )
RGB_Dictionary.Add( "gray6", { 15, 15, 15 } )
RGB_Dictionary.Add( "gray7", { 18, 18, 18 } )
RGB_Dictionary.Add( "gray8", { 20, 20, 20 } )
RGB_Dictionary.Add( "gray9", { 23, 23, 23 } )
RGB_Dictionary.Add( "gray10", { 26, 26, 26 } )
RGB_Dictionary.Add( "gray11", { 28, 28, 28 } )
RGB_Dictionary.Add( "gray12", { 31, 31, 31 } )
RGB_Dictionary.Add( "gray13", { 33, 33, 33 } )
RGB_Dictionary.Add( "gray14", { 36, 36, 36 } )
RGB_Dictionary.Add( "gray15", { 38, 38, 38 } )
RGB_Dictionary.Add( "gray16", { 41, 41, 41 } )
RGB_Dictionary.Add( "gray17", { 43, 43, 43 } )
RGB_Dictionary.Add( "gray18", { 46, 46, 46 } )
RGB_Dictionary.Add( "gray19", { 48, 48, 48 } )
RGB_Dictionary.Add( "gray20", { 51, 51, 51 } )
RGB_Dictionary.Add( "gray21", { 54, 54, 54 } )
RGB_Dictionary.Add( "gray22", { 56, 56, 56 } )
RGB_Dictionary.Add( "gray23", { 59, 59, 59 } )
RGB_Dictionary.Add( "gray24", { 61, 61, 61 } )
```

```
RGB_Dictionary.Add( "gray25", { 64, 64, 64 } )
RGB_Dictionary.Add( "gray26", { 66, 66, 66 } )
RGB_Dictionary.Add( "gray27", { 69, 69, 69 } )
RGB_Dictionary.Add( "gray28", { 71, 71, 71 } )
RGB_Dictionary.Add( "gray29", { 74, 74, 74 } )
RGB_Dictionary.Add( "gray30", { 77, 77, 77 } )
RGB_Dictionary.Add( "gray31", { 79, 79, 79 } )
RGB_Dictionary.Add( "gray32", { 82, 82, 82 } )
RGB_Dictionary.Add( "gray33", { 84, 84, 84 } )
RGB_Dictionary.Add( "gray34", { 87, 87, 87 } )
RGB_Dictionary.Add( "gray35", { 89, 89, 89 } )
RGB_Dictionary.Add( "gray36", { 92, 92, 92 } )
RGB_Dictionary.Add( "gray37", { 94, 94, 94 } )
RGB_Dictionary.Add( "gray38", { 97, 97, 97 } )
RGB_Dictionary.Add( "gray39", { 99, 99, 99 } )
RGB_Dictionary.Add( "gray40", { 102, 102, 102 } )
RGB_Dictionary.Add( "gray41", { 105, 105, 105 } )
RGB_Dictionary.Add( "gray42", { 107, 107, 107 } )
RGB_Dictionary.Add( "gray43", { 110, 110, 110 } )
RGB_Dictionary.Add( "gray44", { 112, 112, 112 } )
RGB_Dictionary.Add( "gray45", { 115, 115, 115 } )
RGB_Dictionary.Add( "gray46", { 117, 117, 117 } )
RGB_Dictionary.Add( "gray47", { 120, 120, 120 } )
RGB_Dictionary.Add( "gray48", { 122, 122, 122 } )
RGB_Dictionary.Add( "gray49", { 125, 125, 125 } )
RGB_Dictionary.Add( "gray50", { 127, 127, 127 } )
RGB_Dictionary.Add( "gray51", { 130, 130, 130 } )
RGB_Dictionary.Add( "gray52", { 133, 133, 133 } )
RGB_Dictionary.Add( "gray53", { 135, 135, 135 } )
RGB_Dictionary.Add( "gray54", { 138, 138, 138 } )
RGB_Dictionary.Add( "gray55", { 140, 140, 140 } )
RGB_Dictionary.Add( "gray56", { 143, 143, 143 } )
RGB_Dictionary.Add( "gray57", { 145, 145, 145 } )
RGB_Dictionary.Add( "gray58", { 148, 148, 148 } )
RGB_Dictionary.Add( "gray59", { 150, 150, 150 } )
RGB_Dictionary.Add( "gray60", { 153, 153, 153 } )
RGB_Dictionary.Add( "gray61", { 156, 156, 156 } )
RGB_Dictionary.Add( "gray62", { 158, 158, 158 } )
RGB_Dictionary.Add( "gray63", { 161, 161, 161 } )
RGB_Dictionary.Add( "gray64", { 163, 163, 163 } )
RGB_Dictionary.Add( "gray65", { 166, 166, 166 } )
RGB_Dictionary.Add( "gray66", { 168, 168, 168 } )
RGB_Dictionary.Add( "gray67", { 171, 171, 171 } )
RGB_Dictionary.Add( "gray68", { 173, 173, 173 } )
RGB_Dictionary.Add( "gray69", { 176, 176, 176 } )
RGB_Dictionary.Add( "gray70", { 179, 179, 179 } )
RGB_Dictionary.Add( "gray71", { 181, 181, 181 } )
RGB_Dictionary.Add( "gray72", { 184, 184, 184 } )
RGB_Dictionary.Add( "gray73", { 186, 186, 186 } )
RGB_Dictionary.Add( "gray74", { 189, 189, 189 } )
```

```
RGB_Dictionary.Add( "gray75", { 191, 191, 191 } )
RGB_Dictionary.Add( "gray76", { 194, 194, 194 } )
RGB_Dictionary.Add( "gray77", { 196, 196, 196 } )
RGB_Dictionary.Add( "gray78", { 199, 199, 199 } )
RGB_Dictionary.Add( "gray79", { 201, 201, 201 } )
RGB_Dictionary.Add( "gray80", { 204, 204, 204 } )
RGB_Dictionary.Add( "gray81", { 207, 207, 207 } )
RGB_Dictionary.Add( "gray82", { 209, 209, 209 } )
RGB_Dictionary.Add( "gray83", { 212, 212, 212 } )
RGB_Dictionary.Add( "gray84", { 214, 214, 214 } )
RGB_Dictionary.Add( "gray85", { 217, 217, 217 } )
RGB_Dictionary.Add( "gray86", { 219, 219, 219 } )
RGB_Dictionary.Add( "gray87", { 222, 222, 222 } )
RGB_Dictionary.Add( "gray88", { 224, 224, 224 } )
RGB_Dictionary.Add( "gray89", { 227, 227, 227 } )
RGB_Dictionary.Add( "gray90", { 229, 229, 229 } )
RGB_Dictionary.Add( "gray91", { 232, 232, 232 } )
RGB_Dictionary.Add( "gray92", { 235, 235, 235 } )
RGB_Dictionary.Add( "gray93", { 237, 237, 237 } )
RGB_Dictionary.Add( "gray94", { 240, 240, 240 } )
RGB_Dictionary.Add( "gray95", { 242, 242, 242 } )
RGB_Dictionary.Add( "gray96", { 245, 245, 245 } )
RGB_Dictionary.Add( "gray97", { 247, 247, 247 } )
RGB_Dictionary.Add( "gray98", { 250, 250, 250 } )
RGB_Dictionary.Add( "gray99", { 252, 252, 252 } )
RGB_Dictionary.Add( "gray100", { 255, 255, 255 } )
RGB_Dictionary.Add( "green yellow", { 173, 255, 47 } )
RGB_Dictionary.Add( "green", { 0, 255, 0 } )
RGB_Dictionary.Add( "green1", { 0, 255, 0 } )
RGB_Dictionary.Add( "green2", { 0, 238, 0 } )
RGB_Dictionary.Add( "green3", { 0, 205, 0 } )
RGB_Dictionary.Add( "green4", { 0, 139, 0 } )
RGB_Dictionary.Add( "greenyellow", { 173, 255, 47 } )
RGB_Dictionary.Add( "grey", { 192, 192, 192 } )
RGB_Dictionary.Add( "grey0", { 0, 0, 0 } )
RGB_Dictionary.Add( "grey1", { 3, 3, 3 } )
RGB_Dictionary.Add( "grey2", { 5, 5, 5 } )
RGB_Dictionary.Add( "grey3", { 8, 8, 8 } )
RGB_Dictionary.Add( "grey4", { 10, 10, 10 } )
RGB_Dictionary.Add( "grey5", { 13, 13, 13 } )
RGB_Dictionary.Add( "grey6", { 15, 15, 15 } )
RGB_Dictionary.Add( "grey7", { 18, 18, 18 } )
RGB_Dictionary.Add( "grey8", { 20, 20, 20 } )
RGB_Dictionary.Add( "grey9", { 23, 23, 23 } )
RGB_Dictionary.Add( "grey10", { 26, 26, 26 } )
RGB_Dictionary.Add( "grey11", { 28, 28, 28 } )
RGB_Dictionary.Add( "grey12", { 31, 31, 31 } )
RGB_Dictionary.Add( "grey13", { 33, 33, 33 } )
RGB_Dictionary.Add( "grey14", { 36, 36, 36 } )
RGB_Dictionary.Add( "grey15", { 38, 38, 38 } )
```

```
RGB_Dictionary.Add( "grey16", { 41, 41, 41 } )
RGB_Dictionary.Add( "grey17", { 43, 43, 43 } )
RGB_Dictionary.Add( "grey18", { 46, 46, 46 } )
RGB_Dictionary.Add( "grey19", { 48, 48, 48 } )
RGB_Dictionary.Add( "grey20", { 51, 51, 51 } )
RGB_Dictionary.Add( "grey21", { 54, 54, 54 } )
RGB_Dictionary.Add( "grey22", { 56, 56, 56 } )
RGB_Dictionary.Add( "grey23", { 59, 59, 59 } )
RGB_Dictionary.Add( "grey24", { 61, 61, 61 } )
RGB_Dictionary.Add( "grey25", { 64, 64, 64 } )
RGB_Dictionary.Add( "grey26", { 66, 66, 66 } )
RGB_Dictionary.Add( "grey27", { 69, 69, 69 } )
RGB_Dictionary.Add( "grey28", { 71, 71, 71 } )
RGB_Dictionary.Add( "grey29", { 74, 74, 74 } )
RGB_Dictionary.Add( "grey30", { 77, 77, 77 } )
RGB_Dictionary.Add( "grey31", { 79, 79, 79 } )
RGB_Dictionary.Add( "grey32", { 82, 82, 82 } )
RGB_Dictionary.Add( "grey33", { 84, 84, 84 } )
RGB_Dictionary.Add( "grey34", { 87, 87, 87 } )
RGB_Dictionary.Add( "grey35", { 89, 89, 89 } )
RGB_Dictionary.Add( "grey36", { 92, 92, 92 } )
RGB_Dictionary.Add( "grey37", { 94, 94, 94 } )
RGB_Dictionary.Add( "grey38", { 97, 97, 97 } )
RGB_Dictionary.Add( "grey39", { 99, 99, 99 } )
RGB_Dictionary.Add( "grey40", { 102, 102, 102 } )
RGB_Dictionary.Add( "grey41", { 105, 105, 105 } )
RGB_Dictionary.Add( "grey42", { 107, 107, 107 } )
RGB_Dictionary.Add( "grey43", { 110, 110, 110 } )
RGB_Dictionary.Add( "grey44", { 112, 112, 112 } )
RGB_Dictionary.Add( "grey45", { 115, 115, 115 } )
RGB_Dictionary.Add( "grey46", { 117, 117, 117 } )
RGB_Dictionary.Add( "grey47", { 120, 120, 120 } )
RGB_Dictionary.Add( "grey48", { 122, 122, 122 } )
RGB_Dictionary.Add( "grey49", { 125, 125, 125 } )
RGB_Dictionary.Add( "grey50", { 127, 127, 127 } )
RGB_Dictionary.Add( "grey51", { 130, 130, 130 } )
RGB_Dictionary.Add( "grey52", { 133, 133, 133 } )
RGB_Dictionary.Add( "grey53", { 135, 135, 135 } )
RGB_Dictionary.Add( "grey54", { 138, 138, 138 } )
RGB_Dictionary.Add( "grey55", { 140, 140, 140 } )
RGB_Dictionary.Add( "grey56", { 143, 143, 143 } )
RGB_Dictionary.Add( "grey57", { 145, 145, 145 } )
RGB_Dictionary.Add( "grey58", { 148, 148, 148 } )
RGB_Dictionary.Add( "grey59", { 150, 150, 150 } )
RGB_Dictionary.Add( "grey60", { 153, 153, 153 } )
RGB_Dictionary.Add( "grey61", { 156, 156, 156 } )
RGB_Dictionary.Add( "grey62", { 158, 158, 158 } )
RGB_Dictionary.Add( "grey63", { 161, 161, 161 } )
RGB_Dictionary.Add( "grey64", { 163, 163, 163 } )
RGB_Dictionary.Add( "grey65", { 166, 166, 166 } )
```

```
RGB_Dictionary.Add( "grey66", { 168, 168, 168 } )
RGB_Dictionary.Add( "grey67", { 171, 171, 171 } )
RGB_Dictionary.Add( "grey68", { 173, 173, 173 } )
RGB_Dictionary.Add( "grey69", { 176, 176, 176 } )
RGB_Dictionary.Add( "grey70", { 179, 179, 179 } )
RGB_Dictionary.Add( "grey71", { 181, 181, 181 } )
RGB_Dictionary.Add( "grey72", { 184, 184, 184 } )
RGB_Dictionary.Add( "grey73", { 186, 186, 186 } )
RGB_Dictionary.Add( "grey74", { 189, 189, 189 } )
RGB_Dictionary.Add( "grey75", { 191, 191, 191 } )
RGB_Dictionary.Add( "grey76", { 194, 194, 194 } )
RGB_Dictionary.Add( "grey77", { 196, 196, 196 } )
RGB_Dictionary.Add( "grey78", { 199, 199, 199 } )
RGB_Dictionary.Add( "grey79", { 201, 201, 201 } )
RGB_Dictionary.Add( "grey80", { 204, 204, 204 } )
RGB_Dictionary.Add( "grey81", { 207, 207, 207 } )
RGB_Dictionary.Add( "grey82", { 209, 209, 209 } )
RGB_Dictionary.Add( "grey83", { 212, 212, 212 } )
RGB_Dictionary.Add( "grey84", { 214, 214, 214 } )
RGB_Dictionary.Add( "grey85", { 217, 217, 217 } )
RGB_Dictionary.Add( "grey86", { 219, 219, 219 } )
RGB_Dictionary.Add( "grey87", { 222, 222, 222 } )
RGB_Dictionary.Add( "grey88", { 224, 224, 224 } )
RGB_Dictionary.Add( "grey89", { 227, 227, 227 } )
RGB_Dictionary.Add( "grey90", { 229, 229, 229 } )
RGB_Dictionary.Add( "grey91", { 232, 232, 232 } )
RGB_Dictionary.Add( "grey92", { 235, 235, 235 } )
RGB_Dictionary.Add( "grey93", { 237, 237, 237 } )
RGB_Dictionary.Add( "grey94", { 240, 240, 240 } )
RGB_Dictionary.Add( "grey95", { 242, 242, 242 } )
RGB_Dictionary.Add( "grey96", { 245, 245, 245 } )
RGB_Dictionary.Add( "grey97", { 247, 247, 247 } )
RGB_Dictionary.Add( "grey98", { 250, 250, 250 } )
RGB_Dictionary.Add( "grey99", { 252, 252, 252 } )
RGB_Dictionary.Add( "grey100", { 255, 255, 255 } )
RGB_Dictionary.Add( "honeydew", { 240, 255, 240 } )
RGB_Dictionary.Add( "honeydew1", { 240, 255, 240 } )
RGB_Dictionary.Add( "honeydew2", { 224, 238, 224 } )
RGB_Dictionary.Add( "honeydew3", { 193, 205, 193 } )
RGB_Dictionary.Add( "honeydew4", { 131, 139, 131 } )
RGB_Dictionary.Add( "hot pink", { 255, 105, 180 } )
RGB_Dictionary.Add( "hotpink", { 255, 105, 180 } )
RGB_Dictionary.Add( "hotpink1", { 255, 110, 180 } )
RGB_Dictionary.Add( "hotpink2", { 238, 106, 167 } )
RGB_Dictionary.Add( "hotpink3", { 205, 96, 144 } )
RGB_Dictionary.Add( "hotpink4", { 139, 58, 98 } )
RGB_Dictionary.Add( "indian red", { 205, 92, 92 } )
RGB_Dictionary.Add( "indianred", { 205, 92, 92 } )
RGB_Dictionary.Add( "indianred1", { 255, 106, 106 } )
RGB_Dictionary.Add( "indianred2", { 238, 99, 99 } )
```

```
RGB_Dictionary.Add( "indianred3", { 205, 85, 85 } )
RGB_Dictionary.Add( "indianred4", { 139, 58, 58 } )
RGB_Dictionary.Add( "ivory", { 255, 255, 240 } )
RGB_Dictionary.Add( "ivory1", { 255, 255, 240 } )
RGB_Dictionary.Add( "ivory2", { 238, 238, 224 } )
RGB_Dictionary.Add( "ivory3", { 205, 205, 193 } )
RGB_Dictionary.Add( "ivory4", { 139, 139, 131 } )
RGB_Dictionary.Add( "khaki", { 240, 230, 140 } )
RGB_Dictionary.Add( "khaki1", { 255, 246, 143 } )
RGB_Dictionary.Add( "khaki2", { 238, 230, 133 } )
RGB_Dictionary.Add( "khaki3", { 205, 198, 115 } )
RGB_Dictionary.Add( "khaki4", { 139, 134, 78 } )
RGB_Dictionary.Add( "lavender blush", { 255, 240, 245 } )
RGB_Dictionary.Add( "lavender", { 230, 230, 250 } )
RGB_Dictionary.Add( "lavenderblush", { 255, 240, 245 } )
RGB_Dictionary.Add( "lavenderblush1", { 255, 240, 245 } )
RGB_Dictionary.Add( "lavenderblush2", { 238, 224, 229 } )
RGB_Dictionary.Add( "lavenderblush3", { 205, 193, 197 } )
RGB_Dictionary.Add( "lavenderblush4", { 139, 131, 134 } )
RGB_Dictionary.Add( "lawn green", { 124, 252, 0 } )
RGB_Dictionary.Add( "lawngreen", { 124, 252, 0 } )
RGB_Dictionary.Add( "lemon chiffon", { 255, 250, 205 } )
RGB_Dictionary.Add( "lemonchiffon", { 255, 250, 205 } )
RGB_Dictionary.Add( "lemonchiffon1", { 255, 250, 205 } )
RGB_Dictionary.Add( "lemonchiffon2", { 238, 233, 191 } )
RGB_Dictionary.Add( "lemonchiffon3", { 205, 201, 165 } )
RGB_Dictionary.Add( "lemonchiffon4", { 139, 137, 112 } )
RGB_Dictionary.Add( "light blue", { 173, 216, 230 } )
RGB_Dictionary.Add( "light coral", { 240, 128, 128 } )
RGB_Dictionary.Add( "light cyan", { 224, 255, 255 } )
RGB_Dictionary.Add( "light goldenrod yellow", { 250, 250, 210 } )
RGB_Dictionary.Add( "light goldenrod", { 238, 221, 130 } )
RGB_Dictionary.Add( "light gray", { 211, 211, 211 } )
RGB_Dictionary.Add( "light grey", { 211, 211, 211 } )
RGB_Dictionary.Add( "light pink", { 255, 182, 193 } )
RGB_Dictionary.Add( "light salmon", { 255, 160, 122 } )
RGB_Dictionary.Add( "light sea green", { 32, 178, 170 } )
RGB_Dictionary.Add( "light sky blue", { 135, 206, 250 } )
RGB_Dictionary.Add( "light slate blue", { 132, 112, 255 } )
RGB_Dictionary.Add( "light slate gray", { 119, 136, 153 } )
RGB_Dictionary.Add( "light slate grey", { 119, 136, 153 } )
RGB_Dictionary.Add( "light steel blue", { 176, 196, 222 } )
RGB_Dictionary.Add( "light yellow", { 255, 255, 224 } )
RGB_Dictionary.Add( "lightblue", { 173, 216, 230 } )
RGB_Dictionary.Add( "lightblue1", { 191, 239, 255 } )
RGB_Dictionary.Add( "lightblue2", { 178, 223, 238 } )
RGB_Dictionary.Add( "lightblue3", { 154, 192, 205 } )
RGB_Dictionary.Add( "lightblue4", { 104, 131, 139 } )
RGB_Dictionary.Add( "lightcoral", { 240, 128, 128 } )
RGB_Dictionary.Add( "lightcyan", { 224, 255, 255 } )
```

```
RGB_Dictionary.Add( "lightcyan1", { 224, 255, 255 } )
RGB_Dictionary.Add( "lightcyan2", { 209, 238, 238 } )
RGB_Dictionary.Add( "lightcyan3", { 180, 205, 205 } )
RGB_Dictionary.Add( "lightcyan4", { 122, 139, 139 } )
RGB_Dictionary.Add( "lightgoldenrod", { 238, 221, 130 } )
RGB_Dictionary.Add( "lightgoldenrod1", { 255, 236, 139 } )
RGB_Dictionary.Add( "lightgoldenrod2", { 238, 220, 130 } )
RGB_Dictionary.Add( "lightgoldenrod3", { 205, 190, 112 } )
RGB_Dictionary.Add( "lightgoldenrod4", { 139, 129, 76 } )
RGB_Dictionary.Add( "lightgoldenrodyellow", { 250, 250, 210 } )
RGB_Dictionary.Add( "lightgray", { 211, 211, 211 } )
RGB_Dictionary.Add( "lightgrey", { 211, 211, 211 } )
RGB_Dictionary.Add( "lightpink", { 255, 182, 193 } )
RGB_Dictionary.Add( "lightpink1", { 255, 174, 185 } )
RGB_Dictionary.Add( "lightpink2", { 238, 162, 173 } )
RGB_Dictionary.Add( "lightpink3", { 205, 140, 149 } )
RGB_Dictionary.Add( "lightpink4", { 139, 95, 101 } )
RGB_Dictionary.Add( "lightsalmon", { 255, 160, 122 } )
RGB_Dictionary.Add( "lightsalmon1", { 255, 160, 122 } )
RGB_Dictionary.Add( "lightsalmon2", { 238, 149, 114 } )
RGB_Dictionary.Add( "lightsalmon3", { 205, 129, 98 } )
RGB_Dictionary.Add( "lightsalmon4", { 139, 87, 66 } )
RGB_Dictionary.Add( "lightseagreen", { 32, 178, 170 } )
RGB_Dictionary.Add( "lightskyblue", { 135, 206, 250 } )
RGB_Dictionary.Add( "lightskyblue1", { 176, 226, 255 } )
RGB_Dictionary.Add( "lightskyblue2", { 164, 211, 238 } )
RGB_Dictionary.Add( "lightskyblue3", { 141, 182, 205 } )
RGB_Dictionary.Add( "lightskyblue4", { 96, 123, 139 } )
RGB_Dictionary.Add( "lightslateblue", { 132, 112, 255 } )
RGB_Dictionary.Add( "lightslategray", { 119, 136, 153 } )
RGB_Dictionary.Add( "lightslategrey", { 119, 136, 153 } )
RGB_Dictionary.Add( "lightsteelblue", { 176, 196, 222 } )
RGB_Dictionary.Add( "lightsteelblue1", { 202, 225, 255 } )
RGB_Dictionary.Add( "lightsteelblue2", { 188, 210, 238 } )
RGB_Dictionary.Add( "lightsteelblue3", { 162, 181, 205 } )
RGB_Dictionary.Add( "lightsteelblue4", { 110, 123, 139 } )
RGB_Dictionary.Add( "lightyellow", { 255, 255, 224 } )
RGB_Dictionary.Add( "lightyellow1", { 255, 255, 224 } )
RGB_Dictionary.Add( "lightyellow2", { 238, 238, 209 } )
RGB_Dictionary.Add( "lightyellow3", { 205, 205, 180 } )
RGB_Dictionary.Add( "lightyellow4", { 139, 139, 122 } )
RGB_Dictionary.Add( "lime green", { 50, 205, 50 } )
RGB_Dictionary.Add( "limegreen", { 50, 205, 50 } )
RGB_Dictionary.Add( "linen", { 250, 240, 230 } )
RGB_Dictionary.Add( "magenta", { 255, 0, 255 } )
RGB_Dictionary.Add( "magenta1", { 255, 0, 255 } )
RGB_Dictionary.Add( "magenta2", { 238, 0, 238 } )
RGB_Dictionary.Add( "magenta3", { 205, 0, 205 } )
RGB_Dictionary.Add( "magenta4", { 139, 0, 139 } )
RGB_Dictionary.Add( "maroon", { 176, 48, 96 } )
```

```
RGB_Dictionary.Add( "maroon1", { 255, 52, 179 } )
RGB_Dictionary.Add( "maroon2", { 238, 48, 167 } )
RGB_Dictionary.Add( "maroon3", { 205, 41, 144 } )
RGB_Dictionary.Add( "maroon4", { 139, 28, 98 } )
RGB_Dictionary.Add( "medium aquamarine", { 102, 205, 170 } )
RGB_Dictionary.Add( "medium blue", { 0, 0, 205 } )
RGB_Dictionary.Add( "medium orchid", { 186, 85, 211 } )
RGB_Dictionary.Add( "medium purple", { 147, 112, 219 } )
RGB_Dictionary.Add( "medium sea green", { 60, 179, 113 } )
RGB_Dictionary.Add( "medium slate blue", { 123, 104, 238 } )
RGB_Dictionary.Add( "medium spring green", { 0, 250, 154 } )
RGB_Dictionary.Add( "medium turquoise", { 72, 209, 204 } )
RGB_Dictionary.Add( "medium violet red", { 199, 21, 133 } )
RGB_Dictionary.Add( "mediumaquamarine", { 102, 205, 170 } )
RGB_Dictionary.Add( "mediumblue", { 0, 0, 205 } )
RGB_Dictionary.Add( "mediumorchid", { 186, 85, 211 } )
RGB_Dictionary.Add( "mediumorchid1", { 224, 102, 255 } )
RGB_Dictionary.Add( "mediumorchid2", { 209, 95, 238 } )
RGB_Dictionary.Add( "mediumorchid3", { 180, 82, 205 } )
RGB_Dictionary.Add( "mediumorchid4", { 122, 55, 139 } )
RGB_Dictionary.Add( "mediumpurple", { 147, 112, 219 } )
RGB_Dictionary.Add( "mediumpurple1", { 171, 130, 255 } )
RGB_Dictionary.Add( "mediumpurple2", { 159, 121, 238 } )
RGB_Dictionary.Add( "mediumpurple3", { 137, 104, 205 } )
RGB_Dictionary.Add( "mediumpurple4", { 93, 71, 139 } )
RGB_Dictionary.Add( "mediumseagreen", { 60, 179, 113 } )
RGB_Dictionary.Add( "mediumslateblue", { 123, 104, 238 } )
RGB_Dictionary.Add( "mediumspringgreen", { 0, 250, 154 } )
RGB_Dictionary.Add( "mediumturquoise", { 72, 209, 204 } )
RGB_Dictionary.Add( "mediumvioletred", { 199, 21, 133 } )
RGB_Dictionary.Add( "midnight blue", { 25, 25, 112 } )
RGB_Dictionary.Add( "midnightblue", { 25, 25, 112 } )
RGB_Dictionary.Add( "mint cream", { 245, 255, 250 } )
RGB_Dictionary.Add( "mintcream", { 245, 255, 250 } )
RGB_Dictionary.Add( "misty rose", { 255, 228, 225 } )
RGB_Dictionary.Add( "mistyrose", { 255, 228, 225 } )
RGB_Dictionary.Add( "mistyrose1", { 255, 228, 225 } )
RGB_Dictionary.Add( "mistyrose2", { 238, 213, 210 } )
RGB_Dictionary.Add( "mistyrose3", { 205, 183, 181 } )
RGB_Dictionary.Add( "mistyrose4", { 139, 125, 123 } )
RGB_Dictionary.Add( "moccasin", { 255, 228, 181 } )
RGB_Dictionary.Add( "navajo white", { 255, 222, 173 } )
RGB_Dictionary.Add( "navajowhite", { 255, 222, 173 } )
RGB_Dictionary.Add( "navajowhite1", { 255, 222, 173 } )
RGB_Dictionary.Add( "navajowhite2", { 238, 207, 161 } )
RGB_Dictionary.Add( "navajowhite3", { 205, 179, 139 } )
RGB_Dictionary.Add( "navajowhite4", { 139, 121, 94 } )
RGB_Dictionary.Add( "navy blue", { 0, 0, 128 } )
RGB_Dictionary.Add( "navy", { 0, 0, 128 } )
RGB_Dictionary.Add( "navyblue", { 0, 0, 128 } )
```

```
RGB_Dictionary.Add( "old lace", { 253, 245, 230 } )
RGB_Dictionary.Add( "oldlace", { 253, 245, 230 } )
RGB_Dictionary.Add( "olive drab", { 107, 142, 35 } )
RGB_Dictionary.Add( "olivedrab", { 107, 142, 35 } )
RGB_Dictionary.Add( "olivedrab1", { 192, 255, 62 } )
RGB_Dictionary.Add( "olivedrab2", { 179, 238, 58 } )
RGB_Dictionary.Add( "olivedrab3", { 154, 205, 50 } )
RGB_Dictionary.Add( "olivedrab4", { 105, 139, 34 } )
RGB_Dictionary.Add( "orange red", { 255, 69, 0 } )
RGB_Dictionary.Add( "orange", { 255, 165, 0 } )
RGB_Dictionary.Add( "orange1", { 255, 165, 0 } )
RGB_Dictionary.Add( "orange2", { 238, 154, 0 } )
RGB_Dictionary.Add( "orange3", { 205, 133, 0 } )
RGB_Dictionary.Add( "orange4", { 139, 90, 0 } )
RGB_Dictionary.Add( "orangered", { 255, 69, 0 } )
RGB_Dictionary.Add( "orangered1", { 255, 69, 0 } )
RGB_Dictionary.Add( "orangered2", { 238, 64, 0 } )
RGB_Dictionary.Add( "orangered3", { 205, 55, 0 } )
RGB_Dictionary.Add( "orangered4", { 139, 37, 0 } )
RGB_Dictionary.Add( "orchid", { 218, 112, 214 } )
RGB_Dictionary.Add( "orchid1", { 255, 131, 250 } )
RGB_Dictionary.Add( "orchid2", { 238, 122, 233 } )
RGB_Dictionary.Add( "orchid3", { 205, 105, 201 } )
RGB_Dictionary.Add( "orchid4", { 139, 71, 137 } )
RGB_Dictionary.Add( "pale goldenrod", { 238, 232, 170 } )
RGB_Dictionary.Add( "pale green", { 152, 251, 152 } )
RGB_Dictionary.Add( "pale turquoise", { 175, 238, 238 } )
RGB_Dictionary.Add( "pale violet red", { 219, 112, 147 } )
RGB_Dictionary.Add( "palegoldenrod", { 238, 232, 170 } )
RGB_Dictionary.Add( "palegreen", { 152, 251, 152 } )
RGB_Dictionary.Add( "palegreen1", { 154, 255, 154 } )
RGB_Dictionary.Add( "palegreen2", { 144, 238, 144 } )
RGB_Dictionary.Add( "palegreen3", { 124, 205, 124 } )
RGB_Dictionary.Add( "palegreen4", { 84, 139, 84 } )
RGB_Dictionary.Add( "paleturquoise", { 175, 238, 238 } )
RGB_Dictionary.Add( "paleturquoise1", { 187, 255, 255 } )
RGB_Dictionary.Add( "paleturquoise2", { 174, 238, 238 } )
RGB_Dictionary.Add( "paleturquoise3", { 150, 205, 205 } )
RGB_Dictionary.Add( "paleturquoise4", { 102, 139, 139 } )
RGB_Dictionary.Add( "palevioletred", { 219, 112, 147 } )
RGB_Dictionary.Add( "palevioletred1", { 255, 130, 171 } )
RGB_Dictionary.Add( "palevioletred2", { 238, 121, 159 } )
RGB_Dictionary.Add( "palevioletred3", { 205, 104, 137 } )
RGB_Dictionary.Add( "palevioletred4", { 139, 71, 93 } )
RGB_Dictionary.Add( "papaya whip", { 255, 239, 213 } )
RGB_Dictionary.Add( "papayawhip", { 255, 239, 213 } )
RGB_Dictionary.Add( "peach puff", { 255, 218, 185 } )
RGB_Dictionary.Add( "peachpuff", { 255, 218, 185 } )
RGB_Dictionary.Add( "peachpuff1", { 255, 218, 185 } )
RGB_Dictionary.Add( "peachpuff2", { 238, 203, 173 } )
```

```
RGB_Dictionary.Add( "peachpuff3", { 205, 175, 149 } )
RGB_Dictionary.Add( "peachpuff4", { 139, 119, 101 } )
RGB_Dictionary.Add( "peru", { 205, 133, 63 } )
RGB_Dictionary.Add( "pink", { 255, 192, 203 } )
RGB_Dictionary.Add( "pink1", { 255, 181, 197 } )
RGB_Dictionary.Add( "pink2", { 238, 169, 184 } )
RGB_Dictionary.Add( "pink3", { 205, 145, 158 } )
RGB_Dictionary.Add( "pink4", { 139, 99, 108 } )
RGB_Dictionary.Add( "plum", { 221, 160, 221 } )
RGB_Dictionary.Add( "plum1", { 255, 187, 255 } )
RGB_Dictionary.Add( "plum2", { 238, 174, 238 } )
RGB_Dictionary.Add( "plum3", { 205, 150, 205 } )
RGB_Dictionary.Add( "plum4", { 139, 102, 139 } )
RGB_Dictionary.Add( "powder blue", { 176, 224, 230 } )
RGB_Dictionary.Add( "powderblue", { 176, 224, 230 } )
RGB_Dictionary.Add( "purple", { 160, 32, 240 } )
RGB_Dictionary.Add( "purple1", { 155, 48, 255 } )
RGB_Dictionary.Add( "purple2", { 145, 44, 238 } )
RGB_Dictionary.Add( "purple3", { 125, 38, 205 } )
RGB_Dictionary.Add( "purple4", { 85, 26, 139 } )
RGB_Dictionary.Add( "red", { 255, 0, 0 } )
RGB_Dictionary.Add( "red1", { 255, 0, 0 } )
RGB_Dictionary.Add( "red2", { 238, 0, 0 } )
RGB_Dictionary.Add( "red3", { 205, 0, 0 } )
RGB_Dictionary.Add( "red4", { 139, 0, 0 } )
RGB_Dictionary.Add( "rosy brown", { 188, 143, 143 } )
RGB_Dictionary.Add( "rosybrown", {-188, 143, 143 } )
RGB_Dictionary.Add( "rosybrown1", { 255, 193, 193 } )
RGB_Dictionary.Add( "rosybrown2", { 238, 180, 180 } )
RGB_Dictionary.Add( "rosybrown3", { 205, 155, 155 } )
RGB_Dictionary.Add( "rosybrown4", { 139, 105, 105 } )
RGB_Dictionary.Add( "royal blue", { 65, 105, 225 } )
RGB_Dictionary.Add( "royalblue", { 65, 105, 225 } )
RGB_Dictionary.Add( "royalblue1", { 72, 118, 255 } )
RGB_Dictionary.Add( "royalblue2", { 67, 110, 238 } )
RGB_Dictionary.Add( "royalblue3", { 58, 95, 205 } )
RGB_Dictionary.Add( "royalblue4", { 39, 64, 139 } )
RGB_Dictionary.Add( "saddle brown", { 139, 69, 19 } )
RGB_Dictionary.Add( "saddlebrown", { 139, 69, 19 } )
RGB_Dictionary.Add( "salmon", { 250, 128, 114 } )
RGB_Dictionary.Add( "salmon1", { 255, 140, 105 } )
RGB_Dictionary.Add( "salmon2", { 238, 130, 98 } )
RGB_Dictionary.Add( "salmon3", { 205, 112, 84 } )
RGB_Dictionary.Add( "salmon4", { 139, 76, 57 } )
RGB_Dictionary.Add( "sandy brown", { 244, 164, 96 } )
RGB_Dictionary.Add( "sandybrown", { 244, 164, 96 } )
RGB_Dictionary.Add( "sea green", { 46, 139, 87 } )
RGB_Dictionary.Add( "seagreen", { 46, 139, 87 } )
RGB_Dictionary.Add( "seagreen1", { 84, 255, 159 } )
RGB_Dictionary.Add( "seagreen2", { 78, 238, 148 } )
```

```
RGB_Dictionary.Add( "seagreen3", { 67, 205, 128 } )
RGB_Dictionary.Add( "seagreen4", { 46, 139, 87 } )
RGB_Dictionary.Add( "seashell", { 255, 245, 238 } )
RGB_Dictionary.Add( "seashell1", { 255, 245, 238 } )
RGB_Dictionary.Add( "seashell2", { 238, 229, 222 } )
RGB_Dictionary.Add( "seashell3", { 205, 197, 191 } )
RGB_Dictionary.Add( "seashell4", { 139, 134, 130 } )
RGB_Dictionary.Add( "sienna", { 160, 82, 45 } )
RGB_Dictionary.Add( "sienna1", { 255, 130, 71 } )
RGB_Dictionary.Add( "sienna2", { 238, 121, 66 } )
RGB_Dictionary.Add( "sienna3", { 205, 104, 57 } )
RGB_Dictionary.Add( "sienna4", { 139, 71, 38 } )
RGB_Dictionary.Add( "sky blue", { 135, 206, 235 } )
RGB_Dictionary.Add( "skyblue", { 135, 206, 235 } )
RGB_Dictionary.Add( "skyblue1", { 135, 206, 255 } )
RGB_Dictionary.Add( "skyblue2", { 126, 192, 238 } )
RGB_Dictionary.Add( "skyblue3", { 108, 166, 205 } )
RGB_Dictionary.Add( "skyblue4", { 74, 112, 139 } )
RGB_Dictionary.Add( "slate blue", { 106, 90, 205 } )
RGB_Dictionary.Add( "slate gray", { 112, 128, 144 } )
RGB_Dictionary.Add( "slate grey", { 112, 128, 144 } )
RGB_Dictionary.Add( "slateblue", { 106, 90, 205 } )
RGB_Dictionary.Add( "slateblue1", { 131, 111, 255 } )
RGB_Dictionary.Add( "slateblue2", { 122, 103, 238 } )
RGB_Dictionary.Add( "slateblue3", { 105, 89, 205 } )
RGB_Dictionary.Add( "slateblue4", { 71, 60, 139 } )
RGB_Dictionary.Add( "slategray", { 112, 128, 144 } )
RGB_Dictionary.Add( "slategray1", { 198, 226, 255 } )
RGB_Dictionary.Add( "slategray2", { 185, 211, 238 } )
RGB_Dictionary.Add( "slategray3", { 159, 182, 205 } )
RGB_Dictionary.Add( "slategray4", { 108, 123, 139 } )
RGB_Dictionary.Add( "slategrey", { 112, 128, 144 } )
RGB_Dictionary.Add( "snow", { 255, 250, 250 } )
RGB_Dictionary.Add( "snow1", { 255, 250, 250 } )
RGB_Dictionary.Add( "snow2", { 238, 233, 233 } )
RGB_Dictionary.Add( "snow3", { 205, 201, 201 } )
RGB_Dictionary.Add( "snow4", { 139, 137, 137 } )
RGB_Dictionary.Add( "spring green", { 0, 255, 127 } )
RGB_Dictionary.Add( "springgreen", { 0, 255, 127 } )
RGB_Dictionary.Add( "springgreen1", { 0, 255, 127 } )
RGB_Dictionary.Add( "springgreen2", { 0, 238, 118 } )
RGB_Dictionary.Add( "springgreen3", { 0, 205, 102 } )
RGB_Dictionary.Add( "springgreen4", { 0, 139, 69 } )
RGB_Dictionary.Add( "steel blue", { 70, 130, 180 } )
RGB_Dictionary.Add( "steelblue", { 70, 130, 180 } )
RGB_Dictionary.Add( "steelblue1", { 99, 184, 255 } )
RGB_Dictionary.Add( "steelblue2", { 92, 172, 238 } )
RGB_Dictionary.Add( "steelblue3", { 79, 148, 205 } )
RGB_Dictionary.Add( "steelblue4", { 54, 100, 139 } )
RGB_Dictionary.Add( "tan", { 210, 180, 140 } )
```

```
RGB_Dictionary.Add( "tan1", { 255, 165, 79 } )
RGB_Dictionary.Add( "tan2", { 238, 154, 73 } )
RGB_Dictionary.Add( "tan3", { 205, 133, 63 } )
RGB_Dictionary.Add( "tan4", { 139, 90, 43 } )
RGB_Dictionary.Add( "thistle", { 216, 191, 216 } )
RGB_Dictionary.Add( "thistle1", { 255, 225, 255 } )
RGB_Dictionary.Add( "thistle2", { 238, 210, 238 } )
RGB_Dictionary.Add( "thistle3", { 205, 181, 205 } )
RGB_Dictionary.Add( "thistle4", { 139, 123, 139 } )
RGB_Dictionary.Add( "tomato", { 255, 99, 71 } )
RGB_Dictionary.Add( "tomato1", { 255, 99, 71 } )
RGB_Dictionary.Add( "tomato2", { 238, 92, 66 } )
RGB_Dictionary.Add( "tomato3", { 205, 79, 57 } )
RGB_Dictionary.Add( "tomato4", { 139, 54, 38 } )
RGB_Dictionary.Add( "turquoise", { 64, 224, 208 } )
RGB_Dictionary.Add( "turquoise1", { 0, 245, 255 } )
RGB_Dictionary.Add( "turquoise2", { 0, 229, 238 } )
RGB_Dictionary.Add( "turquoise3", { 0, 197, 205 } )
RGB_Dictionary.Add( "turquoise4", { 0, 134, 139 } )
RGB_Dictionary.Add( "violet red", { 208, 32, 144 } )
RGB_Dictionary.Add( "violet", { 238, 130, 238 } )
RGB_Dictionary.Add( "violetred", { 208, 32, 144 } )
RGB_Dictionary.Add( "violetred1", { 255, 62, 150 } )
RGB_Dictionary.Add( "violetred2", { 238, 58, 140 } )
RGB_Dictionary.Add( "violetred3", { 205, 50, 120 } )
RGB_Dictionary.Add( "violetred4", { 139, 34, 82 } )
RGB_Dictionary.Add( "wheat", { 245, 222, 179 } )
RGB_Dictionary.Add( "wheat1", { 255, 231, 186 } )
RGB_Dictionary.Add( "wheat2", { 238, 216, 174 } )
RGB_Dictionary.Add( "wheat3", { 205, 186, 150 } )
RGB_Dictionary.Add( "wheat4", { 139, 126, 102 } )
RGB_Dictionary.Add( "white smoke", { 245, 245, 245 } )
RGB_Dictionary.Add( "white", { 255, 255, 255 } )
RGB_Dictionary.Add( "whitesmoke", { 245, 245, 245 } )
RGB_Dictionary.Add( "yellow green", { 154, 205, 50 } )
RGB_Dictionary.Add( "yellow", { 255, 255, 0 } )
RGB_Dictionary.Add( "yellow1", { 255, 255, 0 } )
RGB_Dictionary.Add( "yellow2", { 238, 238, 0 } )
RGB_Dictionary.Add( "yellow3", { 205, 205, 0 } )
RGB_Dictionary.Add( "yellow4", { 139, 139, 0 } )
RGB_Dictionary.Add( "yellowgreen", { 154, 205, 50 } )
return( RGB_Dictionary )
```

Appendix

Review and authors' comments.

WORLD CONSERVATION MONITORING CENTRE

219 Huntingdon Road
Cambridge CB3 0DL
United Kingdom

Telephone: + 44 1223 277314
Fax: + 44 1223 277136
E-mail: Richard.Luxmoore@wcmc.org.uk

FAX MESSAGE

To: Elin Dragland, INSROP

Fax no: 00 47 6712 5047

From: Richard Luxmoore

Date: 26 January 1996

Number of pages (including cover page): 2

Dear Elin Dragland,

Many thanks for the opportunity to review the INSROP report on the design of the GIS.

I found the document to be, in general, well written and comprehensive, though it is very difficult to review such a document without the software to which it relates. My comments are therefore fairly general in nature and do not address the detail of the text.

I would very much welcome the opportunity to review the software together with its datasets. Would it be possible for you to arrange this for me?

It is not clear to me who is the target user for the information system or the software. Without knowledge of this it is very difficult to decide whether the system has been appropriately designed or whether the documentation is pitched at an appropriate level. Use of ArcView, suggests that the system is primarily designed for technical personnel rather than operational managers, ship's captains, etc. If this is correct, what further interface is planned between the technicians and the operational managers?

The system requires the user to have ArcView. This puts a fairly heavy demand on the hardware. I would agree that 8 MB of RAM is too little and would suggest that 16 MB was recommended. This should probably be made explicit at the front of the document, rather than being buried in section 6.2. ArcView does incorporate a data-publisher module (runtime version) and if this were used it would obviate the necessity for users to have ArcView themselves.

In practice, the major problem in the use of environmental information is not its technical manipulation, but correct interpretation. This set of documentation gives no guidance on the content of the datasets, their limitations or how they should

be used.

The English is generally very clear, but there are still a number of minor language errors which could be corrected.

I hope that these few comments have been helpful. I would be most grateful if you could investigate the possibility of transferring a copy of the software and datasets for me to review.

Please note that you appear to have my name incorrect in your records.

Yours sincerely,

Richard Luxmoore, Ph.D.
Head, Habitats Unit
e-mail luxmoore@wcmc.org.uk

Fra: Stig Magnar Lovas
Til: FNI_office.FNI_post(ED)
Dato: 4. mars 1996 12.50
Emne: Review: INSROP Project I.3.1 - INSROP GIS

Hei Elin,

Til informasjon foelger nedenfor kopi av e-mail-korrespondanse
med Richard Luxmoore, WCMC.

Hilsen,
Stig M.

From Stig.M.Lovas@nhl.sintef.no Wed Jan 31 09:48:55 1996
Date: Wed, 31 Jan 1996 09:48:50 +0100 (MET)
From: Stig Magnar Lovas <Stig.M.Lovas@nhl.sintef.no>
To: Richard.Luxmoore@wcmc.org.uk
Subject: Review: INSROP Project I.3.1 - INSROP GIS
Cc: stig.m.lovas@nhl.sintef.no
Content-Length: 3048
Status: RO
X-Lines: 52

Dear Dr. Luxmoore,

Thank you for reviewing the two main discussion papers issued by Project I.3.1 so far. I clearly see the problem of reviewing the project deliverables without the software, but hopefully I can provide some assistance here. I am currently preparing a special installation version with the software, one INSROP GIS (ArcView) project file and a subset of the database (required to run the project file). Altogether this will require about 10 Mb after installation + the size of the installation files. This will enable you to test the functionality of INSROP GIS, without having to install the entire data base. A major part of the system is also devoted to environmental information (birds, fish, mammals, etc) and with special queries and project files devoted to these data. As these data are handled by the Norwegian Polar Institute, I have not included these data in the installation version you will get now. I hope although that the part you get will be sufficient to review the system in its present state.

As you have a nice Web-page, I assume you are able to pick up the installation software through internet (ftp-binary, so I don't have to send you diskettes). Can you confirm this? I will then inform when and from where the installation software can be picked up.

A few comments to your review of the written INSROP GIS documentation:

The target users of INSROP GIS will first be other projects in INSROP (for use as a tool to meet the objectives of INSROP). Several of the INSROP partner institutions in Norway are already using ArcView. When INSROP is completed we plan to release an INSROP GIS version (with data/results) for sponsors of INSROP (decision makers, etc). It is not planned as an operational tool for ship captains etc within INSROP Phase I/II, but it may serve as a basis for creating more operational versions later on.

I will move the INSROP GIS requirements section to a place earlier in the report. I have developed INSROP GIS on a 16Mb PC, and I support your recommendation. Hence I will upgrade the RAM recommendation from the ESRI one (12Mb) to 16Mb. I have read about the recently announced data-publisher module, but due to restricted INSROP budgets and the number of foreseen users (without ArcView) in the near future, I have chosen to postpone further follow-up on this idea. However, I absolutely agree that this will be a good utility for viewing INSROP results.

Regarding providing data and their interpretation, it is not the responsibility of the INSROP GIS design projects. Our task has been to develop a GIS to enable INSROP projects to implement, retrieve, query and display their data, and also make the data available for other projects in a common framework (INSROP GIS/ArcView). We are preparing a paper documenting data we have implemented in a 'standard' we have prepared. This paper (and future updates) is intended to document the INSROP GIS data base and the data sets it comprises.

Best regards,
Stig M. Lovas,
Supervisor, INSROP Project I.3.1

From Richard.Luxmoore@wcmc.org.uk Wed Jan 31 10:16:09 1996
From: Richard.Luxmoore@wcmc.org.uk
Date: Wed, 31 Jan 96 09:12:01 GMT
Subject: re: Review: INSROP Project I.3.1 - INSROP GIS
To: Stig.M.Lovas@nhl.sintef.no
Content-Length: 898
Status: RO
X-Lines: 17

Many thanks for your response. I can certainly pick the files up using ftp and would welcome the chance to review the application. As my main expertise is in the management and use of environmental information I would obviously prefer to review some of the environmental data as well. I have much less expertise in the technical aspects of GIS.

With best wishes

++
++++

Richard Luxmoore, PhD
Head, Habitats Unit Tel: +44 (0)1223 277314
World Conservation Monitoring Centre Fax: +44 (0)1223 277136
219 Huntingdon Road Email: Richard.Luxmoore@wcmc.org.uk
Cambridge CB3 0DL WWW: http://www.wcmc.org.uk
United Kingdom Anonymous FTP: ftp.wcmc.org.uk
++
++++

From Stig.M.Lovas@nhl.sintef.no Thu Feb 15 12:40:12 1996
Date: Thu, 15 Feb 1996 12:39:58 +0100 (MET)
From: Stig Magnar Lovas <Stig.M.Lovas@nhl.sintef.no>
To: Richard.Luxmoore@wcmc.org.uk
Subject: re: Review: INSROP Project I.3.1 - INSROP GIS
Cc: Odd.W.Brude@npolar.no, stig.m.lovas@nhl.sintef.no
Content-Length: 1595
Status: RO
X-Lines: 47

Dear Dr. Luxmoore,

I have now put a demo version of INSROP GIS on the computer at the Norwegian Polar Institute for you to pick up by anonymous-binary ftp.

The ftp address is: maud.npolar.no

When logged in, cd to the pub/insrop/pcSetup directory and copy all files at this directory (using binary ftp) to your local PC. You will need 2.7 Mb to store the installation files and a total of approximately 8.5 Mb to store the installed files.

To install, use File-Run and <yourpath>Øsetup to install everything. You will the get some installation options, and you should click on the following:

Installation options - INSROP GIS modules:
x Customized ArcView ...
x ArcView project library
x Data sets

Installation options - INSROP GIS data sets:
x Base Cartography
x Ice and Snow
x NSR Navigability
x NSR Marine mammals

Installation options - Base Cartography data sets:
x Background Map raster images
x DCW Coastline - 1:1 mill.

Installation options - Ice and Snow data sets:
x AARI Global Sea Ice Data Base - Analyzed

After the installation is completed (as your autoexec.bat file is altered, you will also need to reboot your PC), you must replace your existing ArcView Startup file (Remember to make a backup copy, e.g. startup.v21) with the INSROP GIS Startup file (installed in the main INSROP directory, i.e. C:ØINSROP).

Please contact me immediately if you encounter problems. Be aware that I will be in St.Petersburg from Tuesday 20 February to Friday 23 February.

Good luck!
Stig Magnar Lovas,
SINTEF Civil and Environmental Engineering,
Supervisor INSROP Project I.3.1

From Stig.M.Lovas@nhl.sintef.no Mon Mar 4 09:14:40 1996
Date: Mon, 4 Mar 1996 09:14:32 +0100 (MET)
From: Stig Magnar Lovas <Stig.M.Lovas@nhl.sintef.no>
To: Richard.Luxmoore@wcmc.org.uk
Subject: re: Review: INSROP Project I.3.1 - INSROP GIS
Cc: stig.m.lovas@nhl.sintef.no
Content-Length: 569
X-Lines: 14

Dear Dr. Luxmoore,

Have you been able to install and test the demo-version of INSROP GIS v1.0a I prepared? If you have encountered any problems, please inform me as soon as possible and I will try to solve the problem. As all reports from projects within INSROP Phase I should be completed (and published) by 1 April 1996, and I may need a few days to modify the software and update the final report after the review, I would very much appreciate if I could receive your final review within 1-2 weeks.

Best regards,
Stig M. Lovas,
Supervisor, INSROP Project I.3.1

From Richard.Luxmoore@wcmc.org.uk Mon Mar 4 10:44:06 1996
From: Richard.Luxmoore@wcmc.org.uk
Date: Mon, 4 Mar 96 09:41:52 GMT
Subject: re: Review: INSROP Project I.3.1 - INSROP GIS
To: Stig.M.Lovas@nhl.sintef.no
Content-Length: 244

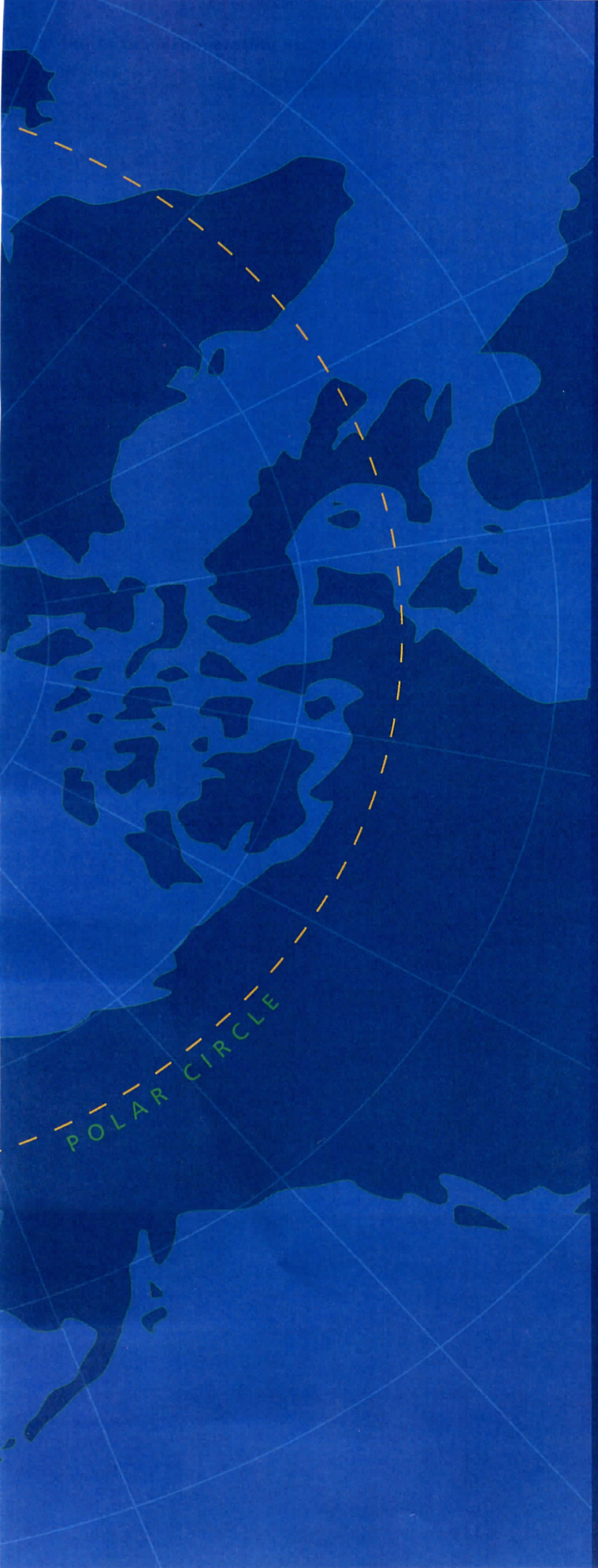
Many thanks for your message. I am afraid that I have been frantically busy for the last few weeks and have not had a chance to look at the INSROP application. I will try to have a look at it over the next couple of weeks.

Richard Luxmoore

From Richard.Luxmoore@wcmc.org.uk Thu Mar 21 14:43:54 1996
From: Richard.Luxmoore@wcmc.org.uk
To: Stig.M.Lovas@nhl.sintef.no
Subject: re: Review: INSROP Project I.3.1 - INSROP GIS
Date: Thu, 21 Mar 96 13:41:14 GMT

Many thanks for your message. I think that you had better proceed on the basis of my earlier comments. I will try to have a look at the aopplication later. Unfortunately both of my staff working on these issues are in Moscow for 3 weeks.

Richard Luxmoore



The three main cooperating institutions of INSROP

Ship & Ocean Foundation (SOF), Tokyo, Japan.

SOF was established in 1975 as a non-profit organization to advance modernization and rationalization of Japan's shipbuilding and related industries, and to give assistance to non-profit organizations associated with these industries. SOF is provided with operation funds by the Sasakawa Foundation, the world's largest foundation operated with revenue from motorboat racing. An integral part of SOF, the Tsukuba Institute, carries out experimental research into ocean environment protection and ocean development.

Central Marine Research & Design Institute (CNIIMF), St. Petersburg, Russia.

CNIIMF was founded in 1929. The institute's research focus is applied and technological with four main goals: the improvment of merchant fleet efficiency; shipping safety; technical development of the merchant fleet; and design support for future fleet development. CNIIMF was a Russian state institution up to 1993, when it was converted into a stock-holding company.

The Fridtjof Nansen Institute (FNI), Lysaker, Norway.

FNI was founded in 1958 and is based at Polhøgda, the home of Fridtjof Nansen, famous Norwegian polar explorer, scientist, humanist and statesman. The institute spesializes in applied social science research, with special focus on international resource and environmental management. In addition to INSROP, the research is organized in six integrated programmes. Typical of FNI research is a multi-disciplinary approach, entailing extensive cooperation with other research institutions both at home and abroad. The INSROP Secretariat is located at FNI.