

Scientific Computer System SCS Version 4.6 **User's Guide**

National Oceanic and Atmospheric Administration
Office of Marine and Aviation Operations
Information Management Division

8403 Colesville Road, Suite 500
Silver Spring, MD 20910

eed.scs@noaa.gov

AUTHORS

John Katebini
Philip Zubaly
Sandy Chang

January 2013

Table of Contents

1	OVERVIEW OF THE SCIENTIFIC COMPUTER SYSTEM (SCS).....	15
1.1	GENERAL INFORMATION	17
1.2	SYSTEM OVERVIEW	17
1.3	SCS DATA FLOW	18
1.4	SCS CONFIGURATION FILE	20
1.5	SCS SYSTEM FILES	23
	<i>Sensor Configuration Files</i>	23
	<i>SCS Database</i>	24
	<i>DeviceConfiguration.XML</i>	24
	<i>NewSensor.XML</i>	24
	<i>Time Stamped DeviceConfiguration.XML and NewSensor.XML</i>	24
	<i>SCS Template Files</i>	24
	<i>RAW Data Files</i>	25
	<i>Compress Data Files</i>	25
	<i>LAB Data Files</i>	25
	<i>Legacy Event Log Data Files</i>	26
	<i>Station, Track line, Coastline, and SHAPE Files</i>	26
1.6	SCS SYSTEM DIRECTORY TREES	27
	<i>SCS Installation Directory</i>	27
	<i>The Executable Directory</i>	27
	<i>Data Log Directory Tree</i>	27
	<i>Ship Directory Tree</i>	28
	<i>The GIS Tree</i>	29
1.7	ONLINE MENU HELP	29
2	INSTALLING SCS.....	33
2.1	PREREQUISITES	33
	<i>Hardware</i>	33
	<i>Operating System</i>	33
	<i>Microsoft .NET Framework</i>	33
	<i>SQL Database Operations</i>	35
	<i>Service Packs</i>	35
	<i>Remove Previous Releases of SCS 4.x</i>	35
2.2	SECURITY INFORMATION FOR SYSTEM ADMINISTRATORS	35
2.3	THE SCS DISTRIBUTION DVD	36
	<i>Installation with Auto-Play</i>	36

	<i>Installation without Auto-Play</i>	37
2.4	INSTALLING SCS SERVER SOFTWARE.....	38
	<i>Start the installation Process</i>	38
	<i>SCS Server Installation</i>	38
	<i>Select Installation Directory</i>	40
	<i>Set Up Ship Directory</i>	40
	<i>Ship Tracker Setup</i>	41
	<i>Datalog and GIS Directories</i>	42
	<i>Release Notes</i>	43
	<i>Finishing the Server Software Installation</i>	43
	<i>SCS Database Installation</i>	44
2.5	INSTALLING SCS REMOTE CLIENT SOFTWARE.....	46
	<i>Prerequisites</i>	46
	<i>Create Drive Mappings</i>	46
	<i>Locate Remote Client Installation Files</i>	46
	<i>Start the Installation Process</i>	47
	<i>Create Drive Mappings</i>	48
	<i>Doing the Installation</i>	48
	<i>Finishing the Installation</i>	49
2.6	POST INSTALLATION TASKS.....	49
	<i>Un-map T Drive (Remote Client)</i>	49
	<i>Set Windows Characteristics (Server and Remote Client)</i>	49
	<i>Modify McAfee Virus Settings (Server and Remote Client)</i>	49
	<i>Installing a Backup Data Log Directory (Server)</i>	50
	<i>Edit SCS Configuration File (Server and Remote Client)</i>	50
	<i>Install SQL Database (Server)</i>	50
2.7	UNINSTALL GUIDE.....	50
	<i>Primary Acquisition Server</i>	50
	<i>Remote SCS Clients</i>	52
3	DATA ACQUISITION.....	57
3.1	STARTING SCS WITH THE SCS MENU.....	57
3.2	SCS CONFIGURATION EDITOR.....	58
	<i>Running the SCS Configuration Program Editor</i>	58
	<i>Editing the File Contents</i>	59
	<i>Adding a New Parameter</i>	60
	<i>Creating a New Parameter</i>	60
3.3	OBTAINING SCS MANAGER STATUS.....	61

3.4	SCS CRUISE MANAGER	61
	<i>The VESSELS Tab</i>	62
	<i>The Cruises Tab</i>	64
	<i>Managing Cruises</i>	65
3.5	SENSOR CONFIGURATION FILES AND THE CONFIGURATION FILE EDITOR	65
	<i>What is Sensor Configuration?</i>	65
	<i>Sensor Configuration Files and SCS Software</i>	65
	<i>Two Golden Rules</i>	67
	<i>Sensor Configuration Files After Data Acquisition</i>	67
	<i>CFE-DB Devices</i>	67
	<i>CFE-DB Message Definitions</i>	69
	<i>Details on Derived Calculation Methods</i>	70
	<i>Manual Message Definition</i>	72
	<i>Simulated Message Definition</i>	72
3.6	SENSOR DATA FIELDS	72
	<i>Special Topics in Sensor Message Acquisition</i>	72
3.7	SENSOR CONFIGURATION EDITOR (CFE-DB)	75
	<i>Introduction</i>	75
	<i>CFE-DB Window Display</i>	76
	<i>The Summary List</i>	77
	<i>“Sensor Device” List</i>	77
	<i>CFE-DB Object Window</i>	77
	<i>Validation, Save and Publish</i>	80
	<i>CFE-DB Edit Menu</i>	81
	<i>CFE-DB View Menu</i>	82
	<i>CFE-DB Object Menu</i>	83
	<i>CFE-DB Output Menu</i>	83
	<i>CFE-DB Logged Data Menu</i>	84
	<i>CFE-DB Sensor device Right Click Immediate Menu</i>	85
	<i>CFE-DB Message Definition Right Click Immediate Menu</i>	86
	<i>CFE-DB Data Field Right Click Immediate Menu</i>	87
	<i>Templates</i>	88
	<i>CFE-DB Describing a Sensor a Step Wise Approach</i>	89
	<i>CFE-DB Parameters</i>	89
	<i>CFE-DB Inventory Parameters</i>	90
	<i>CFE-DB Sensor Device Parameters</i>	94
	<i>Installing a Sensor Device</i>	94

	<i>Common Characteristics Setting Parameter Values</i>	95
	<i>The Validation Tab</i>	95
3.8	PREPARING FOR ACQ STARTUP	97
	<i>Check Free Disk Space on Logging Drives</i>	97
3.9	START ACQ DATA ACQUISITION.....	99
	<i>ACQ Program Description</i>	99
	<i>Starting ACQ: the Main ACQ Window</i>	99
	<i>Monitoring ACQ Operations</i>	102
	<i>Monitoring ACQ Data Logging</i>	105
	<i>ACQ Troubleshooting</i>	109
	<i>Stopping Acquisition</i>	110
3.10	INPUT MANUAL DATA.....	111
	<i>Function</i>	111
3.11	ENABLE/DISABLE MANAGEMENT FUNCTIONS	112
3.12	EXIT MENU	113
4	EVENT LOGGING	115
4.1	INTRODUCTION	115
	<i>Differences in the Two Methods of Event Logging</i>	115
4.2	EVENT LOGGER, “LEGACY” METHOD.....	115
	<i>Event Components</i>	116
	<i>Files Created By the Event Builder and Event Logger</i>	118
	<i>Event Template File</i>	118
	<i>Event Data Files</i>	118
	<i>Event Header File</i>	119
	<i>Button Activity File</i>	119
	<i>Event Save File</i>	119
	<i>Equipment Log File</i>	119
4.3	DEFINING EVENT TEMPLATES WITH EVENT BUILDER.....	120
	<i>Event Builder Main Dialog Overview</i>	120
	<i>Adding a New Component</i>	121
	<i>Editing an Existing Component</i>	121
	<i>General Features of All Add/Edit Dialog Boxes</i>	122
	<i>Add/Edit Dialogs in Detail</i>	123
	<i>The View Menu</i>	142
4.4	RUNNING AN EVENT WITH EVENT LOGGER	143
	<i>Running the Event – The Logger’s Main GUI</i>	144
	<i>Changing the Value of a Metaitem</i>	145

	<i>Button Tab Sheet</i>	146
	<i>Outputs and Monitors Display</i>	146
	<i>Output Cross Reference</i>	147
	<i>Button Activity Window</i>	148
	<i>Stopping the Event</i>	148
	<i>Exiting the Event Logger</i>	148
4.5	(NOT IMPLEMENTED) EVENT LOG MANAGER.....	148
4.6	(NOT YET IMPLEMENTED) EVENT REPORT BUILDING AND GENERATION.....	148
4.7	BUILD BITMAP FORM.....	149
4.8	LABEL/TEMPLATE ASSOCIATION (LTA)	150
4.9	EVENT REPORT BUILDER	151
5	OPERATION EVENT LOGGING (OEL)	153
5.1	INTRODUCTION	153
5.2	OEL BUILDER.....	154
5.3	OEL LOGGER.....	162
5.4	OEL EXTRACTOR.....	167
6	REAL TIME DISPLAYS.....	171
6.1	INTRODUCTION	171
6.2	REAL TIME DISPLAY (NORMAL AND LARGE FONTS)	171
	<i>Function</i>	171
	<i>Procedure</i>	172
	<i>Error Checking</i>	173
6.3	REAL TIME DISPLAY TEMPLATE BUILDER	175
	<i>Function</i>	175
	<i>Main Menu Bar</i>	175
	<i>Available Sensors List</i>	176
	<i>Selected Sensors List</i>	177
	<i>Clear Button</i>	177
	<i>Launch Button</i>	177
6.4	REAL TIME MONITOR.....	178
	<i>Introduction</i>	178
	<i>Real Time Monitor Display Menu</i>	178
6.5	REAL TIME TIME SERIES PLOT.....	180
	<i>Function</i>	180
	<i>The File Menu</i>	181
	<i>The Series Menu</i>	182
	<i>The Zoom Menu</i>	183

<i>The Y Axis Menu</i>	185
<i>The Control Panel Menu</i>	185
<i>Graph Control Panel</i>	186
<i>Graph Behavior</i>	190
6.6 REAL TIME XY PLOT	191
<i>Function</i>	191
<i>The Main Window</i>	192
<i>The File Menu</i>	192
<i>The Series Menu</i>	194
<i>The Zoom Menu</i>	195
<i>The Y Axis Menu</i>	196
<i>The Control Panel Menu</i>	196
<i>The Graph Control Panel</i>	197
<i>Graph Behavior</i>	201
6.7 REAL TIME TRACK LINE PLOT – “GEOTRACK”	202
<i>Introduction</i>	202
<i>Starting Geotrack</i>	202
<i>The Geotrack Main Window: An Overview</i>	203
<i>Controlling the Map with the Mouse</i>	204
<i>The Map’s Context Menu</i>	205
<i>Using the Distance Measuring Tool</i>	205
<i>Controlling the Map with the Layer Legend</i>	205
<i>Working with Tracklines</i>	207
<i>Shape Files</i>	209
<i>Shape Files and Map Layers</i>	209
<i>The Adding Layers Menu</i>	210
<i>Saving and Restoring Views</i>	213
6.8 BUILDING AND USING REAL TIME GAUGE DISPLAYS	214
<i>Gauge Builder</i>	214
<i>Using the Gauge Builder</i>	214
<i>The Gauge Builder Toolbar</i>	214
<i>Gauge Wizard Tutorial</i>	223
6.9 GAUGE DISPLAY	231
<i>Function</i>	231
<i>Procedure</i>	232
6.10 SCS MESSAGE TEMPLATE BUILDER	232
<i>Function</i>	232

	<i>The File Menu</i>	233
	<i>The Message Builder Tab Page</i>	234
6.11	SEND SCS MESSAGE MANAGER	243
	<i>Function</i>	243
	<i>The Message Manager GUI</i>	243
7	QUALITY ASSURANCE PROCESSING	249
7.1	FILES USED FOR QA FUNCTIONS	249
	<i>Raw Data Files (*.RAW)</i>	249
	<i>DeviceConfiguration.XML and NewSensor.XML Configuration Files</i>	250
	<i>Plot Template Files</i>	250
7.2	POST TIME SERIES PLOT	250
	<i>Function</i>	251
	<i>Interface Options</i>	251
7.3	PLOT ALL SENSORS	262
7.4	BUILD NEW COMPRESS FILE (SCS COMPACTOR)	265
7.5	READ DATA FILES	268
7.6	BUILD STATION FILE	269
	<i>The File Menu</i>	269
	<i>Station Builder Main Window Controls</i>	270
	<i>Delete Button</i>	271
	<i>The Format Radio Buttons</i>	271
	<i>Modify Button</i>	272
7.7	DATABASE EXTRACTOR	273
	<i>Function</i>	273
	<i>Main Window GUI</i>	273
	<i>Manual Extraction Mode</i>	274
	<i>Auto Extraction Mode</i>	274
8	MONITOR ERRORS	277
8.1	DATA MON AND SENSOR DATA QUALITY ASSURANCE	277
	<i>Introduction</i>	277
	<i>Incidents</i>	277
	<i>DataMon and Other SCS Software</i>	279
	<i>Timestamps</i>	279
	<i>Summary Window Menus</i>	282
9	SCS UTILITIES	285
9.1	DISPLAY SENSOR INFORMATION	285
	<i>Function</i>	285

<i>The Main Window</i>	285
<i>The Print Button</i>	286
9.2 SCSMAILER	287
<i>Function</i>	287
<i>Using SCSMailer</i>	287
<i>Stop Program</i>	289
9.3 TEST COMMUNICATIONS PORT	289
<i>Introduction</i>	289
<i>Using the Serial Port Diagnostic</i>	290
<i>COM Port Polling</i>	290
9.4 SAMOS MAILER	291
<i>Introduction</i>	291
<i>Setting up the SCS Configuration File</i>	291
<i>Setting up the Sensor Configuration File</i>	292
<i>Setting up the Event Logger Template File</i>	292
<i>Running the Event Logger</i>	292
<i>Running the SAMOS Mailer</i>	292
9.5 TSG TRANSMITTER.....	296
<i>Introduction</i>	296
<i>Setting up the SCS Configuration File</i>	296
<i>Setting up the Sensor Configuration File</i>	297
<i>Setting up the Event Logger Template File</i>	297
<i>Running the Event Logger</i>	297
<i>Running the TSG Transmitter</i>	297
9.6 NODC SUBMISSION APPLICATION.....	301
<i>Operating Instruction</i>	301
<i>Program Interface</i>	301
<i>Settings</i>	302
<i>Archive Directory</i>	303
<i>RAW Data Directory</i>	303
<i>SDAL Address</i>	303
<i>NODC FTP Address</i>	303
<i>NODC FTP Username</i>	303
<i>NODC FTP Password</i>	303
<i>Is Large NOAA Vessel</i>	303
<i>MOA Event Filename</i>	304
<i>Shipname</i>	304

	<i>Cruise / Cruise Leg Selection</i>	304
	<i>Submission Status</i>	305
	<i>Submission Screen</i>	306
	<i>Archive Action</i>	307
	<i>Cruise Leg(s) Date Range</i>	307
	<i>NODC Requested Data</i>	307
	<i>Submitting Data</i>	308
	<i>Quick Start / Recommended SOP</i>	308
10	APPENDIX ONE – THE SENSOR PARAMETER DICTIONARY	311
10.1	FORMAT OF ENTRIES IN THIS DICTIONARY	311
10.2	SENSOR PARAMETERS.....	312
	<i>Average Interval</i>	312
	<i>Average Type</i>	312
	<i>Baud Rate</i>	312
	<i>COM Port</i>	312
	<i>Code Translation</i>	313
	<i>Command Prompt</i>	313
	<i>Comment</i>	313
	<i>Data Bits</i>	314
	<i>Data Field Category</i>	314
	<i>Data Field Position</i>	314
	<i>Decode Type</i>	315
	<i>Delimiters</i>	315
	<i>Delta Check</i>	315
	<i>Delta High</i>	316
	<i>Delta Low</i>	316
	<i>Device Order</i>	317
	<i>Device Type</i>	317
	<i>Eligible Base Data Field</i>	317
	<i>Enabled</i>	318
	<i>Ending Char Position</i>	318
	<i>Extend Label</i>	318
	<i>Installed To</i>	319
	<i>IP Address</i>	319
	<i>Is Template</i>	319
	<i>Label Extenders</i>	319
	<i>Log Sub Folder</i>	320

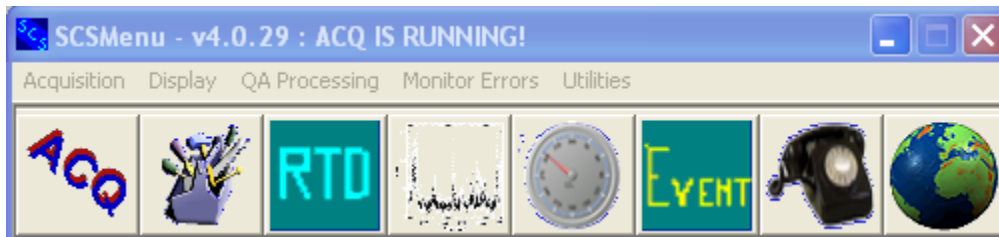
	<i>Logging Rate</i>	320
	<i>Name</i>	320
	<i>Network Protocol</i>	321
	<i>Number Of Samples</i>	321
	<i>Offset</i>	321
	<i>Operator</i>	321
	<i>Parity</i>	322
	<i>Precision</i>	322
	<i>Port Number</i>	322
	<i>Queue Order (NOT YET IMPLEMENTED)</i>	322
	<i>Range Check</i>	323
	<i>Range Low</i>	323
	<i>Record Size</i>	323
	<i>Range High</i>	324
	<i>Sentence Label</i>	324
	<i>Simulation File</i>	324
	<i>Slope</i>	325
	<i>Starting Char Position</i>	325
	<i>Stop Bits</i>	325
	<i>String Position</i>	325
	<i>Synch Check</i>	326
	<i>Synch String</i>	326
	<i>Termination Char</i>	326
	<i>Units</i>	327
	<i>Timeout</i>	327
	<i>Units Field Position</i>	327
	<i>Update Rate</i>	328
11	APPENDIX TWO: INSTALLING SCS WEB SERVICES.....	331
11.1	INSTALL IIS 7.5 ON WINDOWS SERVER 2008 R2.....	332
11.2	CREATE WEB SITE ON IIS 7.5.....	337
12	APPENDIX THREE: SAMOS DATA SUBMISSION PROCEDURES.....	351
12.1	INTRODUCTION AND BACKGROUND.....	351
12.2	SAMOS DATA STANDARD.....	351
12.3	SETTING UP ACQ AND SAMOS DERIVED SENSORS.....	352
12.4	THE SAMOS EVENT LOG.....	353
12.5	RUNNING THE SAMOS EVENT.....	353
12.6	IMPORTANT FILES CREATED BY THE EVENT LOGGER.....	354

12.7	OPERATION OF THE SAMOS MAILER	354
12.8	EXAMINING THE ZIP FILE	355
13	APPENDIX FOUR: NODC DATA COLLECTION PROCEDURES	359
13.1	INTRODUCTION.....	359
13.2	BACKGROUND.....	359
13.3	DATA TO BE COLLECTED	359
13.4	PREPARING FOR DATA COLLECTION	361
13.5	ENDING THE NODC EVENT LOG.....	363
13.6	NODC EVENT OUTPUT	363
13.7	DATA QUALITY ASSURANCE.....	364
	<i>Review Data as Text.....</i>	<i>364</i>
	<i>Review Data as Graph.....</i>	<i>365</i>
13.8	SEND DATA TO NODC	365
	<i>Instructions for uploading SCS data to NODC's FTP server</i>	<i>366</i>
	<i>Copy the data to the CD Server</i>	<i>366</i>
	<i>Burn the CD.....</i>	<i>366</i>
	<i>Label the CD and Mail to NODC.....</i>	<i>367</i>
	<i>Response from the NODC Office.</i>	<i>367</i>
14	APPENDIX FIVE: TSG DATA SUBMISSION PROGRAM	371
14.1	INTRODUCTION AND BACKGROUND	371
14.2	TSG DATA STANDARD.....	371
14.3	THE TSGTRANSMITTER EVENT LOG.....	374
	<i>Template Name</i>	<i>374</i>
	<i>Sensor Metaitems</i>	<i>374</i>
	<i>TsgTransmitter Data Log File</i>	<i>374</i>
14.4	RUNNING THE TSGTRANSMITTER EVENT	375
14.5	IMPORTANT FILES CREATED BY THE EVENT LOGGER	375
14.6	TSG TRANSMITTER OPERATIONS.....	375
15	APPENDIX SIX - RAW DATA SIMULATION	379
15.1	EXTERNAL SIMULATION.....	379
15.2	INTERNAL SIMULATION.....	380

CHAPTER ONE

OVERVIEW

1 OVERVIEW OF THE SCIENTIFIC COMPUTER SYSTEM (SCS)



Preface

Starting with the release of v 4.3 SCS enters the world of database operations. SCS now operates in two modes we'll call DB and Legacy modes.

This Manual for SCS v 4.6 covers SCS Database Mode of Operations. The Legacy mode is not covered since the manual for SCS 4.6.

Note that legacy mode is no longer supported.

SCS Legacy Mode

In legacy mode SCS 4.6 installs and operates just like version 4.2 and prior releases. Legacy mode is the default. In legacy mode:

- SCS v 4.6 behaves exactly like previous releases.
- All template files and configuration files work exactly the same.
- Data logging (ACQ's RAW files and Event Logger's ELG files) are still there.

SCS DB Mode

In DB mode SCS stores sensor configuration in a database. More details showed as below.

Setting the Mode

The mode of operation is controlled in the SCS Configuration File by the SCSDATABASE parameter.

Legacy Mode

You set legacy mode by setting the SCSDATABASE parameter to 0. SCS operates as it did in v 4.2.x and before. All apps having to do with DB operations are removed from the SCSMenu.

DB Mode

You set DB mode by setting SCSDATABASE to 1. A SQL database is now required.

In DB mode ACQ still logs messages to RAW file as it has always done. The DBCONNECTIONSTRING, SCSDATABASELOGGING and SCISOELENABLED parameters now come into play.

- You must set up DBCONNECTION STRING to allow SCS apps access to the SCS SQL Database. See your system administrator for further information.
- If SCSDATABASELOGGING is 0 raw data logging is restricted to RAW files. If it is 1 then raw data is logged both to RAW files and the SQL database.
- If SCSOELENABLED is 0, it is disabled and does not appear on the SCSMenu. If set to 1 the OEL (Operation Event Logger) is enabled.

Life in SCS DB Mode

Essential Features of the SCS Database Design

- The current release was built to work with Microsoft SQL Server 2008 R2, either Standard or Express edition.
- The SCS database schema is multi-ship. That is, it can hold raw data, sensor configurations, event log data and everything else, for a fleet of ships. On each ship of course you will only be storing data for that ship, but shore side databases can store the entire fleet's data and retrieve it by ship.
- Another key concept is the Cruise ID. The administrator updates the cruise ID at the start of a cruise. The ID is stored with the raw messages, and becomes another key to retrieve data by.

Installation

- Installation is exactly the same as for legacy mode. It defaults to legacy mode and you change it after the installation is complete.
- But in addition to SCS you will have a separate installation of the Microsoft *SQL Server 2008 R2 Express Edition*. Refer to Appendix Six, SCS Database Installation, for more details.
- We also include “*SCS Database Supplementary Installation Instruction*” to complete the installation of *SQL Server 2008 R2 Expression Edition*.

Backwards compatibility

- **Data Logging:** Primary and Backup data logging to RAW files take place exactly as before. Logging to the database is done above and beyond the legacy logging activities.
- **Event Logging:** In DB mode the legacy Event Builder and Event Logger are available, and all your old event templates are useable without modification.
- **Sensor Configuration Files:** In DB mode you are allowed a one-time import from your legacy SENSOR.XML file into the database. After that there is no backwards compatibility. Changes to the database sensor configuration cannot be exported back into the legacy SENSOR.SCF file. In this situation your only option is to re-enter your changes manually with the legacy CFE app. Multiple SENSOR.XML files can be imported under different vessel names established by the Cruise Manager app.
- **Template Files** for the various SCS client apps are fully backwards compatible.

1.1 General Information

The Scientific Computer System (SCS) is a data acquisition and display system designed for Oceanographic, Atmospheric, and Fisheries research applications. It acquires sensor data from shipboard oceanographic, atmospheric, and fisheries sensors and provides this information to scientists in real time via text and graphic displays, while simultaneously logging the data to disk for later analysis. SCS also performs quality checks by monitoring I/O, providing delta/range checks and plotting data after acquisition.

VERSION: 4.6

DATE: January 2013

PLATFORMS: Windows 2008 Server, Windows 7, and Windows XP Professional

1.2 System Overview

The Scientific Computer System (SCS) was designed and implemented based on a user requirements study completed in 1986. The first version of SCS was certified in 1989 on the NOAA Ship MALCOLM BALDRIDGE. The general requirements are summarized below:

- Provide real-time data acquisition for an extensive suite of sensors.
- Provide real-time displays and graphics to the scientist.
- Provide real-time quality checks on the data being collected.
- Provide logging of the real time data to disk for data post-processing at sea and in the laboratory environment.

SCS provides this functionality through a set of programs that work in concert to acquire, log and display the data. The system is structured in such a way that each program performs a single function and may be run remotely on client PCs. This simplifies the maintenance of the system while maximizing its flexibility, long-term use, and reliability by isolating display programs from mission critical data collection functions. The programs interconnect using a client/server architecture based on TCP/IP and XML that allows the display and processing programs to run remotely on client computers with the data being provided via sockets and network shares to the acquisition computer.

The ACQ program manages multiple sensors through objects called “*Sensor Device Structures*.” These structures keep track of the RAW Sensor data collected for any given sensor. When a client program needs data it asks the embedded ACQ Server via a network message. The ACQ data server accesses the data from the sensor device structures, formats the data in a standard XML data packet, and then sends the data to the client over the network. The general flow of data through the system is described below and in further detail in section 1.3.

The acquisition program (ACQ) creates sensor device objects for each physical sensor interface supported by the system. These devices independently collect raw data from the attached sensors in real time. ACQ manages and coordinates all of the sensor devices providing data to the internal Data Server when requested. Client applications open socket connections to the ACQ Data Server and request data from ACQ using a standard XML protocol.

The sensor device objects within the ACQ program are also responsible for logging the raw data directly to disk immediately on receipt. This provides a true raw data set containing everything received by the sensors at the log rate requested by the user.

The quality control program (DATAMON) checks all newly acquired logged sensor data against a set of user specified requirements to ensure that the data is within a specific range, within defined delta units, and that the sensor I/O is being successfully completed and logged. The DATAMON program will notify the user with the appropriate error messages and log the errors to disk. This program reads its data to be checked from the raw log files ensuring that the data being validated actually exists.

The real time graphics client programs connect to ACQ and provide a variety of graphic visualizations of the data including; time series plots, XY plots, Trackline Plots, and Gauges.

1.3 SCS Data Flow

The following is a description of the data flow within the SCS system. This description discusses mainly how the data is viewed, logged, and saved for future use. Figure 1 gives a graphic description of the data flow within the Scientific Computing System.

When a sensor device object receives data from a sensor port the message is immediately time-stamped and logged to disk. The sensor device object also stores the data in memory for access by all real time display programs. The ACQ Data Server monitors the network for SCS data requests from client programs and then packages and delivers the data over the network in an XML format. A single client request packet may request multiple pieces of sensor data which improves the overall efficiency of the system and network. The “*Real Time Display*” program display sensor names along with their respective data values in a user configurable format. The “*RT Time Series*” program will plot multiple real-time sensor data against time in graphic form on client PCs. The “*GEOTRACK Track Line*” program will graphically plot the ship’s position, along with latitude and longitude coordinates coastlines, stations and a second position such as an AUV or Net. The “*Send SCS Message*” program outputs data to a remote terminal in an RS232 formatted, comma delimited string.

The raw files will contain the entire message received from the sensor in its original form. The user may define custom log files which contain only key pieces of a sensor message specifically identified in the Sensor Configuration File. The Sensor Configuration File contains all relevant information about the sensors interfaced to the system and is created by the user using the “*Configuration File Editor*”. The “*Event Logger*” (either “*OEL*” or *legacy version*) programs provides custom logging of SCS data in support of real time ship events. The “*DATAMON*” program periodically reads all of the raw log files and the data is checked against specific range, delta, and timeout values specified in the description file. Any error that is detected will be placed in an error log file and it will also show up in “*DATAMON*” error Display window.

Sensor data is logged, in real time, into raw format files (*.RAW) and optionally the database with each message received from a sensor being logged to disk in its entirety. The program provides for custom creation of user-defined ASCII Comma Delimited Files (*.ACO), and data sparse ASCII Space Delimited Files (*.LAB) containing only the critical sensor data extracted from raw sensor messages. These files are generally used for first stage post processing by scientists in standard programs such as Microsoft EXCEL and ACCESS. These custom files are created on demand by the user and are no longer created in real time as in previous version of SCS. Logged RAW data files may be viewed by plotting to either the screen or a printer using the “*Post Processing Time Series*”, and “*GEOTRACK*” Programs. The user has the ability to zoom and pan through a dataset; change graph limits on the fly, and even plot archived data against written data collected by survey staff.

ACQ provides real time redundant logging of the SCS raw data to a second storage device simultaneously. This provides two copies of the data which greatly reduces the risk of data loss and the need for backups. Scientists can access one copy of the data while the second copy is secure.

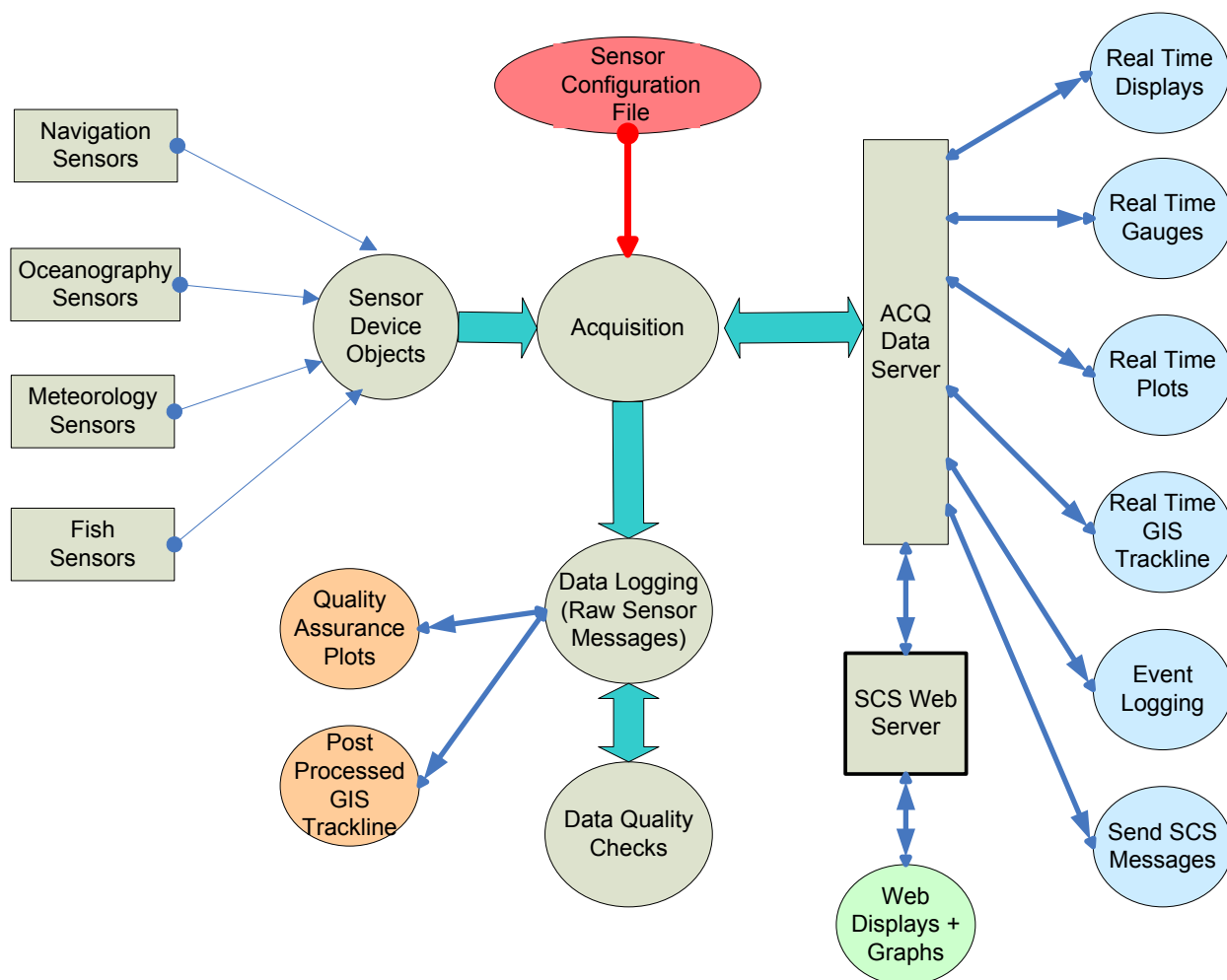


Figure 1 – SCS Data Flow

1.4 SCS Configuration File

The SCS Configuration File SCS40.CFG is a file that contains Configuration Parameters, previously known as Environmental Variables, which allow SCS to operate in many different environments. This file must exist on any PC that runs any SCS application. The file primarily describes directory locations in which specific data and template files will be located (either locally or on a network drive).

The following list outlines the most basic variables used in the system. Note that this list is subject to change, as the SCS configuration file is designed to be highly adaptable to new requirements. This file is edited with the “*SCS Configuration File Editor*” app, started from the SCS Acquisition Menu (Chapter 3).

- SCSAcqEnv is the variable for determining if this machine is actually the ACQ machine responsible for acquiring and logging the data. If this variable is set to 1 then the machine is the ACQ machine. A 0 indicates that the machine is a remote client PC.
- SCSAcqServer is the variable for determining the computer name of the actual ACQ machine responsible for acquiring and logging the data. This variable is ignored if SCSAcqEnv is set to 1. This name may be either a Node Name (i.e. RBNTSCS1) or a TCP/IP address (i.e. 205.156.17.3).
- SCSExe is the variable for the shared folder that contains all of the executable programs used by the SCS system.
- SCSShip is the variable pointing to the directory containing ship specific files, including SENSOR.SCF and calibration files.
- SCSLog is the variable for the shared folder that contains all of the raw SCS files. This is the top level folder containing subfolders for the compress data (COMPRESS) files, event log files (EVENTDATA), and station files.
- SCSLog2 is the variable for the *backup* shared folder that contains a second copy of all of the raw SCS files.
- SCSGIS is the variable pointing to the folder where track-line applications find coastline and shape files.
- ACQSERVERIPADDRESS is the variable specifying the TCP/IP address of the ACQ machine.
- ACQSERVERIPPORT is the variable specifying the TCP/IP port to use to connect to ACQ to request data.
- SCSDATABASE If present and set to “1”, it enables SCS to interface to a database for storage of sensor configuration metadata.
- SCSDATABASELOGGING: If present and set to “1”, and if SCSDATABASE is also set to 1, tells ACQ log all raw sensor data to the database in addition to RAW files.
- SCSOELENABLED: If present and set to 1, enables the Operation Event Logging (OEL) to run and log data to the database. The legacy Event Logger is still available.

- **DBCONNECTIONSTRING:** If **SCSDATABASE** and **SCSDATABASELOGGING** are both set to “1”, it specifies the connection string to a database where SCS stores all sensor data and configuration information.
- **DEFAULTCRUISENAME:** If **SCSDATABASE** is set to “1”, specifies the ship’s cruise identifier.
- **VESSELCODE:** The two-character ship identifier. Used for database operations.
- **ErrorLogToDisk:** If set to 1, ACQ will automatically start logging acquisition metadata to a file in the SCS Logging directory called “SCS4.6_20100830_112233..log” (your name will be slightly different) whenever ACQ is started. If set to 0 logging will not commence at startup. See also the discussion of the “Error Log” in the ACQ sections.
- **DatamonLogLimit:** A numerical value that limits the number of errors being tracked by the Datamon application. See further discussion in Chapter titled “Monitor Errors”.
- **AutoStartLogging:** If set to 1, ACQ will remember the time of the automatic switch of RAW file names occurring every 24 hours. The next time you start ACQ it will remember that time so that you don’t have to keep re-entering it. See the discussion of the “Log Settings” button in the main ACQ window.

Important Notes regarding the SCS40.CFG file:

- Every computer running SCS applications (either as a remote client or the ACQ server) must contain this file in its EXE directory.
- The file can be edited by a text editor, but we strongly recommend you use the SCS Configuration File Editor for editing. It automatically enforces all formatting conventions listed here. See Chapter 3 for details.
- Each line within the file must end in a semicolon (;). Any text after the semicolon is considered a comment and is ignored by SCS software.
- All directory names must end with a backward slash (\).
- The variables names are no longer case-sensitive as of SCS 4.0.
- In the remote PC example above, the X, Y, and Z drives represent mapped network drives linking directly to the ACQ Server (See Chapter 2 - Installation for further details on how to setup a remote client computer for use with SCS)

- A sample *SCS40.CFG* file follows for a primary data acquisition machine:

```
SCSACQSERVER=SCSDEV;
TSG_MAIL_BODY=xxx;
SAMOS_MAIL_SUBJECT=xxx;
TSG_MAIL_SUBJECT=xxx;
```

```

SCSSHIP=C:\SCSServer4.3.3\SHIP40\HQ\;
TSG_MAIL_TO=aoml.tsgdata@noaa.gov;
SHIPTRACKERADDRESS=Shiptracker@noaa.gov;
VESSELCODE=HQ;
SAMOS_MAIL_BODY=xxx;
DEFAULTCRUISENAME=HQ-09-01;
TSG_MAIL_FROM=xxx;
SCSDATABASELOGGING=1;
TSG_EMAILSYSTEMNAME=mail.nems.noaa.gov;
SCSDATABASE=1;
SHIPNAME=HQ;
SCSEXEC=C:\SCSServer4.3.3\EXE40\;
SCSGIS=C:\SCSServer4.3.3\GIS40\;
DBCONNECTIONSTRING=data source=10.49.68.61;initial catalog=scs-jun20;
user id=scs-jun20;persist security nfo=true;
password=+xWACkuOx3A4YkCv3sjwEMkwUkWrnGFDvolqqWk3VDj9DBsyCUNFRQ==;
SCSACQSERVERIPADDRESS=10.49.68.30;
SAMOS_MAIL_TO=SAMOS_DATA@coaps.fsu.edu;
SCSLOG=C:\SCSServer4.3.3\DATALOG40\;
SAMOS_EMAILSYSTEMNAME=xxx;
SAMOS_MAIL_FROM=xxx;
SCSACQENV=1;
SCSACQSERVERIPPORT=505;
ERRORLOGTODISK=0;
DATAMONLOGLIMIT=500;
AUTOSTARTLOGGING=1;

```

- A sample *SCS40.CFG* file follows for a remote client PC:

```

SCSEXEC=X:\SCS4.0_SERVERINSTALL\EXE40\;
SCSACQSERVERIPADDRESS=10.49.5.125;
SCSACQSERVERIPPORT=505;
SCSACQSERVER=SLDC3;
SCSSHIP=Z:\Ship\RONBROWN\;
SCSLOG=Y:\SCS4.0_SERVERINSTALL\DATALOG40\;
SCSGIS=Z:\SCS4.0_SERVERINSTALL\GIS40\;
SCSACQENV=0;

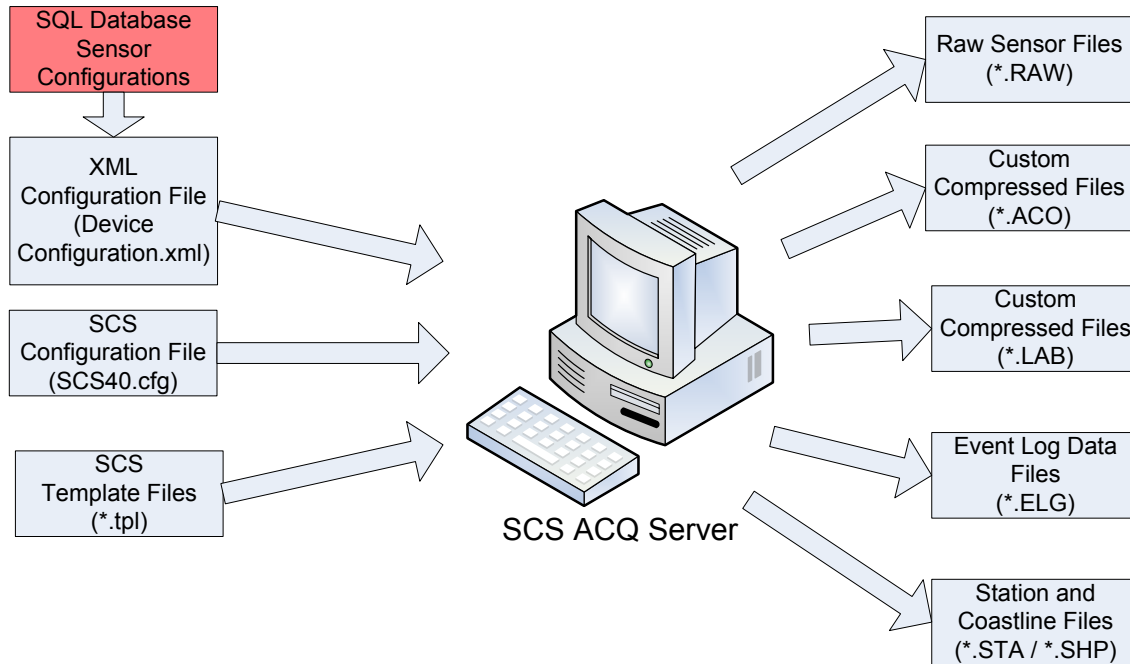
```

1.5 SCS System Files

This diagram illustrates the major types of input and output files used by the various apps in an SCS acquisition server. They are described in detail in the following sections.

System Input Files

System Output Files



Sensor Configuration Files

Sensor Configuration data is the single most important type of data in SCS. SCS terminology defines:

- *Physical Devices* that are the physical sensors in the ship's inventory with metadata such as model and calibration date.
- *Sensor Devices* describe how Physical devices are connected to the ACQ server (through COM ports, TCP sockets, etc). Metadata includes port numbers, baud rate, etc.
- *Message Definitions* are the messages the sensors send out to be acquired. Metadata includes NMEA sentence labels, folder to log the message in, etc.
- *Data Fields* embedded in the messages are the measurements made by the Physical Sensor. Metadata includes position in the message, units of measurement, etc.

Sensor configuration exists in four different places, described below.

SCS Database

The SCS database is the baseline of sensor configuration data and is the archival copy of this data. All XML copies discussed below are derived from the SCS database. The CFE-DB app is used to edit this data.

DeviceConfiguration.XML

CFE-DB writes an XML version of this data to the file *DeviceConfiguration.XML*. Refer to Chapter 3-Acquisition for further information about maintaining Sensor Configuration data with CFE-DB. It is this file that ACQ reads when it starts.

NewSensor.XML

Once ACQ starts and reads DeviceConfiguration.XML, it reformats parts of the contents and rewrites it as *NewSensor.XML* for use by ACQ's client apps. This is another "bridge to the future" file that will disappear once apps are modified to retrieve sensor configuration data directly from the SCS database.

Time Stamped DeviceConfiguration.XML and NewSensor.XML

Because the Sensor Configuration data is critical to the correct interpretation of all incoming data and the resulting computed values, each time the ACQ program is run, copies of the DeviceConfiguration.XML and NewSensor.XML files are saved in the same folder as the logged raw data. The file name is modified by adding a timestamp of the Acquisition Start time. All Raw logged data files names also have this time stamp in their file names, making it easy to associate the XML file with the data files collected.

SCS Template Files

SCS template files allow you to save your SCS development work for later recall. For example a real time series plot template allows you to create a customized display and save display characteristics (for examples, sensors displayed, display colors, axis scales, etc.). Then you can quickly run the time series plot on another computer, load the template and recreate the exact display seen on the first computer.

Display, Plot and Gauge apps all use template files, as does the track-line plot, the message sender, the Event Logger and several other apps. Each app's template has a unique format. While you do not need to be concerned with the exact format of a template file all are ASCII XML files that may be viewed with a text editor.

Note: Editing a template file with a text editor can break a template and is not supported.

Under the <SHIPNAME>\Templates directory is a directory for each app that uses a template file.

Template files are discussed in further detail as needed in the section of the User's Guide that documents the application.

RAW Data Files

ACQ logs messages in their raw format to disk in *raw data files (*.RAW)*. The message is prefixed by a time stamp from the ACQ machine system clock.

RAW files are located in SCSLog logical directory or one of its subdirectories. The optional subdirectory may be specified for a message in the sensor configuration file.

A new set of RAW files is created each and every time Data Acquisition/Data Logger is started, or whenever you execute a “Stop/Start Log” command from within ACQ. Each sensor message has its own associated RAW data file. RAW files names are based on the name of the message, as well as the time stamp of the first data in the file. For example the file PCODE-GPGGA_19980401-123045.RAW contains data from the PCODE-GPGGA message; the first data in the file is at May 01, 1998 at 12:30:45 PM).

RAW files are ASCII text and may in theory be viewed by any text editor. However their large size may cause simple editors such as Notepad or WordPad to crash. It is recommended to use the *Freeware Programmer's File Editor (PFE)* available from the SCS Menu | QA Processing | Read Data Files.

Compress Data Files

These files are generated on demand by the user using the “*SCSCompactor*” program. As opposed to the RAW data files that contain the entire raw parent message, these files contain selected data fields of the message. Also, these files only cover the time period that you select. These files are located in the COMPRESS subdirectory of the SCSLog logical directory, have extensions (*.ACO), and are in ASCII comma delimited format. All compressed files are in plain text ASCII and may be viewed using the Read Data Files program (Chapter 6 – Quality Assurance Processing). Data in these files maybe plotted using the Plot Individual Sensor program (Chapter 6 – Quality Assurance Processing).

The following is the format of these files:

Column #1 = Year, Column #2 = Day number including time of day expressed as a fraction, Column #3 = Day Number, Column #4 = Time of Day expressed as a fraction of a day (e.g. 0.500000 means 12 hours), Columns #5 and following = Data Values.

NOTE: The order of the data values is not given in this file, but is recorded in the associated template file of the same name and extension .TPL in the same directory. These files were originally created years ago due to a limitation in the data load procedure within the COTS graphing application Matlab. Advances in Matlab and general computer power may have made these files unnecessary, but are included in SCS 4.0 for backwards compatibility.

LAB Data Files

These files are generated on demand by the user using the SCSCompactor program, along with the corresponding ACO file. These files are also ASCII and contain exactly the same information as the Compress Data Files (see previous section) except they are space delimited. These files are also located in the SCSLOG/COMPRESS logical directory, but have the extension (*.LAB).

NOTE: The order of the data values is not given in this file, but is recorded in the associated template file of the same name and extension .TPL in the same directory. These files were originally created years ago due to a limitation in the data load procedure within the COTS graphing application Matlab. Advances in Matlab and general computer power may have made these files unnecessary, but are included in SCS 4.0 for backwards compatibility.

Legacy Event Log Data Files

These files are produced whenever the system manager or end-user runs the legacy Event Logger. These files will be located in sub-directories under the SCSLOG\EventData directory (the sub-directories are named exactly by the name of the Event Log template). The extension of these files is *.ELG and each time the Event Log is started for a particular event, a new instance of an Event Log data file will be created.

Also, one header file (*.HDR) is created in the same directory each time an event is run. This file contains user customized metadata about the event and data collected during the event.

(See Chapter 3 – Acquisition, for more detail about the content and structure of these files.)

Station, Track line, Coastline, and SHAPE Files

Station files represent waypoints or other points of interest. You can display these stations using the Geotrack application.

Geotrack can read in coastline data in MAPGEN format and convert it to a Shape File which can be plotted by Geotrack or any other GIS product. Geotrack can convert track line data from a GPS receiver and write it as a Shape File for plotting on another GIS product

(See Chapter 5 – Real Time Displays for further detail about the Shape file format.)

1.6 SCS System Directory Trees

SCS creates four directory trees during installation. The structure of these trees is laid out in the sections that follow.

NOTE: Previous versions of SCS allowed for some variation in the structure of these trees, but SCS 4.0 permanently fixes the structure of all trees. The top level directory names have traditional names which will be used here and which are the default names created by the SCS Installer (chapter 2). You can change these names during installation but it is highly recommended that you use the defaults. These names are recorded in the SCS Configuration File (SCS40.CFG). This file can be edited using the SCS Configuration File editor (Chapter 3).

SCS Installation Directory

The default installation directory is C:\SCSxxx-Server, where the “XXX” represents the release version ID. All SCS software and data trees are located under here, with the exception of certain GIS software (refer to the \Program Files\Common Files\ESRI\ directory below). The following sections describe the subdirectories under the Installation Directory

The Executable Directory

Contains the executable files that are called from the SCS Menu (*.EXE), plus ancillary files (*.DLL) that are required by the EXE files. By default this directory is named EXE40.

Data Log Directory Tree

The default name is Datalog40). The datalog directory tree is where all SCS data is logged. All RAW files logged during data acquisition are written to this directory by default. Other subdirectories will be created as needed to hold RAW files as specified in the SENSOR.SCF configuration file with the Logging Folder parameter (Appendix 1).

- **Compress**
This is where all Compress files (*.ACO, *.LAB) are located, as well as their respective templates (*.TPL).
- **EventData**
The Event Logger creates subdirectories here as needed. The names of these directories correspond to the names of the event templates created by the Event Builder. It is these subdirectories where all the Event Log data files (*.ELG) are located, along with their respective header information metadata files (*.HDR).
- **DataMonLogs**
The Datamon app writes summaries of errors found in data coming into the ACQ app.
- **Reports**
Used by the CFE-DB app to store “Zip Ship” files, cross references files, and other files.
- **SavedGISTracklines**
The Geotrack writes Shape files of track line data (reading from RAW files) as it creates new display layers.
- **JPGS**
Destination for JPEGS images generated by post-processing time-series plot applications.

- Station\

This is where all station files (*.STA) reside. These files are used for the Geotrack track line application within SCS. Whenever Geotrack reads in a Station File it creates a corresponding Shape File that can be read by the ArcView application.

Ship Directory Tree

The default name is Ship40\”ShipName”\ where “ShipName” is a subdirectory of Ship40 named after your ship. The files in this tree generally speaking are those that are inputs to the various SCS applications.

The primary sensor configuration file NEWSSENSOR.XML (in database mode), or SENSOR.SCF and SENSOR.XML (in legacy mode) are located here. Several subdirectory trees are contained in the ship directory

- AcqSimulationData\

RAW files used in conjunction with ACQ simulator mode must be placed in this directory. This is for SCS developer use only.

- Templates\

This directory has subdirectories for all apps that create and use template files. The subdirectories are based on the name of the application.

- ✓ Ship\<ShipName>\Templates\Display\ - Real Time Displays.
- ✓ Ship\<ShipName>\Templates\DualAxisPlotTemplates\ - Real time dual axis graphs.
- ✓ Ship40\<ShipName>\Templates\Gauges\ - Gauge displays
- ✓ Ship40\<ShipName>\Templates\EventTemplates\ - Event Log templates
- ✓ Ship40\<ShipName>\Templates\RealTimeSeriesPlotTemplates\ - Real time Time Series Graphs
- ✓ Ship40\<ShipName>\Templates\TracklineTemplates\ - Real time Trackline templates
- ✓ Ship40\<ShipName>\Templates\PostTimeSeriesPlotTemplates\ - Real time XY templates
- ✓ Ship40\<ShipName>\Templates\XYPlotTemplates\ - Real Time 2Y-Axis Time Series templates
- ✓ Ship40\<ShipName>\Templates\ SendSCS \ - Send SCS Message
- Ship40\<ShipName>\Templates\ PlotAll \ - All Time Series Post Plot template files (*.TPL) reside here. These can be used for the Newspaper Daily plot procedure.
- ✓ Ship40\<ShipName>\Templates\ReportBuilderFiles\ - All Event Log report files (*.BMP, *.FLB, *.RBF, *.RPT).

- SimulatorRawData\

Contains RAW data files used by the Simulation Manager, and other ancillary files. See appendix seven for details

- Sounds\

This directory contains all WAV files required by various alarms in the Event Logger app.

The GIS Tree

The default name is GIS40. The GIS (Geographic Information System) tree contains several subdirectories used to hold various kinds of GIS data files. The following directories are under the GIS tree.

- **ImageFiles**
Contains TIF, TFW and other raster-based data files that can be read in by the Geotrack app.
- **MapgenFiles**
Contains data files in the “Mapgen” format. These are ASCII files that delimit world coastlines. Mapgen files are available for any part of the world at NOAA’s National Oceanographic Data Center (NODC) web site. Mapgen files can be read by Geotrack.
- **ShapeFiles**
This is the default directory for all shape files during an “Add Layer->Shape File” operation in Geotrack. SCS is distributed with three directories having world coastline data at three levels of resolution. They are:
 - ✓ Coastlines-NGDC-HighRes
 - ✓ Coastlines-NGDC-LowRes
 - ✓ Coastlines-NGDC-MediumResOther subdirectories are:
 - ✓ LatLonGrids – Whenever Geotrack constructs a lat-lon grid layer it creates a shape file of the grid in this directory.
- **ShapeFiles\World EEZ**
Contains shap files for the world EEZ in two formats (line and color-filled polygons) outlining the limits of the EEZ for all countries of the world

1.7 Online Menu Help

The first release of SCS 4.0 does not contain a help menu. This feature will be restored in future releases. This document is the primary reference for user help.

CHAPTER TWO

INSTALLING SCS

2 Installing SCS

2.1 Prerequisites

Hardware

For SCS Server you need at least 2 GB free space on the C Drive for temporary storage of files during the installation. This covers the install process only – actual post installation requirements are much less.

It is recommended that your ACQ server's clock speed be at least 2 GHZ.

Operating System

SCS 4.0 is supported under Windows Server 2008, Windows 7, and Windows XP Pro. SCS software is always tested on machines running with all Windows patches applied.

Either SCS (server or client) may be installed on either version of Windows.

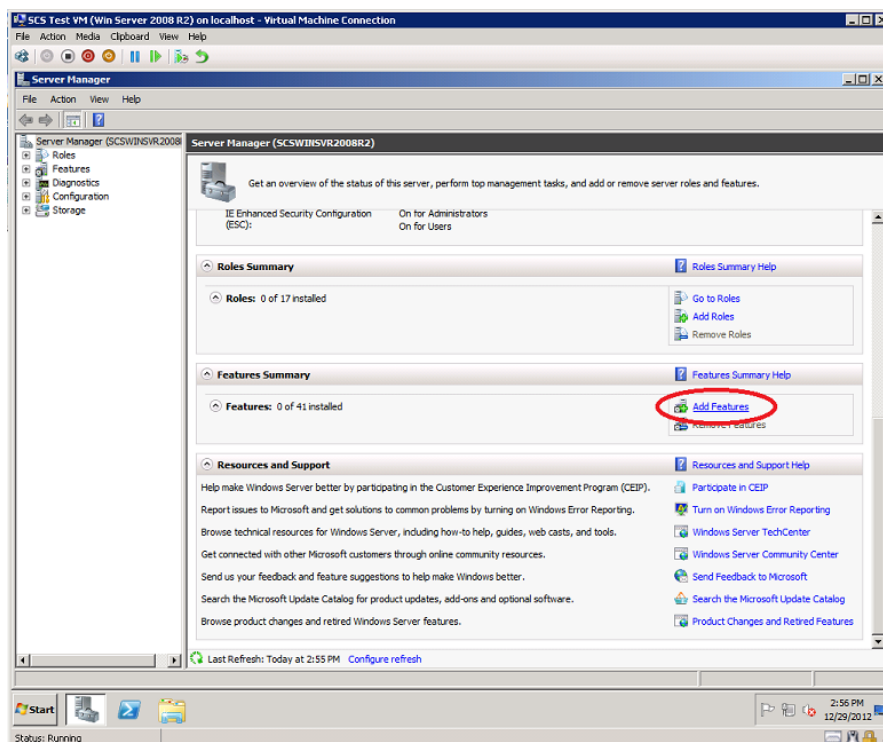
Microsoft .NET Framework

The .NET framework version 3.5 SP1 is required for SCS 4.6. Previous versions may work but SCS 4.6 is not supported under them.

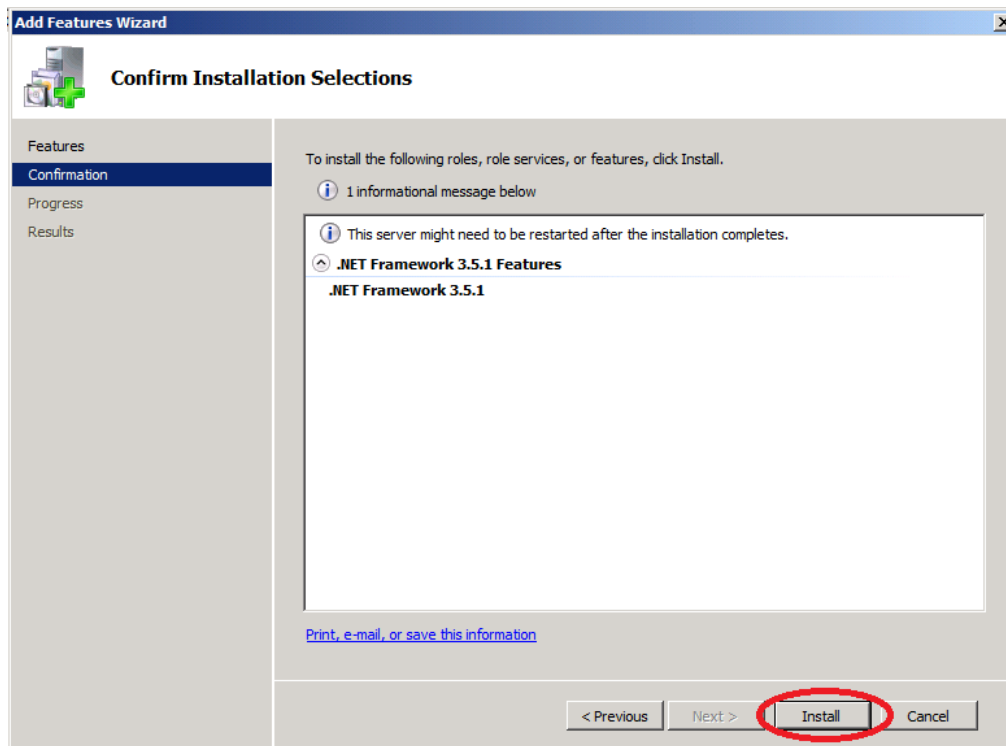
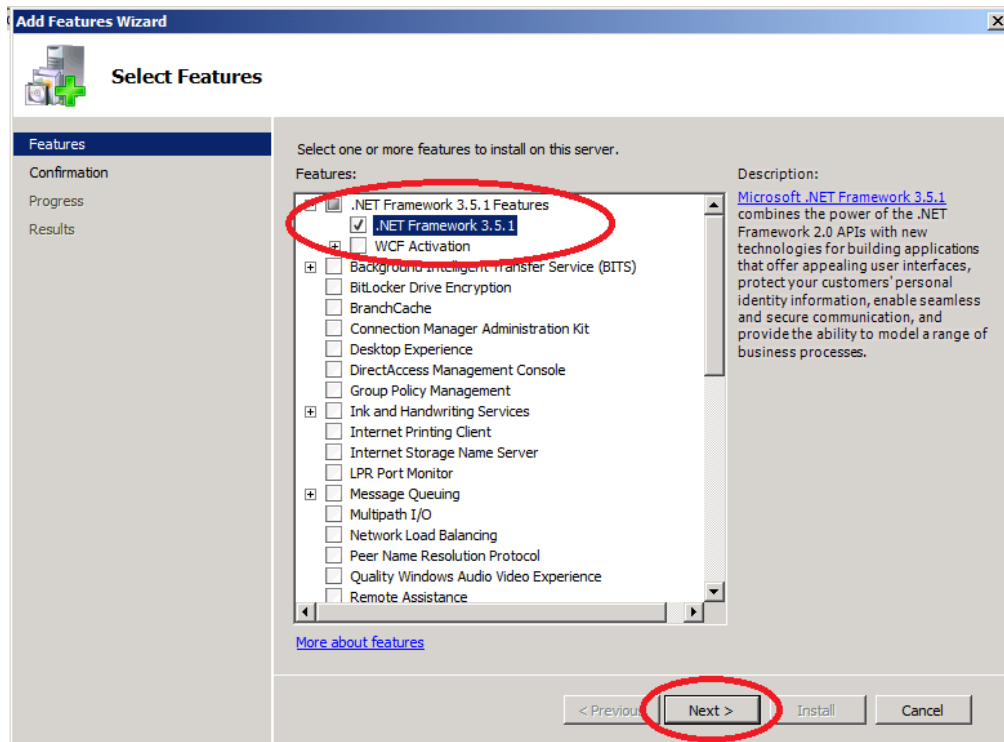
To determine the framework version use the Control Panel, Add and Remove Programs, wait for the list to be populated, and check under “Microsoft .NET Framework”.

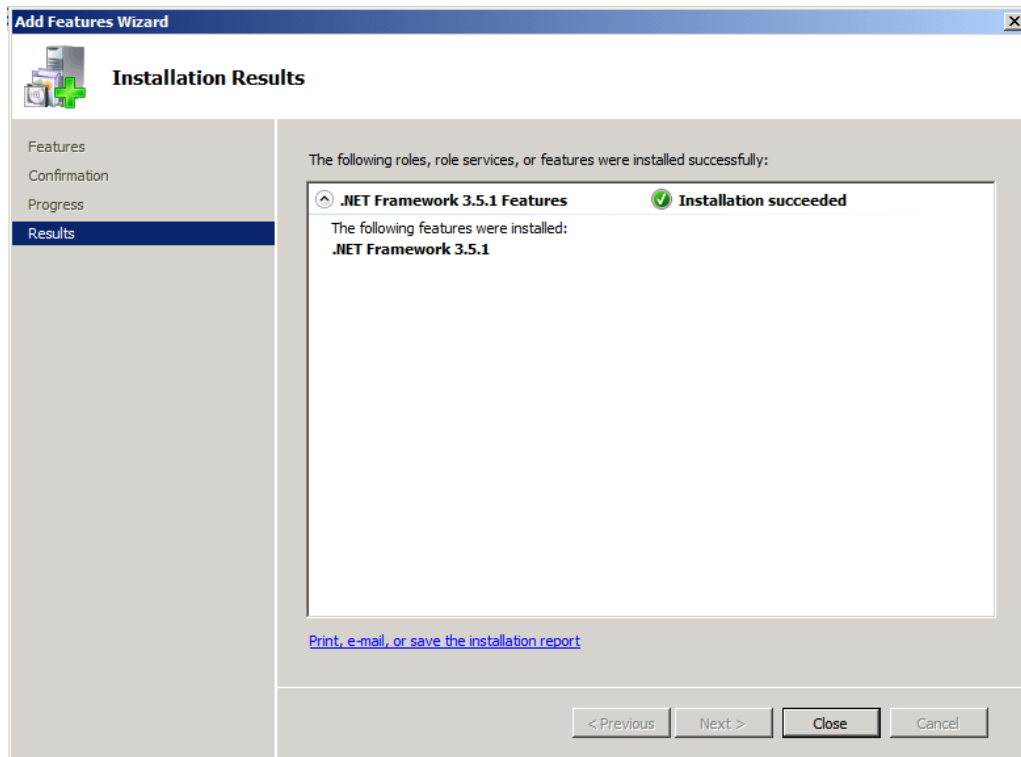
The .NET Framework is an optional Windows module that is available from the Microsoft Windows Update web site. On the Installation CD in the Supplementary Files directory is a Framework installer that you can invoke before beginning the SCS install.

If the installation computer is running Windows Server 2008, installation of the .NET framework version 3.5 SP1 is handled by activating the feature in the Server Manager. Open Server Manager and select *Add Features*.



Check the .NET Framework 3.5.1 checkbox and click *Next*.





SQL Database Operations

If SCS Server is being installed the installation process will automatically install SQL Server Express Edition and the SCS database schema. If you would like to use the full standard edition of SQL server, contact the system administrator for instruction on how to install the database.

Service Packs

All service packs must be installed for the operating system, the .NET Framework, and SQL server software.

Remove Previous Releases of SCS 4.x

If any previous release of SCS 4.x has been installed using the auto installation procedure you must remove it before installing a newer release. Refer to the Uninstall Guide later in this chapter for details.

2.2 Security Information for System Administrators

SCS v4.0 is written in the Microsoft “NET Framework” environment. By default .NET sets up a number of protections that make it more difficult for .NET applications to be malicious.

Installing SCS 4.0 on your computers will install other software. It may also be necessary to change certain Windows security settings on SCS machines. This section covers a number of miscellaneous security issues.

Third-Party Software : ESRI “MapObjects” v 2.4

The SCS Geotrack app is built on a third-party commercial off the shelf software called “Map Objects” by ESRI, a leader in Geographic Information Systems software. During SCS installation a separate ESRI installation package is executed, which loads a collection of DLLs called the “Map Objects Runtime” to the C:\Program Files\Common Files\ESRI directory and registers them in the Windows registry. The ESRI software is required to run the Geotrack application. Running Geotrack requires read/write access to the ESRI directory tree.

The ESRI install package writes an activity log file, C:\Windows\Mo21.log, summarizing the changes to Windows it has made.

SCS and Anti-Virus Software

Several SCS apps use TCP port 25 to send out email messages. If anti-virus software has been installed on the ACQ server, it may prevent these apps from using this port, and it may be necessary to open port 25 in the anti-virus software to enable email.

2.3 The SCS Distribution DVD

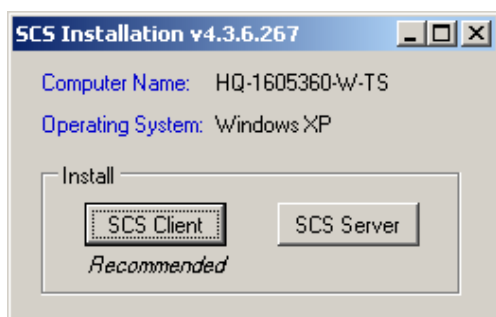
SCS version 4.0 is installed using an automated installation procedure on both the primary SCS acquisition server and all SCS remote clients. Two installation programs are found on the DVD - one for the ACQ Server and one for SCS Clients.

In addition to the software, please note the following contents on the CD in the \Documentations directory

- A Release Notes file containing up-to-the-minute instructions, bug reports, and other useful information.
- A digital copy of the SCS User’s Guide in Adobe PDF format.

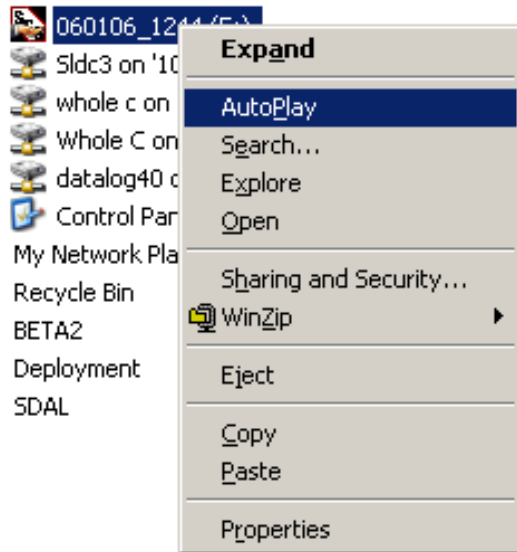
Installation with Auto-Play

The SCS DVD is equipped with the Windows auto-play feature. So, unless you have disabled auto-play, when you insert the DVD into the machine a window will automatically pop up a window that allows you to select which install procedure you want, Server or Client. A sample is shown here:



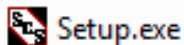
The title bar shows the version of SCS you are about to install. The installer detects the type of Windows you are running (XP in this case) and based on that recommends the SCS client be installed. However, you may override that.

If the auto play does not start when you insert the DVD, open Windows Explorer, right click on the DCD Drive letter, and select “Auto Play” from the popup menu.

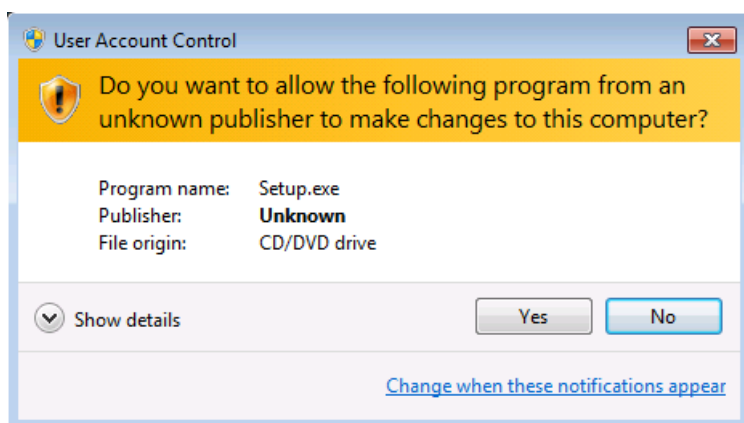


Installation without Auto-Play

If you cannot use the auto-play feature of the DVD you can start the SCS installation by hand. On the root of the SCS Distribution DVD is a **Setup.exe** file. Simply double-click this file to start the installation.



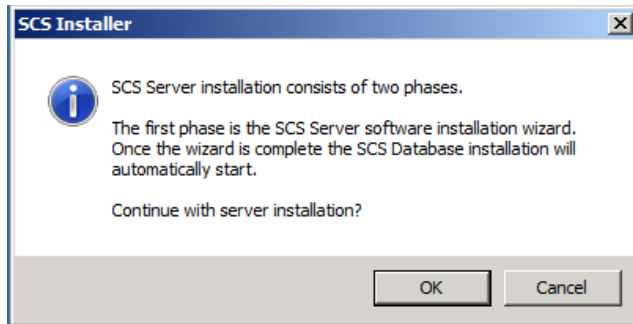
A “User Account Control” message box appears if the user is not logged in as Administrator. Click on “Yes” to continue.



2.4 Installing SCS Server Software

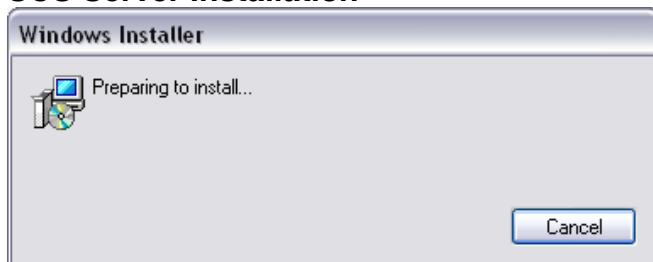
Start the installation Process

After clicking the “SCS Server” button on the initial installation screen the server installation will begin. Note that the SCS Server installation consists of two phases. The first phase is the SCS Server software installation wizard. After the server software installation wizard is complete, the SCS Database installation will automatically start as indicated in this initial informational dialog.

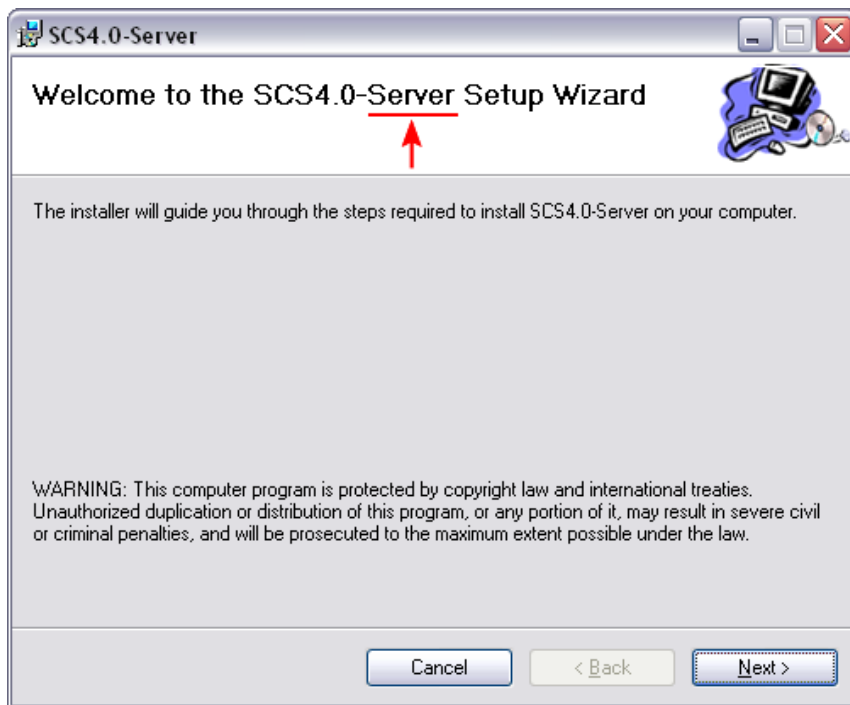


After acknowledging this informational screen by clicking the OK button, the process continues. Depending on the local machine setup, some prerequisites may be installed automatically. To complete these prerequisite installations simply follow the instructions that are presented. Once all automatic prerequisites are installed (if any are required) the installation continues with the SCS Server software installation.

SCS Server Installation

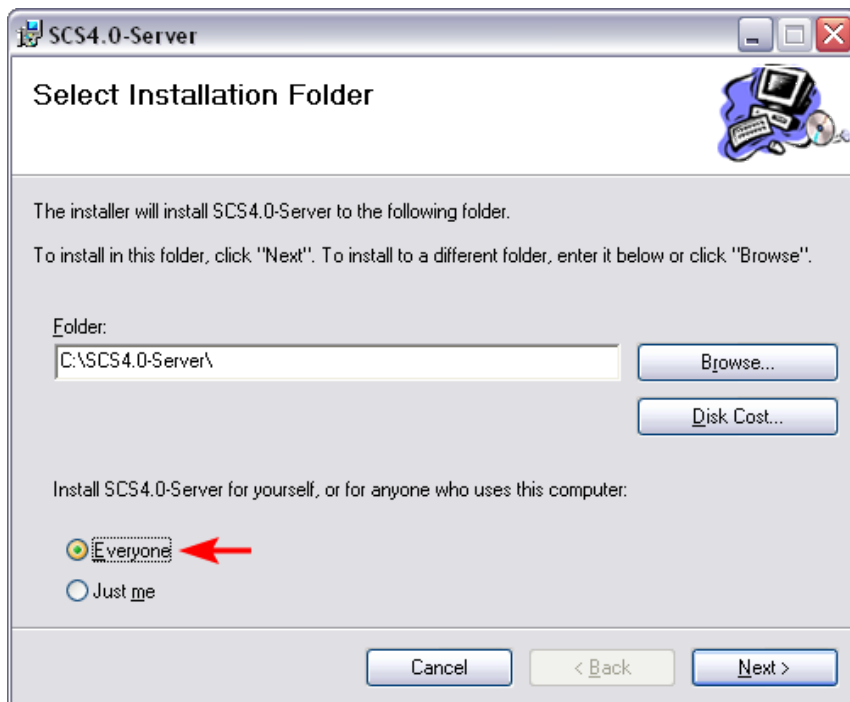


NOTE: It may take a few minutes to get past this “Windows Installer” screen...



NOTE: Please ensure that the title bar references the "Server" install...

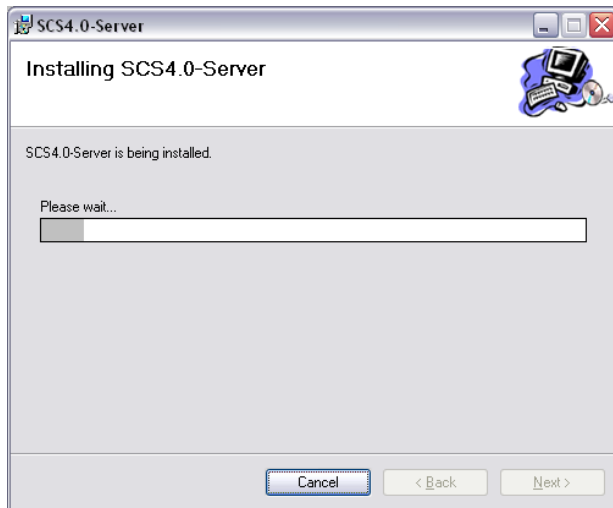
Click next to continue...



Select Installation Directory

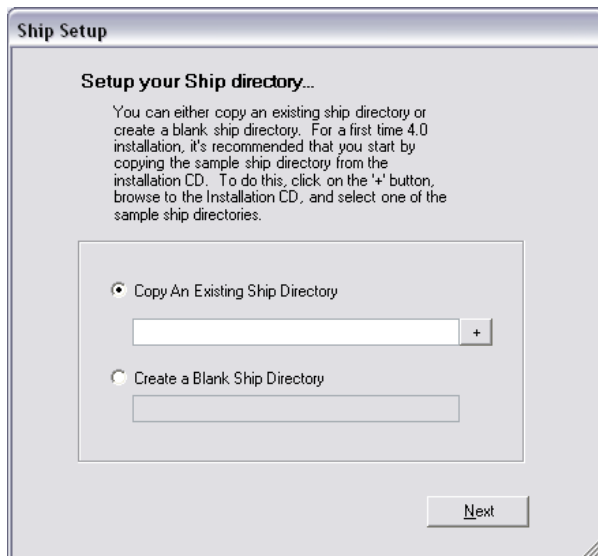
Select the directory you wish to have SCS ACQ Server installed to. You may use the browse button to view/create directories trees. You might want to modify the directory name to include the precise release ID that is shown on the installation DVD.

Also, ensure that the “Everyone” option has been selected. Once you have selected the appropriate directory, hit next to continue...



Set Up Ship Directory

After a few moments of installation, you will be prompted to set up the Ship directory. A dialog will help guide you through this.



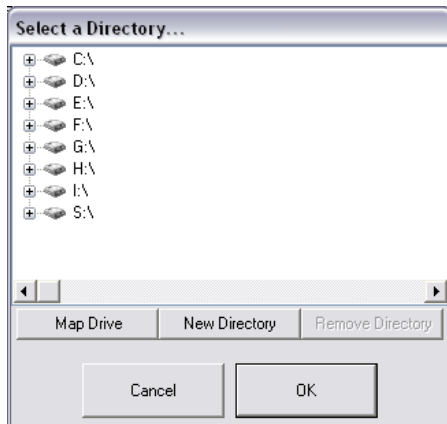
2.4.1 Create Blank Ship Directory

For first time SCS 4.x installation, create a blank ship directory tree. Enter the name of your ship into the text box. This directory and all subdirectories will be created under the SHIP40 directory created during installation.

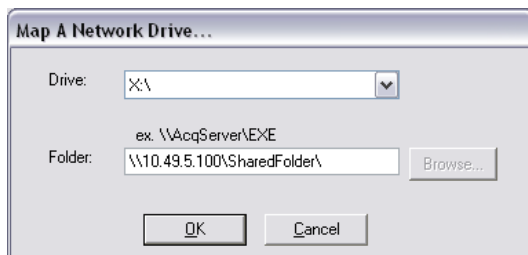
2.4.2 Copy Existing Directory

If you have chosen to copy over an existing ship directory by hitting the plus sign, then you will be prompted to select the directory from which you wish to copy. Do not copy the “Ship40” folder – copy the subfolder named after your ship. You may map drives here, or create new folders.

*** Note: do not copy a ship directory from a SCS 3.x installation – the structure of the directory tree is much different and the files in the tree will be of no use to SCS 4.x.*

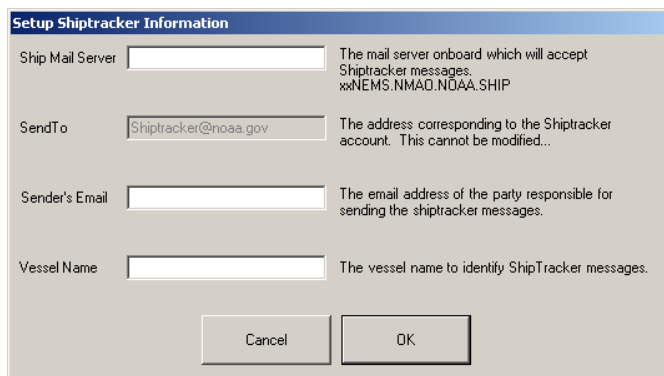


If you want to map a drive, you will be presented with the following dialog. It will allow you to specify the drive letter you want to use and the remote share you want to connect to.



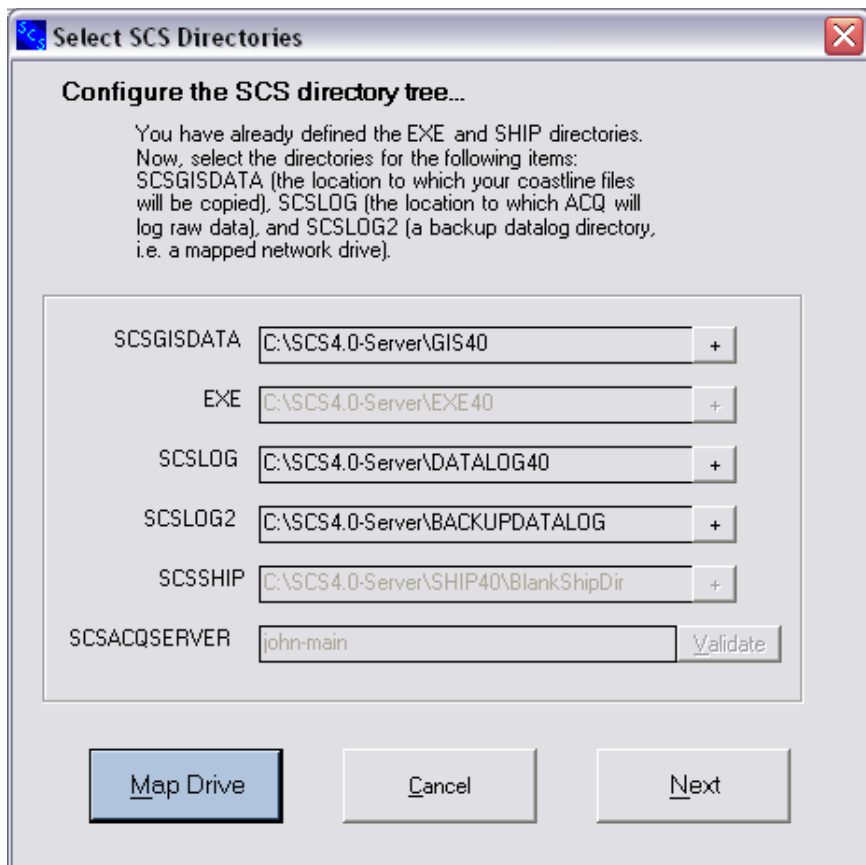
Ship Tracker Setup

The following dialog window appears. Information you enter here is copied to the *SCS40.CFG* file, and is used to construct email messages to the Shiptracker web site maintained by NOAA. Fill in the required information and click on OK.

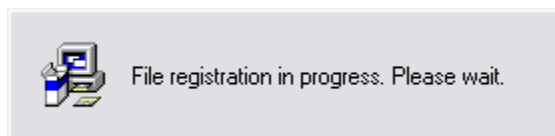
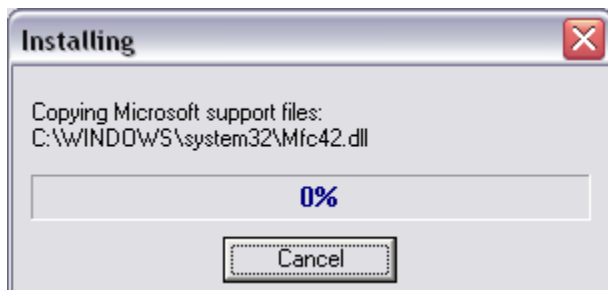


Datalog and GIS Directories

Once you've configured your ship directory you will be allowed to select your Datalog and GIS directories. Please confirm that the value for SCSACQSERVER is the DNS name of the machine you want to use for your ACQ Server (it's the local machine name). You may map drives here also, see above for more information on mapping.



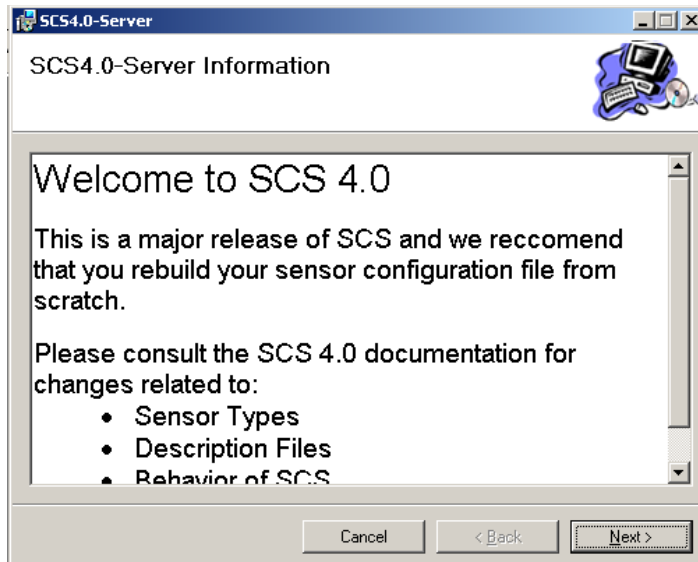
Once you've selected your directories, please hit Next to continue. You may see various additional installation screens, none of which require user input.



Release Notes

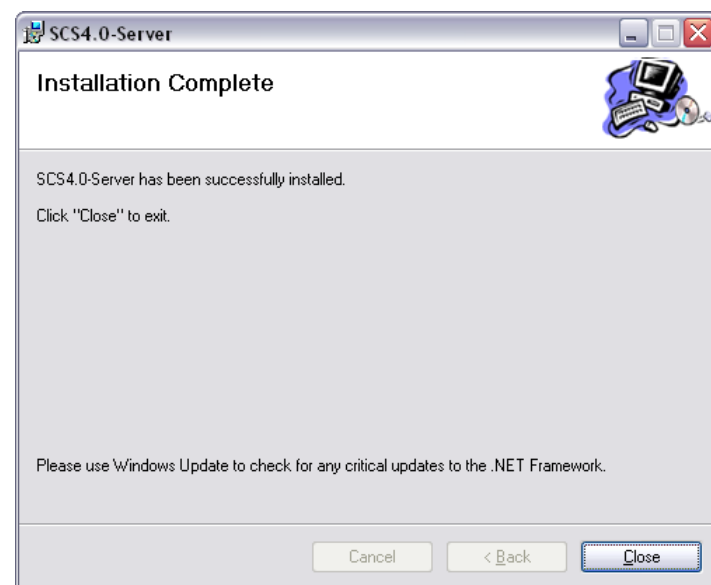
On the DVD in the Documentation directory there is a ReleaseNotes.txt file that contains valuable information on the bug fixes and new features of this version of SCS.

The sample dialog below shows a bulletin displayed by the installer. It contains a general introduction to SCS 4.



Finishing the Server Software Installation

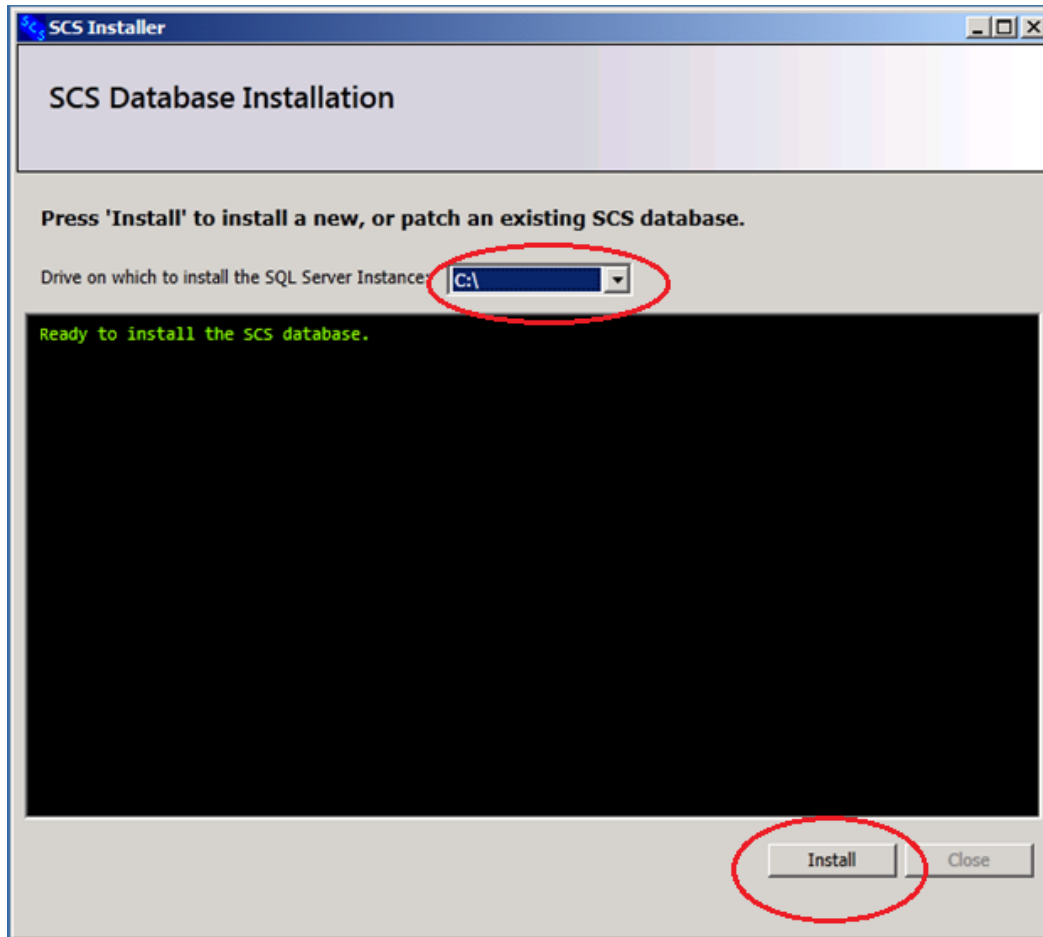
When the server software installation wizard is complete you will be presented with the "Installation Complete" dialog shown below. Click the *Close* button to continue with the SCS Database installation.



NOTE: There may be a slight delay before the database installation starts...

SCS Database Installation

After completing the SCS Software installation by clicking the OK button, the process continues with the SCS database installation.

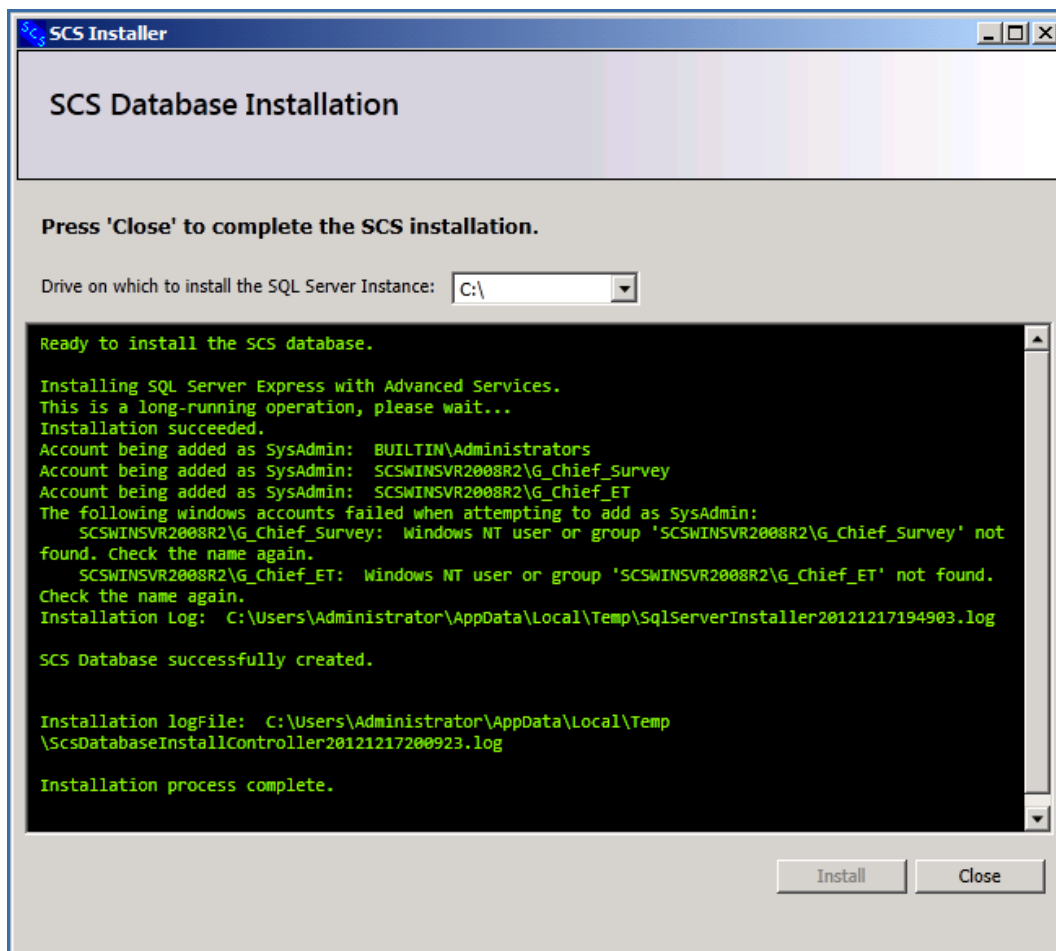


The database will be installed onto the C drive.

Note: If a second fixed drive is available on the computer, installing the database instance to that drive can improve performance.

The database instance includes the files that contain the SCS data. The default is to install both the database software and the database instance files onto the indicated drive. To install the database instance to an alternate drive, select that drive in the “*Drive on which to install the SQL Server Instance*” drop-down box. Then click the *Install* button to continue with the installation.

After clicking the *Install* button, a number of SQL Server installation forms will be displayed to show progress. This database installation can take several minutes, but no user intervention is required or permitted. Once the database installation is complete, the final display entry will be “*Installation process complete.*”



Any errors encountered will be displayed in the output. If the screen does not indicate that the database has been successfully created, please contact your system administrator for assistance. At this point the SCS Server system is installed and the form can be closed.

2.5 Installing SCS Remote Client Software

Client installation is much simpler than ACQ Server installation. The major activities are:

- Install ESRI “Map Objects Runtime” software to the C:\Program Files, Common Files\ESRI directory. A log of this installation activity can be found in the file C:\WINDOWS\MO21.LOG.
- Set up the file *SCS40.CFG* on the client machine.
- Create mapped drives to the data directory trees on the SCS client.
- Copy all application executable files to the client machine on C:\SCS4.6-Client\EXE40.
- Install a shortcut to the SCS Menu app on the remote desktop.

Prerequisites

Please *carefully read section 2.1* in this chapter.

Create Drive Mappings




Create the four mapped drives needed for installation and operation of the remote client machine. The drive names can be any letter, but drives S-V is the SCS default. The default mappings are as follows:

Mapped Drive Letter on Client	Share Name on SCS Server
S:\	GIS40
T:\	EXE40
U:\	LOG40
V:\	SHIP40

Note: If the default drive letters and server shares are unsatisfactory you may change them to suit your needs.

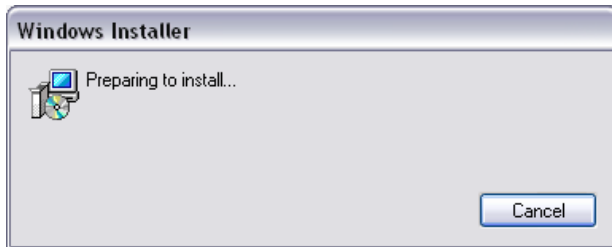
Locate Remote Client Installation Files.

If your DVD is not configured to support Auto-play, open Windows Explorer, open the DVD and proceed to the “\Installation Files\Client” directory.

 InstallRemoteSCS.msi	49,216 KB
 Setup.Exe	108 KB
 Setup.Ini	1 KB

Start the Installation Process

Double click on the Setup.Exe file; this will start the remote installation process.

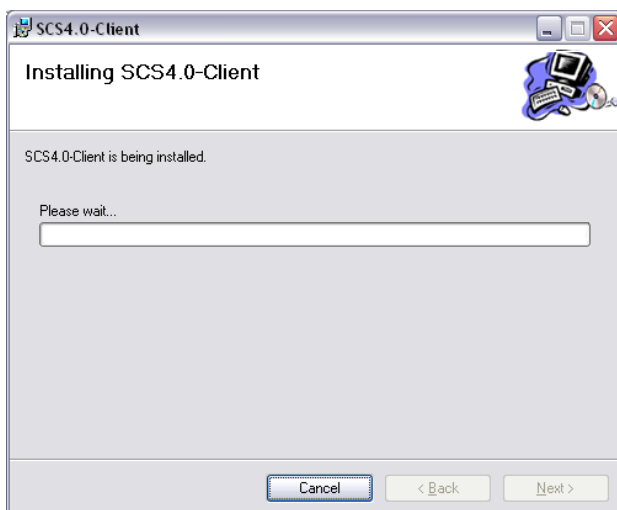


NOTE: It may take a few minutes to get past this “Windows Installer” screen...



NOTE: Please ensure that the title bar references the “Client” install...

Click “Next” twice to begin installation...

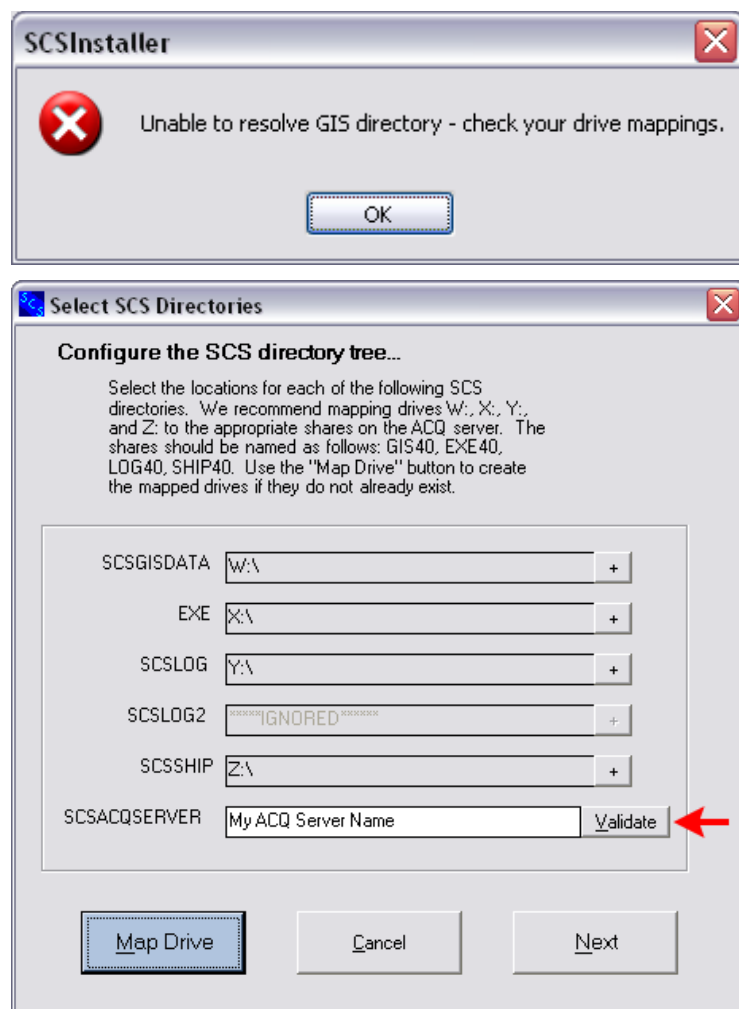


Create Drive Mappings

Once the basic installation has completed, you will be asked to create the drive mappings required to allow the remote client to communicate with the server.

NOTE: Due to difficulties in automating this procedure under all possible circumstances please do not let the installer attempt to map drives. Drives should be mapped before the installation starts as described in the Prerequisites section above.

In the section for SCSACQSERVER, please change *My ACQ Server Name* to the name of the SCS Server. If you type a DNS name, you may click validate to ensure that it resolves to the correct IP for the server. Any errors will be displayed before you are able to continue with the installation...



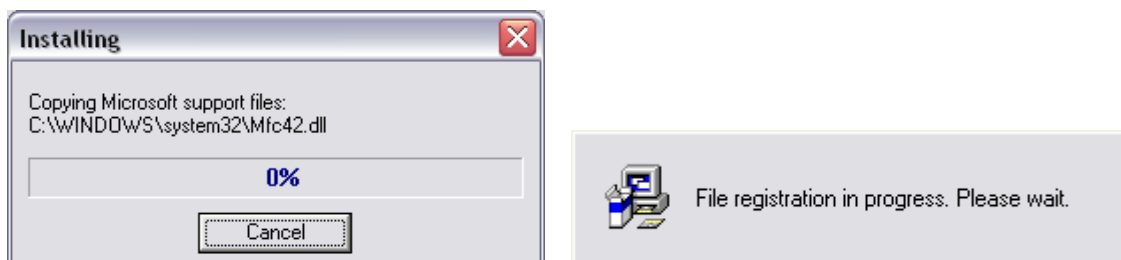
Doing the Installation

During the installation ESRI software will be copied to your machines C:\Program Files\Common Files\ESRI folder.

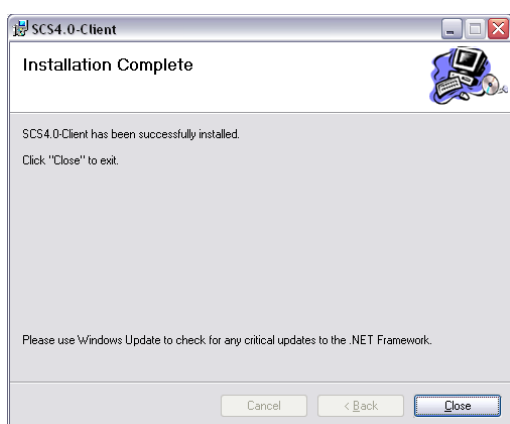
SCS software will be copied from the T Drive mapped in the Prerequisites section above. The software will be copied to C:\SCSClientXXX\EXE40. In this path, "XXX" represents the version number of SCS.

Finishing the Installation

You may see various additional installation screens, none of which require user input.



When all is complete you will be presented with the final dialog. At this point the installer has put a shortcut to the SCS Menu app on your Windows desktop. Use the menu to start up SCS applications. Refer to Chapter three, Acquisition for more information on SCS Menu.



2.6 Post Installation Tasks

Un-map T Drive (Remote Client)

This is optional. The Remote Client installer copies all executable files from the T: drive that you map to the ACQ server into a folder called C:\SCSClientXXX\EXE40, and after installation all SCS software is run from there. Therefore the T drive mapping is no longer needed by SCS.

Set Windows Characteristics (Server and Remote Client)

In order to prevent the system from stopping operations on Remote Client computers (e.g. Real Time Plots), you must set all Hibernation and System Standby options to “Never” or “Off”. Depending on the version of Windows in use, there may be more than one Windows applet where this needs to be done.

Modify McAfee Virus Settings (Server and Remote Client)

To send email out using the Event Log Mailer app, you must enable mail output from your SCS server on port 25.

1. Right Click on McAfee shield in task bar, select Virus Scan Console.
2. Double-Click Access Protection
3. Uncheck “Prevent Mass Mailings...”

Installing a Backup Data Log Directory (Server)

If you are going to set up backup data logging:

1. Map a drive to the backup logging directory on your backup server
2. Run SCS Configuration Editor from SCS Menu->Acquisition
3. In the editor, Add Parameter named SCSLOG2 and set it equal to your mapped drive.

Edit SCS Configuration File (Server and Remote Client)

(Client) It is important that the SCSDATABASE, SCSDATABASELOGGING, and SCISOELENABLE parameter values match those in the ACQ server's SCS Configuration file. The value of SCSDATABASEK especially is important because it causes client apps (such as display and charting) to look for the sensor configuration in different places. If set to 0 clients look to SENSOR.XML, and if set to 1 to NEWSENSOR.XML.

(Server) Make sure the values of SCSDATABASE, SCSDATABASELOGGING and SCISOELENABLE parameters are set to correctly define your intended operating mode (legacy or database) Refer to Chapter One, Overview for more information.

(Server) SAMO data must be mailed to SAMOS_DATA@coaps.fsu.edu. Because of fleet security policies it is impossible for this app to send email outside the noaa.gov domain. Therefore ships of the NOAA fleet should change the value of SAMOS_TO to SamosShip.SCS@noaa.gov, a special account that simply forwards the SAMOS data on to the to the COAPS office for processing.

Install SQL Database (Server)

If you intend to use SCS in database mode, you must install a SQL database. Appendix Six has an outline to install Microsoft SQL Server or SQL Express, the standards for the NOAA fleet.

2.7 Uninstall Guide

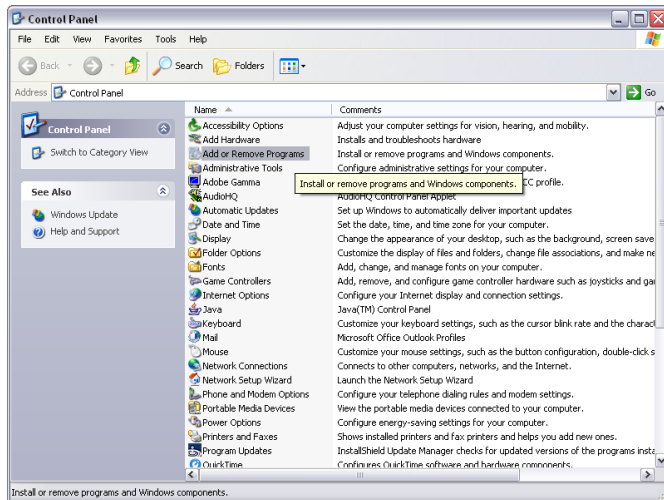
Previous SCS version 4.0 installations should be removed from the system prior to installation of any other versions. The SCS uninstall procedure is as simple and straightforward as the install for both the Server and all remote clients.

Primary Acquisition Server

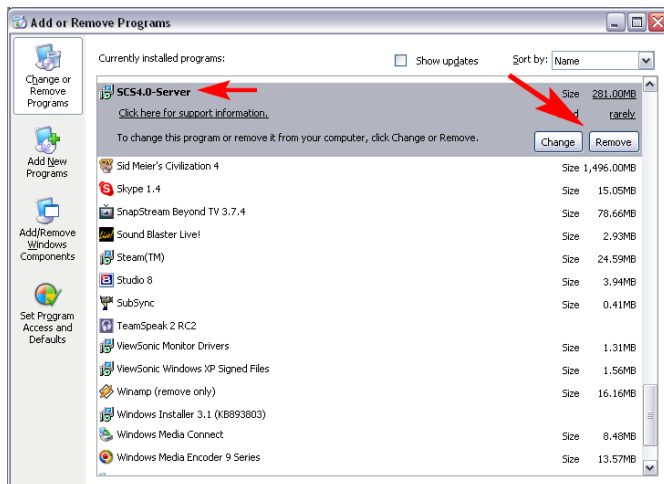
Please remove all connections from remote clients before uninstalling the Primary Acquisition Server instance. This will ensure a safe shutdown and will aid in the ease of the uninstall process.

2.7.1 Execute Windows Uninstall

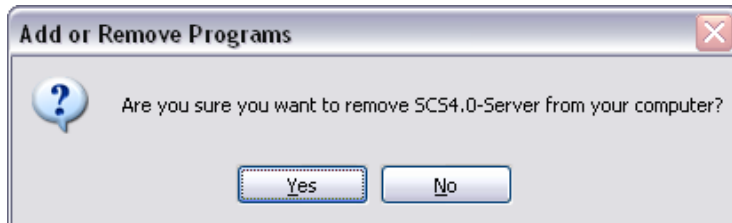
Control Panel → Add Remove Programs



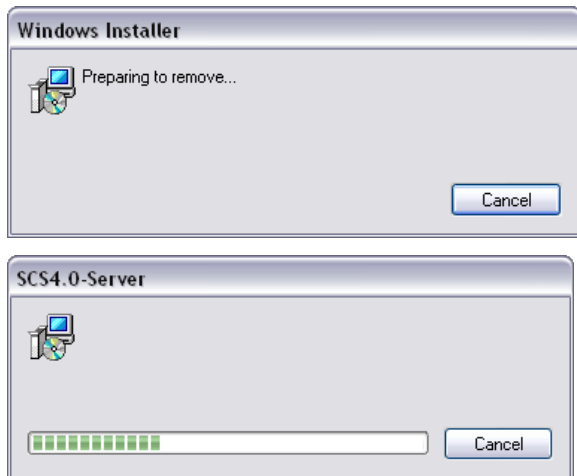
Locate the *SCS Server* installation in the list of installed applications. Once you have found the software (and clicked on it) click on the *Remove* button.



After clicking on *Remove* a prompt will come up asking for confirmation. Click *Yes*.



The following dialogs will appear.



Once these dialogs have completed, SCS Server should be removed from the system. Please ensure that the software is no longer shown in the Add/Remove Programs listing.

2.7.2 Delete GIS Software

For final cleanup, please delete the entire ESRI directory (this is a 3rd party tool used for some graphic applications). This can be located at *C:\Program Files\Common Files\ESRI*.

Do NOT delete the *Common Files* folder; only delete the ESRI subdirectory located in that folder! Also, you may delete the files *ESRI*.TIF* from the *C:\Windows\Fonts* folder.

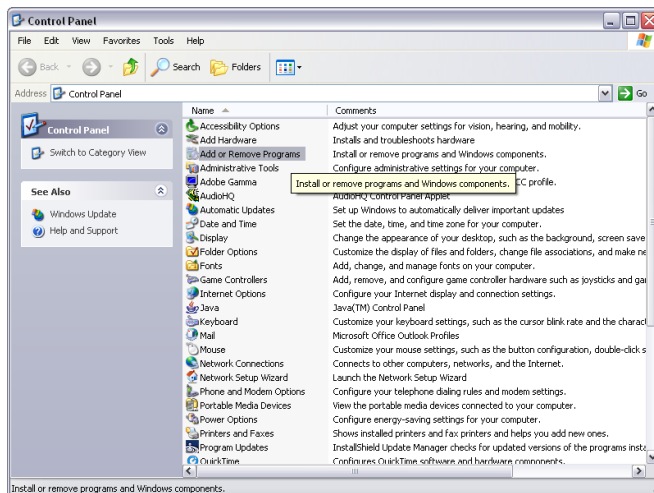
2.7.3 Delete SCS Configuration File

The SCS40.cfg file has been left on your C:\ directory in case it is needed for future reference. You may delete it if it is no longer needed.

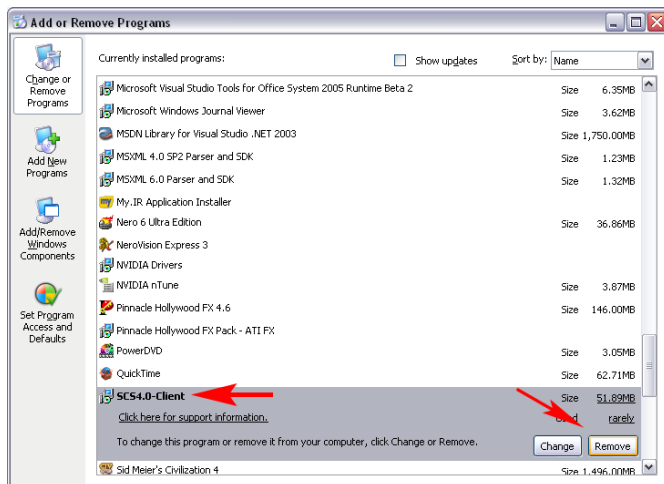
Remote SCS Clients

2.7.4 Execute Windows Uninstall

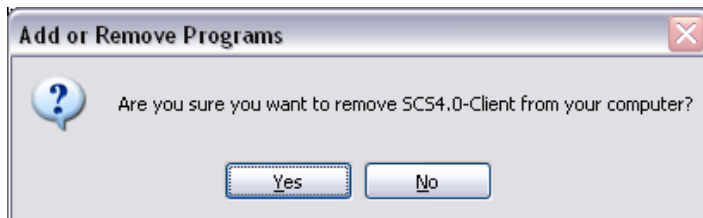
1. Open the Windows Add/Remove programs dialog Control Panel → Add Remove Programs



2. Locate the *SCS Client* installation in the list of installed applications. Once you have found the software (and clicked on it) click on the *Remove* button.



After clicking on *Remove* a prompt will come up asking for confirmation. Click *Yes*.



The following dialogs will appear.



Once these dialogs have completed, SCS Client should be removed from the system. Please ensure that the software is no longer shown in the Add/Remove Programs listing.

2.7.5 Delete GIS Software

For final cleanup, please delete the entire ESRI directory (this is a 3rd party tool used for some graphic applications). This can be located at *C:\Program Files\Common Files\ESRI*. Do NOT delete the *Common Files* folder; only delete the ESRI subdirectory located in that folder!

2.7.6 Delete SCS Configuration File

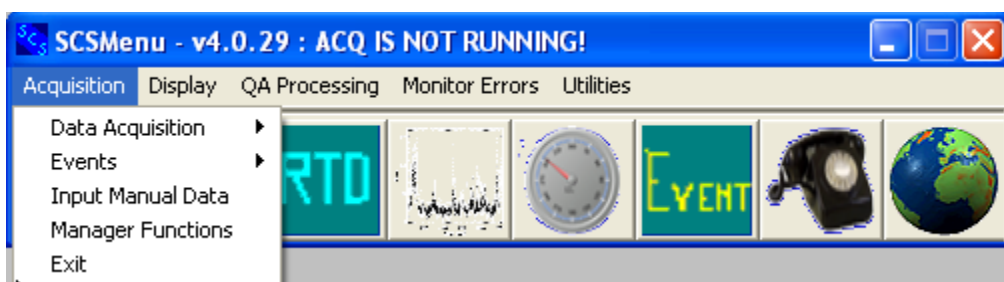
The SCS40.cfg file has been left on your C:\ directory in case it is needed for future reference. You may delete it if it is no longer needed.

CHAPTER THREE

DATA ACQUISITION

3 DATA ACQUISITION

The Acquisition menu covers activities needed to set up data acquisition operations, and then to acquire and log sensor data. The logged data is then readily available for use by other programs that display, plot, or otherwise manipulate the data in either real-time or post-processing applications.



3.1 Starting SCS with the SCS Menu



The SCS Installer places a shortcut to the SCSMenu app on your desktop. This app provides the gateway to all other SCS apps. The functions of SCSMenu are:

- To provide convenient access to SCS apps, i.e. to relieve the user from having to double-click on exe files in the EXE40 folder, and
- To permit or deny access to certain applications based on the values of the SCSACQENV, SCSDATABASE and SCSEOELENABLE parameters in the SCS configuration file.

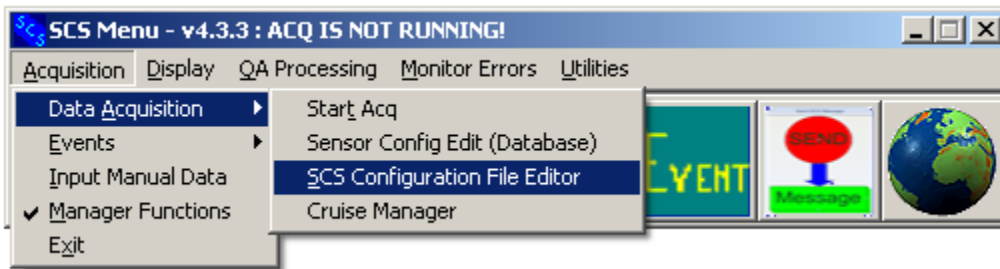
NOTES:

The SCSMenu is independent of any and all SCS apps that may be running. SCSMenu may be stopped and restarted at any time without affecting the currently running apps. To restart the menu, double click on the SCS Menu Shortcut or run the program from the SCSEXEC directory. Re-enter the password in Manager Functions if necessary.

It is possible to start multiple copies of SCSMenu. This usually happens by accident. There is no harm to anything. In this case either one could be used to start other apps.

Simply stop one of them and reenter the management password if necessary.

3.2 SCS Configuration Editor

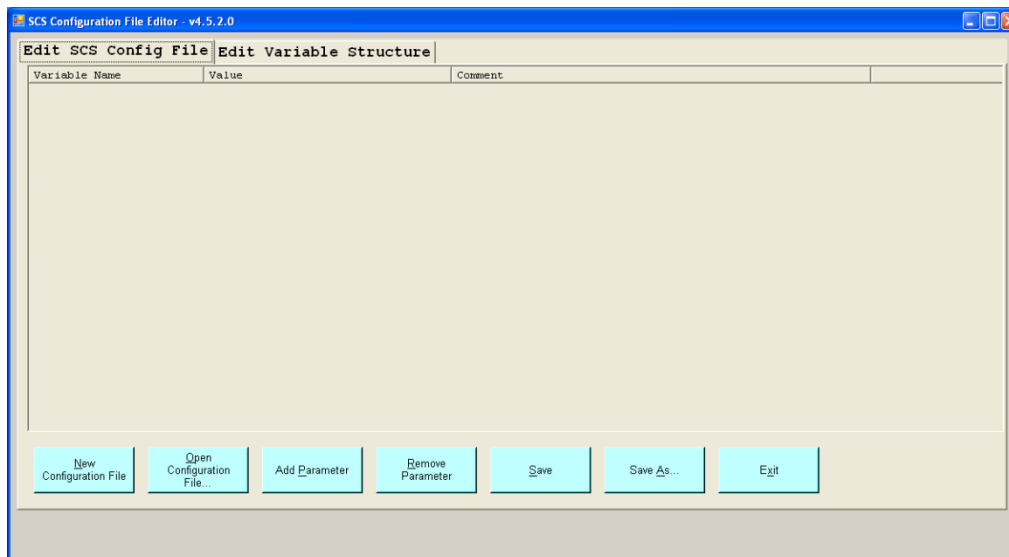


Use this program to edit the *SCS40.CFG* file (see Chapter One for a full discussion of its contents.) This file is created and set up automatically based on your input to the Setup Wizard. After that it typically does not need editing except in special circumstances.

This is an ASCII file with roughly the same structure as in SCS 3, but with some significant differences. There is nothing to stop you from editing this file with any text editor, but it is highly recommended not to. Unintended consequences to manual change may render SCS useless.

Running the SCS Configuration Program Editor

The editor is available from the SCSMenu --> Acquisition menu. The initial window is shown below.

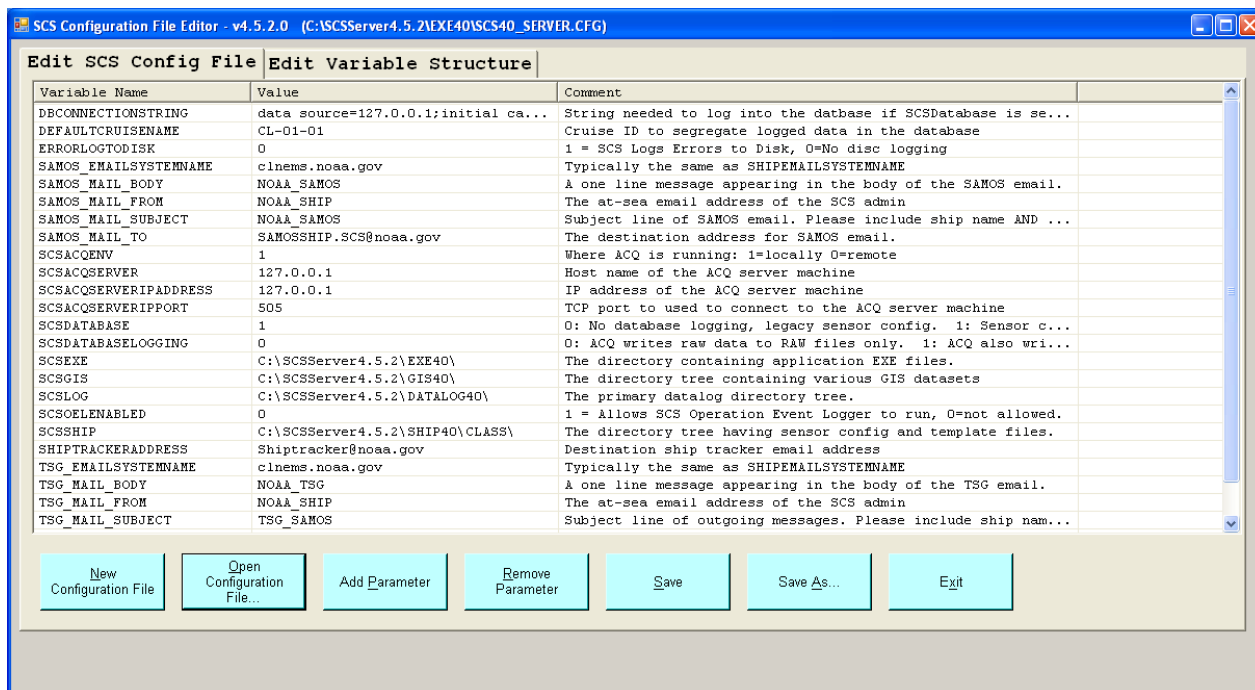


The window is a big tab sheet with two tab selections, handling the two basic editing activities for this file.

1. Editing the contents of the file. This is the default activity, and the one used by administrators except in very special and rare cases.
2. Creating new parameter names for inclusion in the file, activated by clicking on the “Edit Parameter Structure” tab. You would use this only under the direction of OMAO Software Engineering Group.

Editing the File Contents

Begin by clicking on the Open Configuration File Button at the bottom. The default file name to open is C:\SCS40.CFG, but you can work with any file name you want by changing the suggested name. Press Open to display the file. A file opened for editing is shown below.



To change the value of a parameter, double-click on it. This will pop up a dialog box to allow you to enter the new value. There are four kinds of parameters:

1. **String-valued:** the general purpose kind. The editor does not do any checks on your input to verify correctness. An example is SCSACQSERVER, which is the network host name of the acquisition machine and data server.
2. **List-valued.** A parameter that can take one of a limited number of values. For example, SCSACQENV by definition only has two possible values, 0 and 1. Editing this parameter will display a dropdown list showing all legal values, and you must select one of the list entries. Selecting from this list minimizes the chances of your making a typographical error.
3. **File-valued.** Some parameters are used to define directories where SCS keeps its files, for example SCSEXE. The editor will not allow you to type in a directory path; you must click the Browse... button and make a selection from the list of directories. The selection box also allows you to create new directories on the fly. Again, this method minimizes the chance of your making a typo.
4. **Database Connection.** Sets up your connection to the SQL database for accessing sensor configuration, raw data, etc.

You must enter the database server IP address, the name of the database, the user login name and password. Only SQL authentication is allowed. The password you enter is not displayed and is encrypted before being saved to the configuration file. The “Instance” text box is only used if you are running a SQL Express database. In that case the DB installer specified an instance name during the installation, and that name must be included here.

Adding a New Parameter

Click the Add Parameter button to add a new parameter. This is typically not necessary as the SCS Install Wizard sets up the configuration file with all necessary parameters. A list of available parameters is displayed. Pick one and define its value.

The list of available parameters is a pre-defined list created by the SCS installer. You can define your own parameter names and add them to the list by editing the parameter structure, described in the next section.

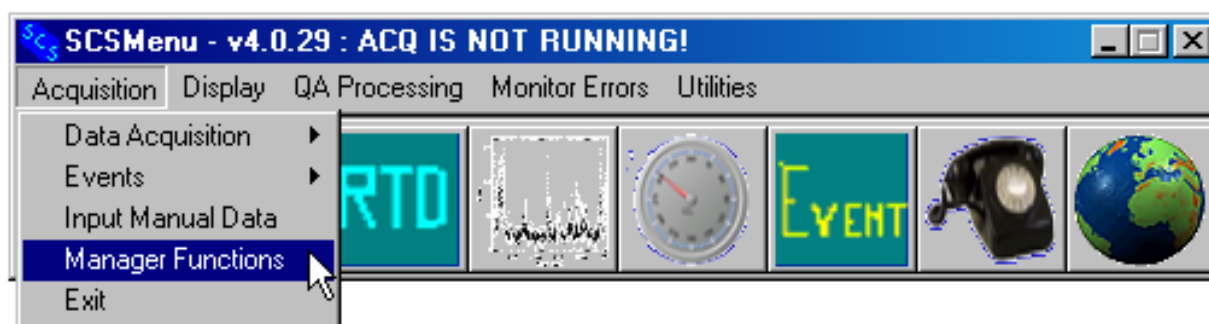
Creating a New Parameter

The editor’s **Edit Variable Structure** tab allows you to create new parameter definitions.

Normally there is never any need to do this; the SCS Setup wizard creates all parameters needed by any SCS application. Non-NOAA users may want to create one for a custom app that they are writing.

Parameter definitions are contained in `C:\SCSConfigurationTemplate.XML`. This file is created and populated by the SCS installer. Changes made in the list with the editor are saved in the file, but the file is overwritten by the installer at the next install.

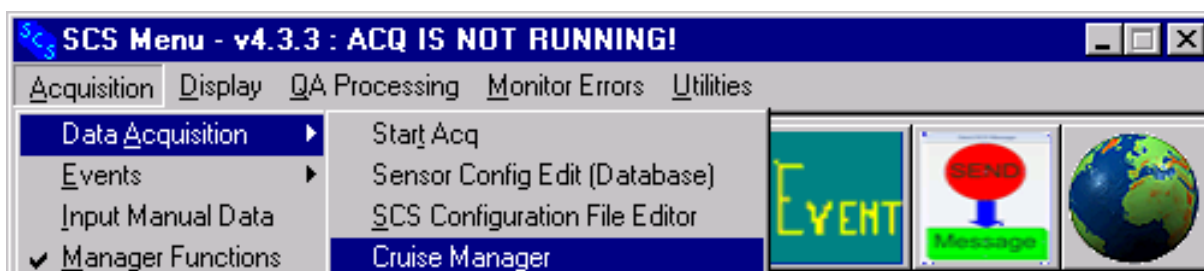
3.3 Obtaining SCS Manager Status



In order to start or stop ACQ, or configure any sensors, you must have SCS manager status. Select Manager Functions from the Acquisition menu and enter the password.

The password is *kkool* (case sensitive). It is hard-coded into the SCS software and cannot be changed.

3.4 SCS Cruise Manager



The SCS Cruise Manager app is used to enter metadata about vessels and cruises into the SCS database's VESSELS and CRUISES tables.

Also, the Cruise Manager tells SCS where to read and write data (such as raw data) to the SCS database. Using this app you set your ship ID and Cruise Name,. All data SCS writes to the data base is segregated by Ship ID and Cruise Name for easy retrieval later.

The VESSELS Tab

Execute the app and the following GUI appears. The default view is of the Vessel List.

The screenshot shows the 'Cruise Manager - v4.5.2.0' application window. The 'Vessels' tab is selected, and the 'Vessels' section is active. It features four buttons: 'Add...', 'Edit...', 'Delete', and 'Make Current'. Below these buttons is a table with the following data:

Vessel ID	Ship ID	Current	Center Vessel ID	Center Name	Owner Name
26	CL	*	OMAO	OMAO	OMAO

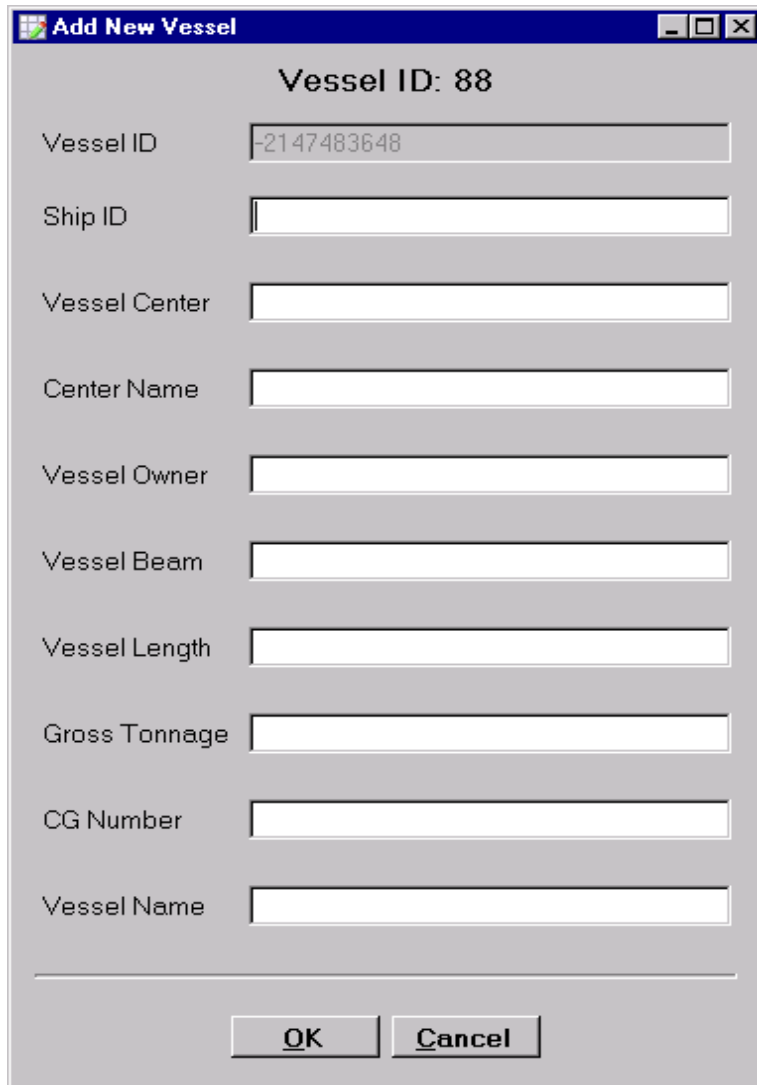
At the bottom of the window is a red 'Save' button.

The SCS database is fleet-oriented. That is, the database can store all data for multiple ships simultaneously. By selecting one ship as “current” and saving the result you make available data for that ship.

NOTE: Even though the SCS database can carry configurations and data for several ships at once, on NOAA ships at sea this feature will not be used. That is, every ship will have its own entry in the vessel table, but no others.

3.4.1 Adding a Vessel

You add a vessel to the list by pressing the “Add...” button. The following dialog pops up



The image shows a Windows-style dialog box titled "Add New Vessel". At the top, it displays "Vessel ID: 88". Below this, there are ten input fields, each with a label to its left: "Vessel ID" (containing "-2147483648"), "Ship ID", "Vessel Center", "Center Name", "Vessel Owner", "Vessel Beam", "Vessel Length", "Gross Tonnage", "CG Number", and "Vessel Name". At the bottom of the dialog are two buttons: "OK" and "Cancel".

The *Vessel ID* is assigned automatically by SCS and cannot be modified. (It is the primary key to the SQL database VESSELS table.)

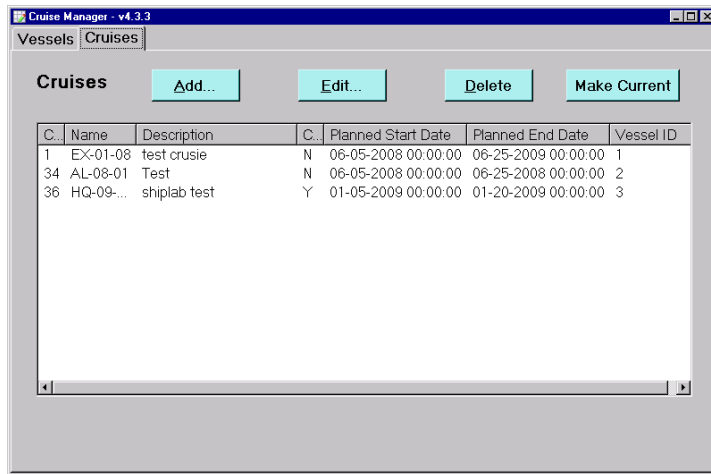
The other items are metadata, stored in the database Vessels table but reserved for future use.

3.4.2 Making a Vessel “Current”

The most important entry is for *Ship ID*. It has the two-letter code for the ship name that is copied to the SCS Configuration File. The SCS database may have several names in it (such as IDs for testing or IDs from other ships. The *Current Vessel* has an important place in the SCS database. Also, CFE-DB will bring up the sensor configuration data for the *current vessel*. When you declare a ship ID as current, the value is copied into the SCS Configuration File SCS40.CFG. It is this value that CFE-DB uses to display

The Cruises Tab

Click on the *Cruises* tab and you will see the list of cruises that have been defined.



To add another click on the *Add...* button and the following popup appears.

Add Cruise

Cruise ID: -2147483648

Cruise Name:

Description:

Planned Start Time:

Planned End Time:

Vessel ID:

OK Cancel

- The *Cruise ID* cannot be changed by the user, as it is the primary key in the CRUISES table.
- The *Cruise Name* must be in the NOAA fleet standard format: two characters, hyphen, two digits, hyphen, and two digits. The first group of characters specifies the ship ID, the second group are the last two digits of the year, and the third group the cruise number for that year.
- *Description* is freeform text.
- *Planned start and end dates* are entered in the format shown in the above figure. Times are optional and default to 00:00:00.
- *Vessel ID* comes from the desired vessel in the Vessels Tab. Note that is the numeric Vessel ID, not the Ship Name or Vessel Name.

3.4.3 Making a Cruise Current

Click on a cruise in the list then click the *Make Current* Button. Remember that one and only one cruise can be current. The current cruise is used together with the Current Ship ID to segregate raw data by ship and cruise.

Managing Cruises

As mentioned before a NOAA ship will have only a single entry in the vessel table, for itself. But it will have an entry for each cruise it sails on. Also, the cruise list can extend over multiple years.

It is important to update the “current cruise” when one cruise ends and the next one starts. When SCS logs raw data to the database it will record the current cruise with each message. This can then be used to easily extract raw data from the database after the cruise is complete.

3.5 Sensor Configuration Files and the Configuration File Editor

What is Sensor Configuration?

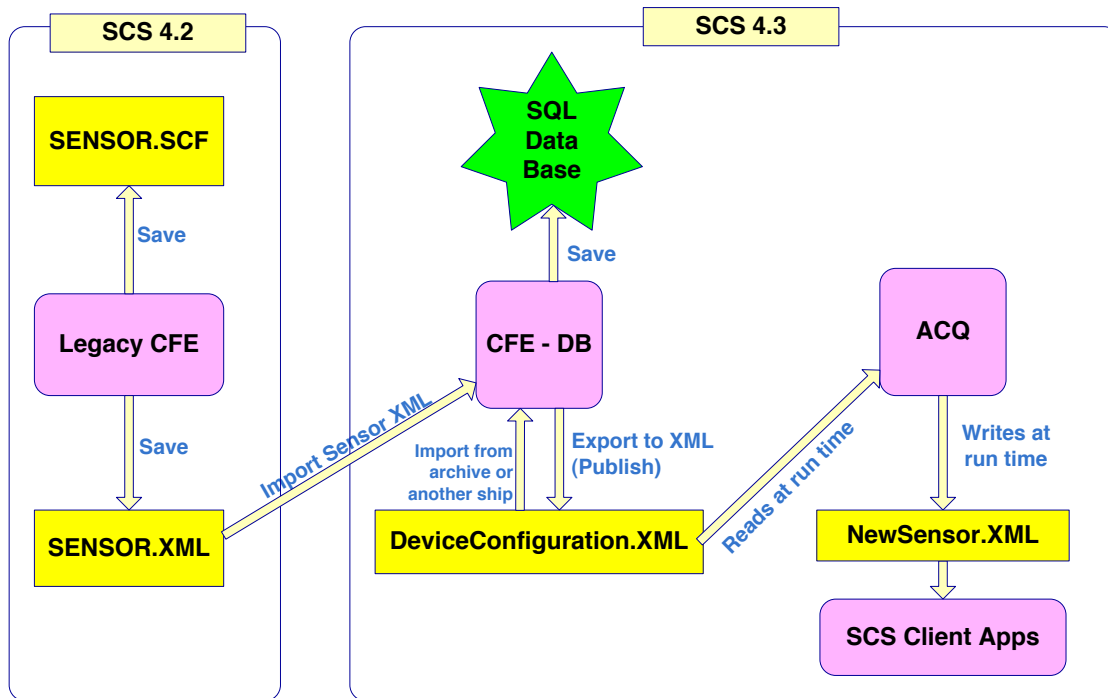
When you configure sensors, you are specifying the following:

- What sensor equipment is connected to SCS,
- The source of the sensor data (COM port, network socket, keyboard entry, etc.)
- What messages the sensors are supplying,
- How each message is divided into data fields,
- How the sensor data will be displayed in real time,
- The quality checks SCS will perform on each sensor, and
- How SCS will log the data to disk.

Sensor Configuration Files and SCS Software

The following diagram shows the flow of sensor configuration information from the database, into ACQ and from there into ACQ’s client apps.

It displays the transition path from pre-database ops (SCS 4.2) to database ops (SCS 4.3 and following). The NOAA fleet will finish this transition by the end of 2010.



As you study this diagram keep in mind the following.

The diagram is complicated because SCS 4.6 is a bridge between the pre database legacy and a pure database future. Eventually *all* apps getting sensor configuration information directly from the database; the various XML files will no longer be needed.

In the pre-database SCS (v 4.2) the legacy CFE created a SENSOR.SCF (the archival record of the sensor configuration) and a copy called SENSOR.XML, which was used by ACQ and all its client apps.

In moving to the implementation of the SCS database, the sensor configuration is initially populated from SENSOR.XML file. This is done by CFE-DB's Import from XML function. This is a one-time transfer, and after that SENSOR.SCF and SENSOR.XML should be archived into a safe place.

Once you migrate to DB Mode, any changes are made to the database configuration cannot be sent back to the legacy configuration files.

The database now replaces the legacy SENSOR.SCF as the archival repository of all sensor configuration information. This includes data stored by previous SCS releases in the SENSOR.SCF file. The new CFE-DB allows physical sensor metadata such as calibration dates and location on ship to be stored in the database, which was not possible in prior releases of SCS.

CFE-DB has been rewritten from scratch to talk to the database. In SCS 4.3 it is the only app that reads and writes sensor configuration information to the database. To make this information available to ACQ CFE-DB has a "Publish", aka "Export to XML" function which writes all sensor data into the DeviceConfiguration.XML file. This file is read by ACQ when it starts.

To make configuration information available to other SCS apps ACQ then reformats it into a file called "*NewSensor.XML*". ACQ has the capacity to acquire data for a maximum of 500 devices and thousands of discrete data elements.

Two Golden Rules

The diagram and discussion in the previous section have three points which must be remembered.

- 1 After changing the sensor configuration you must not only save your work to the database, but you must “Publish” to DeviceConfiguration.XML so that ACQ sees the changes.
- 2 SCS apps needing sensor configuration information must go to the NewSensor.XML file. You must run ACQ to transfer the new DeviceConfiguration.XML to NewSensor.XML. SCS Client software will not see published changes until ACQ is run.

Sensor Configuration Files After Data Acquisition

Apart from the Sensor Configuration Files discussed in the previous diagram, ACQ creates backup copies whenever it runs.

When ACQ starts, it reads the DeviceConfiguration.XML file previously published from CFE-DB and creates a copy of this file in the *DATALOG* directory, appending the acquisition timestamp so that its name is similar to this: NEWSSENSOR_19990830-1028.XML. The timestamp “19990830-1028” signifies that ACQ was started at 10:28 am on August 30 1999.

ACQ also writes time-stamped copy of NewSensor.XML for the same purpose. Since this is the file read by ACQ’s client apps, it is a focal point of any post processing efforts.

The timestamps also allow you to match the Acquisition start session with the last published sensor configuration date in the database which corresponding log files, which have the same timestamp.

CFE-DB Devices

3.5.1 CFE-DB Physical Devices

In CFE-DB a Physical Device describes the kinds of actual devices (i.e. electronic equipment) that may be interfaced to the system. The database encapsulates all of the information about the physical sensor including the make, model, installation and calibrations details.

3.5.2 CFE-DB Sensor Devices

In the database, and in CFE-DB, a Sensor Device encapsulates the interface of a physical device to ACQ. You can also think of it as describing the source of a physical device’s messages, without knowing anything about the physical device itself. A sensor device has parameters such as the baud rate, TCP/IP port and others that further describe the interface.

A sensor device does not contain details about the data in the message – this is the function of its associated message definitions. Refer to the following section for details.

SCS has nine Sensor Device types, each one distinct in how it is interfaced to its data source.

1. COM device: Connects to its physical device via a COM port using the RS-232 protocol. A COM device receives messages without prompting the physical device.

2. Polled COM: Same as a COM device, except the physical device sends a message only when ACQ prompts it to.
3. Network device: Connects to a physical device over a network using a TCP or UDP protocol.
4. Manual. The physical device is a keyboard that passes data to ACQ over a TCP connection. The custom SCS app “Input Manual Data” allows the user to send observational data from his keyboard to ACQ for logging, display, etc.
5. Simulated. The source of simulated data is previously logged raw data files having raw physical sensor messages) ACQ reads these files and processes them as if they were coming from external sources.
6. Derived Arithmetic. See below.
7. Derived Average. See below.
8. Derived Linear Equation. See below.
9. Derived True Wind. See below.

3.5.3 Derived Sensor Devices

COM, network, manual sensor devices receive messages from external sources. Derived sensor devices do not have source external to ACQ. Derived sensors use data from other sensors (“base sensors”) and combine them to calculate new data values. A derived sensor device then constructs a message containing the derived values which appears to SCS to have come from an external physical device. The message can be logged and parsed into data fields as if it were coming from an external source.

There are four types of derived sensor devices

1. Arithmetic: Takes two data field values from other sensors and either adds, subtracts, multiplies or divides them.
2. Average: Takes a data field value and performs a time series average of the previous few data samples.
3. Linear Equation: Takes a data field value, multiplies it by a constant then adds a second constant.
4. True Wind: Uses heading, course over ground, speed over ground and relative wind speed and direction to compute true wind speed and direction.

CFE-DB Message Definitions

Each Sensor Device may deliver multiple distinct messages that are transmitted from the Physical sensor. Message definitions define all of the information needed to acquire a specific message from a physical sensor. A sensor may transmit multiple messages with each being described separately. Message definitions are sometimes closely coupled with the kind of Sensor Devices and other times message definitions are re-used by multiple Sensor Devices. For example a COM Sensor Device and a Network Sensor Device may both have the same kind of message definition, where a true wind sensor device will have a very specific message defined for that device.

A Sensor Definition may contain multiple message definitions, encapsulating a GPS receiver that may be sending multiple NMEA messages (GGA, VTG, etc.)

3.5.4 SERIAL Message Definition

A Serial Message is in fixed format with each data field in the same position in each message. The message is either terminated by a special character or is defined at a fixed length.

3.5.5 POLLED SERIAL Message Definition

A Polled Serial Message is in fixed format with each data field in the same position in each message. The message is only output when a specific command is transmitted from SCS to the Sensor. The message is either terminated by a special character or is defined at a fixed length.

3.5.6 NMEA Message Definition

A NMEA message is always in the NMEA-183 Standard format or equivalent. In contrast to serial messages, NMEA messages may have variable record length. The order of the data fields is constant, but the number of characters in each field can change. Commas serve to separate all data fields. Also, a physical device often sends more than one message, each distinguished by a unique identifier (called a “Sentence Label”). You must create a message definition for each message that ACQ will be accepting from the device.

Thus, interfacing a Trimble GPS receiver sending \$GPGGA and \$GPVTG messages would require two NMEA messages: one for the Trimble \$GPGGA message and the other for the Trimble \$GPVTG message.

3.5.7 Manual Message Definitions

Manual Message data field values are entered by keyboard using the Input Manual Data app in the Acquisition menu. The message is in NMEA format. You have control over the sentence label. The data values are comma separated.

3.5.8 General Derived Message Definitions

All types of Derived Sensors generate their messages in a fixed format, and you cannot change it. For all derived sensors:

- Messages are in a NMEA-183 format
- The sentence label is specified using CFE-DB.
- The first field(s) in the message always is (are) the computed value(s).
- Additional fields if they exist show base sensor values and possibly some intermediate calculations, for debugging purposes.

3.5.9 Specific Derived Message Definitions

1. For Derived Arithmetic the data fields are: Computed value, value of base sensor 1, value of base sensor 2.
2. Derived Average can be computed in two ways: simple arithmetic and polar.
 - a. For simple arithmetic, the data fields are: derived average value, most recent base sensor values, sum of base sensors values used in the average, number of samples in the average.
 - b. For polar average, the data fields are: derived average value, most recent value of base sensor, sum of sine of base sensor values, sum of cosines of base sensor values, number of samples in the average.
3. For Derived Linear Equation the data fields are: Computed value, base sensor value.
4. For Derived True Wind the data fields are: True wind speed, true wind direction, base relative wind speed, base relative wind direction, base speed over ground, base course over ground, base heading

Details on Derived Calculation Methods

3.5.10 Effect of Base Sensor Message Timeout Parameter

All sensor messages have a `Timeout` parameter which declares how old a sensor value can be before it becomes stale. ACQ will not compute a derived sensor value if any of its base sensor messages is stale. Logging to derived sensor RAW files ceases until the timed-out message is brought back to a current status.

3.5.11 Derived Average

A Derived Average Message averages the most recent data from its defined base data value (data field). A typical use for the Average message is to smooth out data from a Wind Direction sensor, whose time series plot is usually very noisy.

Averaging can be done two ways: by sample or by time. By sample means the N most recent samples from the base sensor are averaged, without regard to the length of time covered by the samples.

By time means all samples received in the last N seconds are averaged, without regard to the number of samples in the average. Latitude and longitude values can be averaged as any other sensor, but the averaged value is output in decimal degrees.

NOTE: only those values that have not timed out will be used, regardless of the number of seconds specified in the averaging sensor's definition.

There are also simple arithmetic and polar averages to choose from. Select the polar method for base sensors which report some type of compass data, e.g. course over ground, wind direction, gyro heading. Select arithmetic averaging for any other kind of sensor value.

3.5.12 Derived Arithmetic

This performs an arithmetic operation on the two base sensor values. Available operations are addition, subtraction, multiplication and division.

3.5.13 Derived Linear Equation

Multiplies the data value of its base sensor by a constant (the *Slope*) and adds the result to another constant (the *Offset*). Mathematically, $y = mx + b$, where y is the derived sensor value, m is the *Slope*, x is the base sensor value, and b is the *Offset*.

A Derived Line Equation sensor could be used to convert a measurement from nautical miles to kilometers (Slope = 1.852, Offset = 0), or a Centigrade temperature to Fahrenheit (Slope = 1.8, Offset = 32), or the indicated sonar depth to the true depth (Slope = 1, Offset = depth of transducer)

3.5.14 Derived True Wind

A Derived True Wind Messages computes the true wind speed or direction, given a ship's course over ground, heading and speed over ground, and relative wind speed and direction.

The algorithm is found in the paper “Establishing More Truth in True Winds”, second revision, by Shawn R. Smith et al. It is published in the *Journal of Atmospheric and Oceanic Technology*, and is also available on the Florida State University COAPS web site.

When you set up a derived sensor, only some categories of sensor data values will be eligible to be base values. In order to create a new derived message, at least one of every base sensor data value required by that sensor must already exist.

Manual Message Definition

A Manual Device is used to allow SCS users to enter observation data into the system via a specialized application called the Manual Data Entry Program.

Simulated Message Definition

A Simulated Message should never be used in an operational system and should only be used to replay data for purpose of testing, training, or development.

3.6 Sensor Data Fields

Every Sensor Device passes one or more Messages to ACQ. Each message in turn has one or more data fields associated with it, one for each measurement that the physical device makes.

SCS allows you to define data fields for each message. Each data field has parameters that establish how to decode and monitor one measurement in the sensor device message. These parameters specify such things as where the measurement is in the message, what units it has, and what kind of quality checks to do. Sensor data field definitions describe critical information of not only where the data is located in the message but also information such as the name, units, quality parameters, of the data.

Once ACQ isolates a data field it can pass on the value to other SCS apps that can display them, pass them on for derived sensor calculations, or use them in other ways described in later chapters.

Since serial messages have fixed format, serial data fields are delimited by their start and end character positions in the message.

NMEA Messages are variable length where the data fields are delimited by a comma. NMEA data fields may also be attributed special features of translation to allow a code embedded in the message to be translated to a more meaningful value.

Special Topics in Sensor Message Acquisition

3.6.1 Grouping Polled Serial Sensors with RS-485

You may configure a number of Polled Serial Devices to operate as a group. Physically, the sensors in this group are daisy chained together using an RS-485 serial interface and are connected to the same serial port. You may set up any number of queued groups chained to different serial ports. Perform the following steps in CFE-DB for each group you wish to create.

1. Create a “Polled COM” sensor device.
2. Add a message definition below the device for each message in the polled queue group. Fill in the Queue Order value for each message indicating the order that polling will occur.
3. Add the data field definitions to each message as appropriate.

NOTE: The polling sequence depends only on the Queue Order values. It does not depend on the order of the Message definitions under the Device.

3.6.2 Use of Label Extenders for Non-Standard NMEA Messages

**** NOT AVAILABLE IN THE CURRENT VERSION OF ACQ ****

The NMEA-183 message spec allows sensor manufacturers to compose proprietary message formats that look like NMEA messages but do not conform to the 183 standard in one or more ways. Manufacturers may indicate this by using the characters “\$P” at the start of the sentence label. *Note: “P” standing for “proprietary”.*

In the NOAA fleet the SIMRAD PI4x net mensuration sensor is in use on several fisheries trawl ships. This message violates the NMEA standard in that its sentence label is not sufficient by itself to describe the information present in the message. The complete description is specified only when you know the value of one or more fields in the message.

SCS NMEA Messages by default must be completely described by the sentence label. In order to accommodate the scenario presented by the SIMRAD sensor SCS has implemented the Extend Label and Label Extension parameters in the NMEA Message definition

To quote from the SIMRAD manual:

All PI3X/4X/5X-specific NMEA sentences start with”\$PSIMP”, followed by a 1or 2 character field specifying which PIXX sentence is being sent. The sentence specifies are

C: Configuration sentence

D: Data sentence (obsolete)

D1: Data sentence

F: Sensor definition sentence (obsolete)

F1: Sensor definition sentence

S: Frequency spectrum sentence

T: “Transparent” or system sentences, for communication between DSP and external equipment without interfering with the PI3X/4X/5X display unit.

It gets worse: field one of the sentences indicates the *basic* sensor type. There is also a “sensor type variant” in field four of the message. These two fields are letters, and together with the sentence label, determine the content of the rest of the fields in the message.

EXAMPLE: Field 1 = “D1” indicates this is a data sentence. Field 4 = “H1” indicates the data is “head rope height” while Field 4 = “T1” indicates the data is temperature.

In CFE-DB, this situation is handled as follows:

- In the Sentence Label box, insert “PSIMPD1T1”. This is the concatenation of the sentence label and the values in fields 1 and 4 that describe the data in the temperature message.
- Check the Extend Label Box

- For the Label Extenders box, enter “1,4” indicating the two fields that determine the data content. The field numbers here are comma-separated and must correspond to the values of the fields specified in the Sentence Label.
- Define data fields(s) to pick up the expected data from this message type. You will need to consult the sensor’s manual to determine the field definitions.

This procedure must be done for all combinations of values in fields 1 and 4 you wish ACQ to pick up.

When ACQ runs it can now look at the sentence label and fields 1 and 4, and identify a valid message associated with specific data fields and how to parse them.

Converting NMEA-formatted LAT and LON values to Decimal Degrees

Use the Derived Average sensor. LAT and LON values can be averaged like any other sensor values, and a side effect is to convert the input values (in NMEA format) to decimal degrees.

NOTE: Do not use polar averaging for the longitude. You must use arithmetic averaging for both. Set the number of samples to 1. Averaging one sample returns the sample itself.

Sensor Timeouts and Stale Data

If a sensor that stops sending messages, ACQ will continue to use the last received data until the data’s age exceeds a *timeout* limit. At this point the data becomes *stale* and is no longer used. There are several ways this happens.

The Timeout is defined for all sensor messages in units of seconds. It defaults to 30 sec when the message is created but must be modified to suit the project’s needs.

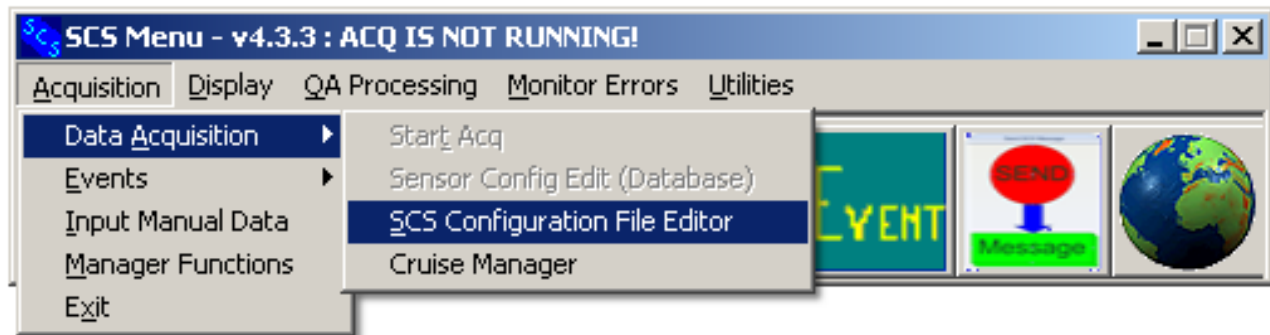
When the time of the last message received exceeds Timeout ACQ stops logging data to the RAW file. You will see a hole in the stream of messages in the RAW file, until the sensor comes back on line.

ACQ will not use stale data will to compute any derived values. Derived sensor RAW files will have gaps until base sensor comes back on line.

The “*Send SCSMessage*” app will not send stale data. The Event Logger will not log stale data to an ELG file.

The Real Time Display uses its own definition of “Stale”. The last received value will be displayed until it is more than 10 minutes old. Then it is marked stale in the display window by printing a warning message instead of the last received. The warning message will remain until the sensor comes back on line.

3.7 Sensor Configuration Editor (CFE-DB)



Introduction

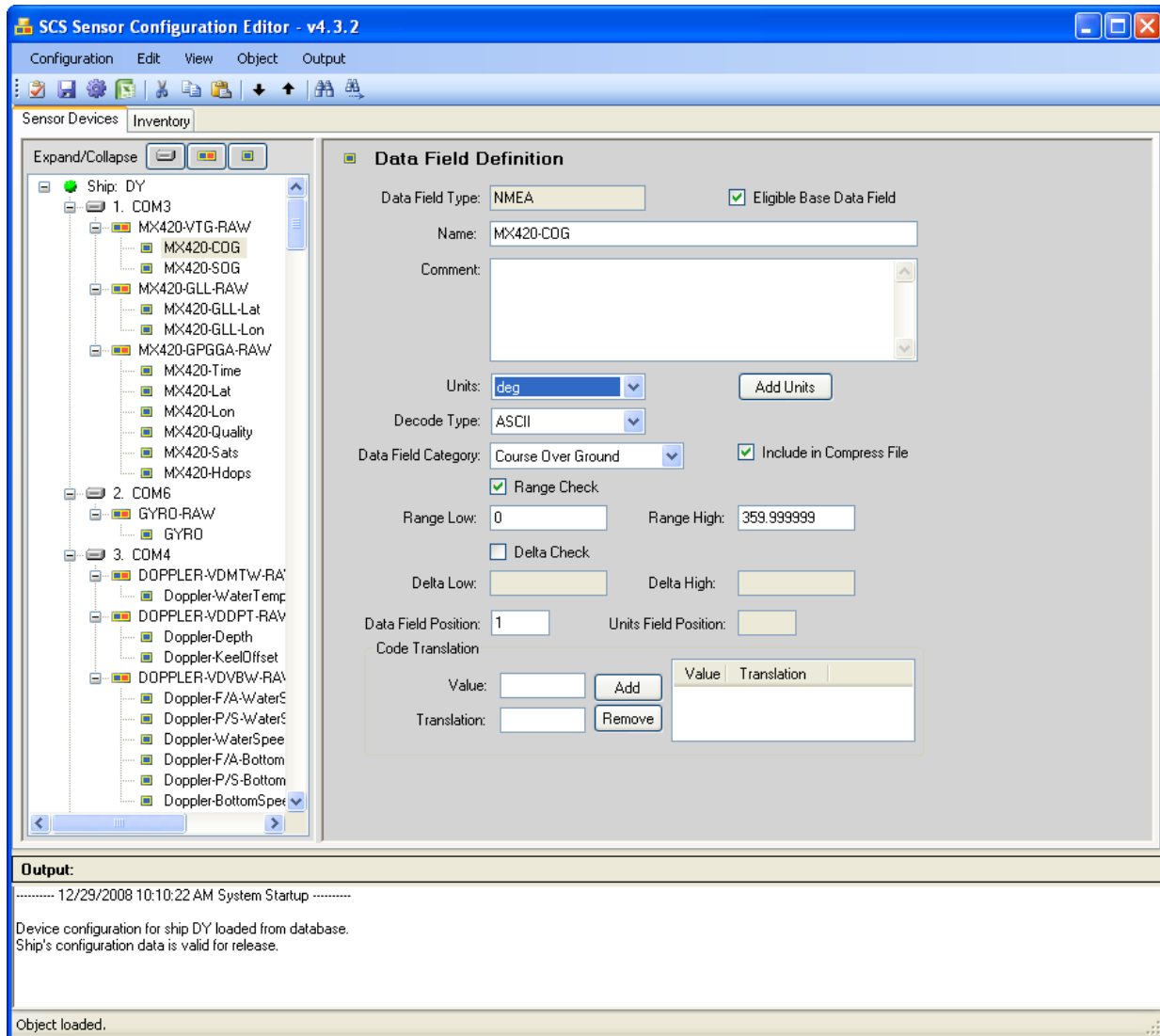
SCS release 4.3.3 introduces a completely new CFE-DB. The new GUI design is meant to mesh with the organization of the SCS SQL database. Users of the previous version will recognize most concepts, although some have changed names. For example, "*Parent Sensor*" becomes "*Message*", and "*Child Sensor*" becomes "*Data Field*".

Relationships between parameters have changed too. For example in pre-database releases of SCS the COM port number of a sensor was a property of the parent sensor. In SCS database mode the COM port has a collection of one or more messages, for example \$GPGLGA and \$GPVTG messages from a GPS receiver.

Sensor Configurations are created and edited directly in a database which is installed on the SCS ACQ system with the sensor *Configuration Editor* (CFE-DB) application. CFE-DB requires Microsoft SQL Server 2008 or Microsoft SQL Server 2008 Express to operate. This application is tightly coupled with the business rules of the database to provide consistent and quality sensor configuration while providing extensibility of the history of sensor devices over time

CFE-DB Window Display

Here is a sample CFE-DB Main Window, showing the principle components.



The CFE-DB main window has three panels.

1. On the left is the Summary List. The contents vary as explained in the next paragraph.
2. To the right is the Object Editor where you enter information that configures your sensors and devices.
3. At the bottom is the Output Panel, showing status and error messages from CFE-DB.

There are two TABs at the top of the Summary List. Select "Sensor Devices" and the Summary List and Edit Panel allow you to configure the input devices, sensor messages and data fields.

Select "Inventory" to describe the Physical Device and meta-such as serial number and calibration date.

The Summary List

As mentioned above the Summary List displays either sensor configurations or physical device metadata.

“Sensor Device” List

Select the “Sensor Device” tab to display the device and sensor configurations.

3.7.1 The Vessel ID

At the root level of the list is the name of your ship. In the sample screen above it is “DY”. The architecture of the SCS database allows sensor configurations from multiple ships to be stored, and this determines which ship’s configuration will be displayed.

The VESSELCODE parameter in the SCS Configuration file controls which ship’s sensor configuration is displayed. This parameter can be set either directly with the SCS Configuration File Editor. Also, using the Cruise Manager can also change this parameter. See the sections on the Cruise Manager app in this chapter for details.

3.7.2 Validation Status

To the left of the Vessel ID is an LED indicating the validation status of the configuration. Green means there are currently no errors detected and red means errors exist that must be corrected.

3.7.3 Tree Structure

Previous versions of CFE had a two level tree showing Parent Sensors and their Child Sensors. The new CFE-DB shows a three-level tree showing Sensor Devices, Sensor Messages and Data Fields.

CFE-DB Object Window

When reading the descriptions of CFE-DB operations, remember that the term “object” refers to one of Sensor Device, Message Definition, and Data Field Definition.

These terms were discussed in the section on Configuring Sensors previously.

The large window placed to the right of the Sensor Device window displays all of the specific information needed to configure the objects needed to configure your system. Most information and activities in this application will occur in this window. CFE-DB will change the content of this panel depending on the kind of object you select in the Summary List.

3.7.4 CFE-DB Output Window

CFE-DB window at the bottom of the application is the output window where the application will provide information about the configuration and confirm actions taken by the user. Errors that CFE-DB detects in your inputs will all be displayed here. Any logic errors generated by CFE-DB will also be displayed here.

3.7.5 Splitter Bars

The panels are separated by ‘splitter’ bars that allow you to resize each panel within the window.

3.7.6 CFE-DB Command Menus

All the commands you need to create and maintain Sensor Configurations are contained in the menu structure. The following sections describe each menu selection in turn.

You can also use the toolbar buttons to invoke the most commonly used commands. In the following sections, you will see the image of the toolbar button for any menu selection that has one.

3.7.7 Alternatives to the Command Menus

Keyboard Accelerators

Some commands have keyboard accelerators defined for them. Many are Windows standards, for example Ctrl/X for Edit/Cut. Others are specific to CFE-DB:

- Keypad up and down arrow keys move up and down the Sensor List.
- Keypad left and right arrows collapse and expand child sensor displays in the Sensor list.

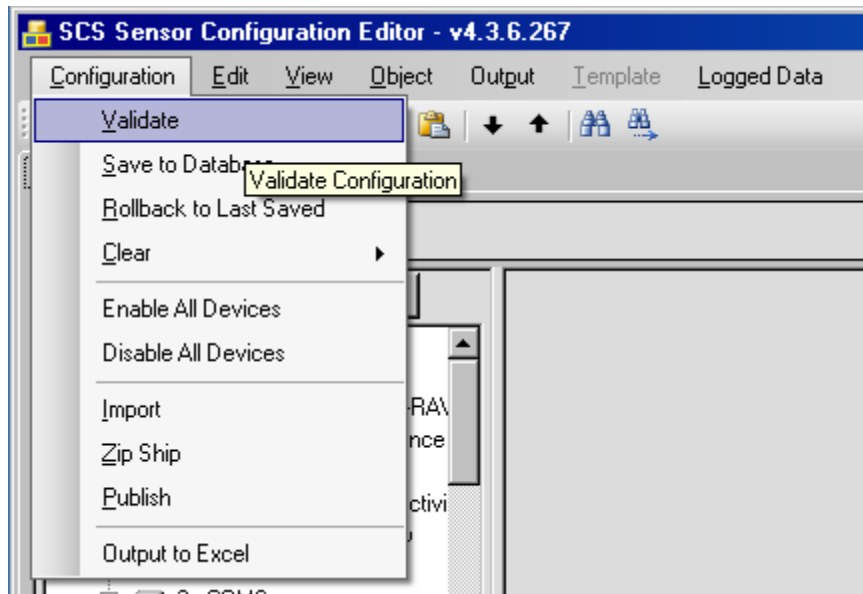
Context Menus

Right-click on an object in the Summary List to invoke an immediate menu, also known as a context menu. A context menu has the most common commands associated with that object. Context menus are discussed in detail in the following sections.

Tool Bar

Below the Menu Bar is CFE-DB's *Tool Bar*, a collection of buttons that you can click on to performing the most common actions. You will recognize many of them (e.g. Copy and Paste) from other Windows apps. A brief description of the button's function pops up as you move the mouse cursor over a button and pause without clicking.

3.7.8 CFE-DB Configuration Menu



This menu is used to manipulate the database configuration and provide a tool for importing, exporting sensor configurations and converting them to Excel spreadsheets.

- **Validate:** used to determine if the current configuration is valid or is not complete
- **Save To Database:** Save all the current changes to the database permanently. See below for details.
- **Rollback:** drop all changes since the last save and start over
- **Clear:** delete all configuration data for this ship (Warning: this is extreme and not revocable!)

- **Enable All Devices:** Just what it says.
- **Disable All Devices:** Just what it says.

- **Import:** Allows for the importing of DeviceConfiguration.XML files, legacy Sensor.XML files, and Zip Ship files (see below).

Note: Legacy Sensor.XML files do not include sensors declared as disabled in the legacy CFE application.

- **Zip Ship:** Publishes the sensor configuration info, and then inserts it into a ZIP file along with all images stored in the sensor inventory. The zip file is written to the *Reports* folder under the DATALOG folder.
- **Publish:** Create a new DeviceConfiguration.XML file in the SCSShip folder, an ASCII version of the configuration in the DB. ACQ reads this file whenever it starts up.
- **Output to Excel:** provide an Excel version of the sensor configuration file.

Note this application does not open or exit from this menu. As described above the application utilizes the configuration to pre-configure with the define ship. Users exit the application simply by using the X in the top right corner.

Validation, Save and Publish

The Validation, Save and Publish commands are intertwined in the following ways.

1. The *Save* performs a validation (though incomplete). After you fix any errors it fixes, CFE-DB saves the current configuration to the database.

WARNING: Saving does not make it visible to ACQ or any other SCS app. That requires a Publish command.

2. The *Publish* operation does the following:
 - Performs a Full validation.
 - Allows you to enter a tag for this configuration release.
 - Writes the DeviceConfiguration.XML file to the ship directory.
 - Archives any images in the configuration within the database.
3. The *Validate* command validates the configuration and notes errors in the output window. This command does the same validation as the Publish command. See below for more detail.

Publish puts aside entire the configuration in case you ever need to recover some lost work.

3.7.9 The Validation Process

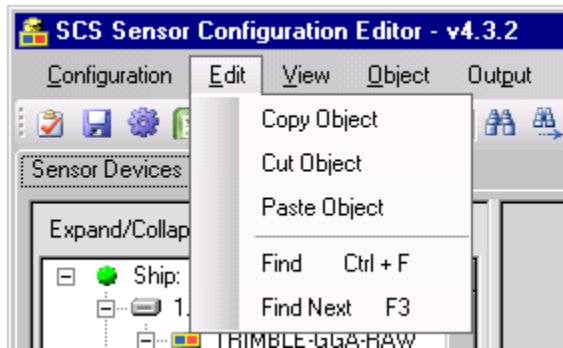
Validation is CFE-DB's way of insuring all parameter values are "reasonable". CFE-DB performs different kinds of validation at different points in your editing session:

- when you move from one data entry element to another,
- when you move from one sensor device list object to another,
- when you save the configuration, and
- When you attempt to publish the configuration for system use.
- During editing some but not all parameters may be validated. Errors detected here are flagged with a pop-up error dialog box, and must be corrected immediately
- During a Save additional validation is performed. A Save is permitted even with validation errors.
- During a Publish: During Publish a full validation is performed. All validation rules must be satisfied; publish does not happen otherwise.
- You may request a Validation at any time. This is the complete validation that also happens during a Publish.

3.7.10 Error Messages

During Save or Publish errors will appear in output panel at the bottom of the CFE-DB app window. You can copy the errors then paste them in a text editor window so that you can read them while you are correcting.

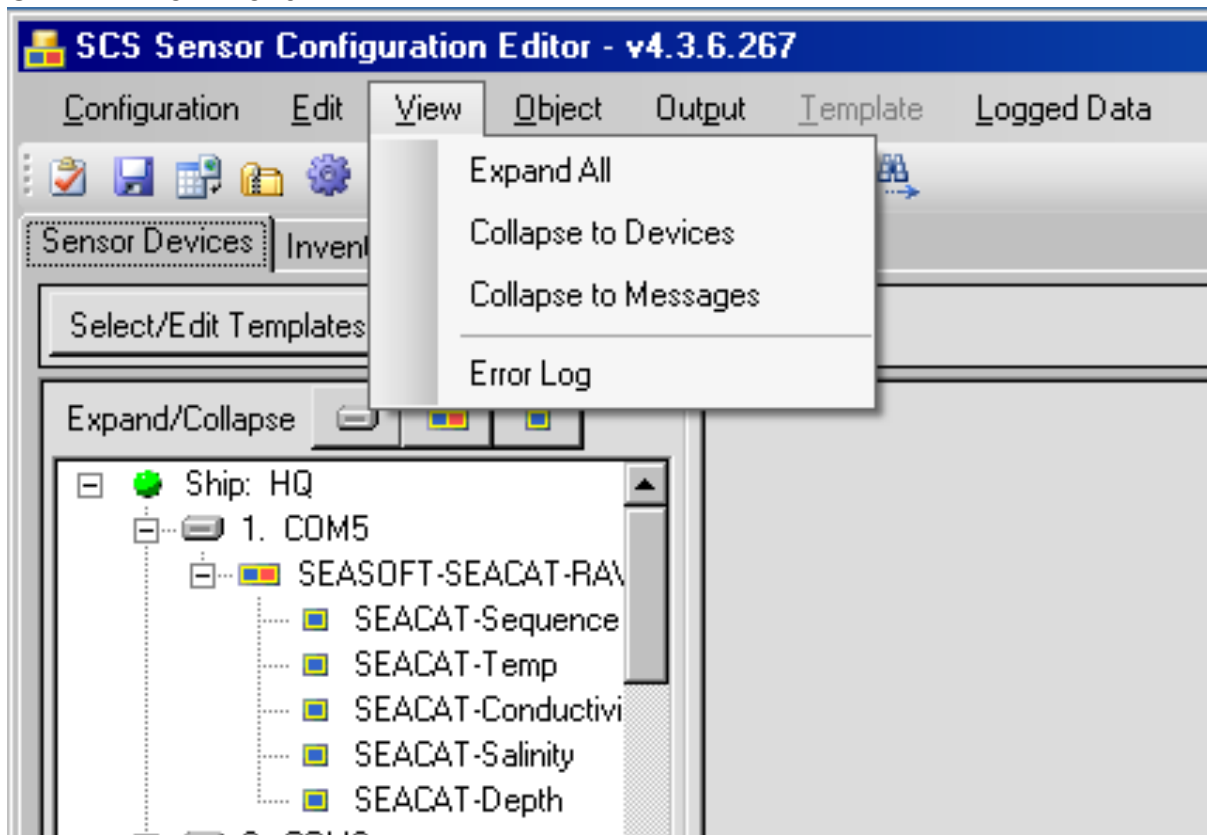
CFE-DB Edit Menu



The edit menu is utilized to manipulate the Sensor Device, Message Definitions or Data Fields listed on the Sensor Device window described above. Similar to normal Windows Applications the edit menu applies to the object selected. These functions may also be found on the short cut menu near the top of the application.

- Copy Object: Copy the object and all definitions for use in the future
- Cut Object: Remove the object selected and save for use in the buffer
- Paste Object: Insert the object saved by copy or cut into the Sensor Device Window Tree at the selected location
- Find Object: Provide a window to search for an object keyed
- Find Next: search for the next occurrence of the object being searched

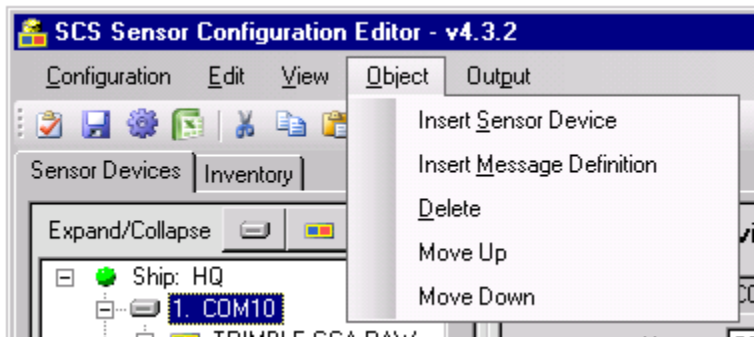
CFE-DB View Menu



The view menu is used to manipulate the Sensor Device or Inventory window to expand and contract the tree for visibility of items sought. The error log will pop up a window showing all errors logged in the application.

- **Expand All:** Expand the whole tree on the Tab selected
- **Collapse To devices:** Collapse the whole tree on the Tab selected to the Sensor devices hiding the message definitions and data fields
- **Collapse to Messages:** Collapse the whole tree on the Tab selected to Sensor Devices and message definitions hiding all the Data field definitions.
- **Error Log:** The error log has all messages generated by CFE-DB regarding warning and errors as they occur. These messages are not the result of user error; OMAO support personnel may ask you to send a copy of this file in the event of a CFE-DB application error.

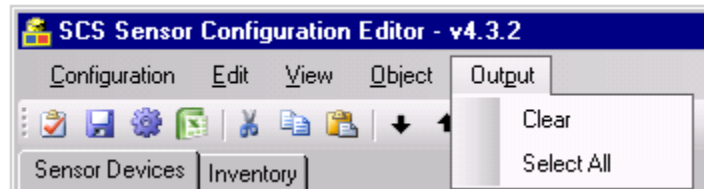
CFE-DB Object Menu



The object menu is used to move or insert Data Field objects. It is also used to move whole Sensor Devices or message Definitions up and down in the Sensor Device Tree. The user selects the object targeted for action or location of insert.

- Insert Data Field: The user selects the location to insert. A new empty data field will be created below the selected object
- Delete: The user selects the object to be deleted. Sensor device and message Definition objects will delete all child objects
- Move Up: Selected object are moved up in the tree
- Move Down: Selected object is moved down in the tree

CFE-DB Output Menu

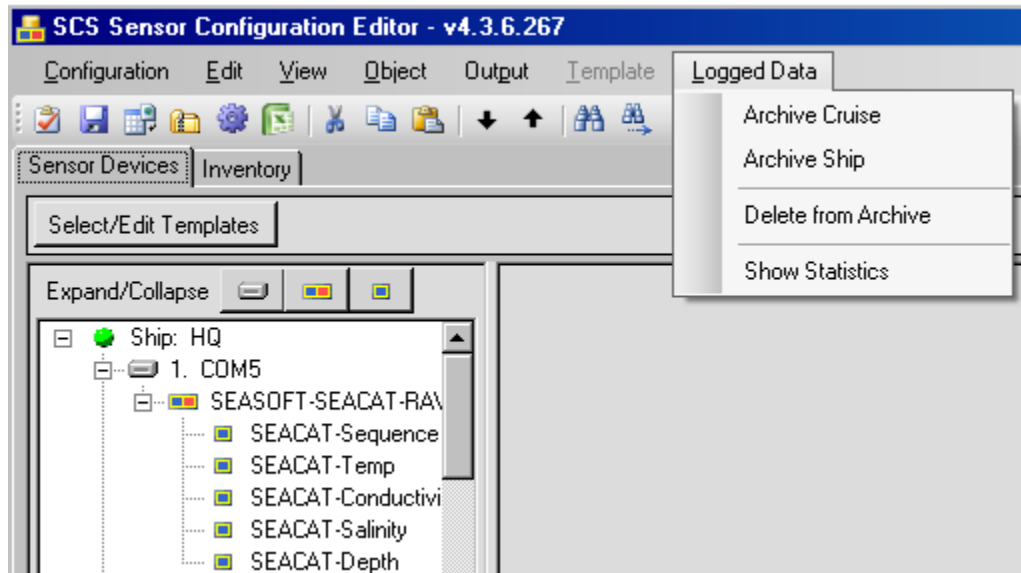


The output window is a diagnostic window to help you track the progress of the editing session. This menu allows the user to reset this window and also copy the content of this window in the event of a problem and send in an e-mail for assistance.

- Clear: Delete the contents of the output window for visibility
- Select All: Select all contents of the output window to copy to other application.

CFE-DB Logged Data Menu

This menu includes commands to manage the logged messages in the database.



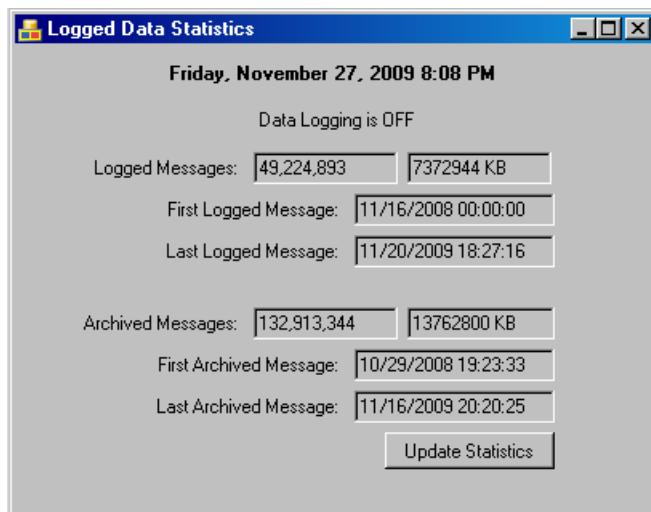
Archive Cruise: Moves the logged messages from its primary storage place in the database to the archive area of the database. This empties the primary storage and improves response time when SCS is started up again.

Archive Ship: Same as Archive Cruise, but operates on all cruises for a ship.

Delete from Archive: Deletes raw sensor messages from the archive portion of the database.

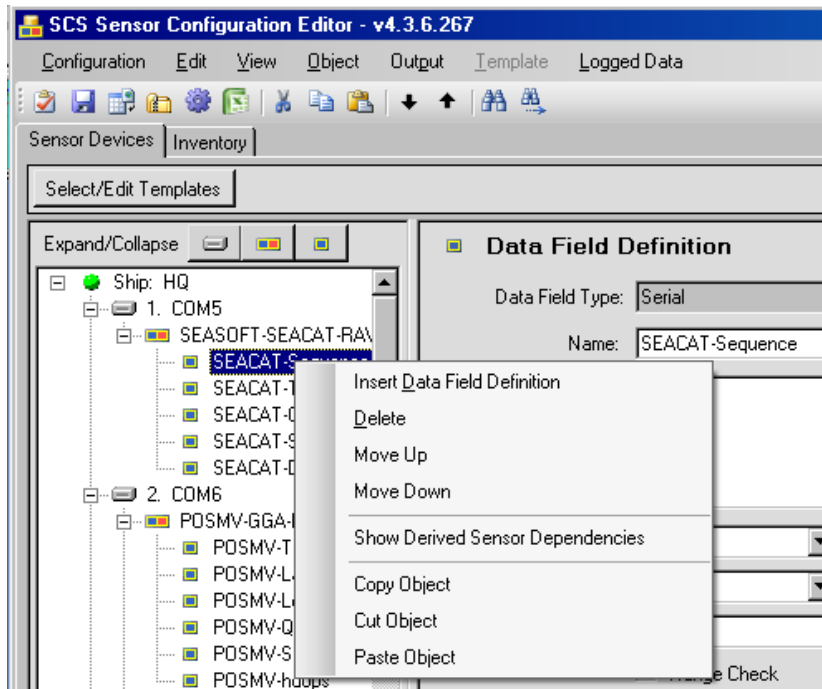
**Warning: this operation cannot be reversed, and requires the manager password to be invoked.*

Show statistics: brings up a summary window like the following. Note: depending on the number of raw sensor messages stored in the database, this operation may take some time to complete.



CFE-DB Sensor device Right Click Immediate Menu

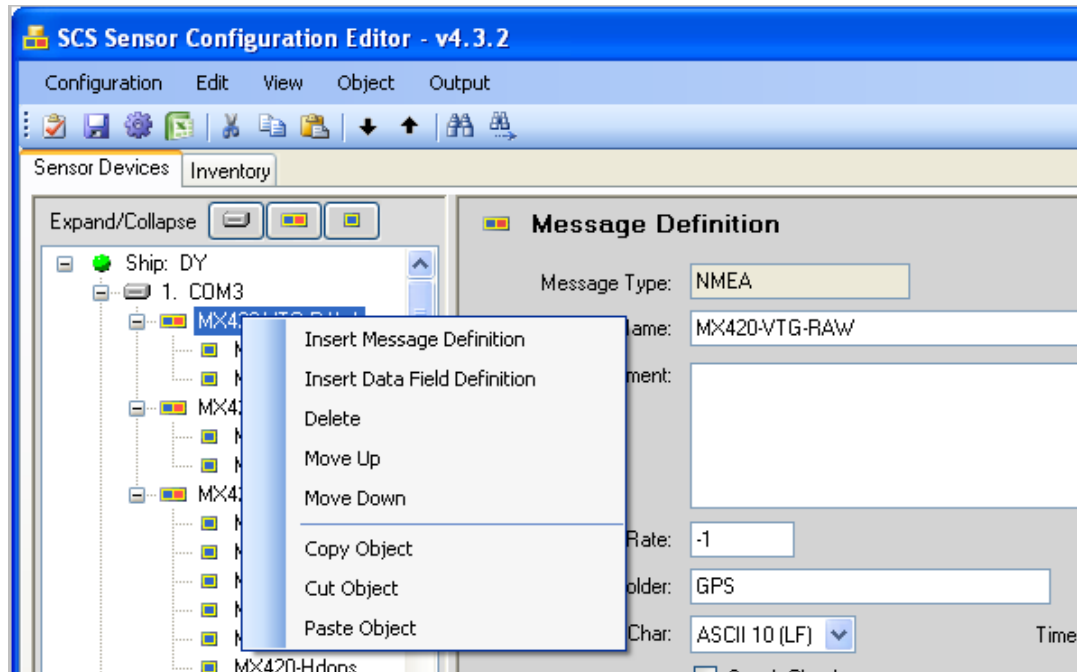
The following actions are available by right-clicking on a Sensor Device object.



- **Insert Sensor Device:** This action allows the user to create a new Sensor Device to begin describing a sensor. The Sensor device will be inserted immediately after the last message definition for that Sensor device. The Application ensures that the insert is consistent with all configuration rules of the database.
- **Insert message Definition:** This action allows the user to create a new Message Definition associated with the Sensor Device immediately above.
- **Delete:** This action deletes the entire sensor device and all message definitions and data fields described for this Sensor device
- **Move Up:** This action Moves the entire Sensor Device and all associated message Definitions and Data Fields up in the tree
- **Move Down:** This action Moves the entire Sensor Device and all associated message Definitions and Data Fields down in the tree
- **Show Derived Sensor Dependencies:** Displays a list of all derived sensors that use this data field as a base sensor.
- **Copy Object:** This action Copies the Object for use
- **Cut Object:** This action deletes the object selected and retains it for use elsewhere in a buffer
- **Paste Object:** This action allows the saved Sensor Device object to be inserted immediately below the selected Sensor device. The application does not allow disruption of objects owned by the Sensor Device such as Message Definitions and Data Fields.

CFE-DB Message Definition Right Click Immediate Menu

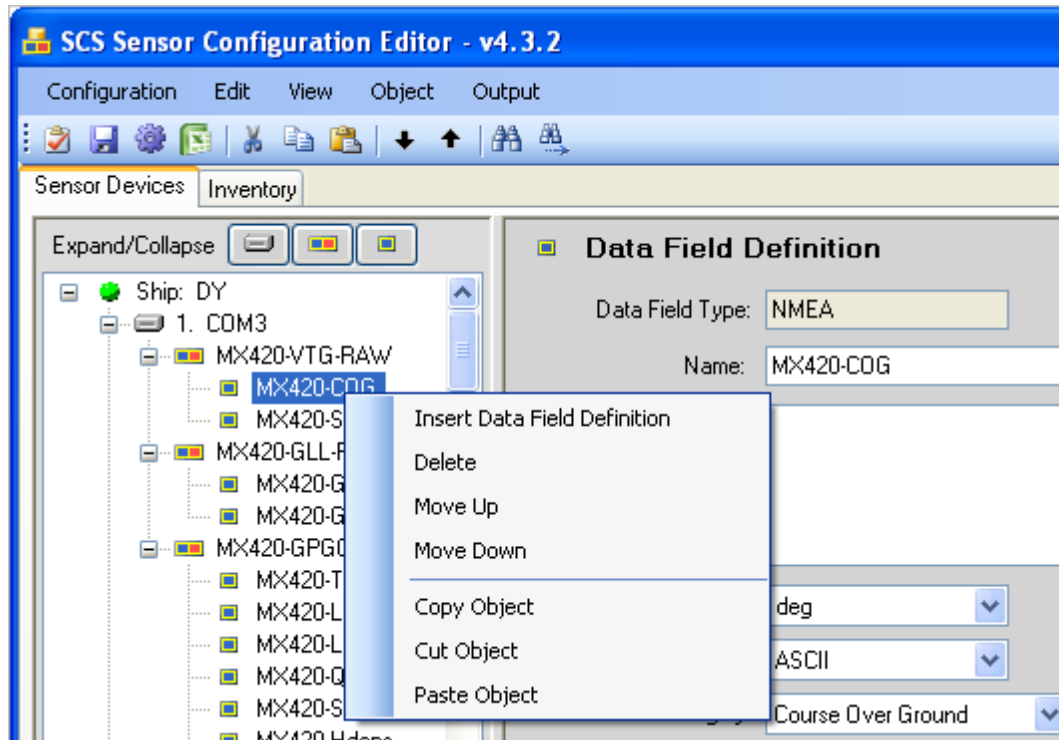
The following actions are available by right-clicking on a Message Definition object.



- **Insert message Definition:** This action allows the user to create a new Message Definition associated with the Sensor Device immediately above.
- **Insert Data Fields:** This action allows the user to create a new Data Field associated with the Message Definition immediately above.
- **Delete:** This action deletes the entire Message Definition and all data fields described for this Message Definition
- **Move Up:** This action Moves the entire Message Definition and all associated Data Fields up in the tree
- **Move Down:** This action Moves the entire Message Definition and all associated Data Fields down in the tree
- **Copy Object:** This action Copies the Object for use
- **Cut Object:** This action deletes the object selected and retains it for use elsewhere in a buffer
- **Paste Object:** This action allows the saved Message Definition object to be inserted immediately below the selected Message Definition. The application does not allow disruption of objects owned by existing Message Definition such as Data Fields.

CFE-DB Data Field Right Click Immediate Menu

The following actions are available by right-clicking on a Data Field object.



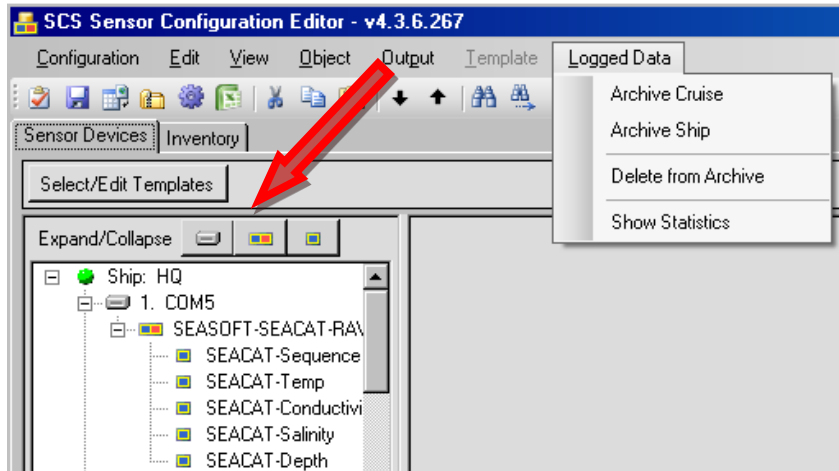
- **Insert Data Fields:** This action allows the user to create a new Data Field associated with the Message Definition immediately above.
- **Delete:** This action deletes the data field
- **Move Up:** This action Moves the entire Data Field up in the tree
- **Move Down:** This action Moves the entire Data Field down in the tree
- **Copy Object:** This action Copies the Object for use
- **Cut Object:** This action deletes the object selected and retains it for use elsewhere in a buffer
- **Paste Object:** This action allows the saved Data Field object to be inserted immediately below the selected Data Field.

Templates

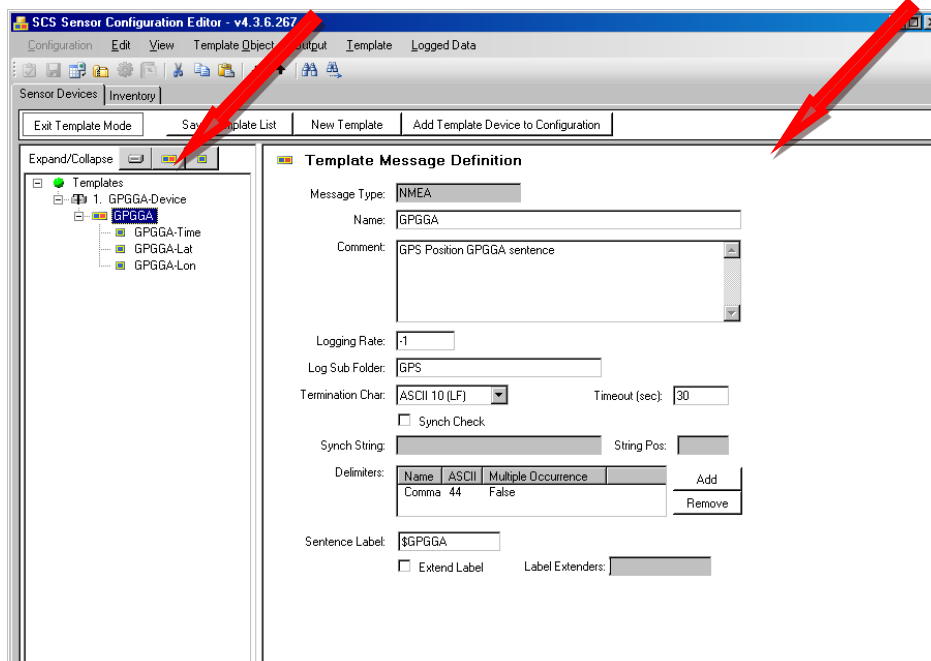
CFE-DB includes a feature where you can define templates for a class of commonly used sensors, then copy the template in the sensor list. You only have to type the template sensor once, but then you can use it as many times as you need. It is especially useful for NMEA sensors such as GPGGA or other common sentence labels.

Creating or Editing a Template

Click on *Select/Edit Templates*, show below, to put the form into template mode and to see the list of defined templates in your sensor configuration.



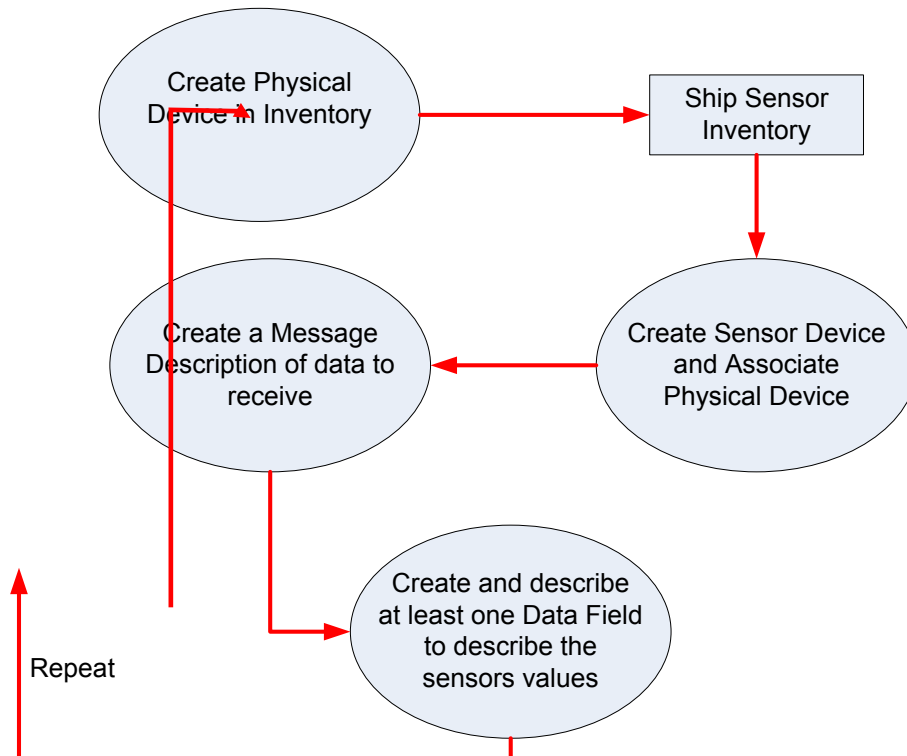
Note that the data looks just like the sensors in the Sensor Devices



Create sensor definitions just like in the regular sensor list. Pick an object and press Add Template Device to Configuration to move it into your sensor list. When you are finished, press “Exit Template Mode” to return to editing your sensor list.

CFE-DB Describing a Sensor a Step Wise Approach

This diagram illustrates a good way to completely configure and describe a sensor.



1. Create a Physical Device in inventory that will be associated with the data being collected.
2. Create a single Sensor Device and associate that sensor Device with the installed Physical device just defined. This Sensor Device when created will be of one of the types described earlier in this document. Based on the Sensor Device Type a set of parameters will be presented in the Object Window to fully define this Sensor Device.
3. Add at least one Message Definition to the Sensor Device to describe the message that will be acquired by the sensor.
4. Add at least one data field to be extracted from the sensor message.

CFE-DB Parameters

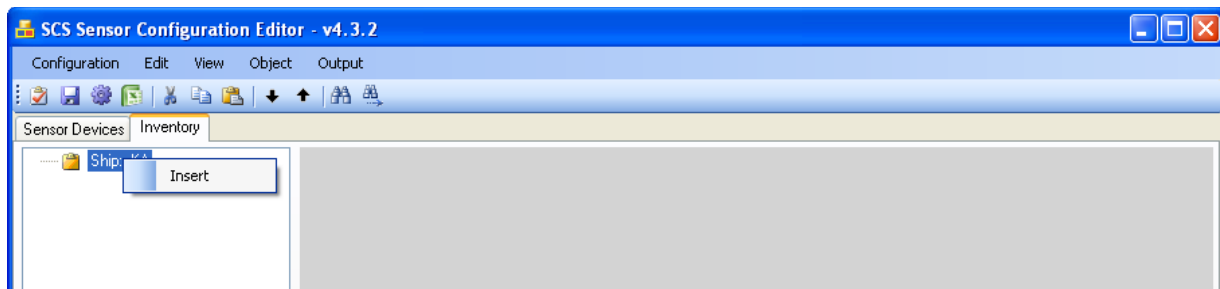
CFE is based on the user filling in critical descriptive parameters in one of four primary objects; Physical Devices, Sensor Devices, Message Definitions and Data Field Definitions. Examples would be a physical device's Calibration Date, sensor device's Baud rate, message's Sentence Label, and data field's Decode Type.

CFE-DB Inventory Parameters


3.7.11 Creating Physical Device Inventory with CFE-DB

The inventory tab is a new addition to the CFE-DB v4.3. You will need to populate the Inventory with Physical Devices. Once populated the inventory provides a standardized location for tracking sensor systems, serial numbers and calibration dates that will be easy to update as changes are made. Having this one location will help ensure no information is lost as crew rotates. CFE-DB will also identify sensors that are in need of calibration.

After activating CFE-DB and importing the sensor.xml file, click on the Inventory Tab to begin creating the Inventory. Highlight the Ship ID and with the mouse, right-click. The option of Insert will appear.


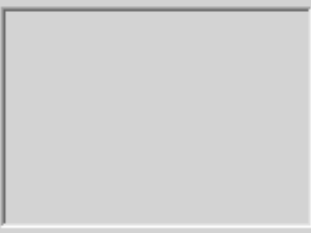


This will create a New Physical Device.


Physical Device

Warning -- calibration date is more than one year old or is not set.

Name: Primary GPS
Manufacturer: Magnavox
Model No.: MX-200
Serial No.: 115625348
CD No.:
Type: GPS
Comment:

0 Images. 


Gen. Location: Flying Bridge Aft

Clear

Survey Location

Measurement Reference: Gyro Room Survey Mark

Select/View

X: 20 Y: Z:

Latest Calibration

☐ Cannot be calibrated

Upload

View

File:

Select Date: 1/23/2012 11:58:38

Date: 11/23/2012 11:58:38.000

History

Delete

Latest Test

Select Date: 10/11/2011 11:58:38

Date: 11/11/2011 11:58:38.000

History

Add New

Tested By: Chief ET

Delete

Results: Passed

Not installed.

With the exception of the Dates and Calibration File, each section will accept text input. Calibration Files can be uploaded using the Upload button. Once uploaded the file is available for viewing using the View button. Calibration Date and Test Date will bring up pull down calendars:

91

Latest Calibration

☐ Cannot be calibrated

File:

Select Date: > Date:

History

Delete

Latest Test

Tested By:

Results:

History

Add New

Delete

January, 2012

Sun	Mon	Tue	Wed	Thu	Fri	Sat
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Today: 1/23/2012

After the date has been selected, click on the right-arrow to save the information. IF YOU DO NOT CLICK ON THE RIGHT-ARROW, THE DATE WILL NOT BE SAVED.

Latest Test

Select Date: > Date:

History

Add New

Delete

Tested By:

Results:

January, 2012

Sun	Mon	Tue	Wed	Thu	Fri	Sat
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Today: 1/23/2012

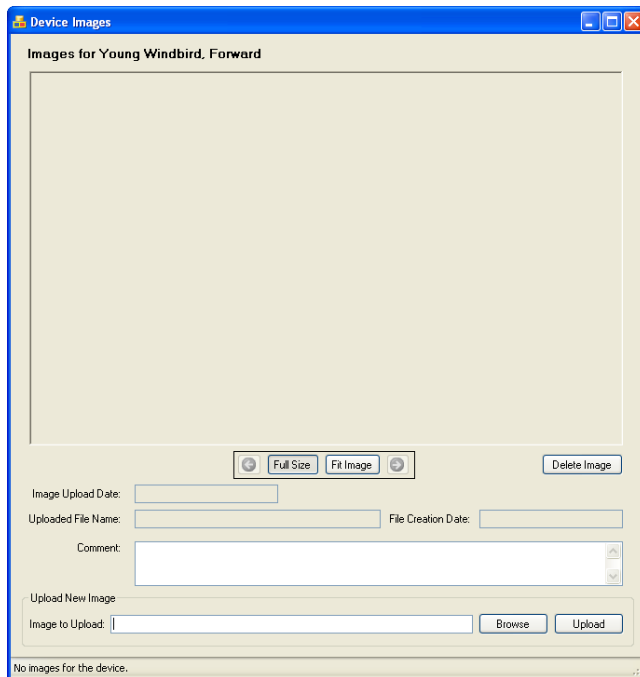
After filling in the device information, save the change and the inventory will show the physical device name. Note that the icon for the physical device will show an exclamation point if the calibration date is empty or more than one year old.

The device will continue to be listed as *Not Installed* at this time (see Installing)

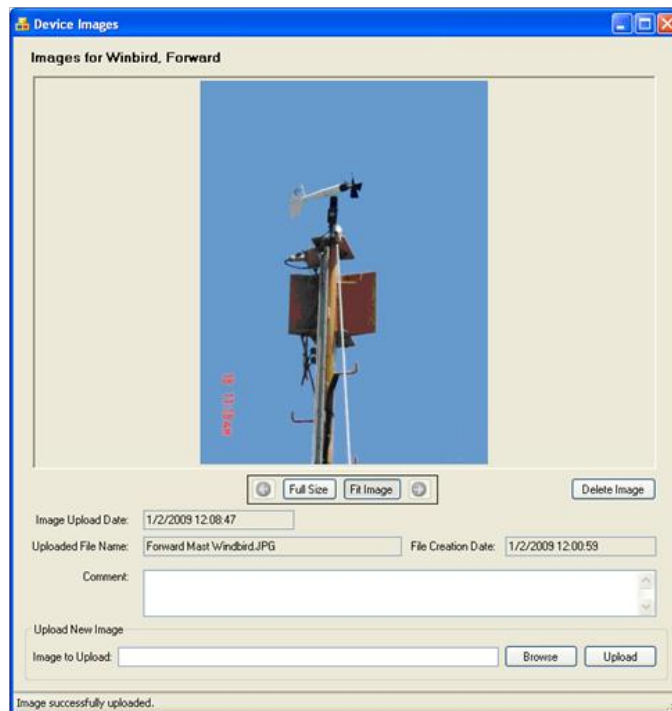
After the first device has been created, additional devices can be made by Inserting or Copying, or the order of a device can be changed by Cutting and Pasting

Adding Images:

Near the Top/Right of the Inventory page there is a button labeled Images. Clicking on this button will open a new pop-up



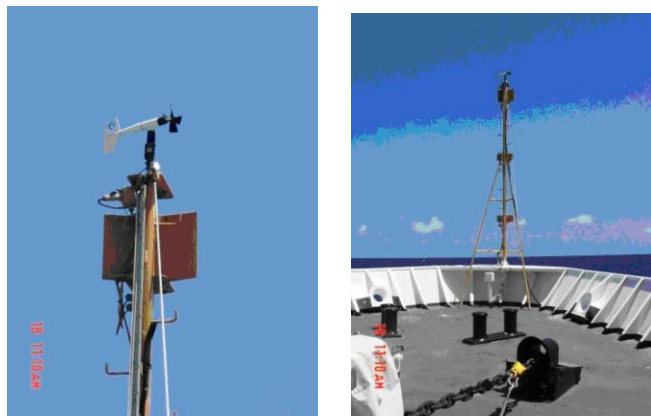
Selecting **Browse**, navigate to the image folder. Select the image file by double-clicking on it and then click **Open**. The selected picture is copied into the CFE inventory section of the SCS Database.



You may upload multiple images for each sensor, using the arrows under the image to move between them. Selecting between Full Size and Fit Image will shift the image shown to either be as large as it was or reduced in size to fit the box.

A note on good imagery: Including multiple images of the sensor, both close up and from enough of a distance that other landmarks on the ship are visible, is a good way to ensure that sensor can be correctly identified.

In the example below, the left image clearly shows that the sensor in question is an anemometer or the following actions are available by right-clicking on a Message Definition object.



The following actions are available by right-clicking on a Message Definition object.

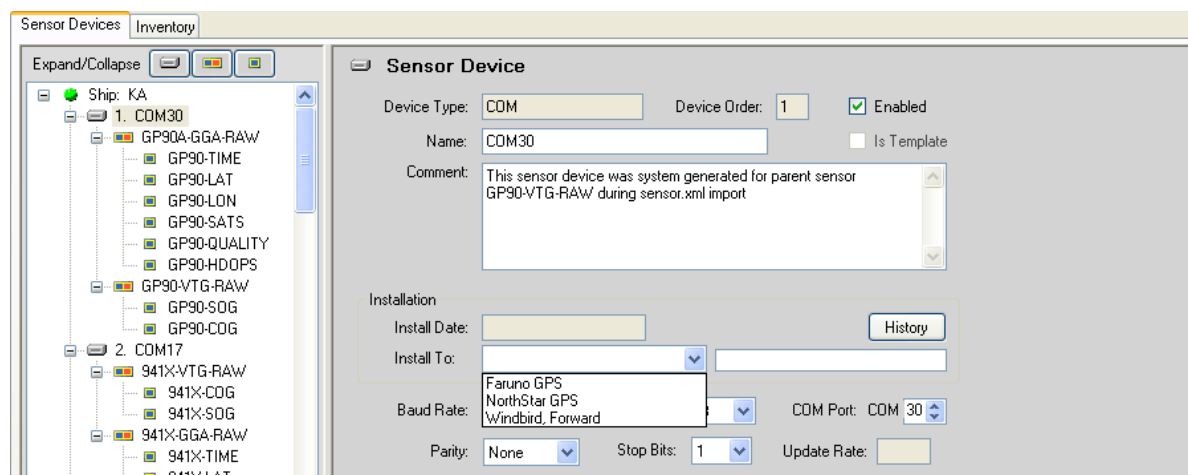
CFE-DB Sensor Device Parameters

Some Sensor Device parameters are common to all Types of Sensor Devices for example the name and installation parameters shown below.

Installing a Sensor Device

Installing a Sensor Device is the process of connecting a physical device (the box sending the message(s)) with the corresponding sensor device (describing the messages being sent).

To do this, populate the inventory, and then return to the Sensor Device menu. For any Sensor Device, there is a Pull Down menu *Install to:* NOTE: this pull down is available only for COM, Manual, Network or Polled COM device types.



This will list all currently *uninstalled* Physical Sensors. Select the Physical Device from the Inventory list that corresponds to the Sensor Device.

To replace an existing sensor with a new one, create the new Physical Device first and then under Sensor Device you can install the new device. After that can you delete the old device. (Note, an Installed Physical Device cannot be deleted or cut, but can still be copied)

Common Characteristics Setting Parameter Values

CFE-DB displays parameter values in *Check Boxes*, *Drop-down Lists* or *Edit Boxes*. Check Boxes display parameters that are in essence true or false. A parameter that assumes a limited number of values is displayed a Drop Down List. Some lists never change, as with the Stop Bits parameter; CFE-DB may also generate list values “on the fly” according to other information in the Sensor List, as with any list of base sensors. A parameter with wide ranging values is displayed in an Edit Box. Edit Boxes give you the freedom to enter anything you wish, but CFE-DB will check anything you type to ensure it is reasonable.

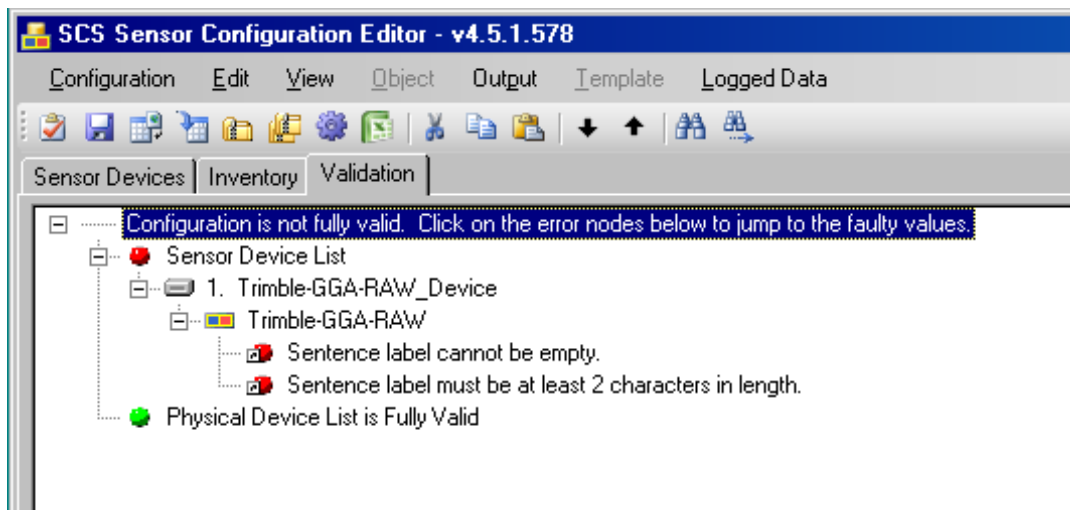
Once you have clicked your mouse somewhere in the sensor pane, the Tab and “Shift+Tab” keys are available to move from one parameter to the next in a regular order.

The Validation Tab

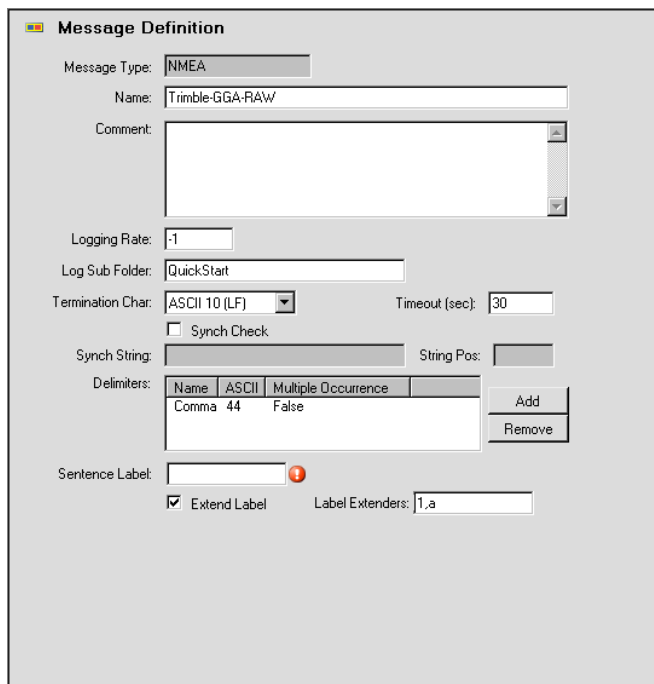
This tab has a list of all messages generated by the Validation process. Validation is invoked by selecting Validate from the menu or tool bar, or also by selecting Save or Publish. CFE-DB

auto-starts a validation when you click on the Validation tab. If there are any validation errors they will appear here. You may then select a line in the list and be transferred directly to the object definition where you can resolve the issue.

The following figures illustrate the procedure. An error has been introduced into the MX-200 message object by removing the sentence label value. When validation is selected the validate tab contains the following messages.



Double click on one of the “red” error messages and you are passed back to the message definition, with the red exclamation point marking the source of the error as shown below.



Message Definition

Message Type: NMEA

Name: Trimble-GGA-RAW

Comment:

Logging Rate: -1

Log Sub Folder: QuickStart

Termination Char: ASCII 10 (LF) Timeout (sec): 30

☐ Synch Check

Synch String: String Pos:

Name	ASCII	Multiple Occurrence
Comma	44	False

Add
Remove

Sentence Label: !

☒ Extend Label Label Extenders: 1,a

3.8 Preparing for ACQ Startup

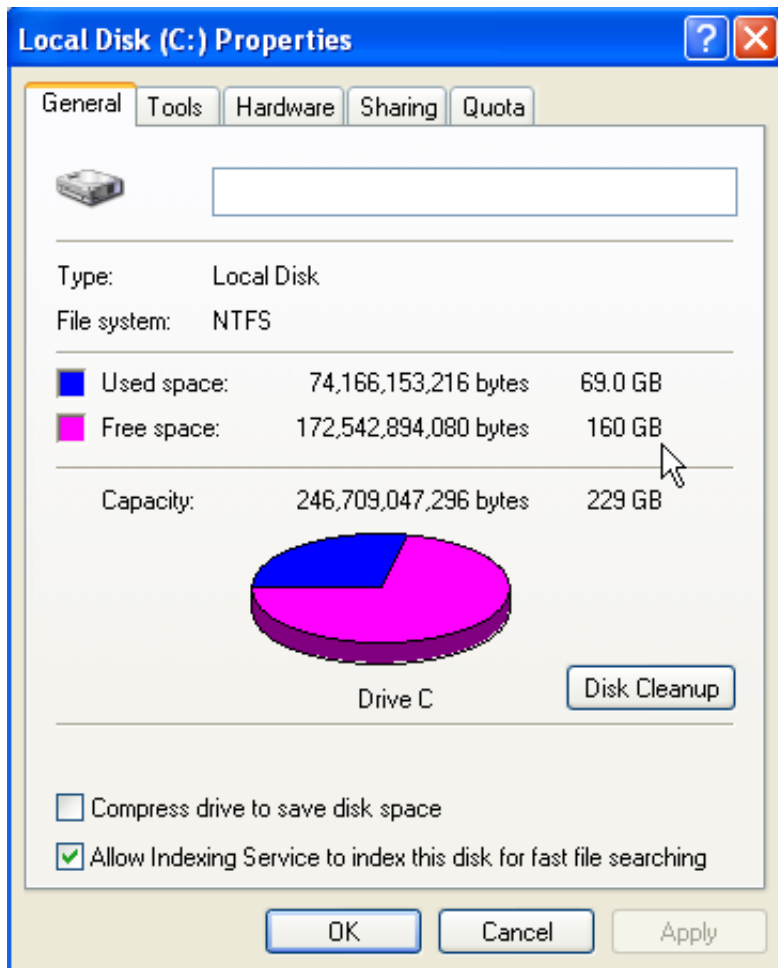
To start ACQ properly, the following guidelines should be followed:

- The following file must be in the logical SCSShip directory: DeviceConfiguration.XML. This file is a reformatted copy of the sensor configuration database, created by CFE-DB during the Export to XML function. This is the file that is read by ACQ.
- If ACQ is currently running it must be stopped prior to running ACQ again.
- All sensors should be interfaced to the system as described in the previous sections.
- Time on all computers should be synchronized to GPS network clock time if available or manually to GPS time otherwise. GMT is the assumed time zone for all SCS data.
- *SCS40.CFG* file should reflect the SCS40 directories and environment being used.

Check Free Disk Space on Logging Drives

You must ensure sufficient free space remains on the log drive(s).

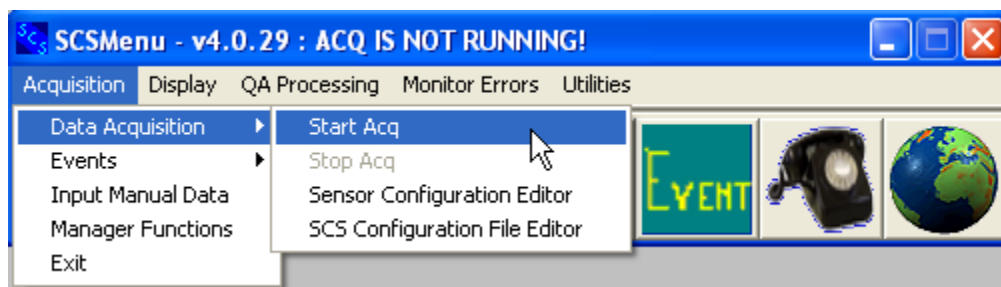
Note: ACQ shuts down logging autonomously when there is less than 5 MB free space remaining on the log disk. Thereafter, ACQ will continue to run when the disk is full but the status circle lights will no longer cycle to green and DATAMON errors will be seen.



You can determine the free space on a disk by opening Windows Explorer. Select the log disk drive, right click on it and select properties. You will see the following window describing the properties of the disk including disk space.

The secondary data disk (SCSLog2) should also be checked to ensure it is not full and accessible across the network if applicable. Checking disk space across the network may require physically connecting to that device.

3.9 Start ACQ Data Acquisition



Use this menu selection to begin the ACQ application, which acquires sensor data based on the sensors described in the Sensor Configuration File (SENSOR.SCF). ACQ stores the acquired data so that other programs may use the data. Once data acquisition has begun, data logging and various real time displays may begin.

ACQ Program Description

****NOTE ON TERMINOLOGY****

This manual describes ACQ functions using the out-of-date terms “*Parent Sensor*” and “*Child Sensor*”. The current terminology is “*Message*” and “*Data Field*”, respectively.

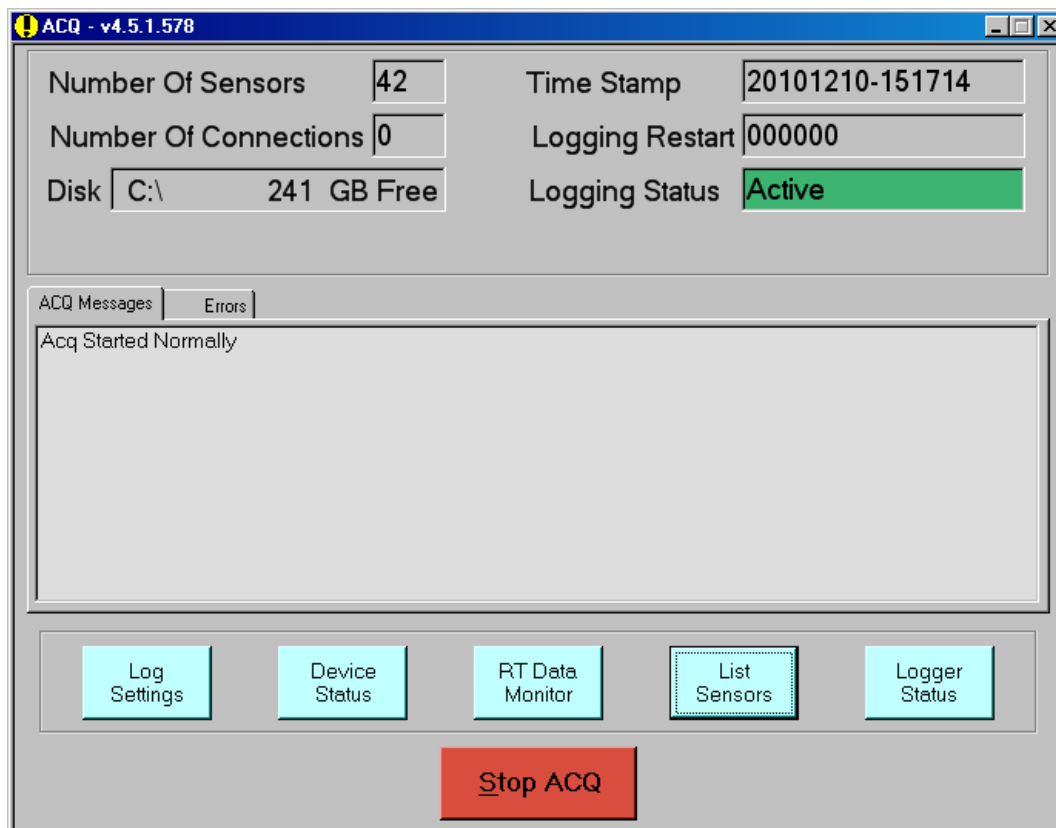
The ACQUISITION program (ACQ) is the heart of the SCS system. This is the program that acquires all sensor data, extracts the sensor elements from the parent serial messages and calculates the values of the derived sensors. Any and all data manipulated by the SCS system is first acquired by the ACQ program.

The ACQ program execution is based on the information located in the Sensor Configuration Files called SENSOR.SCF and the SENSOR.XML (located under the SCSShip logical directory). It is the configuration file where all of the sensors and sensor elements in the system are described by a common set of parameters.

For a detailed explanation of this file, please refer to Section 1.3. SENSOR CONFIGURATION FILES, SENSOR CONFIGURATION XML FILE AND THE CONFIGURATION FILE EDITOR. The ACQ program will run indefinitely until terminated by the user. When this program is terminated, all current I/O requests and all of the SCS programs currently using data from the global section of memory are terminated as well.

Starting ACQ: the Main ACQ Window

When these guidelines are met, the data acquisition process may begin. To start, click on the button to Start Acquisition as shown above. ACQ will immediately start and the following window will be seen:

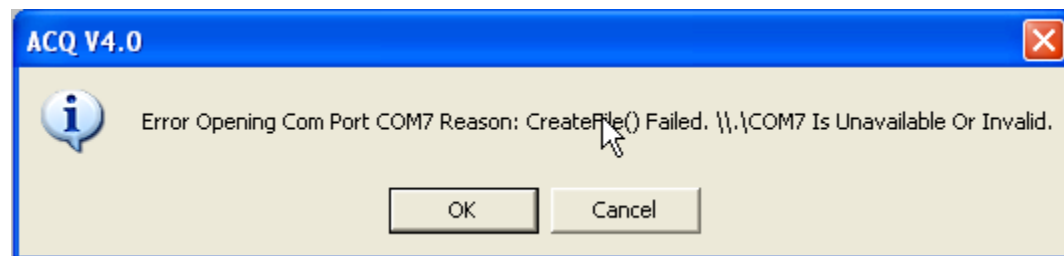


There is a tab sheet with ACQ Messages and Errors tabs

ACQ Messages tab is a window containing all ACQ messages is presented at the center of the main ACQ window. The following messages should be seen when ACQ starts normally:

- ACQ Started Normally: The program has successfully read the SENSOR.XML configuration file and allocated all I/O input devices as described by the SENSOR.SCF
- Sensor Device Objects: A Sensor Device Object is created for each physical device in the system.

In the event that ACQ detects a problem regarding a sensor's device an error message will be shown as follows:



ACQ will continue to run with all possible sensors even if one or more sensors have failed. Careful inspection of ACQ and associated windows should be made on startup to ensure all sensors are being collected.

After all of the Sensor Device Objects are created successfully:

- ACQ then Opens all of the Raw Data Files on the primary and secondary Log Folders
- ACQ then starts the internal data server to communicate with client applications.
- ACQ listens on each sensor device for sensor messages and acquires them as Parent Sensors
- ACQ parses all Parent Sensors for Child Sensors

The ACQ Main window also displays the following information:

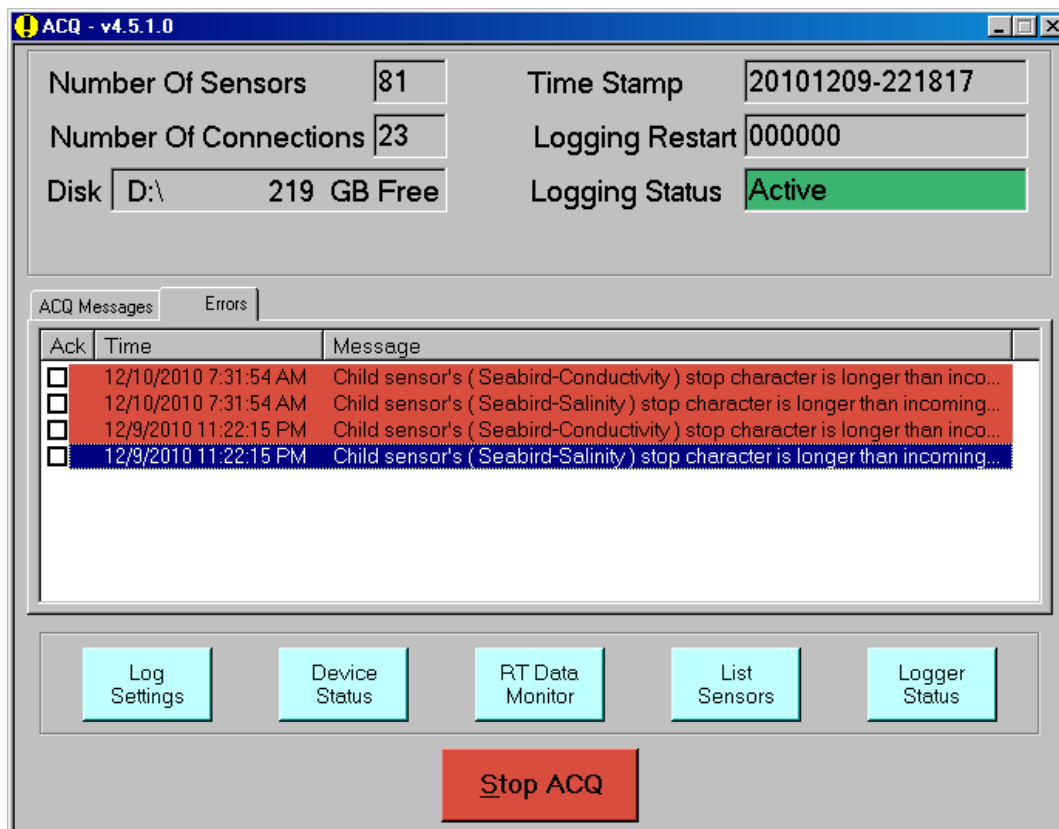
- **Number Of Sensors:** displays the total number of sensors ACQ is acquiring both parent and child sensors.
- **# Connections:** The number of client connections active to the internal ACQ data server
- **Disk Space:** The number of gigabytes free on the primary logging disk. When the free space falls below one GB the display shows the number of free megabytes on the disk.
- **Backup Disk Space:** If SCSLOG2 is defined in the SCS Configuration File a second text box is visible underneath the primary logging disk free space.
- **Time Stamp:** The actual time stamp used for all files created by ACQ. This also equates to the start time of ACQ.
- **Logging Restart:** Describes the status of automatic logging stop/starts.
- **Logging Status:** Describes if logging is presently active.

Monitoring ACQ Operations

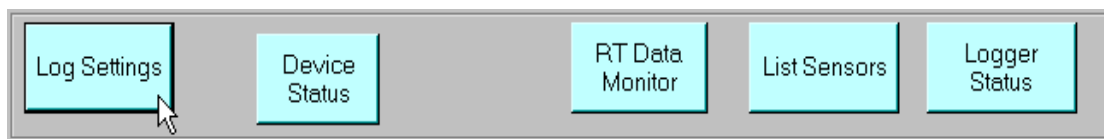
There are two major aspects to monitor: data acquisition and data logging. In the ACQ window there is the Error Tab and the Blue Buttons

3.9.1 The “Errors” Tab

The Errors tab in the ACQ main window shows errors as they happen. Here is a sample



You may right-click in the message list to select a number of options such as get more detail, etc. These messages are also available in the “log file” in your data logging folder with the name “SCS <version> <timestamp>.LOG. See also the ERRORLOGTODISK parameter in the SCS Configuration File.



The button bar at the bottom of the ACQ window provides quick access to monitor and control the various aspects of ACQ operations.

3.9.2 Monitoring ACQ Data Acquisition

Three buttons in the button bar look at various aspects of data acquisition:

- RT Data Monitor
- List Sensors
- Device Status

3.9.3 RT Data Monitor



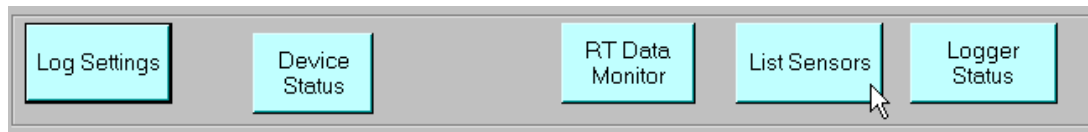
The REAL TIME DATA MONITOR button launches the client app REAL TIME DATA MONITOR providing a quick view of all data being collected by ACQ. A sample data Monitor screen is show below. This program is discussed in detail in chapter five of this manual.

The screenshot shows the 'RealTimeMonitor - v4.0.22' application window. The title bar includes standard Windows window controls (minimize, maximize, close) and menu options 'Display' and 'Help'. The main area contains a table with two columns: 'Sensor Name' and 'Sensor Data'. The table lists ten sensors, each preceded by a red circular icon. The sensors are: GPS1Mx200-GGA, GPS1Mx200-VTG, Youngf-TrueWind-Raw, Youngf-WindSpeedTrue, Youngf-WindDirTrue, AvgWindRaw, GYRO-HDT, YoungWind-forward, Math-Plus-Raw, Math-Subtract-Raw, Math-Times-Raw, Math-divide-Raw, avg, and WeatherObs. The 'Sensor Data' column is currently empty.

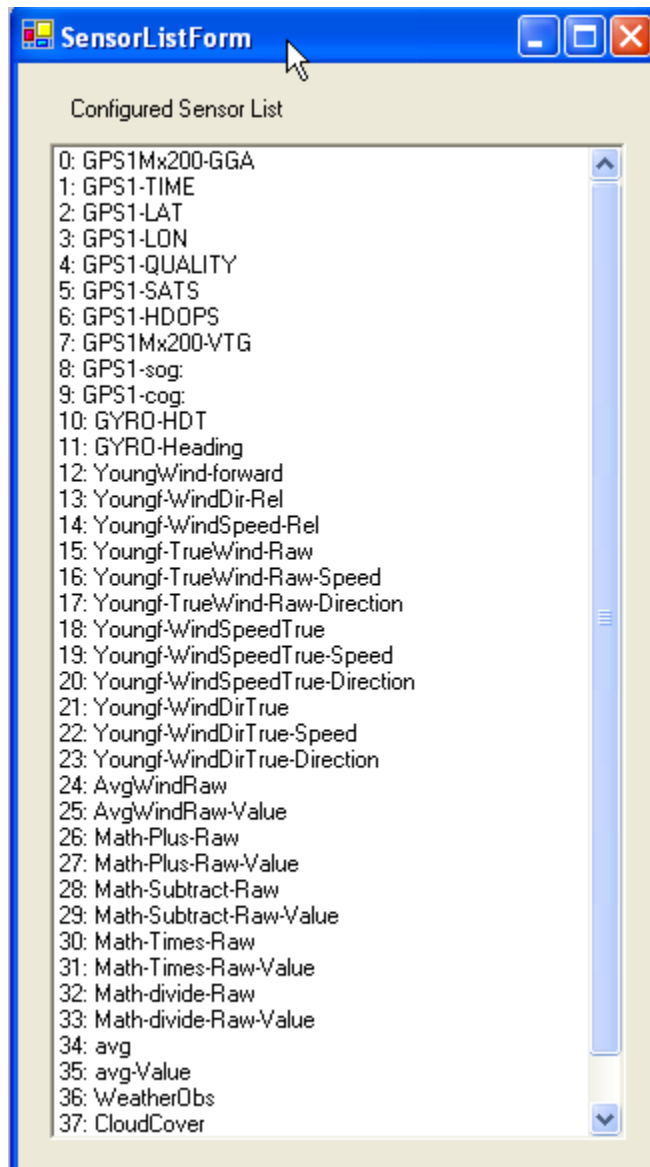
Sensor Name	Sensor Data
GPS1Mx200-GGA	
GPS1Mx200-VTG	
Youngf-TrueWind-Raw	
Youngf-WindSpeedTrue	
Youngf-WindDirTrue	
AvgWindRaw	
GYRO-HDT	
YoungWind-forward	
Math-Plus-Raw	
Math-Subtract-Raw	
Math-Times-Raw	
Math-divide-Raw	
avg	
WeatherObs	

An 'Exit' button is located at the bottom right corner of the window.

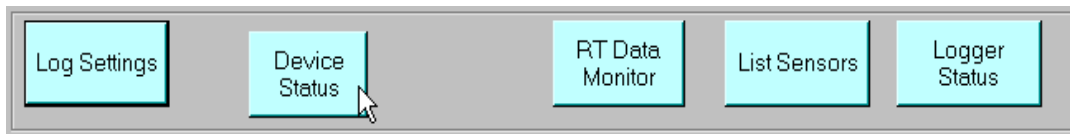
3.9.4 List Sensors



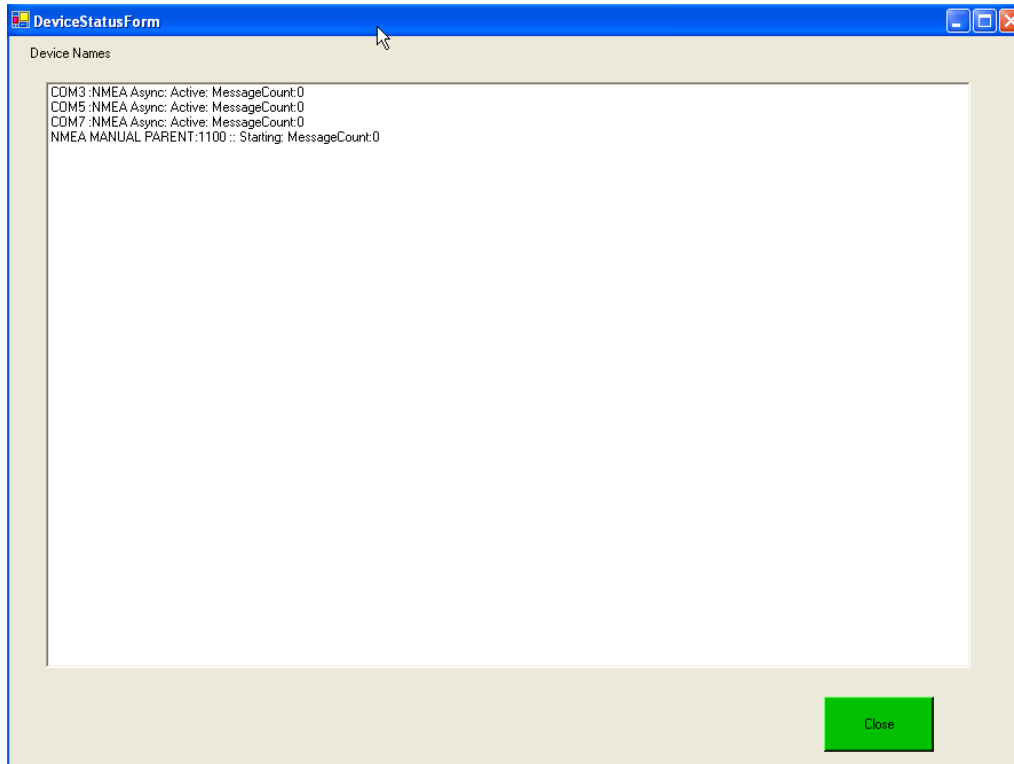
The “*LIST SENSORS*” button provides a quick list of all sensors in the sensor configuration file which have their “Enable” box checked. This list includes parent and child sensors. A sample sensor list appears below:



3.9.5 ACQ Device Status



The DEVICE STATUS button provides a window to show the historical and current status of all devices being controlled by ACQ. A sample screen is shown below:



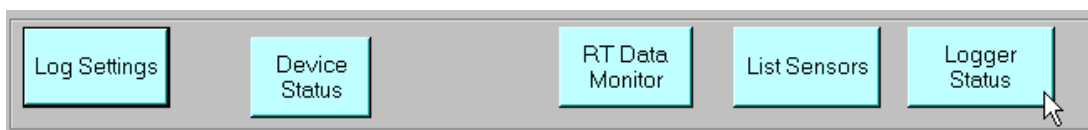
Monitoring ACQ Data Logging

Previous versions of SCS (v1 – v3.3c) had separate applications to control acquisition (ACQ) and data logging (Logger). In SCS 4.0 both activities are handled by a single application, ACQ.

The ACQ button bar includes two buttons to monitor and control the various aspects of data logging:

- Logger Status monitors logging activities
- Log Settings controls data logging operations

3.9.6 Monitoring ACQ Logger Status



The **LOGGER STATUS** button is used to display the status of all log files presently active in ACQ.

NOTE: This window shows status of logging to RAW files. It DOES NOT show logging status to the database (if it is enabled).

This pops up the Logging Status window. The status window may be closed at any time without affecting logging operations. A sample status window appears below:

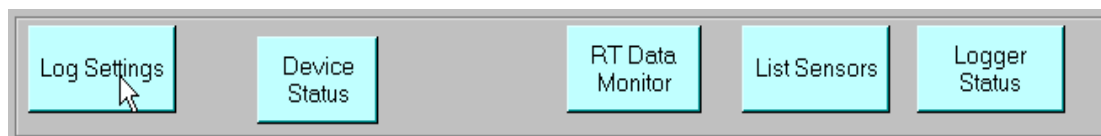
	Raw File Name	Log File Time	Log Flag	Log File Size
●	c:\Datalog\GPS1Mx200-GGA_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/GPS1Mx200-GGA_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\GPS1Mx200-VTG_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/GPS1Mx200-VTG_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\GYRO-HDT_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/GYRO-HDT_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\YoungWind-forward_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/YoungWind-forward_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\Youngf-TrueWind-raw_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/Youngf-TrueWind-raw_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\Youngf-WindSpeedTrue_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/Youngf-WindSpeedTrue_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\Youngf-WindDirTrue_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/Youngf-WindDirTrue_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\AvgWindRaw_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/AvgWindRaw_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\Math-Plus-Raw_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/Math-Plus-Raw_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\Math-Subtract-Raw_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/Math-Subtract-Raw_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\Math-Times-Raw_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/Math-Times-Raw_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\Math-divide-Raw_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/Math-divide-Raw_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\avg_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/avg_20051211-195855.Raw	00010101-000000	0	0
●	c:\Datalog\WeatherObs_20051211-195855.Raw	00010101-000000	0	0
●	c:/Datalog2/WeatherObs_20051211-195855.Raw	00010101-000000	0	0

The Logger Status window has columns describing information about each of ACQ's log files:

- The first column is a status indicator: The circle is red when no data has ever been logged to the file. This is the case for all log files when the program is first started. The status circle blinks green when data has just been written to disk. And the status circle turns **yellow** when no data has been written to the disk recently. Yellow circles are valid provided that they change to green on a regular basis. All red or all yellow circles for any significant period of time indicate a logging or system problem.
- Raw File Name: The full file name of the RAW file created by the ACQ program

- Log File Time: The time the last piece of data was logged (Log Time) to each file
- Log Flag: The number of records written to the log file.
- Log File Size: The number of bytes written to the log file.

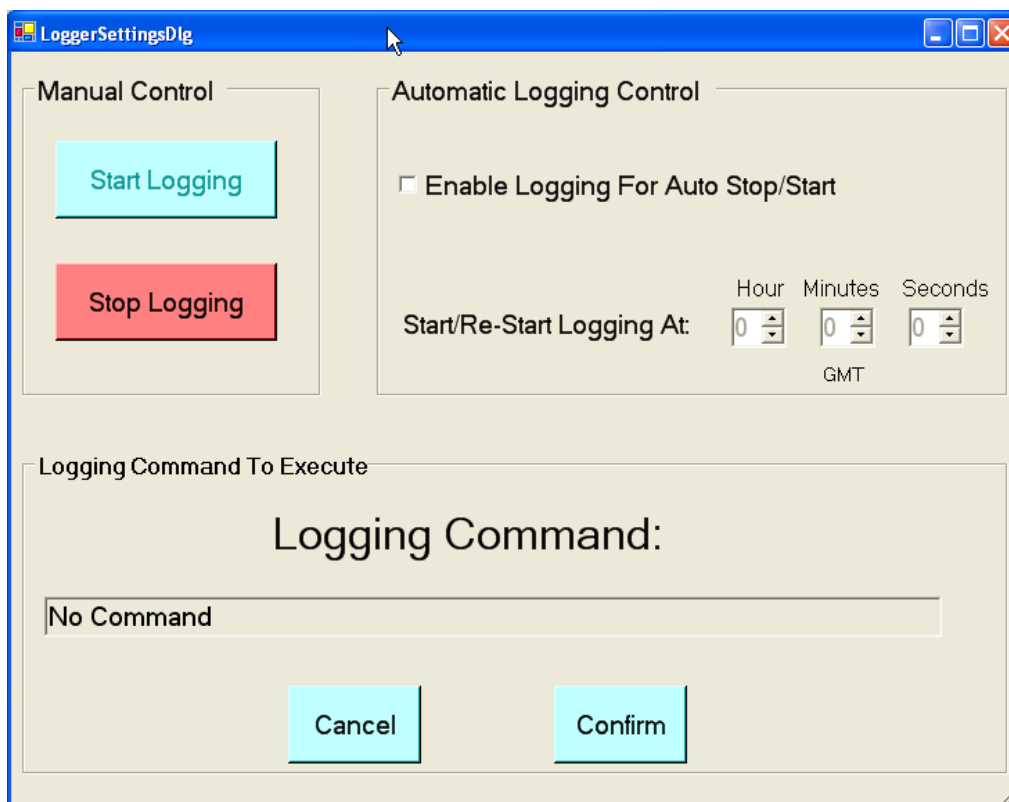
3.9.7 Controlling Logger Operations



This discussion covers logging to RAW files. It does not cover logging to the SCS database.

ACQ is capable of logging all data to a single set of files for the duration of a vessel's cruise, but you can override this in the Log Settings.

The Log Settings button in the ACQ button bar brings up the Logger Settings window. A sample window is shown below. This window has controls to start and stop the logging of all raw data files, and then start the logging up again.



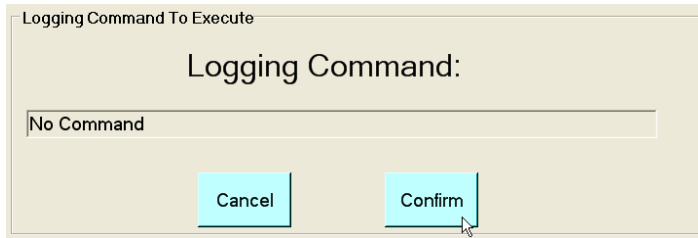
In general, when logging stops all raw files are closed out permanently. Recall that all raw files have a common timestamp built into their names, the time that logging was initiated. Then when logging is restarted a new set of files is created with a different time stamp, the time that logging was restarted. This allows access to the old raw files while the new files are being written. The

ACQ Time Stamp on the main window will be updated with the new timestamp of current log files whenever logging is restarted.

NOTE: When Logging is stopped data is still acquired and made available to client display apps.

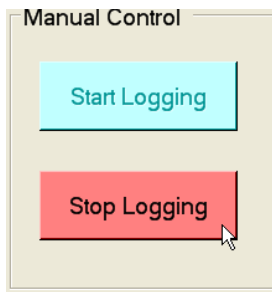
Manual Controls allow you to start and stop logging at will. Auto start/ stop allow you to automatically stop logging once a day at the desired time and immediately restart logging.

- Logging Commands Must Be Confirmed



You will be prompted for verification of any change to the logger. You must press the CONFIRM button to cause the log command to be executed. Changes will not be executed if you press the CANCEL button.

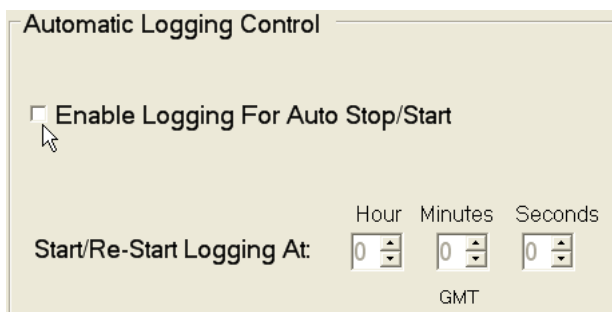
- Control Logging Manually



The Manual Control buttons provide you the means to stop and restart logging raw data files at any time. Usually this is done when it becomes a mission priority to close the files to gain access to critical data just collected. You should immediately re-start Logging to ensure data collection resumes without loss of data.

- Control Logging Automatically

The Automatic Logging Control handles the settings for ACQ to close and re-open log files on a periodic basis such as once a day at noon or midnight.



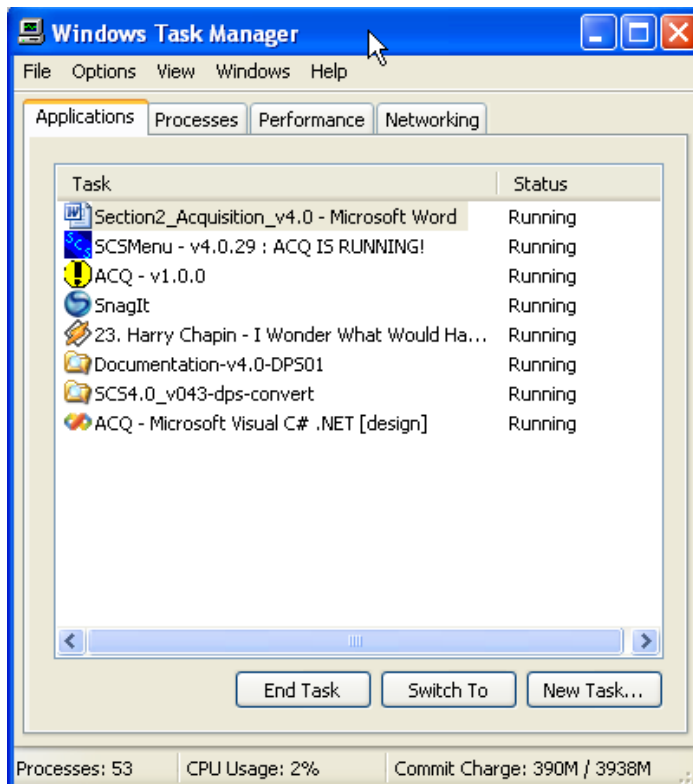
If automatic logging is enabled then at a specified time all log files will be closed and reopened with a new ACQ time stamp. This allows the files to remain of a size that is manageable by most programs and easier to back up.

By default, stopping ACQ and restarting will wipe out any restart time you have set here. But the parameter `AutoStartLogging` in the SCS Configuration File allows you to set the changeover time and have it persist over ACQ restarts.

ACQ Troubleshooting

If the Acquisition program started up with errors, in which the “ACQ Started Normally” message did not appear or if the system starts dramatically slowing down for a sustained period of time on the initial startup of Acquisition, there are certain steps that should be taken to STOP acquisition manually. These steps go as follows:

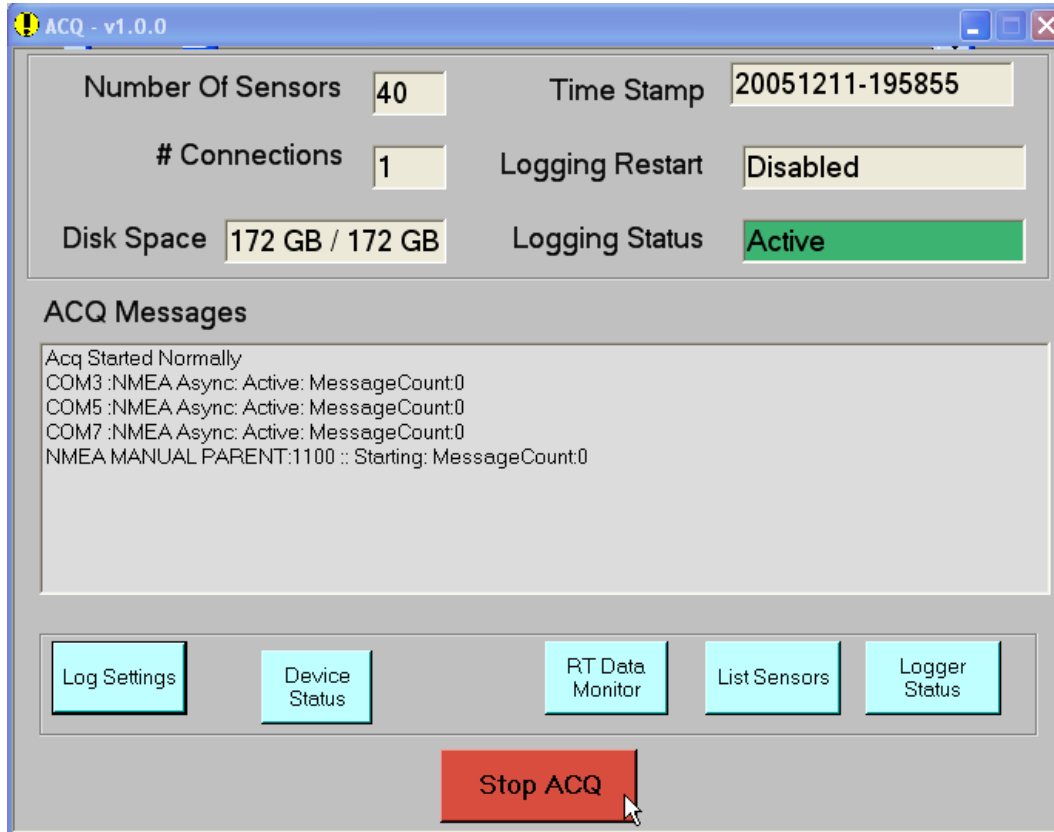
1. EXIT out of the SCSMENU - which is done by selecting EXIT in the ACQUISITION menu. And Retry running ACQ
2. Check that the Sensor Configuration file is correct and specifies properly working serial ports.
3. Check the Serial Ports with HyperTerminal, or with the SCS Serial Port Diagnostic app in the SCS Utilities menu.
4. Check if ACQ was stopped properly, Use the Task Manager to locate the Applications Running and look for ACQ. Stop This Application if it is running and then restart ACQ from the Menu.



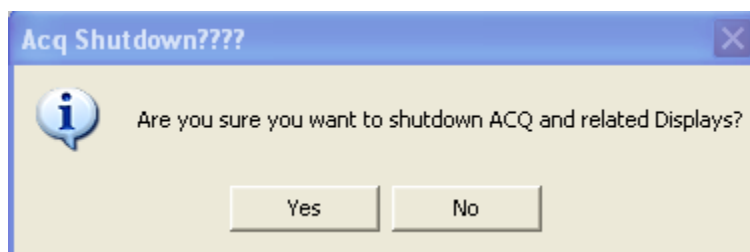
Stopping Acquisition

To stop ACQ:

- Click on the red Stop ACQ button, or
- Click on the ACQ window's Exit box at the right of the Title Bar.



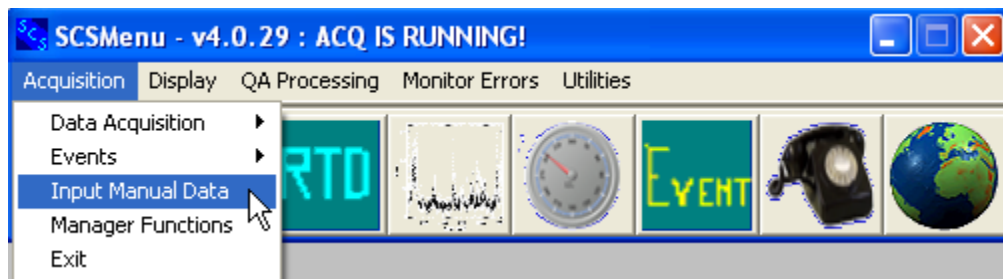
In either case confirmation is required to begin the shutdown process.



When you stop the ACQ program, the following things happen:

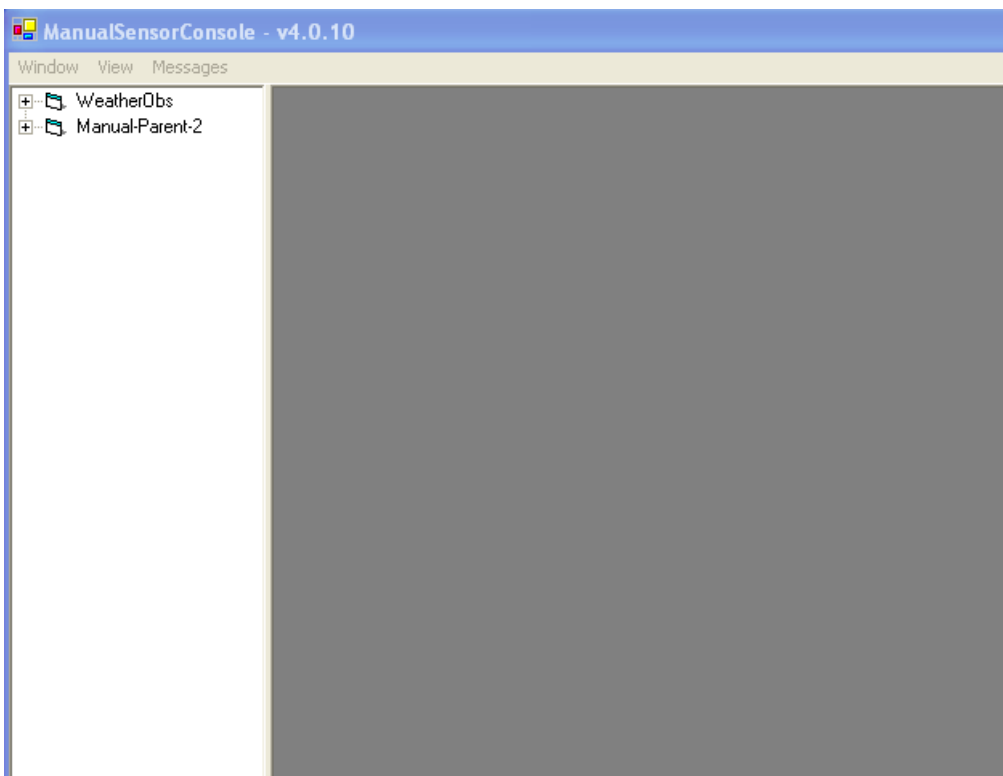
- Data Logging stops and raw sensor files (*.RAW) are closed.
- All client programs (displays, charts, etc.) terminate themselves when their connection to the ACQ Data server has been lost. Their windows disappear from the desktop.

3.10 Input Manual Data

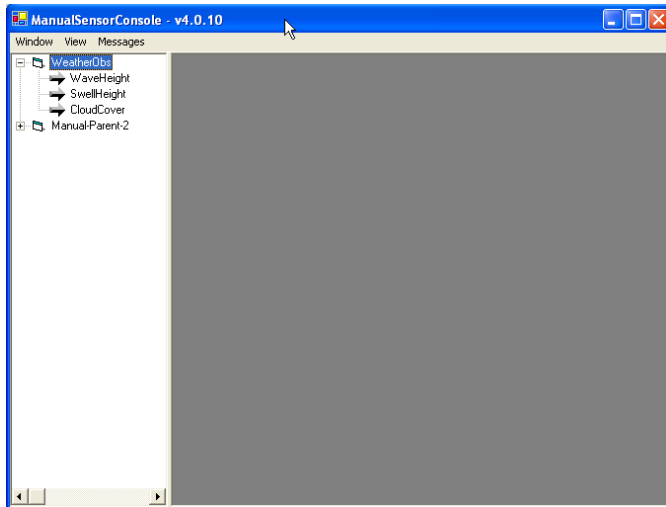


Function

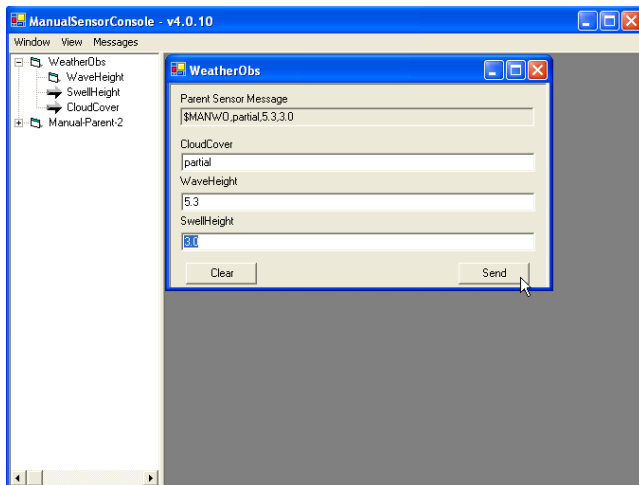
This will allow the user to specify values for all manual sensors, which then acquired and logged into the SCS system.



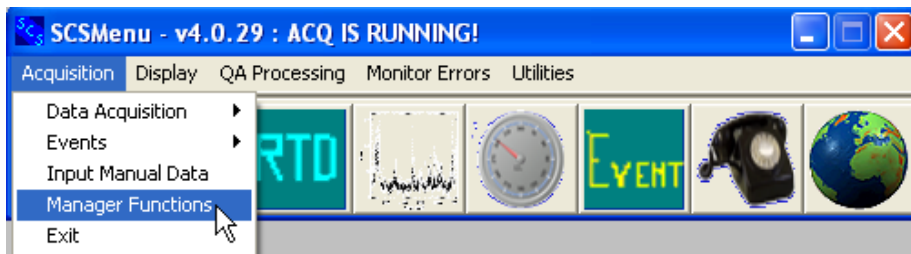
This program displays all sensors available for manual data entry, i.e. all NMEA Parent sensors whose Data Source is MANUAL. The program operates as a client application using a dedicated socket to ACQ for communication. When the program starts it asks ACQ for a list of all manual sensors and then displays these sensors in a tree format on the left of the main window. Manual sensors may be setup in multiple parents connected on different ACQ network ports. Each Discrete Manual Parent is shown as a separate group. These groups are represented as folders in the tree.



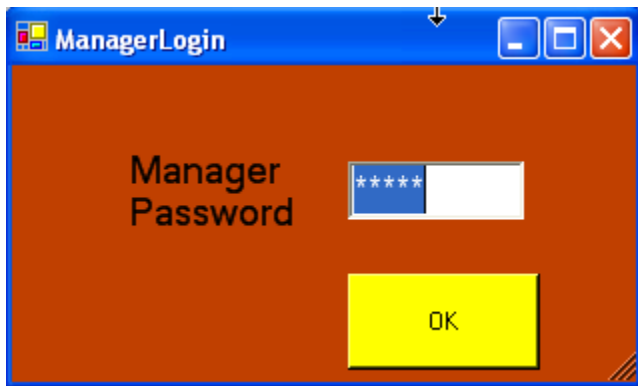
The user selects the group and sensor to update by double clicking the sensor. This activates a dialog window for entering the manual observation. The user then fills in one or more data fields with the appropriate observational data. Note while the user is entering data a NMEA style parent sentence message is being developed in the first text box. When the user hits the send button the parent message is transmitted to ACQ over the network socket and is treated like all other parent sensors with the child sensors being parsed and distributed. The user is not required to enter all child data. Only the data entered will be updated by ACQ.



3.11 Enable/Disable Management Functions

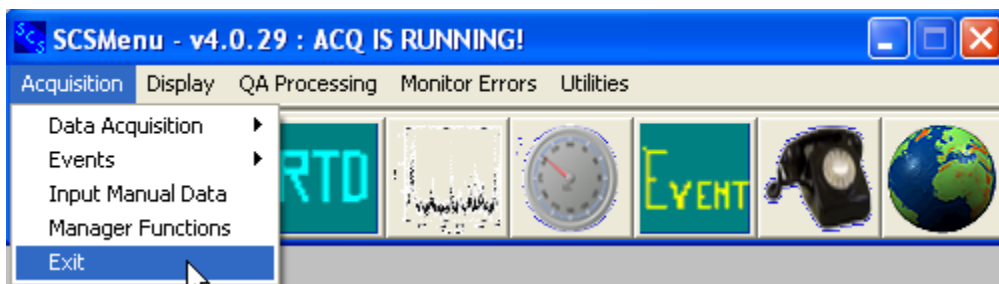


This mini app enables the ACQ and CFE-DB apps to run. The SCS Menu is protected by this manager function from allowing general users from selecting menu items that may corrupt or damage the SCS system while it is collecting data.



- Once this window is displayed on screen, move the cursor with the mouse over the window and proceed to enter the manager password. Note the following:
- The manager password is kk00l (case sensitive). The password is hardwired into the app and cannot be changed by a user or administrator.
- The letters being typed will show up as ‘*’ characters to ensure secrecy. Press the OK button to complete the login dialog.
- If the password is correct the manager’s options will be enabled and a check box will be shown to the left of the manager function menu. Otherwise no visual clue will be shown to indicate an incorrect password entry.
- Selecting Management Functions a second time and entering the correct password will disable the functions in the menu.

3.12Exit Menu



This menu selection stops the SCSMenu app. Stopping SCSMenu does not affect the operation of other SCS apps that may be running. If necessary you can restart SCSMenu at any time without affecting other SCS apps in any way.

CHAPTER FOUR

EVENT LOGGING

4 Event Logging

4.1 Introduction

SCS offers two styles of data logging. The traditional style is continuous ACQ logging, which records complete raw sensor data in RAW files. Each sensor is logged to its own file, and in post processing it is necessary to correlate the contents of any two sensor log files using timestamps.

The second style is *Event Logging*, which provides a means of collecting data sets that are completely separate from the raw data collected by the system. In a typical situation, you may only be interested in sensor data collected *only* during specific times within the cruise. Typically these times represent specific *events* that may take place during the cruise, such as a CTD cast, a trawl tow, a balloon launch, a buoy deployment, etc. Even Logging enables each and every scientist or end-user to collect a customized set of data during a cruise. At the end of the cruise, the user will walk away with exactly the data he/she desires, without the need to sort through the entire raw data set.

Starting with SCS 4.3 two methods of Event Logging are available:

- The “*Legacy Method*”, which has been part of SCS for several years.
- The *Operation Event Logger* or OEL for short.

All aspects of the legacy event builder and event logger remain unchanged.

Section 4.2 and Section 4.3 cover “Legacy” Event Builder and Event Logger. Sections 4.6 and 4.7 cover *the OEL Builder* app and the OEL app.

Differences in the Two Methods of Event Logging

There are several major differences between the two methods, which are covered in more detail in the sections to follow.

- **Data Files.** The legacy logger writes data to flat files with .ELG extensions. The OEL writes data to the SQL database.
- **Template Files** do not exist in OEL. All data about the definition of an event is stored in the SCQ database.
- **Meta Items:** Summary metaitems are not available in the OEL; Manual and sensor metaitems are still available.
- **Buttons:** OEL button setup is much more sophisticated than in the legacy event logging. It is possible to control in fine detail the order in which buttons can be pressed, and how many times they can be pressed. This allows fine control of how an event is to proceed.

4.2 Event Logger, “Legacy” Method

Event Logging is accomplished using two applications: the *Event Builder*, with which you define an event, and *Event Logger*, which you use to collect data and otherwise manage event operations. The Logger in turn makes use of the ACQ data server to access any sensor data it may need.

A third app (not yet available) is the *Event Logger Manager* can be used to conveniently start, monitor and stop multiple event loggers from a single window.

In this documentation, *event* refers to a particular operation on the ship, such as a fish trawl or a CTD cast. SCS Event Logging allows the user to define specific actions within an event, such as “CTD in water”, “Net deployed”. For any action you can choose to record the current value of sensor data, manually entered data, or clock time. When the event is run, each action in the event is marked by the user by clicking on buttons in the Logger GUI.

Event Components

You use the Event Builder app to construct a template for an event. In the template there are five major components: Meta-items, Outputs, Buttons, Monitors and Alarms. You use the Builder to create these components and establish relationships between them.

The following sections give introductory component descriptions and provide explanations of terminology crucial to your understanding. Full details about components and their interactions are provided later in this Guide.

The different components are closely interdependent, so understanding one component requires the understanding of the others. Therefore, it is suggested that you read through the five component types once for an overview, and a second time to fill in details. The following figure illustrates the text. In the figure components are enclosed in ellipses and external entities are enclosed by rectangles.

Italicized words are standard terminology that is used throughout this document.

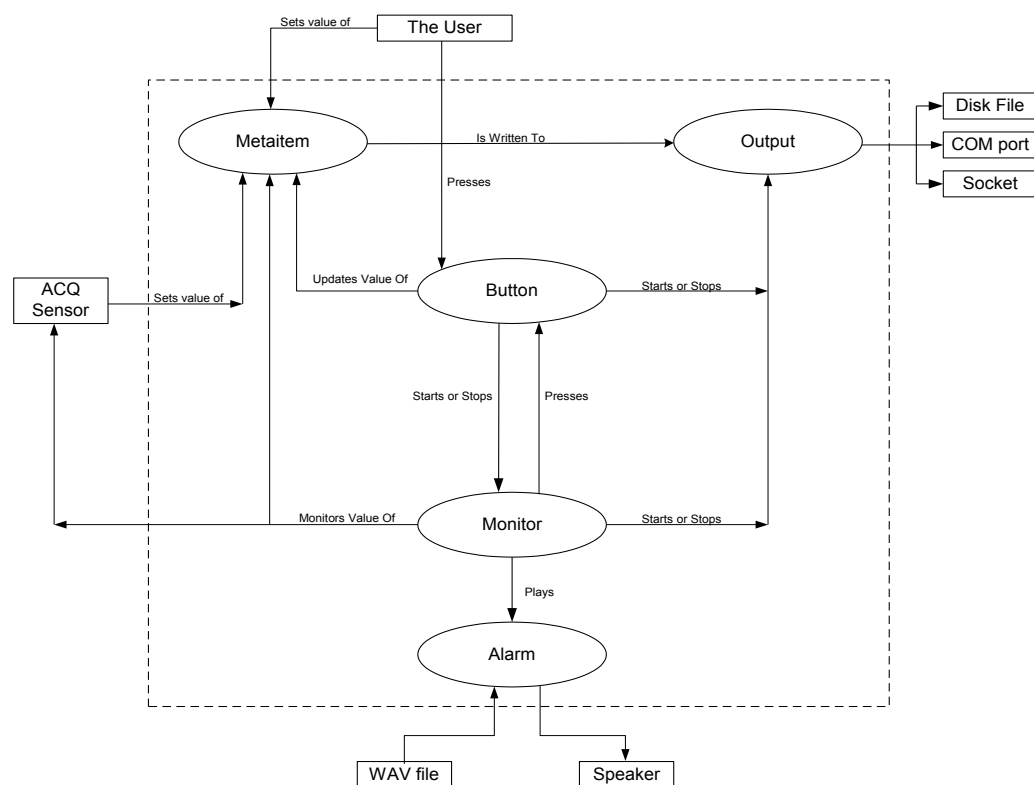


Figure 2.1
Event Components And Their Relationships

4.2.1 Meta-Items

Meta-items are atomic pieces of data. The heart of each meta-item is its value, which is written to data files or monitored by the Logger during an event. There are three types of meta-items:

- A *manual meta-item*'s value must be entered by you from the keyboard during or before an Event. The Builder allows you to define different data types (numbers and strings), place limits on the range of values, or define a dropdown list of values from which the user must choose in the Logger.
- A *sensor meta-item*'s value comes from a sensor being acquired by the SCS Acquisition Data Server. The Logger obtains this using information saved in the Event Template file and in the Sensor Configuration File created by the CFE-DB application. SCS acquisition software must be started before you run any event defining a sensor meta-item.
- A *summary meta-item* value is computed automatically by the Logger at the end of an event. Using the Builder, you define the type of value to be calculated (average, minimum, time duration, etc.), and which meta-items or sensor data form the basis of that calculation.

4.2.2 Outputs

The function of an *Output* is to log data (meta-items or sensor values). You can set up multiple Outputs, writing different sets of data to different files during an Event. Individual meta-items / sensor values are known as the *elements* of the Output.

During an event the Logger logs its elements as a set of time stamped, comma delimited values. The element set is written in ASCII format to the Output's *destination*, which is typically a disk file, but may also be a COM port or a TCP/IP socket. Outputs may be individually started, stopped and restarted during an event. Once started, logging may be *continuous* (elements are written periodically) or a *snapshot* (only one set of elements is written, then the Output is stopped).

4.2.3 Buttons

Buttons appear as controls on the Logger's Graphical User Interface (GUI). Typically you will click on a button in response to external events that you have predefined, such as "Net in water".

When you define a Button in the Builder, you assign meta-items, outputs and monitors to it. Then, when you click on a button Logger performs these actions:

- Updates the values of assigned meta-items.
- Starts or stops logging to assigned Outputs.
- Starts or stops assigned Monitors.

You may also arrange for a Button to be automatically clicked for you by a Monitor, when certain conditions arise.

4.2.4 Monitors

Using Builder, you define a *Monitor* to watch for a *condition* and make *responses* for you. During an event the monitor will then act on your behalf so that you can step away from the console. The condition can be one of the following:

- A length of time has passed,
- A sensor value goes out of range, or comes back in range,
- The ship's position enters or leaves any of several circular areas.

In Logger, a Monitor *responds* to a condition by pressing a Button and/or sounding an audible Alarm. A monitor is started or stopped when you press any button to which it has been assigned.

4.2.5 Alarms

You define an *Alarm* in the Builder by specifying the name of a WAV file on disk. WAV is an industry standard format for digitizing and recording soundtracks. Logger will play this sound whenever an associated Monitor is responding to a condition.

Files Created By the Event Builder and Event Logger

Several files play important roles in the Event Logging environment.

Event Template File

The Event Template file defines the event. The Builder creates or edits it in the SCSEventData logical directory, with extension (*.TPL). Logger reads template files to set up its GUI and logging operations.

Event Data Files

Also called “Log File”. Logger writes these files during an event; they contain a series of time tagged meta-item and sensor values as defined in the Output that writes it. Event Data files are ASCII and comma delimited, suitable for import into a spreadsheet. Each line in the file is terminated with <CR><LF> characters.

Entries in this file have the form “02/28/2001,10:26:31,001,210102,102615, ...” where “02/28/2001,10:26:31” is the timestamp (which is always there), and “001”, “210102”, and “102615”, are sample elements of the Output associated with this file.

Knowledge of the list of the output elements is required to be able to identify each value. Therefore, the first line in the file is a comma-separated list of the elements being written. This line will form the column headers when the file is imported into a spreadsheet.

Event Data files reside in a subdirectory of the SCSEventLog logical directory – the subdirectory takes its name from the name of the template file. The path to this file is has the following form:

“C:\DataLog\EventData\TrawlEvent\TrawlData_006.ELG”

where “C:\DataLog\EventData\” is the value of SCSEventLog in the C:\SCS.CFG file, “TrawlEvent” is the name of the event template, and “Trawl” is the name of the Output File defined in the template, and “_006” is an event index number auto incremented by Logger every time it runs TrawlEvent. The event number is always three digits, leading zeros as needed. If the event number exceeds 1000 four digits are used in the file name.

Event Header File

Written by the Logger at the end of an event, it contains the current values of all manual and sensor meta-items, followed by the computed values of all summary meta-items. The entries in this file have the form

CruiseNumber=0045

Where “CruiseNumber” is the name of a meta-item and “0045” is its value at the end of the event.

Header files reside in the same directory as the Event Data Files, with HDR extensions. The name of an Event Header file follows the same convention as the Event Data file – Logger creates Header and Data files as a set, with the same index number, so they can always be matched.

Button Activity File

Button Activity Files have a record of all button presses made during an event. They reside in the same directory as the Event Header and Data Files. The file name has the form

ButtonActivity_006.TXT

Where the “_006” is the event index that matches the Data and Header files. A typical entry in this file looks like:

03/06/2001,20:34:58, Start Event Button has been pushed.

The entry has a time stamp and the name of the button that was pressed.

Event Save File

Logger creates this file and saves the value of all meta-items to it at the end of an event. When you run the event again a meta-item will be initialized to this value if its “Restore Previous” property was checked in the Builder.

This file is written to the Event Template folder under the Ship Directory. The file is named after the event name and has an ESL extension. It is written when Stop Event is pressed, and is always written even if no metaitems are marked for restore.

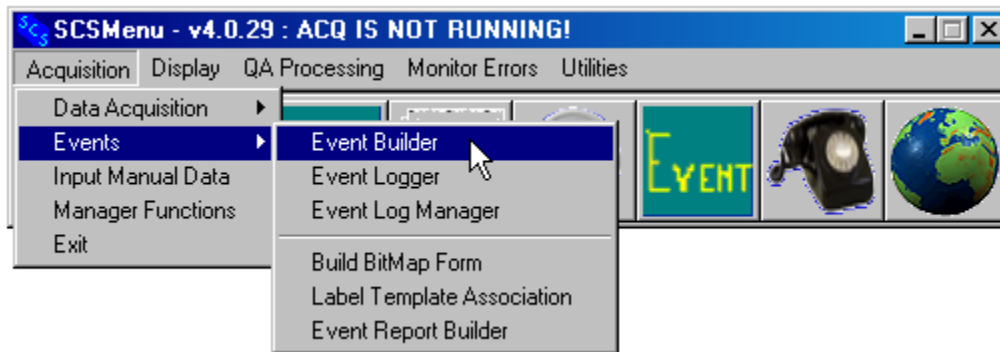
Equipment Log File

This file is found with the other output files under the name MetadataSensorDescription_NNN.CSV. It contains a list of all metaitems. For each metaitem the following information is given:

- For Manual metaitems, the most recent value of the metaitem is given.
- For Sensor metaitems, the name of the sensor supplying the data is given.
- For Summary metaitems, the name(s) of all base metaitems used to compute its value is given.

The equipment log file is written at Event Start. At Event Stop the file is written again, and changes made to manual metaitems during the event are reflected in the updated version.

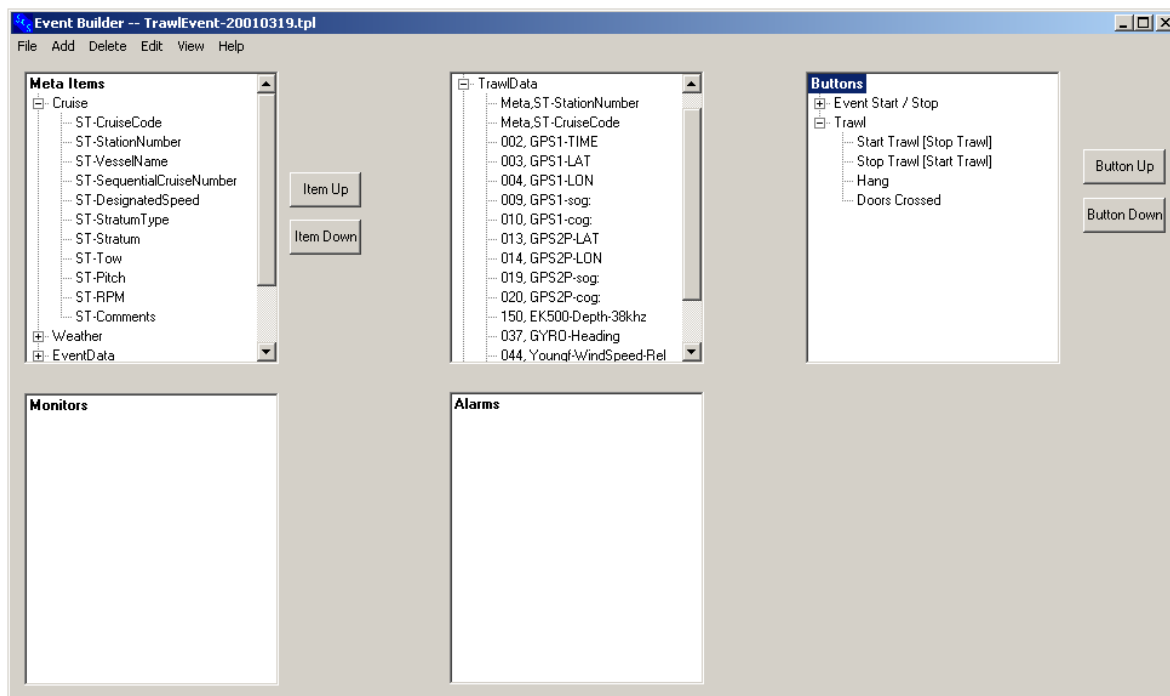
4.3 Defining Event Templates with Event Builder

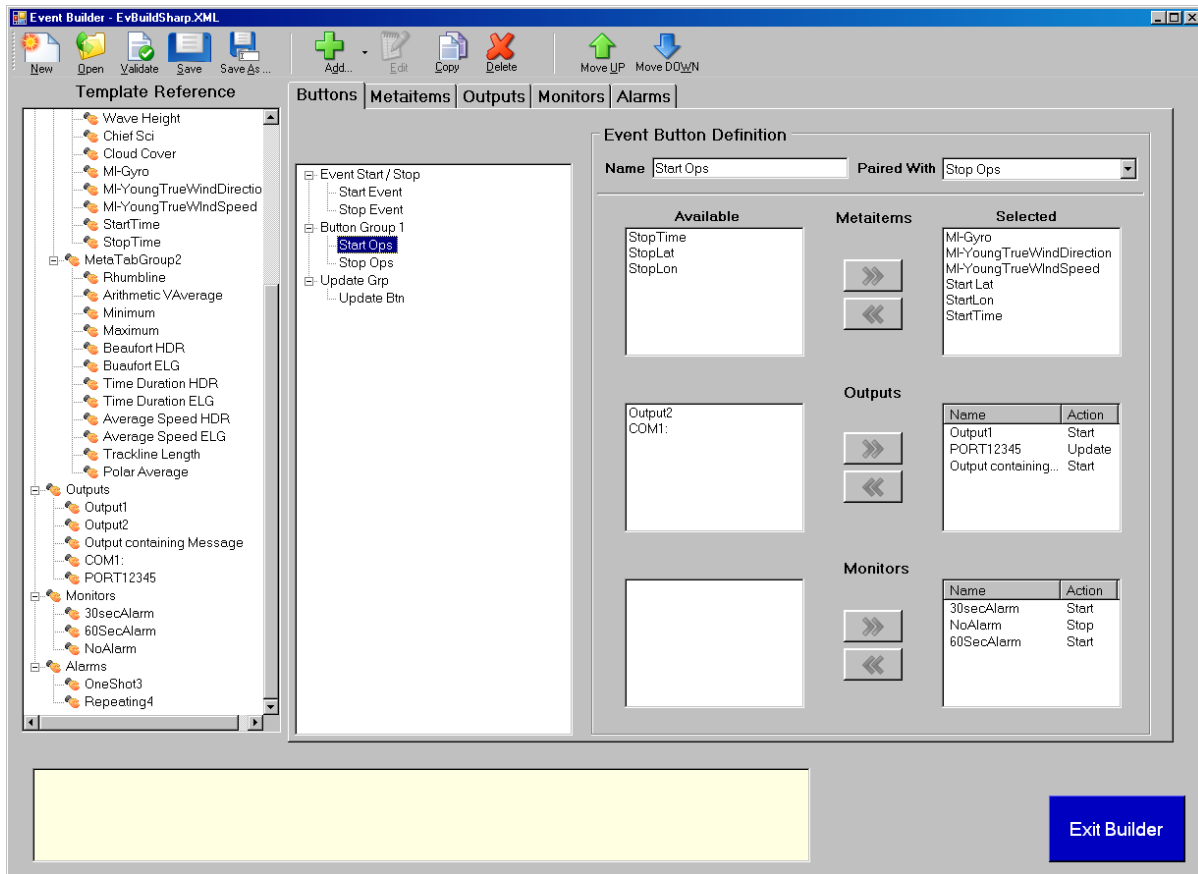


The Event Builder application creates all components of an event and sets up their interactions. Use it to create new event templates and modify existing templates. To run the Builder, start up the SCS Menu application, and from the Acquisition Menu select Events -> Event Builder

Event Builder Main Dialog Overview

The Main Dialog GUI contains a tree for each of the five types of event components: Meta-item, Output, Button, Monitor and Alarm. The items in each tree represent the various components of the event, taking their name from the Label of the component. The figure below shows a sample Main Dialog.





The trees for Monitors and Alarms represent simple lists and have only one level. The Output tree has two levels for easy reference; the top level shows the Outputs, the second level the outputs' elements. Trees for Meta-items and Buttons have two levels because they can be organized into *Groups*.

For Meta-items, Buttons and their groups, Up and Down pushbuttons exist to move components up and down within their groups, and for moving groups up and down within the tree. Note that Meta-items and Buttons cannot be moved to a different group – they can only be deleted, and then recreated in the desired group.

In the Logger the ordering of Meta-items, Buttons and their groups determines the order in which they are displayed. Section 3.7.2, Running an Event, provides additional detail.

Adding a New Component

First, click on an item on the relevant tree. Second, from the Add menu, select the type of component you want to add – this will bring up an Add dialog for the selected component type.

The new component will be added to the corresponding tree *immediately after* the selected component.

Alternatively, you can right click on the component, and then select Add from the popup menu.

Editing an Existing Component

There are several ways to edit a component.

- Double click on the component.

- Right click on the component, then select Edit from the popup menu
- Left click on the component, and then select the appropriate component type from the Edit menu.

General Features of All Add/Edit Dialog Boxes

The popup Dialog Boxes to add or edit a component type are identical – the dialog title bar shows the actual operation. Within the dialog is a set of controls (edit boxes, buttons, etc.) needed to define the component.

- The Label Edit Box
All components have *labels*, or names, which serve two purposes.
- Builder displays the label for each component in the corresponding tree in its GUI.
- Components reference other components using the label.

Labels must be unique within the tree of a given component. For example, two outputs could not have the same label, but an output and a monitor could.

Builder accepts Labels in upper and lower case, but does not consider case when comparing two labels for duplication.

The Label edit box is always at the very top of the Add/Edit dialogs.

• Relevant and Irrelevant Controls

Some GUI controls may be *irrelevant* to a component's definition; then the control is grayed out and will not accept any input. Controls can change from relevant to irrelevant, depending on the value of other controls. The reason for this will become clear as you read through the detailed descriptions of the various Add/Edit Dialogs.

• The OK Button

When you finish defining the component, click on the dialog's OK button to save your changes. At that time Builder checks your input looking for missing or erroneous values.

Certain controls must be filled in before you press OK. The component's Label is the only thing that is *always* required. Other controls may be required depending on the situation – Builder will always inform you if you have left a required label blank, and will not pop down the dialog until you have entered a value.

Other controls may be optional for Builder; that is, they are not required to press OK, but must be filled in before you run Logger. This feature exists to deal with a chicken-egg problem: to define some components you must refer to other components, and if you are building an event from scratch you may not have had time to define those components yet.

• The Cancel Button

Clicking the Cancel button discards any changes you have made to any of the controls during an Edit dialog.

Add/Edit Dialogs in Detail

The following dialog descriptions give a description of each control, and describe interactions between the controls, that is, whether a control can be made irrelevant by another control's value.

4.3.1 Add/Edit Meta Tab Group Dialog

All meta-items must be part of a *Meta Tab Group*. In Builder, the Meta-item Tree has two levels, the first for meta tab groups, and the second for the meta-items themselves. In Logger meta tab groups are organized into tab sheets, each displaying their meta-items. There can be a maximum of 16 meta tab groups.

This dialog has one edit box control, for the group label.

4.3.2 Add/Edit Manual Meta-Item Dialog

The figure below shows a sample dialog to add or edit a manual meta-item. The controls in this dialog are:

Label edit box: The label of a meta-item is its name. Other event components may refer to it by this name.

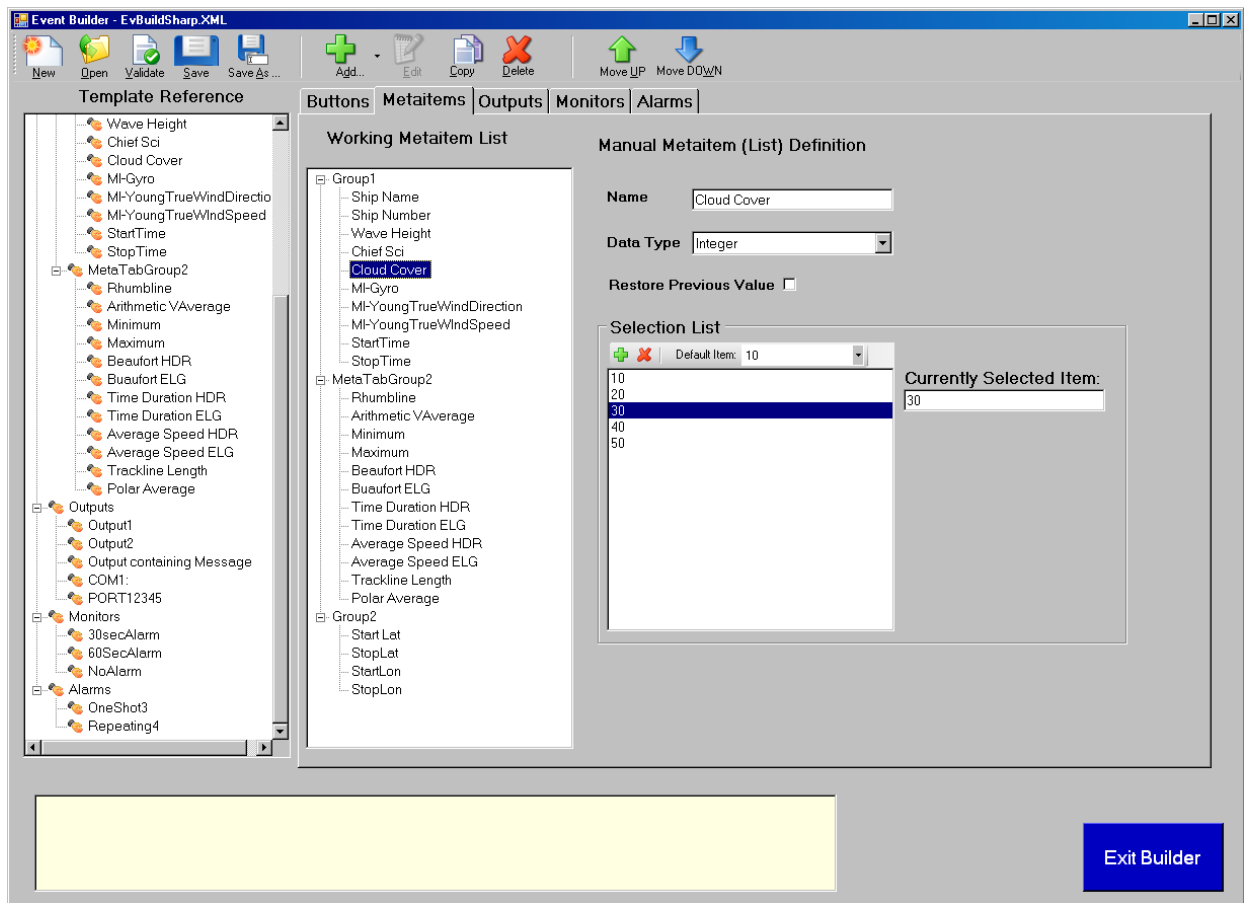
The Default Value edit box: Has the initial value for the Meta-item at the start of an event, but may be left blank. The default value is overridden if the Restore Previous Value is checked.

The Data Type dropdown list: Specifies the values this meta-item can take on during an event.

- **String:** Any character may be used in the value.
- **Integer:** The value must be an integer.
- **Integer (Auto increment):** Same as integer, except at the start of an Event Logger will automatically increment the value from the previous Event by one.
- **Real:** The value can have whole number and fractional parts, and a decimal point.

The screenshot shows the 'Event Builder - Edit Manual Meta Item' dialog. The 'Label' field contains 'ST-VesselName' and the 'Default Value' field contains 'Albatross IV'. The 'Data Type' is set to 'String'. The 'Editable' checkbox is unchecked, and 'Restore Previous Value' is also unchecked. The 'List Of Values' checkbox is checked, revealing a list area with several control buttons: 'Enter New Value', 'Add to List', 'Make Default', 'Move Value Up', 'Move Value Down', and 'Delete Value'. The 'OK' and 'Cancel' buttons are at the bottom.

The Add/Edit Manual Meta-item Dialog



The Editable check box: specifies whether Logger will allow you to type in a new value for the meta-items during an event. This affects the way Logger displays this meta-item.

The List of Values checkbox: Allows you to create dropdown a list of values from which the user will select during an Event. Logger will then allow no other value.

The Minimum Value and Maximum Value edit boxes: For real or integer types, it sets limits on the value the user can enter during an event. Either or both can be left blank, indicating there is no minimum or maximum constraint.

The String Length edit box sets the maximum number of characters in a string-type meta-item value the user can enter during an event.

The Restore Previous Value checkbox: Specifies whether Logger will load the last known value back into the meta-item the next time you run the event. (Logger automatically saves the final values of all meta-items at the end of any event.) If checked, the saved value overrides the default value.

The List of Value controls are grouped at the bottom of the dialog. By default they are irrelevant, but become relevant when the List of Values checkbox is checked. The Logger will then present the entered values in a dropdown list, forcing the user to pick one of those values.

To add a value to the list, fill the Enter New Value edit box, then click on the Add To List button. The new value is added immediately after the currently selected list item. To change the order of the list entries select an item on the list. Click on the Move Up and Move Down buttons changes the position of the selected item in the list. When you finish list creation, select one item and then press the Make Default button – the selected value will be copied to the Default Value edit box. The meta-item's Data Type determines what kind of list entries is acceptable (string, integer, real).

Interactions among the dialog controls:

- Data Type determines relevancy of Minimum Value, Maximum Value and String Length. Numeric data types require Minimum and Maximum Values; the string data type requires a String Length.
- If Editable is checked, Default Value may be left blank and Minimum Value, Maximum Value and String Length are all relevant; otherwise a Default Value is required.
- If List of Values is unchecked all controls in the List of Values group box are irrelevant; if checked at least one entry must be in the list.

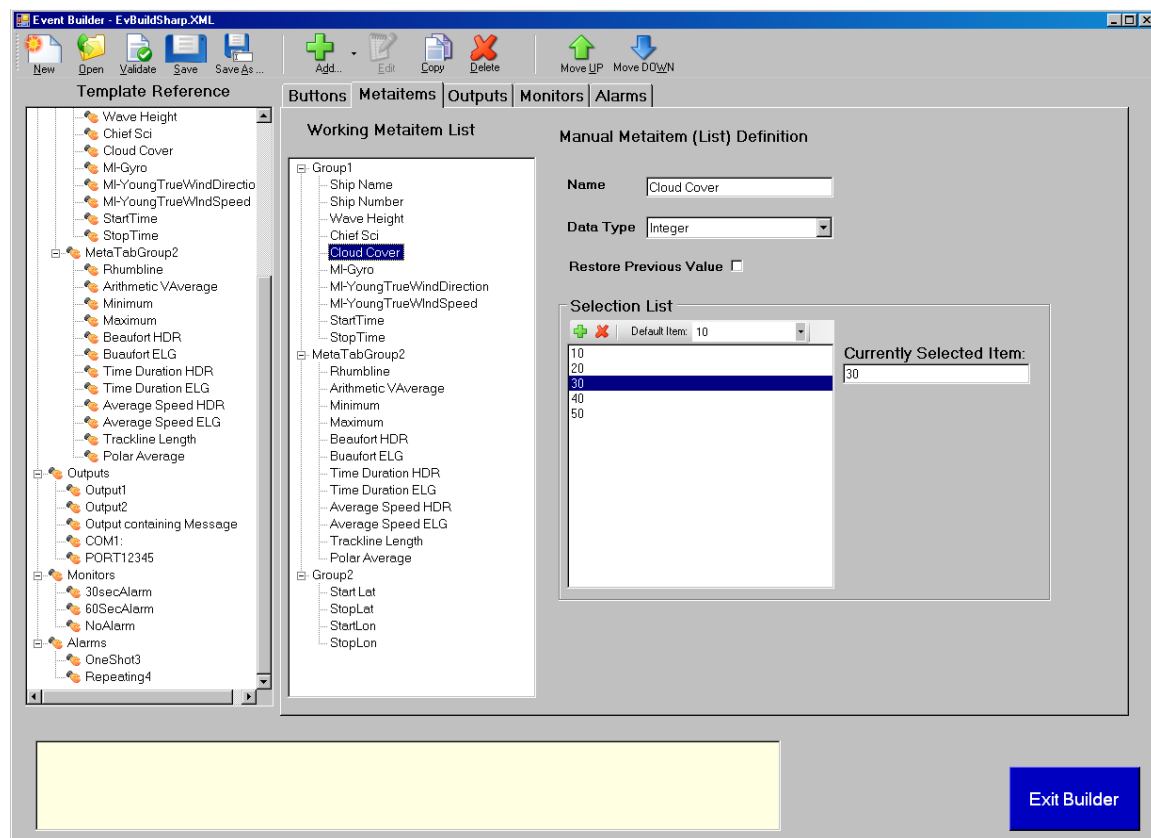
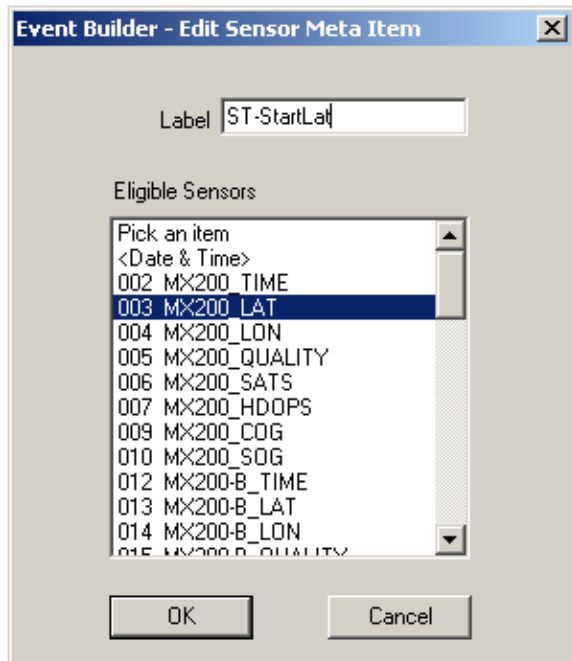
4.3.3 Add/Edit Sensor Meta-Item Dialog

The following figure shows a sample dialog to add or edit a sensor meta-item.

Label edit box: This is the name of the meta item. Other event components such as buttons refer to the metaitem by its label.

Sensor ID List: This list is populated with sensors from the Sensor Configuration File SENSOR.DAT. All Data Fields are listed; the first list item is always "<Date & Time> ", meaning the meta-item value is updated with the current time from the system clock of the computer Event Logger is running on.

The Logger will update the value of the meta-item with the current value of the selected sensor whenever the associated button is pushed. Further detail is in the sections on running the Event Logger.



The Add/Edit Sensor Meta-item Dialog

4.3.4 Add/Edit Summary Meta-Item Dialog

Figure 2.5 shows a sample dialog to add or edit a summary meta-item.

Event Builder - Edit Summary Meta Item [X]

Label

Data Type

Data Source

Base Meta Item(s)

1 Meta Item

Statistical Data Filter

☐ Enable

Sigma

Range Filter

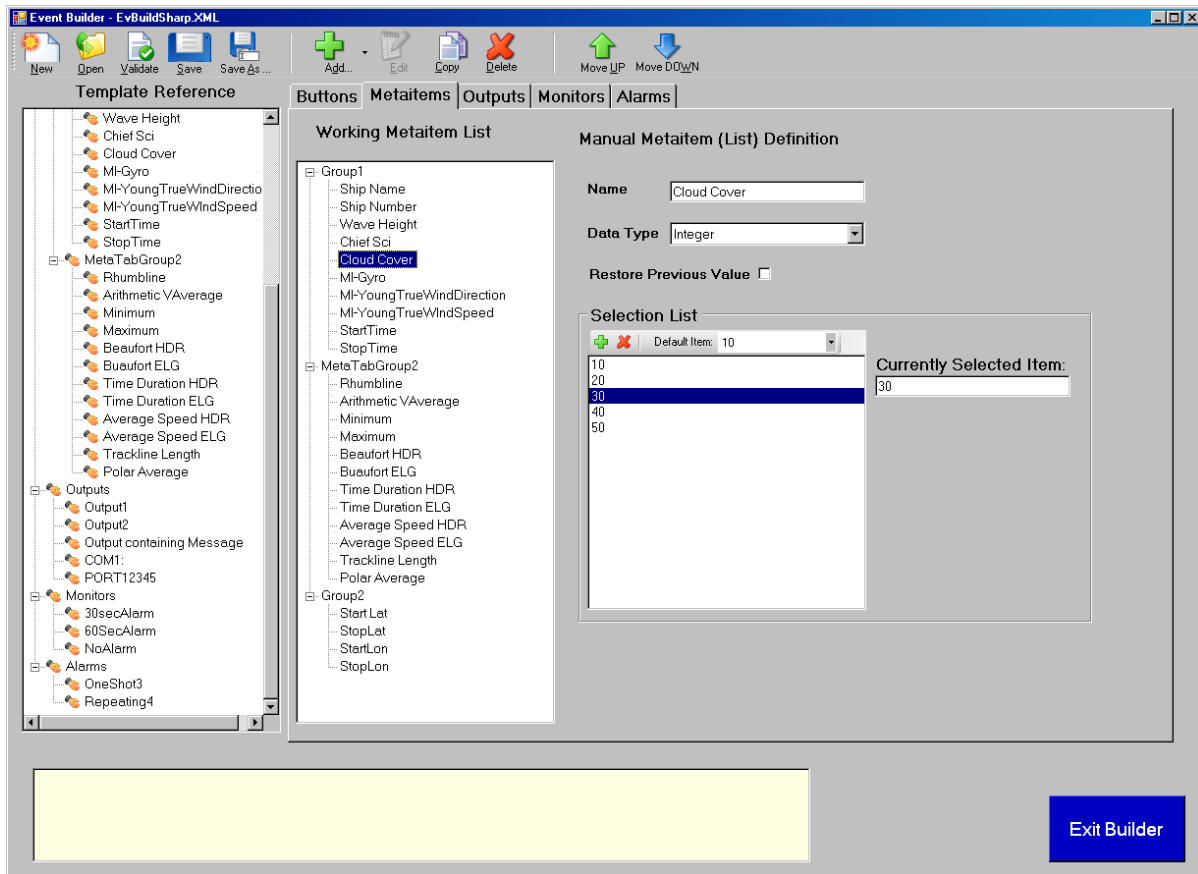
☐ Enable

Min

Max

OK Cancel

The Add/Edit Summary Meta-item Dialog



The controls in this dialog are:

Label edit box: This is the name of the meta item. Other event components such as buttons refer to the metaitem by its label.

Data Type dropdown list: Defines the nature of the calculation that produces the meta-item's value. The following types are supported:

Rhumb line – Computes the straight-line distance between two lat/lon pairs on a Mercator projection. Result is in nautical miles.

Average Value – Computes the average value of a meta-item time series.

Minimum – Selects the minimum value in a meta-item time series.

Maximum – Selects the maximum value in a meta-item time series.

Beaufort Scale – Converts an average wind speed to a number on the Beaufort Scale

Time Duration – Computes the number of minutes and fraction of minutes between two date/time meta-items.

Average Speed – Computes the average vessel speed using the time series of latitude and longitude meta-items. Result is in knots.

Track line Length – Computed by summing the distances between consecutive locations within a time series of latitude and longitude meta-items.. Result is in nautical miles.

Polar Average – same as Average Value, except it is intended for use with “zero-to360” sensors such as Gyro, Compass, Wind Direction or COG.

See table below for additional information on how these values are calculated.

Data Source dropdown list: Contains the labels of all currently defined Outputs that write to disk files, and also that contain only data fields; any output containing a complete message is disqualified. These are the Event Data files (*.ELG). There is also an additional item in the list, “HEADER FILE”, which tells Logger that the summary value will be computed from metadata in the event header file (*.HDR). If an ELG file is selected, Logger will read through the time-series in this file to compute the value of this meta-item. See the table below for further information on the effect of choosing the header file versus a data file.

4.3.5 Base Meta-item Dropdown Lists

Base Meta-items are the ones that are used to compute the summary meta-item value. There are six lists in the dialog window, but number of relevant lists varies, depending on the Data Type. For example, for a Minimum data type only one meta-item is used, but Time Duration requires two meta-items (start and stop times), and a Rhumb line requires four base meta-items (Start Lat, Start Lon, Stop Lat and Stop Lon).

The number of Base Meta items also depends on the Data Source. For example, an Average Speed requires six base meta-items if you are computing it from the header file (Start Lat, Start Lon, Stop Lat, Stop Lon, Start Time, and End Time). But if you are using an ELG file, only two are required: Latitude and Longitude. The starting Lat and Lon are the first in the file, the Stop Lat and Lon are the last in the file, and the times are taken from the timestamps.

Items included in each Base Meta-item list are:

- Meta-items, but only integer or real data types (all summary data types are derived from numeric data). In addition, for Time Duration summary meta-items - only <Date/Time> sensor meta-items are in the list.
- For any Output that includes this meta item in its list of elements, the Output’s sensor output elements are also included.

Refer to the table below for a summary of methods for computing summary meta-item values.

Data Type	Can Compute From Header File?	Can Compute From Data File?	Range Filter Available?
Rhumb line	Yes. Distance in NM is computed on a Mercator projection. Convergence of meridians is factored in by multiplying delta Lon by the cosine of the average Lat.	No	No
Average	No	Yes. The average of all (filtered) values is computed for the output element	Yes
Minimum	No	Yes. The minimum of all (filtered) values for the output element is selected.	Yes
Maximum	No	Yes. The maximum of all (filtered) values for the output element is selected.	Yes
Beaufort Scale	Yes. The value of the base Meta-item is converted using the “Beaufort Scale” Table below,	Yes. The average of all (filtered) values for the output element is computed, then converted to a Beaufort scale using Table 2.7	Yes
Time Duration	Yes. The interval between the two base Meta-items is computed and converted to minutes and decimal fraction.	Yes. The interval between the first and last timestamps in the file is computed. Note that the “Base Meta-item” is immaterial.	No
Average Speed	Yes. The Rhumb line is computed between the start and stop positions and is divided by the interval between the start and stop times. Result is in knots.	The trackline length is computed from the base Lat and Lon values, and then divided by the Time Duration. Result is in knots.	No
Trackline Length	No	Yes. The distance in NM traveled is computed from one position fix to the next. These distances are then summed.	No
Polar Average	No	The average is computed from the values in the time series. It is assumed the values are in decimal degrees, and vary from 0.00 to 360.00.	No

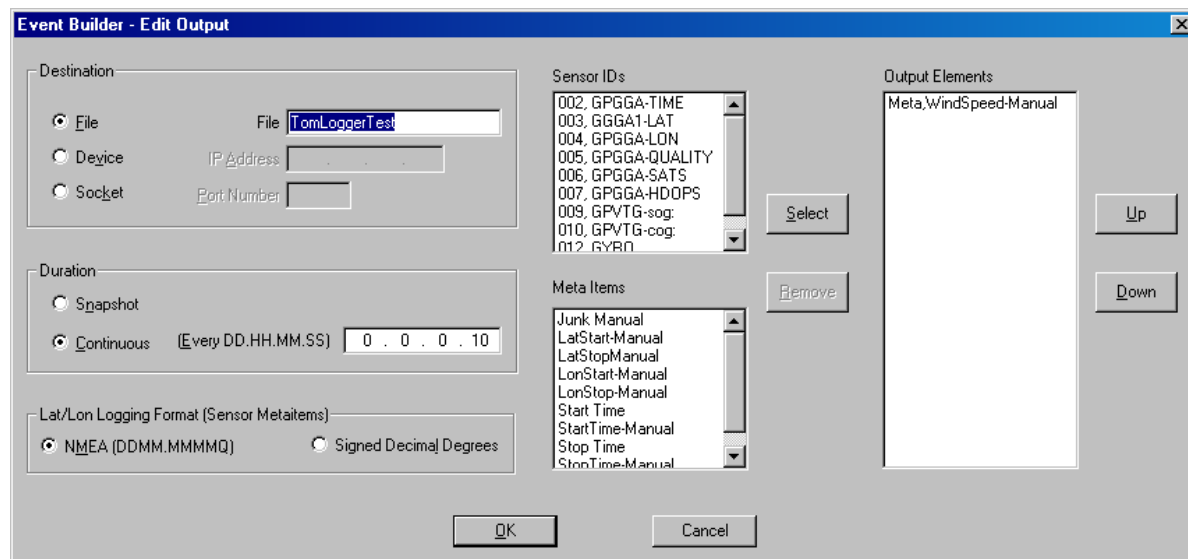
~Summary Meta-item Calculation Options~

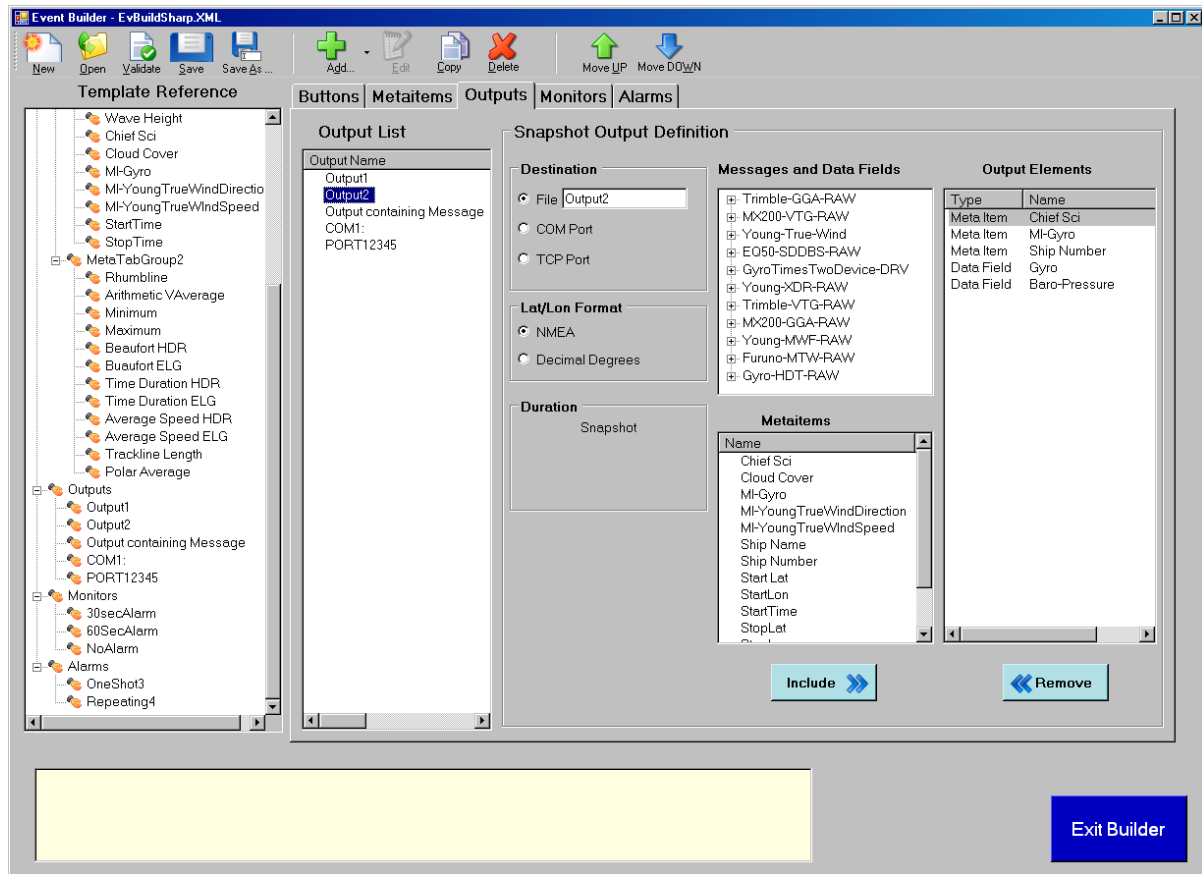
Wind Speed (knots)	Beaufort Scale
< 1	0
1-3	1
4-6	2
7-10	3
11-16	4
17-21	5
22-27	6
28-33	7
34-40	8
41-47	9
48-55	10
56-63	11
> 64	12

~The Beaufort Scale~

4.3.6 Add/Edit Output Dialog

The figure below shows a sample dialog to add or edit an output.





The Add/Edit Output Dialog (File Type)

The controls in this dialog are:

The Destination radio buttons: Specify where the Output will write its data. You can specify a disk file, a COM port or a TCP/IP socket port; the latter two allowing you to send data to external computers.

Select the Destination using the radio button controls. Selecting one causes the appropriate controls to become relevant or irrelevant – you must then fill in the relevant controls to complete the definition of the Output.

For file destinations, fill in the name of the file. The Windows operating restricts the characters you can use in this name, and the Builder will not allow you to enter those characters. Logger will give an .ELG extension to this name and write it to a subdirectory of the SCSEventLog logical directory. The subdirectory is named after the Event. The SCSEventLog directory must exist, but if the event subdirectory does not exist Event Logger will create it.

For Device (COM port) destinations, specify a port in the form “COM1:”. The COM in upper case and the trailing colon are required.

For a socket port destination, enter the port number to send the data out on.

When you are running and event, Outputs are started in two ways:

- An Output is assigned to a Button, and is started when the user presses that button in the Logger GUI.
- An Output can be assigned to a Monitor, and is started when the Monitor detects any of several conditions.

Outputs can also be stopped in the same two ways.

Once started, an Output's *Duration* determines how often the output data is written to its destination. A snapshot output appends its elements once whenever started, and the logging interval is not relevant. A continuous output appends its output elements until it is stopped; if you select continuous output, you must specify the logging interval.

You have a choice of format when lat/lon data is written. NMEA format is the form in which latitudes or longitudes are reported by GPS receivers. For example, 45-12-30 N is reported as 45.1250N, where "1250" indicates 12.50 minutes. Alternatively you can select Signed Decimal degrees, which would write the above latitude as 45.20833. In this convention South latitudes and West longitudes are written as negative numbers.

On the right side of the dialog box you select which data will be written to the output destination. You are presented with two lists on the left, ACQ Sensor Ids from the SENSOR.SCF file, and Meta-items. Select one or more from each list, and then press the Select button to move those items to the Output Elements list on the right. Output elements may be removed by selecting them and pressing the Remove button. Elements are written in the order in which they appear on this list. Their order in the list may be modified by selecting an element, then moving it with the Up and Down buttons. Note that you can always tell a Meta-item from an ACQ sensor – a Meta-item is prefixed by "Meta,", while an ACQ sensor is prefixed by its sensor ID.

The list of Sensor Ids includes both parent and child sensors. Selecting a child sensor will result in its value appearing in the output file. Selecting a parent sensor inserts the entire sensor message into the output file. If an output contains a parent sensor as one of its elements you will not be able to define any summary Metaitems based on any data in this files. This is because a parent sensor message may include an unknown number of commas, which would confuse Logger when it tries to locate the base time series to compute the summary value.

4.3.7 Add/Edit Button Group Dialog

Similar to Meta Tab Groups, Button Groups allow you to organize Buttons in the Logger GUI into tab sheets so that related buttons are displayed together. Each Button Group corresponds to one tab page. Up to ten groups are allowed, with up to 16 buttons per group. Refer to the sections on running the Event Logger for additional detail.

4.3.8 Add/Edit Button Dialog

The dialog to add or edit a Logger Button is shown in the next figure. The items in this dialog are described below.

The “Label” is the label that appears on the button when you run the Event Logger.

The “Paired With” dropdown list allows two buttons to be connected as a pair. Its use is strictly optional. The Logger will position paired buttons in adjacent positions and surround them with a frame. Typically you would use this feature to group a start and a stop button for a given situation. On Builder’s Button tree paired buttons appear together with their partners in square brackets.

The dropdown list contains all Buttons other than this one. Selecting a partner button from this list pairs it to this button. If the partner button had already been paired with another button that pairing is broken, and the previous partner becomes unpaired.

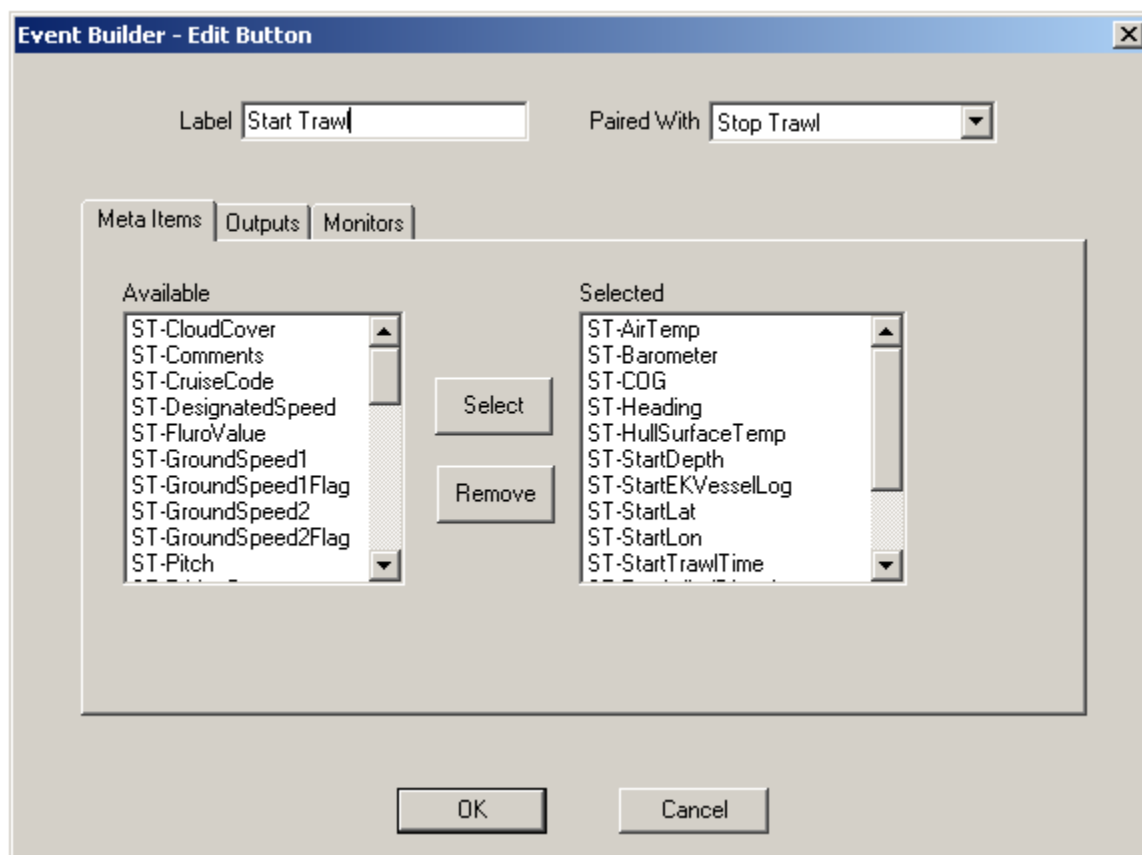
See the description of the Event Logger app for more information about how Logger treats button pairs during an event.

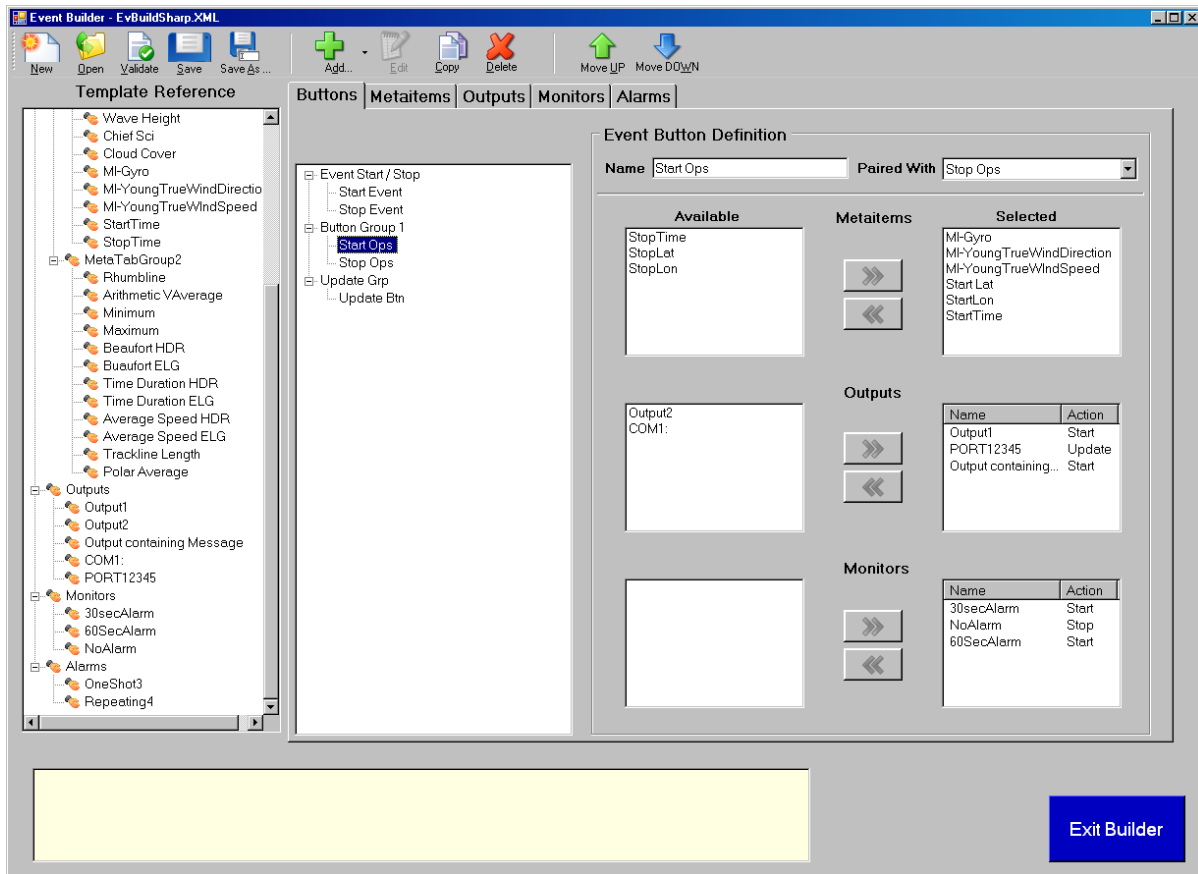
Tab Sheets

The central part of the dialog consists of three tab sheets, allowing you to assign the Meta-items, Outputs or Monitors to a Button. They are similar in operation.

Meta-item Tab Sheet

The following figure shows a sample dialog to assign meta-items to a button.





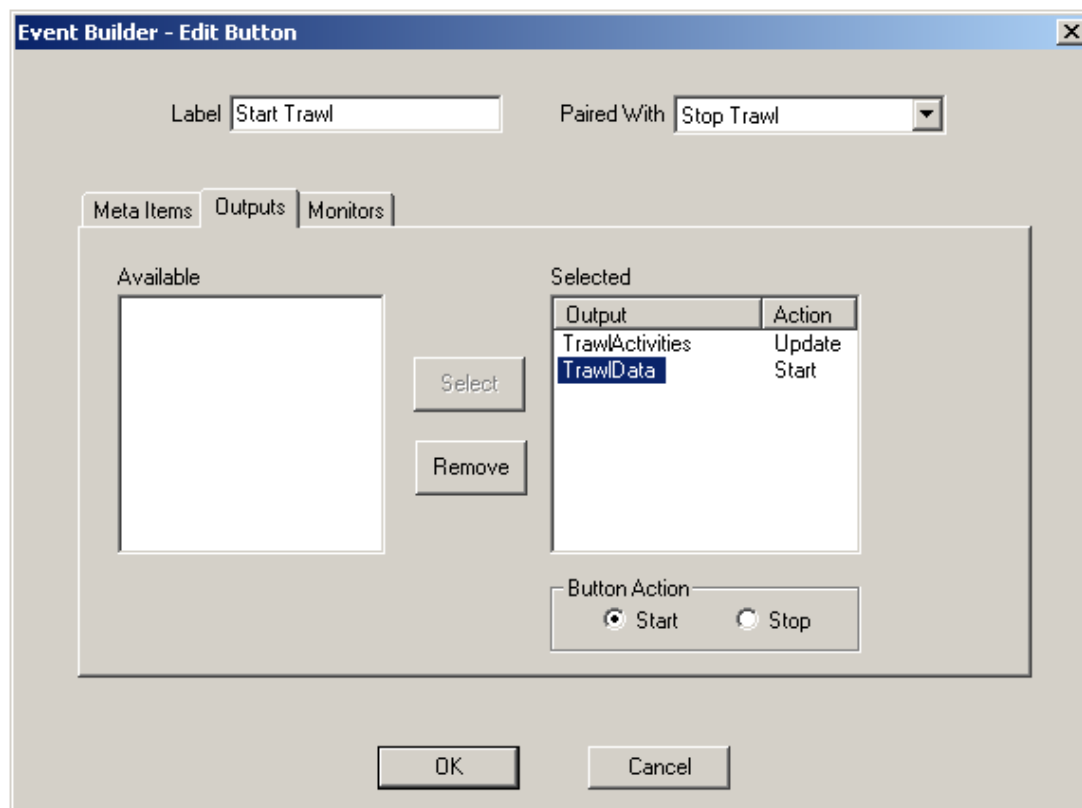
The Add/Edit Button Dialog, Meta-items Tab Sheet

On the left is the Available List, containing all meta-items not already selected. Only sensor meta-items are shown. Select sensors from this list, and then press the Select button to move those meta-items to the Selected List on the right. Click on selected meta-items, and then press the Remove button to move meta-items back to the Available List.

When this Button is pressed during an event, the value of the selected meta-items will be updated with the latest value from the associated sensor. These values will then be written on any subsequent Output logging operations.

Output Tab Sheet

The following figure shows a sample dialog to assign outputs to a button.



The Add/Edit Button Dialog, Outputs Tab Sheet

On the left is the Available List, containing all Outputs not already selected. Click on Outputs from this list, and then press the Select button to move them to the Selected List on the right. Click on Outputs in the selected list, and then press the Remove button to move Outputs back to the Available List.

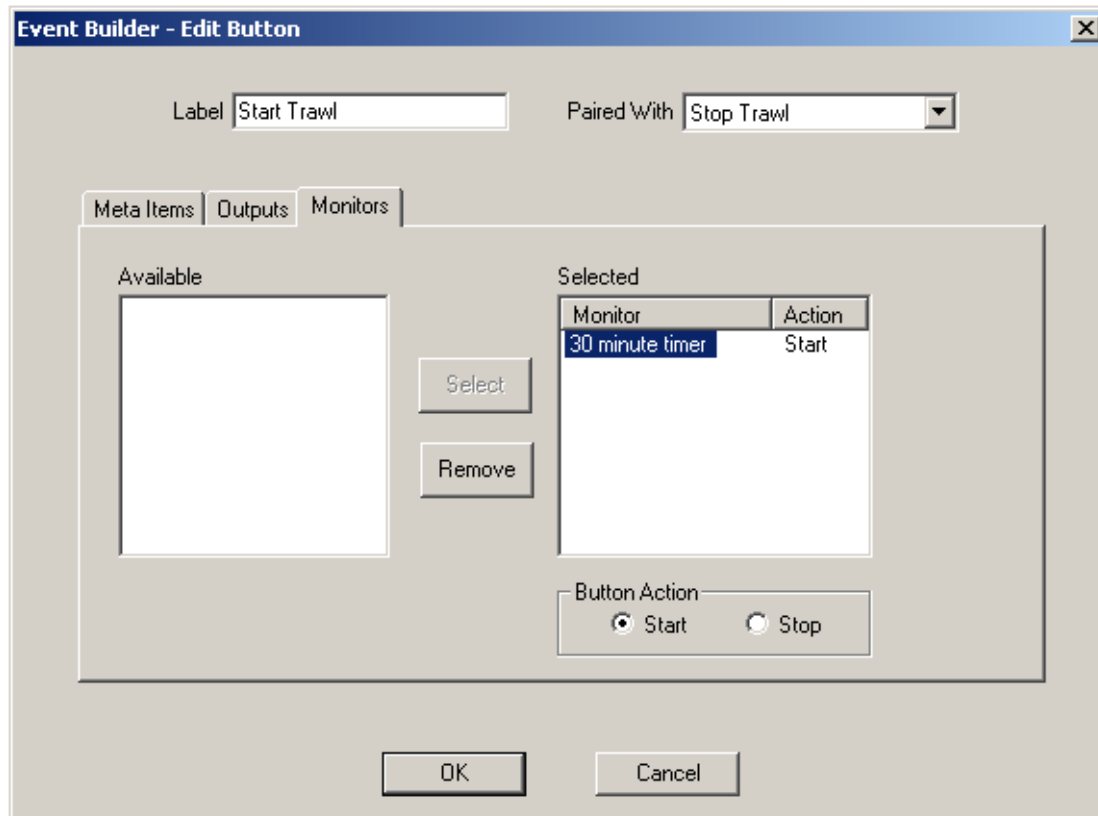
The second column in the Selected Output list is one of three possible actions: Start, Stop or Update.

For a continuous Output you will see Start or Stop in this. By default, when you move a continuous Output to this list, the action will be Start, indicating that when this button is pressed in the Logger, the Output will commence writing its data. To change the action for an output, click on it, which will enable the Start and Stop radio buttons beneath the list. Now you can click on one of them to instruct the Logger to either start or stop the Output when the Button is pressed.

If a selected output is a snapshot you will always see Update in this column, and the Start and Stop radio buttons are always irrelevant.

4.3.9 Monitor Tab Sheet

The following figure shows a sample dialog to associate monitors with a button.



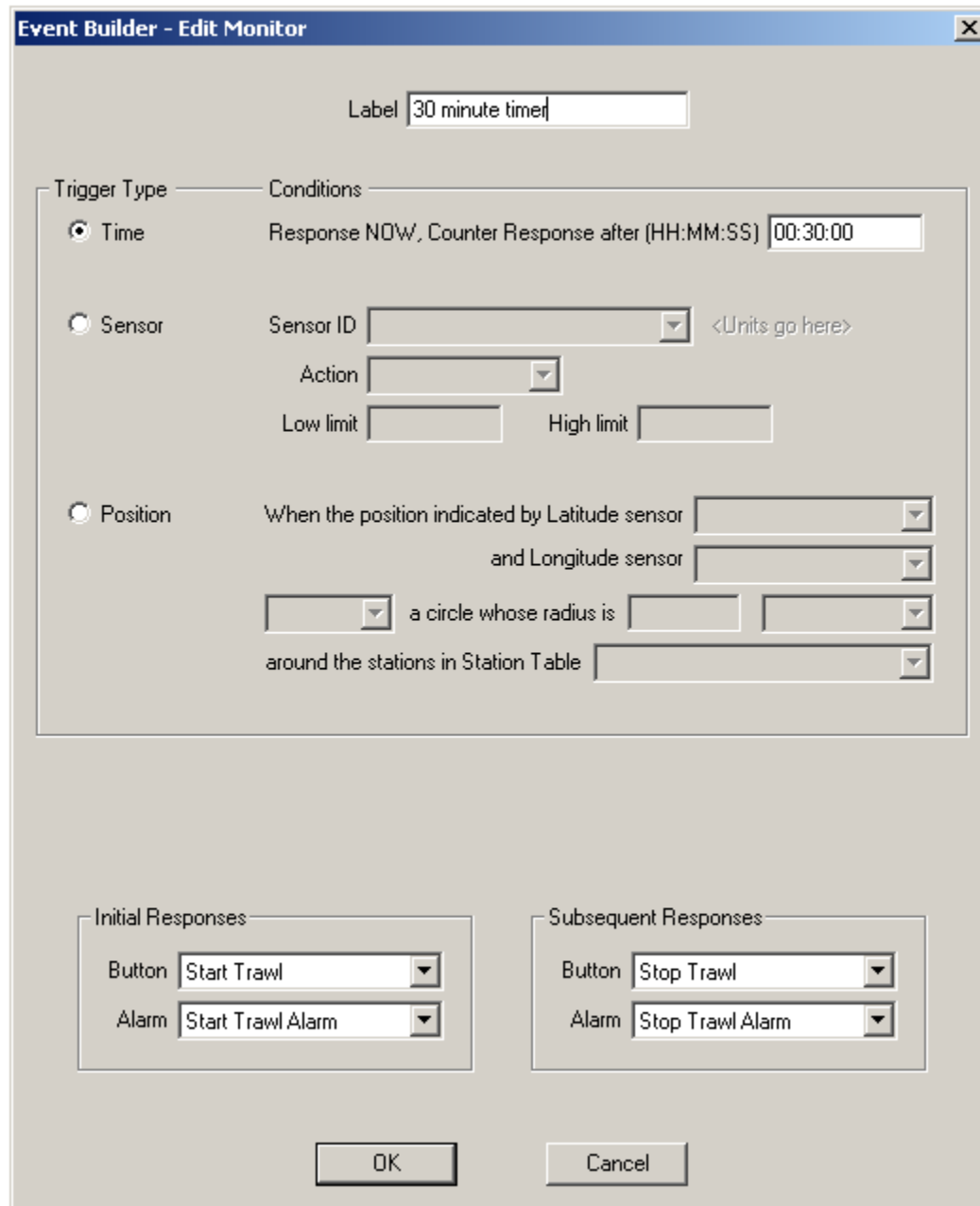
The Add/Edit Button Dialog, Monitors Tab Sheet

On the left is the Available List, containing all Monitors not already selected. Click on Monitors from this list, and then press the Select button to move them to the Selected List on the right. Click on Monitors in the Selected list, and then press the Remove button to move Monitors back to the Available List.

Associated with each selected Monitor in the Selected list is an action, Start or Stop. Selecting a monitor in the list makes the Start and Stop radio buttons relevant; you then click on one of them to instruct the Logger to either start or stop the Monitor when the Button is pressed.

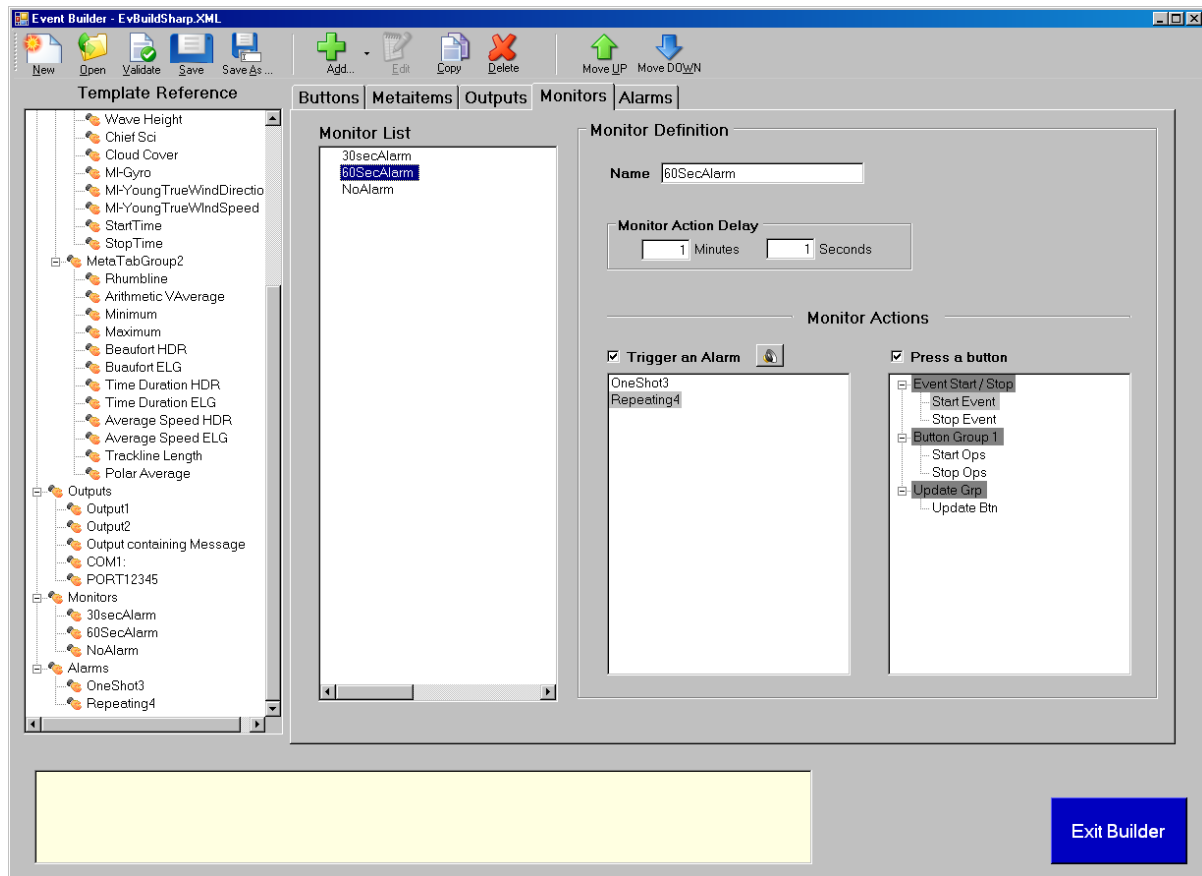
4.3.10 Add/Edit Monitor Dialog

The figure below shows a sample dialog to add or edit a monitor.



The dialog box is titled "Event Builder - Edit Monitor". It contains the following fields and controls:

- Label:** A text field containing "30 minute timer".
- Trigger Type:** A group box containing three radio buttons: "Time" (selected), "Sensor", and "Position".
- Conditions:** A group box containing:
 - Time:** A text field for "Response NOW, Counter Response after (HH:MM:SS)" containing "00:30:00".
 - Sensor:** A "Sensor ID" dropdown menu, a "<Units go here>" text field, an "Action" dropdown menu, and "Low limit" and "High limit" text fields.
 - Position:** A group of fields including "When the position indicated by Latitude sensor" and "and Longitude sensor" dropdowns, a "a circle whose radius is" dropdown, and "around the stations in Station Table" dropdown.
- Initial Responses:** A group box containing "Button" (Start Trawl) and "Alarm" (Start Trawl Alarm) dropdowns.
- Subsequent Responses:** A group box containing "Button" (Stop Trawl) and "Alarm" (Stop Trawl Alarm) dropdowns.
- Buttons:** "OK" and "Cancel" buttons at the bottom.



The Add/Edit Monitor Dialog

Label Edit Box:

The controls below the Label box set the type of the Monitor, and supply the rest of the information needed to define monitor of that type. The three types of monitors are:

- Time (watches the clock)
- Sensor (watches a sensor's value) **NOT IMPLEMENTED**
- Position (watches the ship's lat/lon) **NOT IMPLEMENTED**

Press one of the three radio buttons to select the type; Builder will then make relevant the controls needed to complete the monitor definition.

For a time monitor, you must specify the countdown interval. The monitor begins counting down when it starts up (through a button press). When it reaches zero, the Monitor initiates a response (see below for details).

(NOT IMPLEMENTED)

Clicking the Sensor radio button makes the following controls relevant:

The sensor dropdown list: Populated with all sensors that supply a value (sub sensors, derived, and manual). Select one.

The Action dropdown list: Describes what the sensor value will be doing: increasing, decreasing, going into or out of range.

The Low and High Limit edit boxes: These are relevant depending on the Action. Into/Out of Range requires both; the other actions require only a single value.

When the sensor value behaves according to the parameters you have set up, the Monitor initiates a response (see below for details).

A Position Monitor watches the position of the ship, and when the ship enters or leaves the area of any station in the Station File, the Monitor initiates a response. The following controls become relevant.

The latitude sensor and longitude sensor dropdown lists: Have all ACQ sensors of NAVSEC type, i.e. those that supply latitude or longitude values. Select a sensor from each list.

The Action dropdown list: Select “Enters” or “Leaves”

The Circle Radius edit box: Enter the numerical value of the radius of the circle that defines the area around the stations in the Station File.

The Radius Units dropdown list: Select from Meters, Kilometers or Nautical Miles

The Station Table dropdown list: Populated with all *.STA files in the ScsStations logical directory. Select one file. See section TBD for additional information about Station Tables

Monitor Responses

Any Monitor, regardless of type, can take two actions in responding to a condition. Both responses are optional.

- It can push a button (just one).
- It can sound an audible Alarm.

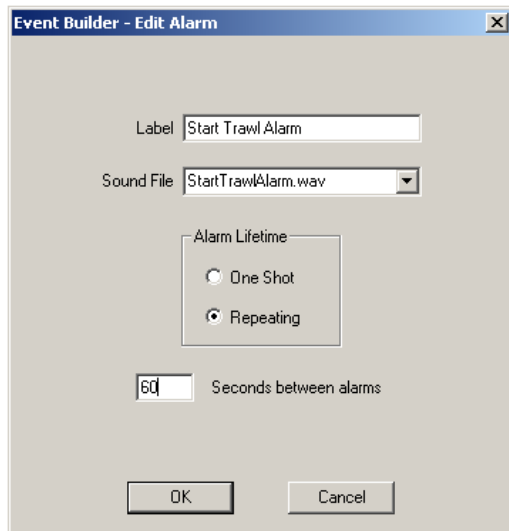
Moreover, monitors are actually capable of two responses: Initial and Follow-up. Remember, you can choose not to set a response for either.

- Time Monitor: Makes the initial response immediately upon starting. When the countdown timer reaches zero, it initiates the follow-up response, then deactivates itself.
- (NOT IMPLEMENTED)Sensor Monitor: Makes the initial response when the specified condition becomes true. When the condition becomes false again, the Monitor makes the follow-up response then deactivates itself.
- (NOT IMPLEMENTED)Position Monitor: Makes the initial response when the specified condition becomes true. When the condition becomes false again, the Monitor makes the follow-up response, and then deactivates itself.

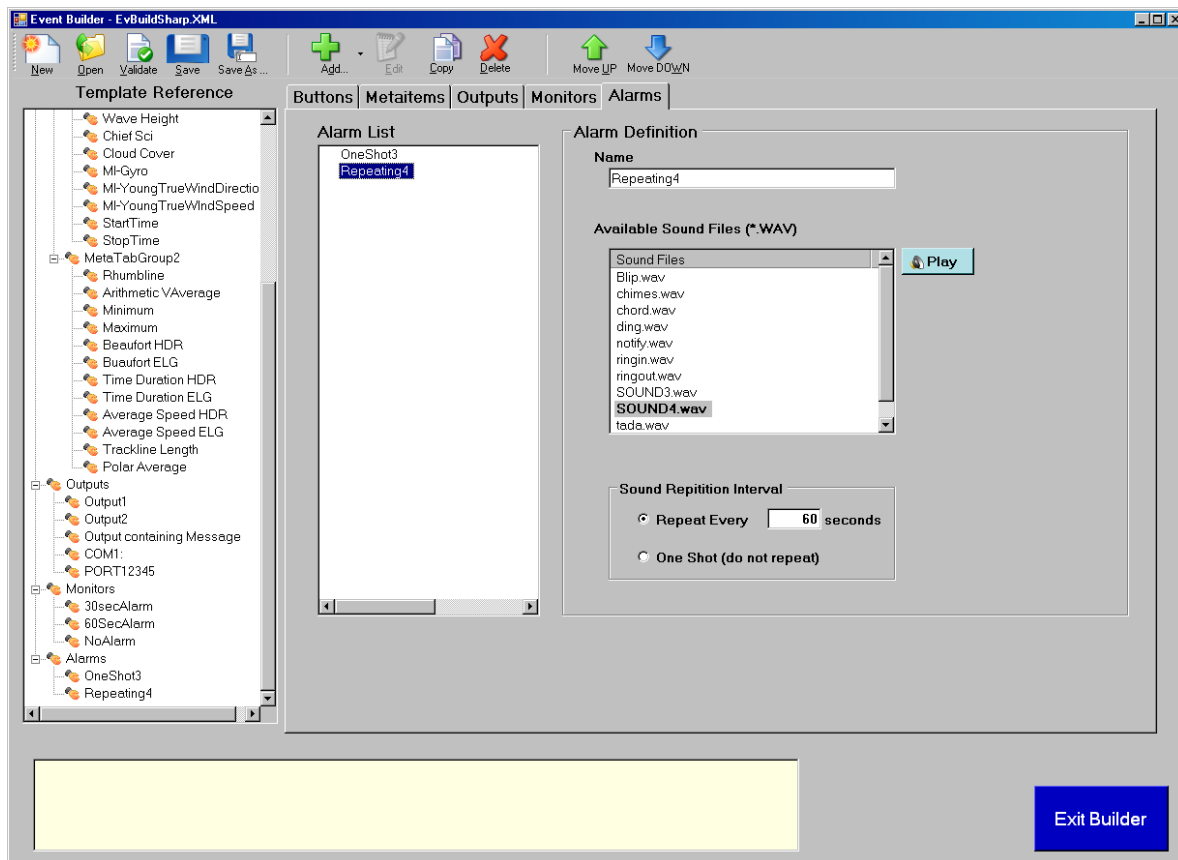
All monitors deactivate themselves (that is, cease their watches) after the follow-up response, or after the initial response if no follow-up is specified. Self-deactivation prevents Monitors from engaging in any type of annoying behavior *ad infinitum*.

4.3.11 Add/Edit Alarm Dialog

Figure 2.14 shows a sample dialog to add or edit an alarm.



The Add/Edit Alarm Dialog



Label edit box:

Sound File Dropdown List: Builder populates the list with all *.WAV files in the Sounds directory, directly under the SCSShip logical directory. The selected file will be played as part of a Monitor response.

Alarm Lifetime radio buttons: Reserved for future use. Currently all alarms are one shot, that is, they play the file once and then deactivate themselves.

Seconds Between Alarms edit box: Reserved for future use. Relevant only for repeating Alarms.

The View Menu

In the view menu are several selections to make your life easier.

4.3.12 View Cross References

You can view a cross reference of any component of your event. For example, the Metaitem cross reference will show all outputs that include that metaitems, and (for sensor metatitems) all buttons that change the value of it. Similar cross references are available for Outputs and the other component types.

4.3.13 View Equipment Log

This selection generates a preview of the Equipment Log file written by the Event Logger. See 3.4.7 for a description of this file.

4.3.14 View Dependency Checks

In order to help you keep track of unfinished work, and in order to prevent Logger from trying to deal with incompletely defined event templates, Builder and Logger implement the *Dependency Check*. The Dependency Check scans all components, searching for references to other components that do not exist. A report is made for you in a popup dialog, listing all such references, and you must fix them before you can run the Logger.

You can execute a Dependency Check at any time in the Builder by going to the View menu and selecting Dependency Report. In addition, Builder automatically runs a Check any time you save a template file, and informs you if any errors exist (but does not give the full report).

Whenever you run Logger, the same check is made, and Logger will not run the event if there are errors – you must go back to Builder to fix them.

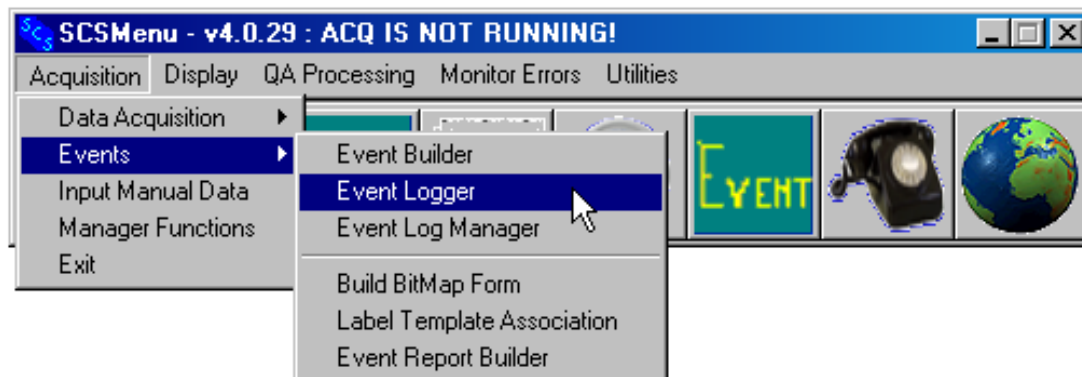
You can get an idea of the checks performed from figure 2.1, which shows the relationships among the event components; every arrow in this figure represents a reference to be checked. Moreover, checks go beyond the idea of references to other components. For example, the Dependency Check also verifies that Logger has permission from Windows to write a file to the Event Log directory.

The total number of checks is too great to be listed here. Suffice it to say that every attempt is made to insure that Logger will not run into unrecoverable errors during an event

4.4 Running an Event with Event Logger

The Logger receives the Event Template file from the Builder and executes the real-time logging activities specified there. There are three ways to start an event with the Logger.

From the SCS Menu



To start a single instance of the Logger, go to the SCS Menu, and from the Acquisition menu, select Event Logger -> Event. Logger responds by popping up a standard Windows File Open dialog showing all Event Template files in the SCSEventData directory. Select the desired template and press the OK button.

(NOT IMPLEMENTED) From the Event Log Manager

Events can be started in a batch using the Event Log Manager app, new to SCS v3.3a. The Event Log Manager is described in section 3.8

From MS-DOS Batch Files

You can start one or more instances of Logger from a batch file. To do this you create a DOS batch file with Notepad or some other text editor. You will enter one line for each event template you want to start. Each line in the batch file should look like this:

```
C:\AllSCS\Exe\EventLogger.exe trawl
```

Here, "C:\AllSCS\Exe" is the directory containing all the SCS executable files (substitute your own directory if different), "EventLogger" is the name of the Event Logger executable file, and "trawl" is a sample name of an event template with extension ".TPL". Note that you *do not* specify the TPL extension. NOTE: See Chapter One for further detail about the SCS directory trees.

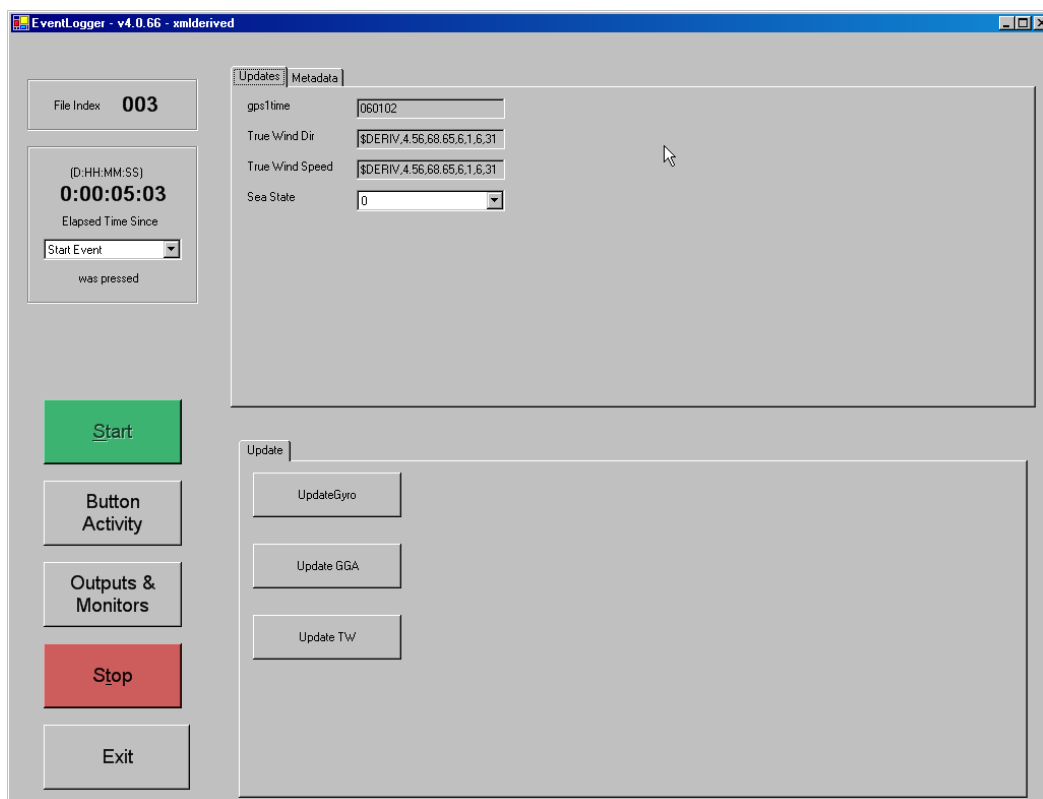
To execute the batch file, first bring up a "DOS command window". First, click on the Windows "Start" Button and select "Run... ". Then enter "CMD" in the "Open" box. In the DOS window, execute the batch file.

A Logger Main Dialog Window will appear for each instance of Logger started – the title of the window will have the name of the template file. This window is described below.

Running the Event – The Logger’s Main GUI

Figure 2.15 shows the Graphical User Interface (GUI) of the Logger Main Dialog window. The title bar of this window contains the program’s version number and the name of the event template being executed. Most of the GUI is taken by two tab sheets, one for meta tab groups, the other for button groups as defined in the Event Builder. Along the left side is a column of buttons and displays.

When the Logger Main Dialog first appears, the event is not yet running – it starts for real when you press the Start Event button.



Event Logger Main Dialog, Meta-item Activities

4.4.1 File Index Number

At the top left corner, this number is appended to the name of all output log files, header files and Button Activity files created by the event. Every time you run an event this number increases by one.

4.4.2 Elapsed Time

This display allows you to select any button defined for the template, and then shows the time elapsed since you last pressed that button. It is updated every second.

4.4.3 Start Button

Running the Event Logger and opening a template file do not by themselves start anything. Data logging cannot happen until you press Start Event. No user-defined buttons can be pressed either. You can however change editable manual Metaitems as you wish.

4.4.4 Meta-Item Tab Sheet

In the upper right of the GUI is the Meta-item Tab Sheet. The tabs across the top are named after the Meta Tab Groups you created in the Builder. Click on a tab to display the meta-items in that Tab Group. Tab sheets in the Logger's Metaitem display correspond to the meta tab groups you set up in the Event Builder.

The way a meta-item's value is displayed in its tab sheet depends on how you defined the meta-item in the Builder. Manual meta-items are normally displayed in edit boxes. In the previous figure the Cruise ID meta-item is editable, and the edit box background is white, indicating you can change the value during the event. The Time metaitem is not editable and its edit box has a gray background.

A list oriented meta-item, such as Sea State, it is displayed in a drop down list. Click on the "down arrow" at the right of the displayed value to bring up the list and make a selection.

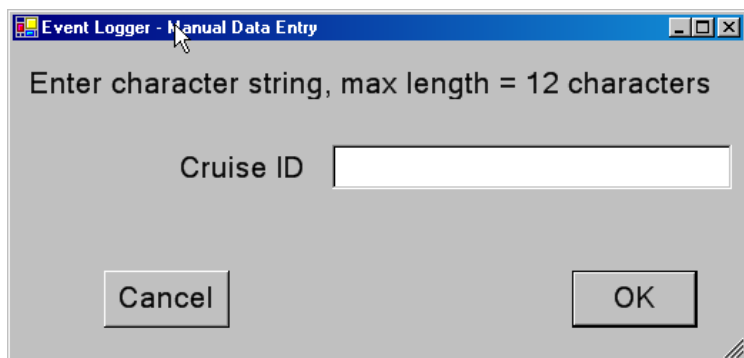
A sensor meta-item such as True Wind Dir or True Wind Speed is displayed in a text box, grayed out or not depending on if it is editable or not. There is no way of telling a sensor meta-item from a manual meta-item except by Label naming conventions.

Summary meta-items (none shown in this tab sheet) are also displayed in grayed out text boxes, which are blank until you press the Stop Event button. After that you will see the computed values displayed.

Changing the Value of a Metaitem

4.4.5 Manual Meta-item

Only editable manual meta-items can be changed during an event. Click on an editable meta-item value pops up an "Enter New Value" dialog.



Enter a new value, keeping in mind the restrictions on string length, and maximum and minimum values specified in the Builder. The updated value takes effect when you click on the OK button, and is then used for any subsequent Output logging activity.

4.4.6 Sensor Meta-item

A sensor meta-item gets its value from the SCS Acquisition data server. This value is updated whenever you press a Button that the meta-item is assigned to. Note that any meta-item can be assigned to multiple Buttons. The new value is then used for any subsequent Output logging activity. If you declared a sensor meta-item editable in the Builder you can change its value during the event. This allows you to recover from a dead sensor situation, but it should be used with care.

4.4.7 Summary Meta-item

Summary meta-items by definition are not editable. They are computed when you press the Stop Event button, and have no value until then. There is no way to re-compute a summary metaitem later.

Button Tab Sheet

The lower right of the GUI contains the Button Tab Sheet. Like meta-items, buttons are arranged in tab sheets that correspond to the Button Groups you defined in the Builder.

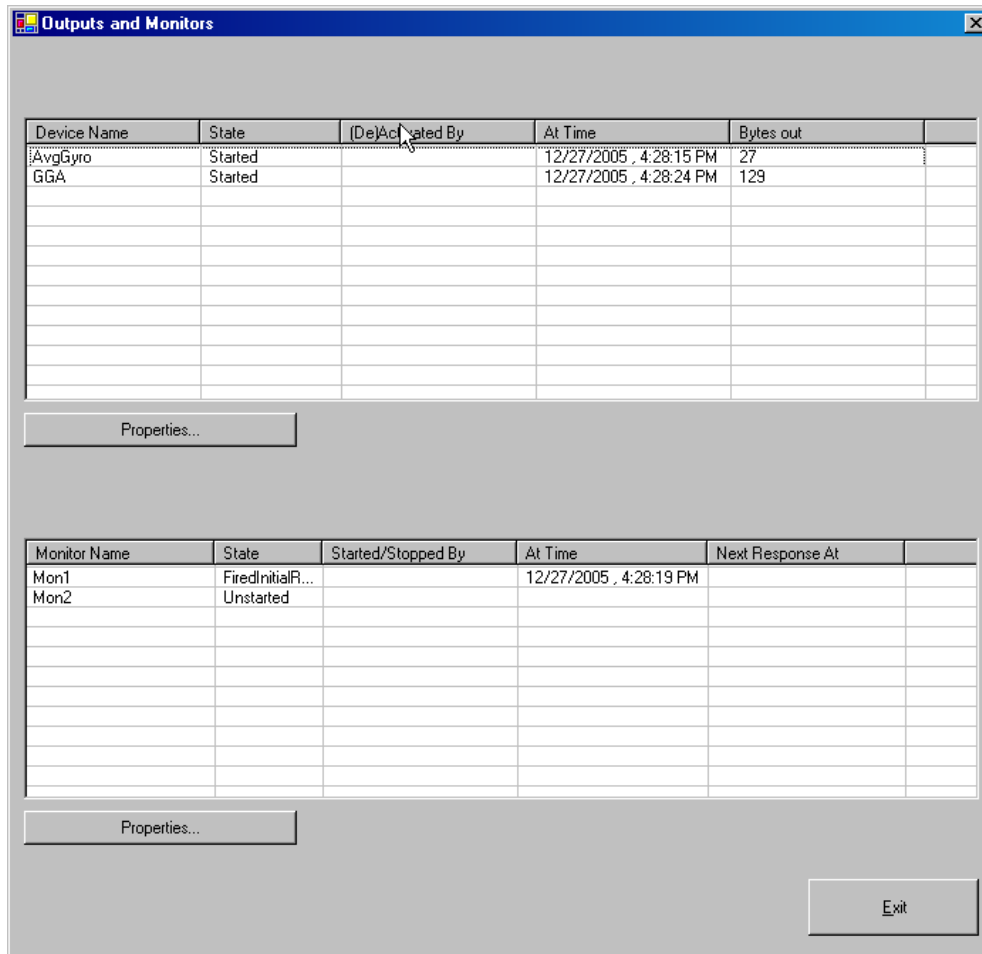
On a tab sheet buttons are displayed, either alone or as a pair within a group box. When the display is built, all paired buttons are displayed first, then all singleton buttons. Singleton buttons are typically associated with Snapshot outputs, and paired buttons with Continuous outputs.

When two buttons are paired the first is enabled while the second is disabled. When you press the first button it becomes disabled and the second becomes enabled. Then, when you press the second button the reverse happens. As the idea behind button pairs is to coordinate start and stop activity, this feature prevents you from starting or stopping an activity twice in a row. It also gives immediate visual feedback as to which button was last pressed.

Outputs and Monitors Display

Click on the Outputs & Monitors button to pop up this window. The following figure shows a sample display. Outputs are listed with several important items of information about them.

- The “Output Device Name” is the Label you gave to the Output
- The State of an Output is either “Started” or “Inactive” (relevant for Continuous Outputs only)
- The (De)Activated column shows the button that was pressed to bring the Output into its current state.
- The “At Time” shows the time that the (de)activating button was pressed (either by you or a Monitor)
- The “Lines / Bytes Out” shows two numbers, the number of lines written to the Event Data file, and the total number of bytes written.



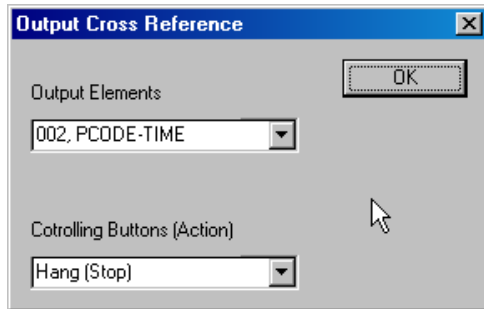
Outputs and Monitors Display

For all monitors the following information is given:

- The “Monitor Name” is the Label you gave to the Monitor.
- The “State” column shows whether the Monitor is actively watching anything.
- The “Started By” column shows the name of the Button that started the Monitor
- The “Start Time” column shows the time that the Monitor was started
- The “Next Response At” column shows the time of the next response, either the initial response or the follow-up.

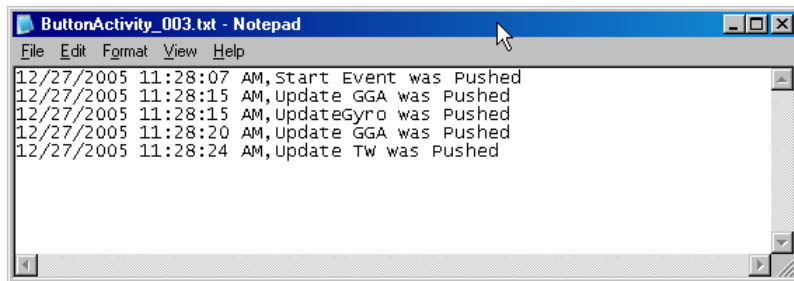
Output Cross Reference

Click on any output in the list, then press the Properties... button. This brings up a dialog box containing details about the output. A drop down lists displays a list of all output elements and another list shows all buttons that control the output. Click on a button in the list to show the kind of action taken when that button is pressed.



Button Activity Window

Press the Button Activity Button on the left side of the Logger GUI to bring up a summary of all button presses during this event. A sample window is shown in figure 2.19. This data is also written to the ButtonActivity_NNN.TXT file (where NNN is the file sequence number) in the same directory as the data files.



Button Activity Display

Stopping the Event

Press the Stop Event Button in the lower left of the GUI to stop the event. When you do this the following tasks are executed:

- Active Outputs are stopped and their Log files closed out.
- The current value of all manual and sensor meta-items is written to the Header file.
- The values of all summary meta-items are computed and appended to the Header file.
- The Button Activity file is written, having the time of each button press during the event.
- A Save file is written, having the last recorded values of all meta-items. It will be used to initialize meta-items at the start of the next run of an Event.
- A Message file, containing all the messages to the user that Logger generated during the Event, is written to the Event's output directory.

You may now review the Messages list, summary meta-item values, etc, before exiting the Logger application.

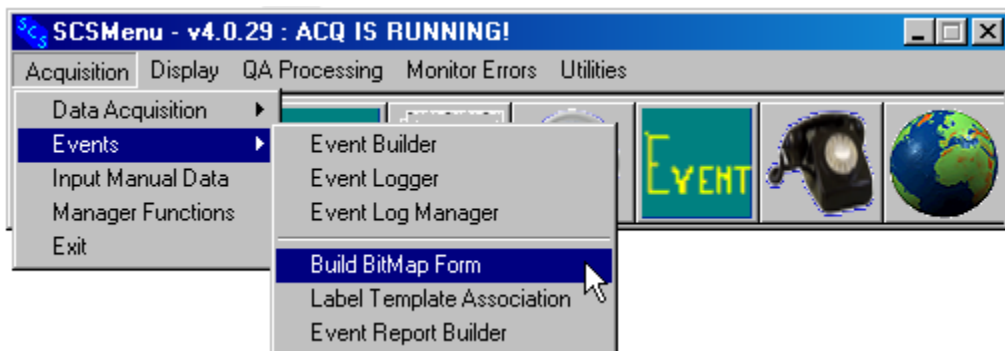
Exiting the Event Logger

Press the Exit button to close the Logger application. The Exit Button will not function if there is an event running – you must press the Stop Event button first. No calculations or file activities take place when you exit Logger.

4.5 (NOT IMPLEMENTED) Event Log Manager

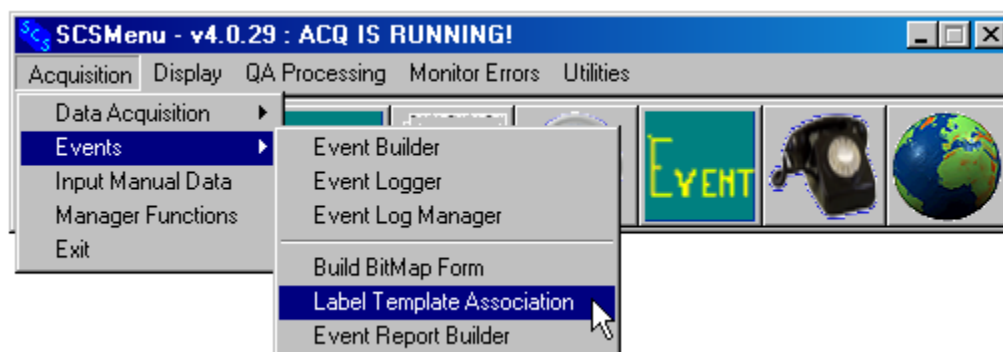
4.6 (NOT YET IMPLEMENTED) Event Report Building and Generation

4.7 Build Bitmap Form

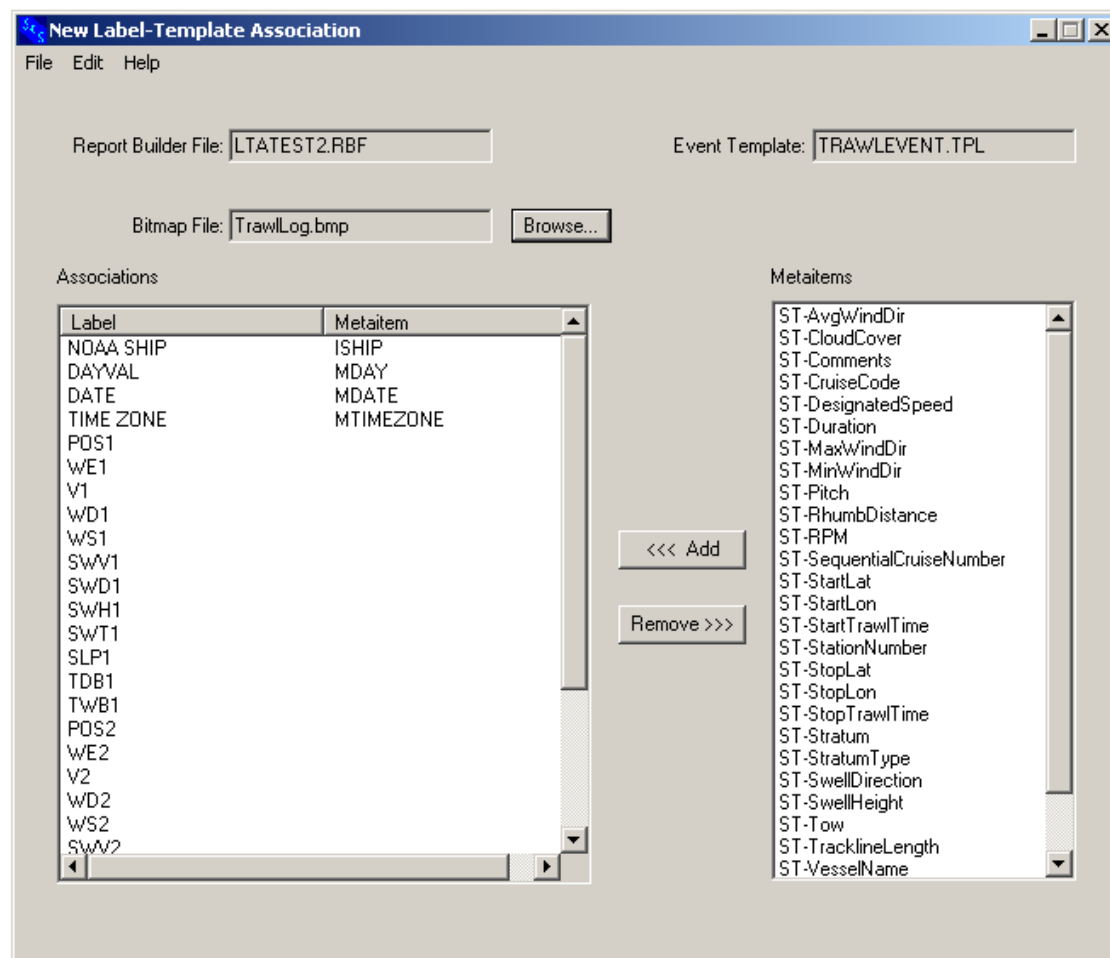


The purpose of this program is to first bring in a scanned in bitmap image of the desired paper form. With this bitmap in place, the user will then proceed to place the cursor at the exact pixel positions where the data will be filled in on the form. Each one of these positions will get a label value; and each one of these labels will have an associated X/Y coordinate pixel position. All of this information is saved to a Report Builder File (*.RBF), which will then be used by the Label/Template Association program. The following shows the dialog window:

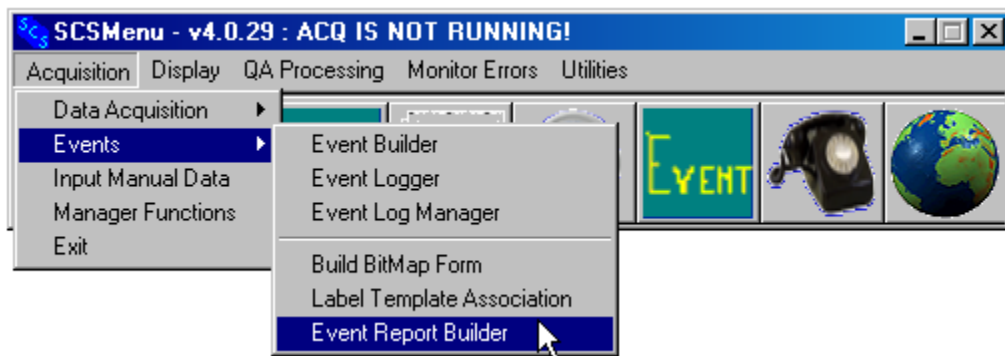
4.8 Label/Template Association (LTA)



This is the second of three programs that need to be run in order to build and generate an Event Log Report. This program first loads in the Report Builder File (*.RBF) created from the Build Bitmap Form application. Then, the system manager/user must associate which Event Log meta-items will be associated with each and every label. Once this is complete, this association information is added to the same Report Builder File for use then in the Event Report Builder application. The following shows the dialog window:



4.9 Event Report Builder



This is the last of three programs that needs to be run in order to automatically generate a paper Event Log report based on any scanned in form. This program first reads in the finalized Report Builder File (*.RBF) created from the Label/Template Association program. Then, before the report is generated and printed, the needed meta-item data from the specific Event Log Header File (*.HDR) is read into the program. This is the data that is used to fill in the form before it is ready for printing. The following shows the dialog window:

The screenshot shows a window titled "Report Builder: S:\Ship\McArthur\ReportBuilderFiles\LTATEST2.RBF; Hdr Data File: ; BitMap: ...". It has a menu bar with "File", "Edit", "View", and "Help". Below the menu bar are icons for ".rbf", ".bmp", and a save icon. The main area displays a NOAA Form 77-13d, "DECK LOG - WEATHER OBSERVATION SHEET". The form includes fields for "NOAA SHIP", "DAY", "DATE", and "TIME ZONE". It also has a table for weather observations with columns for "TIME", "POSITION (Lat. and Long.)", "PRESENT WEATHER", "VISIBILITY (Miles)", "WIND (Dir. (True) Speed (Kts-))", "SEA WAVE HEIGHT (Ft-)", "SWELL WAVES (Dir. (True) Height (Ft-))", "SEA WATER TEMP. (C-)", "SEA LEVEL (Pressure (mb))", and "TEMPERATURE (C-)" (Dry Bulb, Wet Bulb). The table has rows for times 01 through 17. The status bar at the bottom says "Ready".

TIME	POSITION (Lat. and Long.)	PRESENT WEATHER	VISIBILITY (Miles)	WIND		SEA WAVE HEIGHT (Ft-)	SWELL WAVES		SEA WATER TEMP. (C-)	SEA LEVEL (Pressure (mb))	TEMPERATURE (C-)	
				DIR. (True)	SPEED (Kts-)		DIR. (True)	HEIGHT (Ft-)			DRY BULB	WET BULB
01												
02												
03												
04												
05												
06												
07												
08												
09												
10												
11												
12												
13												
14												
15												
16												
17												

CHAPTER FIVE

SCS OPERATION EVENT LOGGER (SCSOEL)

5 OPERATION EVENT LOGGING (OEL)

5.1 Introduction

SCSOEL is a suite of three applications, SCSOEL Builder, SCSOEL Logger and SCSOEL Event Extractor. The OEL is a variation of traditional SCS Event Logging, and was written for the Fisheries Scientific Computer System (FSCS). SCS OEL is an adaptation of the FSCS OEL.

In purpose functionality it is closely related to traditional SCS Event Logging, but there are some major differences.

- SCSOEL apps all read and write data to the SCS database, and thus cannot be used in SCS's legacy mode.
- The OEL Builder does not create templates; instead, it writes the templates to the SCS database.
- OEL Logger does not create any traditional data files (*.ELG), nor does it create a HDR file. This data is written to the SCS database and is extracted in post processing by the OEL Extractor application.

Terminology Differences:

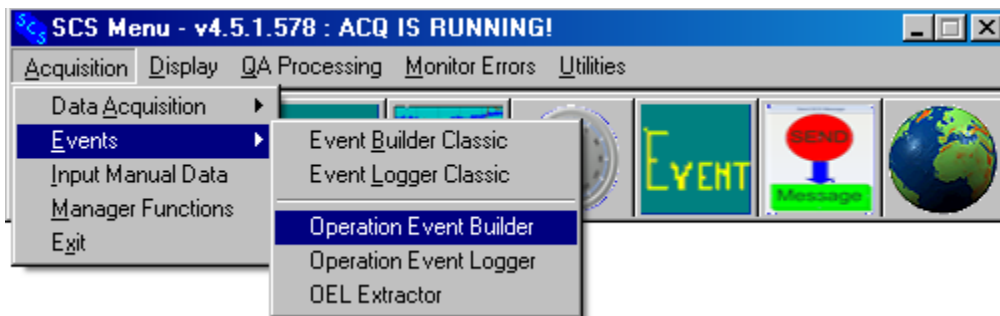
- Classic Output File (*.ELG)" is now called a Stream in OEL
- Classic Sensor Metaitems, Manual Metaitems, Monitors, and Alarms now fall under the term OEL "Data Component"
- Monitor and Alarm terminology remain the same
- Classic Event Template names are called Operation Types in the OEL Builder and Logger apps.

New Features

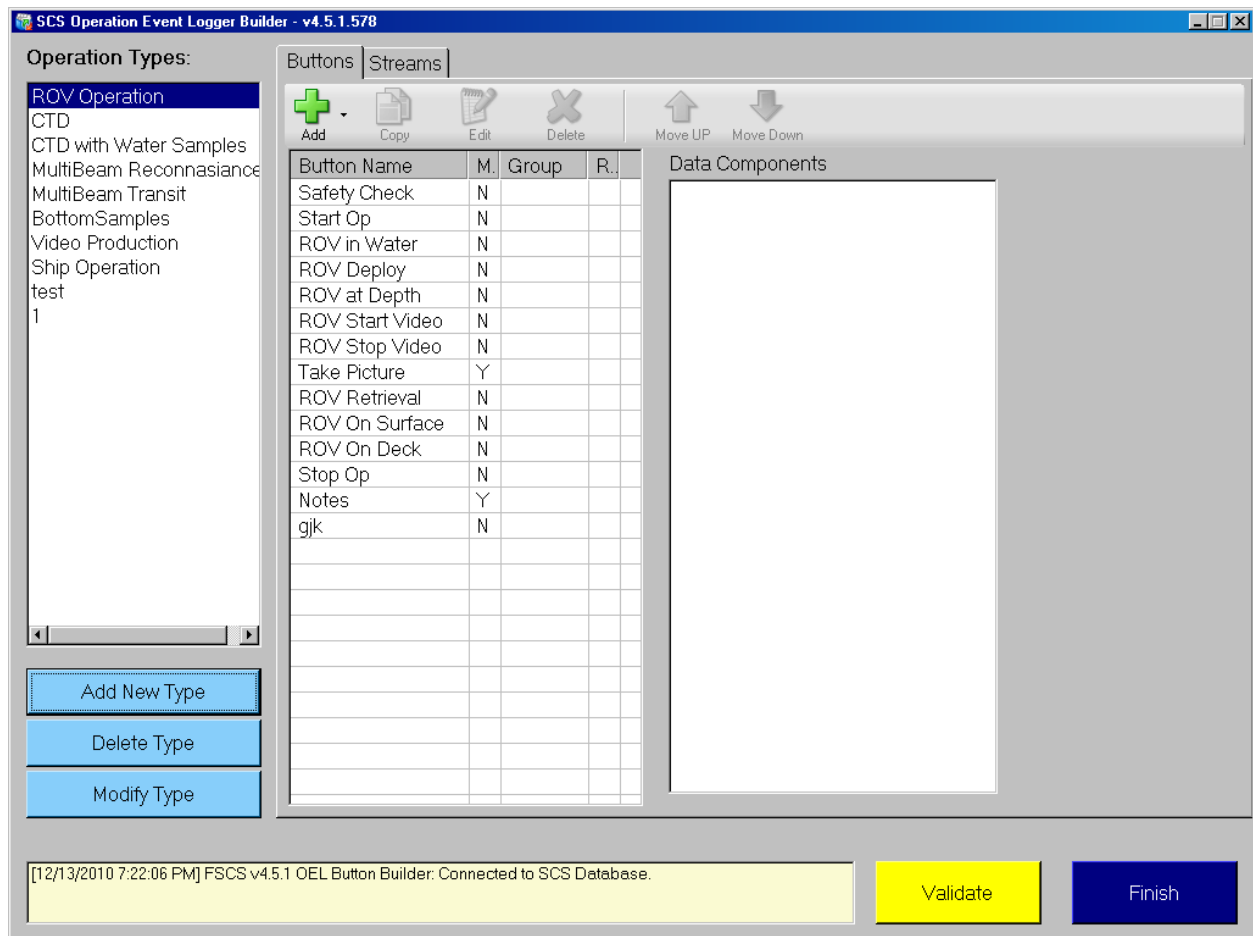
- Button Sequencing: Classic Event Logging defined only primitive button pairing. OEL can define long sequencing of buttons with options for forcing buttons to be pressed in a pre-defined order. It can also designate buttons as "single press" (may only be pressed once during an event, or "multi press" (able to be pressed more than once).
- Summary Metaitems are not defined in the OEL Builder and Logger. This follows the standard set for FSCS 2.0 that all summary metaitems would be computed by NOAA Office of Fisheries in post processing.
- When a button is pressed in the OEL Logger, all metaitem and sensor metaitem values are displayable immediately on the Logger main window.

The following pages show the set of GUI windows displayed in the different aspects of creating or editing an Operation Type with the OEL Builder.

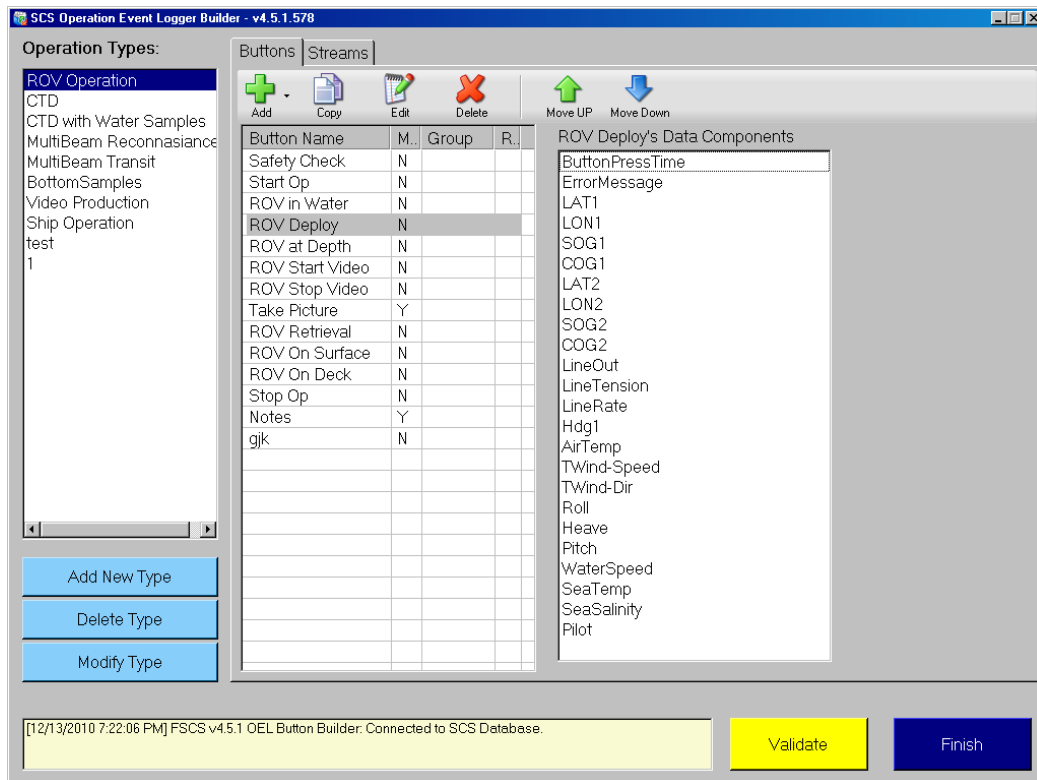
5.2 OEL Builder



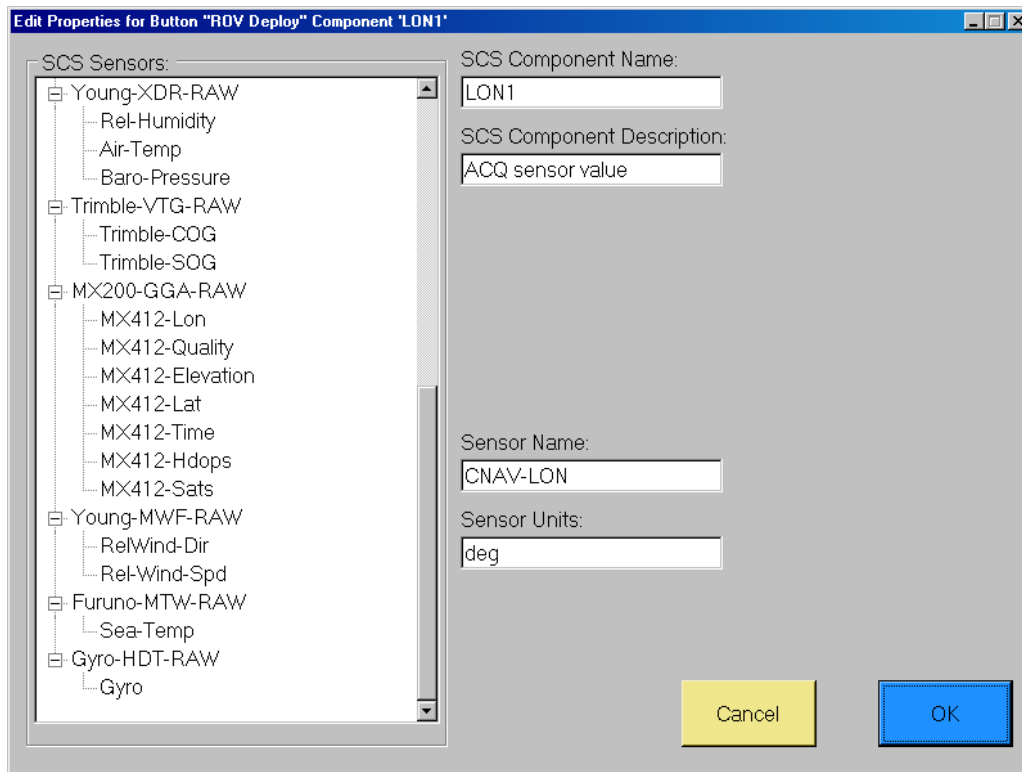
Initial screen



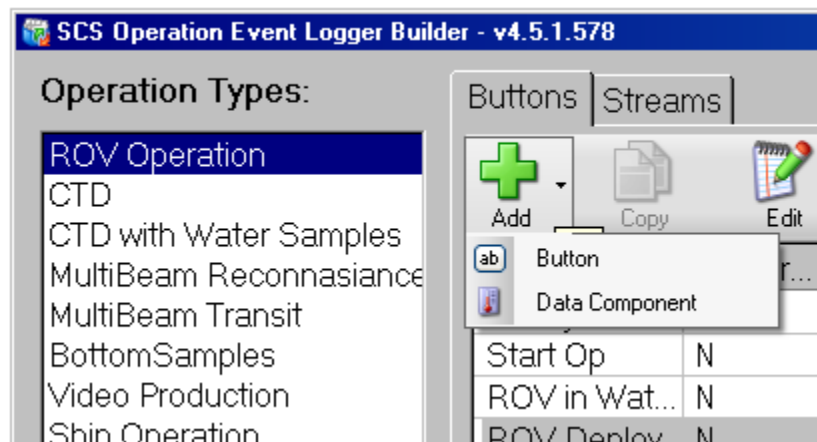
Pick a button



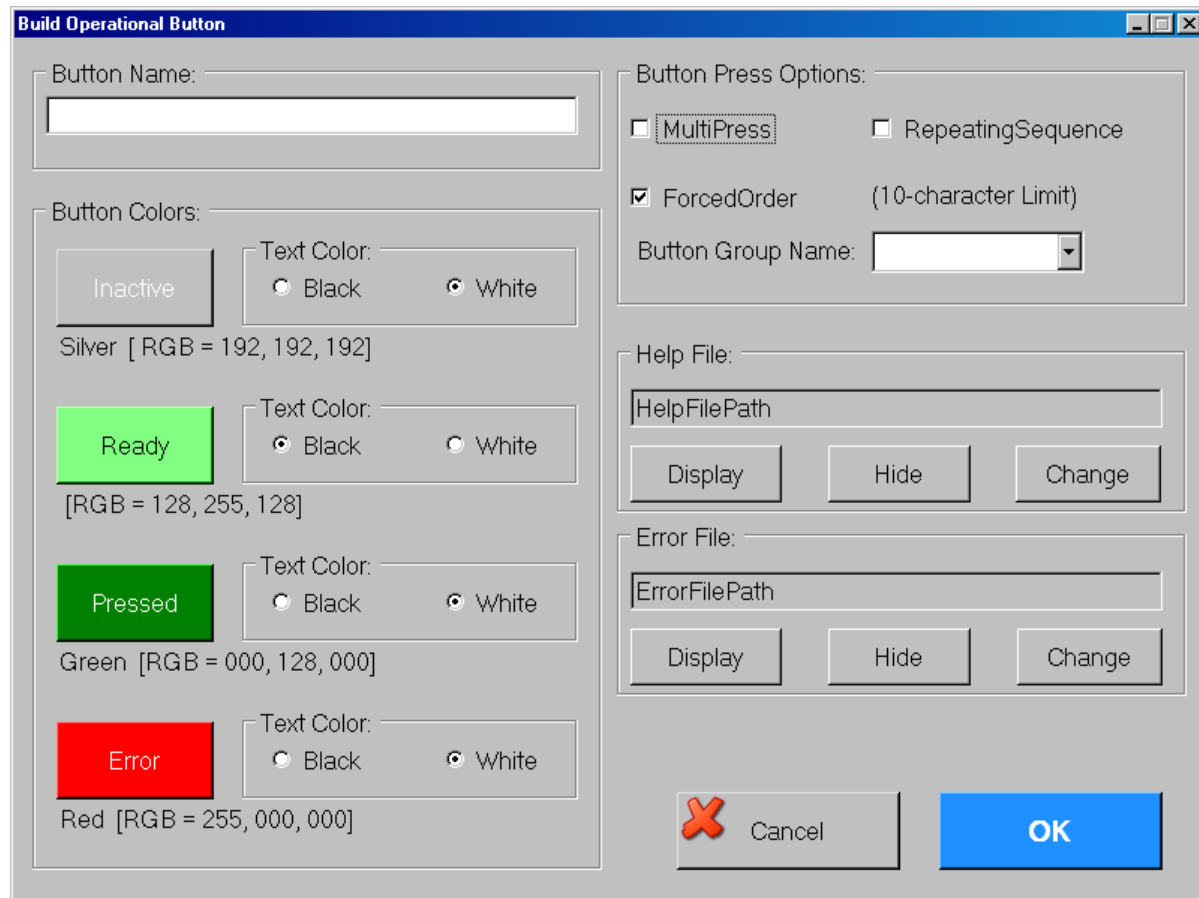
Pick Lon1 component and edit



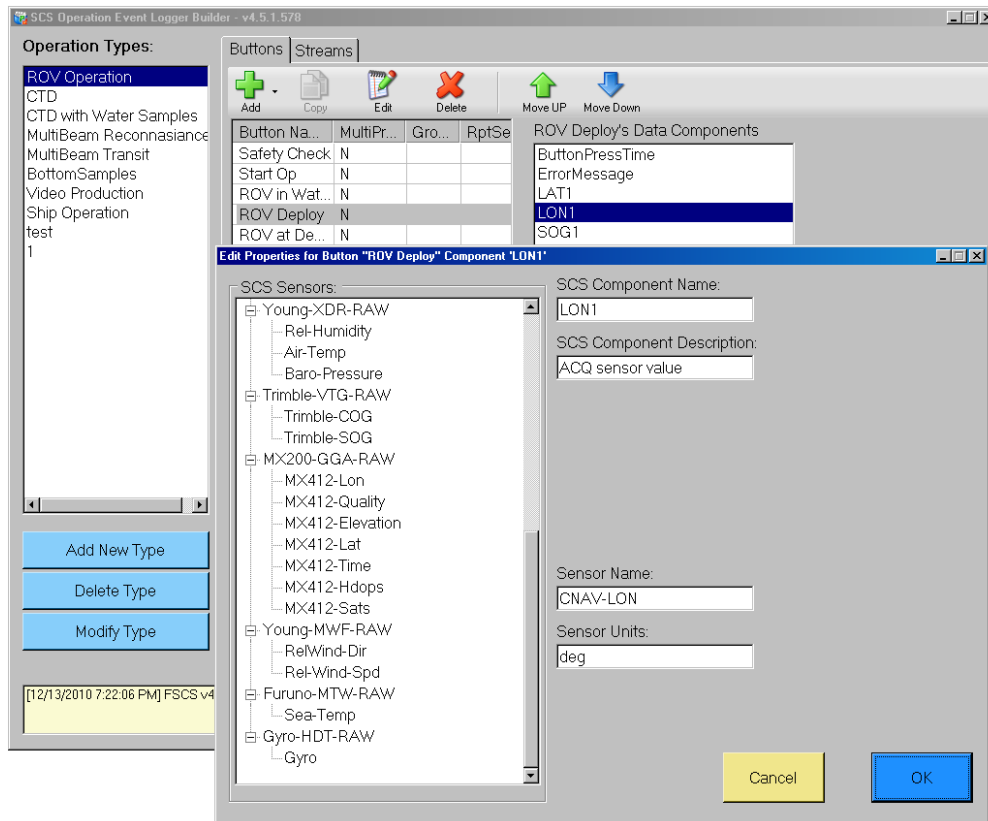
Adding a button or data component



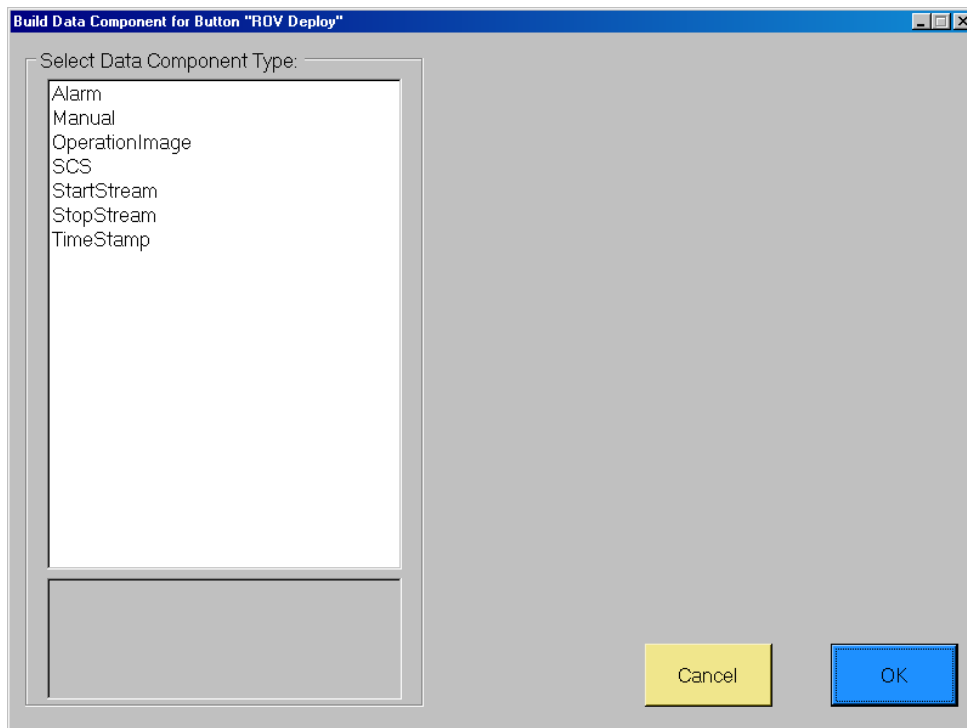
Add a new button



Pick a button and edit



Add a data component



Add alarm data component

The screenshot shows a Windows-style dialog box titled "Build Data Component for Button 'ROV Deploy'". On the left, under "Select Data Component Type:", a list contains "Alarm", "Manual", "OperationImage", "SCS", "StartStream", "StopStream", and "TimeStamp". "Alarm" is selected. Below this list is a text box with the instruction "Set an Alarm and specify Message and Sound". On the right side of the dialog, there are several fields: "Component Name:" with an empty text box; "Interval - Minutes:" with a spinner box set to "10"; "Component Description:" with an empty text box; "Seconds:" with a spinner box set to "0"; "Alarm Message to Display:" with a large empty text area; "Alarm Sound File:" with an empty text box; a "Play" button; a "Change" button; a checked checkbox labeled "Primary Alarm Display when Pressed"; and at the bottom, "Cancel" and "OK" buttons.

Add manual data component

The screenshot shows the same dialog box as above, but with "Manual" selected in the "Select Data Component Type:" list. Below the list, the "Manual Data Type:" dropdown menu is open, showing "String - alphanumeric character string" as the selected option. On the right side, the fields are: "Manual Component Name:" with an empty text box; "Component Description:" with an empty text box; "Default Value:" with an empty text box; "Input Length:" with a spinner box set to "40"; and at the bottom, "Reset", "Cancel", and "OK" buttons.

Add operational image data component

Build Data Component for Button "ROV Deploy"

Select Data Component Type:

- Alarm
- Manual
- OperationImage
- SCS
- StartStream
- StopStream
- TimeStamp

Component Name:

Component Description:

Collect information about an Operation Image

Cancel OK

Add scs data component

Build Data Component for Button "ROV Deploy"

SCS Sensors:

- Trimble-GGA-RAW
 - Trimble-Quality
 - Trimble-Lat
 - Trimble-Lon
 - Trimble-Hdops
 - Trimble-Time
 - Trimble-Sats
 - Trimble-Elevation
- MX200-VTG-RAW
 - MX412-SOG
 - MX412-COG
- Young-True-Wind
 - Young-True-Wind-SPEED
 - Young-True-Wind-Direction
- EQ50-SDDBS-RAW
 - EQ50-Depth-fm
 - EQ50-Depth-m
 - EQ50-Depth-ft
- GyroTimesTwoDevice-DRV
 - GyroTimesTwo-Value
- Young-XDR-RAW
 - Rel-Humidity
 - Air-Temp

SCS Component Name:

SCS Component Description:

Add >>

<< Remove

Sensor Name:

Sensor Units:

Selected Components:

Reset Cancel OK

Add start stream or stop stream data component

Build Data Component for Button "ROV Deploy"

Select Data Stream:

Component Name:

Component Description:

Reset Cancel OK

Streams tab

SCS Operation Event Logger Builder - v4.5.1.578

Operation Types:

- ROV Operation
- CTD
- CTD with Water Samples
- MultiBeam Reconnaissance
- MultiBeam Transit
- Bottom Samples
- Video Production
- Ship Operation
- test
- 1

Add New Type

Delete Type

Modify Type

Buttons Streams

Stream Name	Log Interval	Start
log2	5 sec	Auto

Add SCS Stream

Edit SCS Stream

Remove

Stream Data Components:

LAT1
LON1

Add Data Component

Edit Data Component

Remove

Move Component:

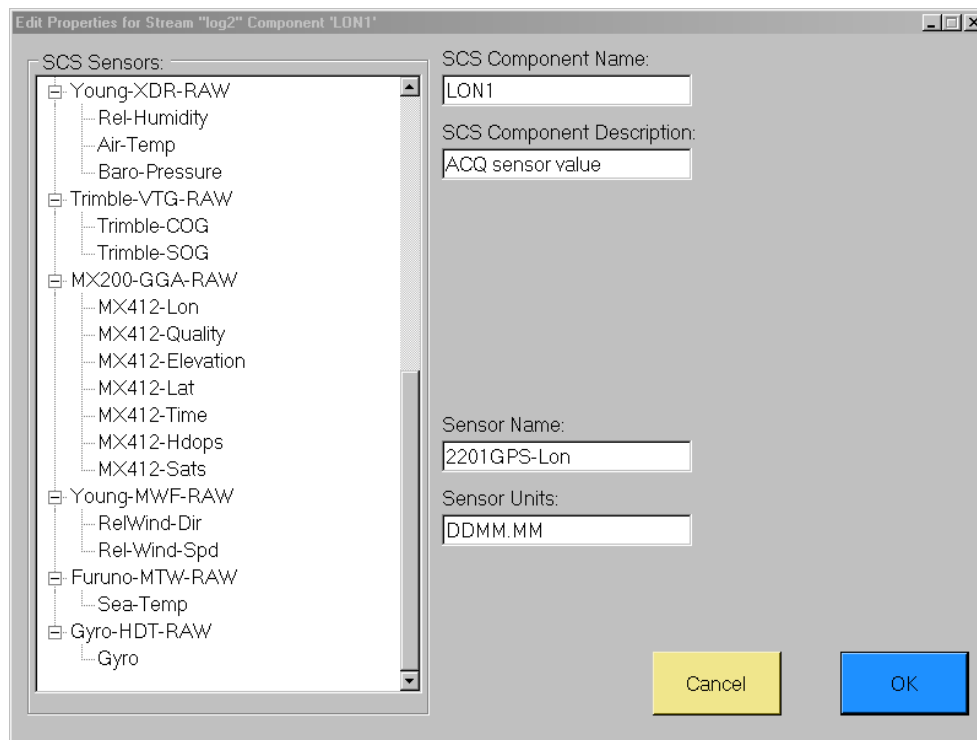
Up

Down

[12/13/2010 7:22:06 PM] FSCS v4.5.1 OEL Button Builder. Connected to SCS Database.

Validate Finish

Edit Stream data component



Dialog box titled "Edit Properties for Stream 'log2' Component 'LON1'".

SCS Sensors:

- Young-XDR-RAW
 - Rel-Humidity
 - Air-Temp
 - Baro-Pressure
- Trimble-VTG-RAW
 - Trimble-COG
 - Trimble-SOG
- MX200-GGA-RAW
 - MX412-Lon
 - MX412-Quality
 - MX412-Elevation
 - MX412-Lat
 - MX412-Time
 - MX412-Hdops
 - MX412-Sats
- Young-MWF-RAW
 - RelWind-Dir
 - Rel-Wind-Spd
- Furuno-MTW-RAW
 - Sea-Temp
- Gyro-HDT-RAW
 - Gyro

SCS Component Name: LON1

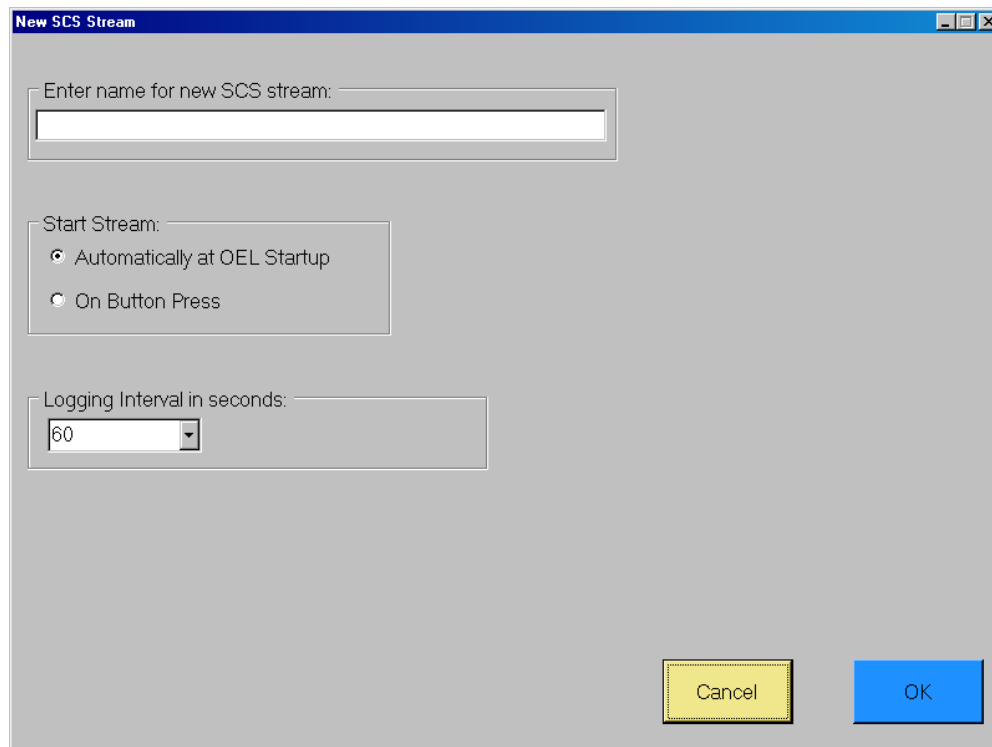
SCS Component Description: ACQ sensor value

Sensor Name: 2201GPS-Lon

Sensor Units: DDMM.MM

Buttons: Cancel, OK

Edit existing or Add new scs stream



Dialog box titled "New SCS Stream".

Enter name for new SCS stream:

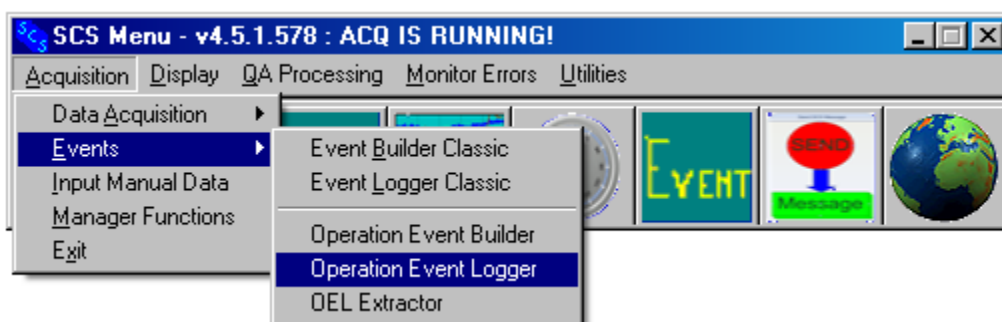
Start Stream:

- ☒ Automatically at OEL Startup
- ☐ On Button Press

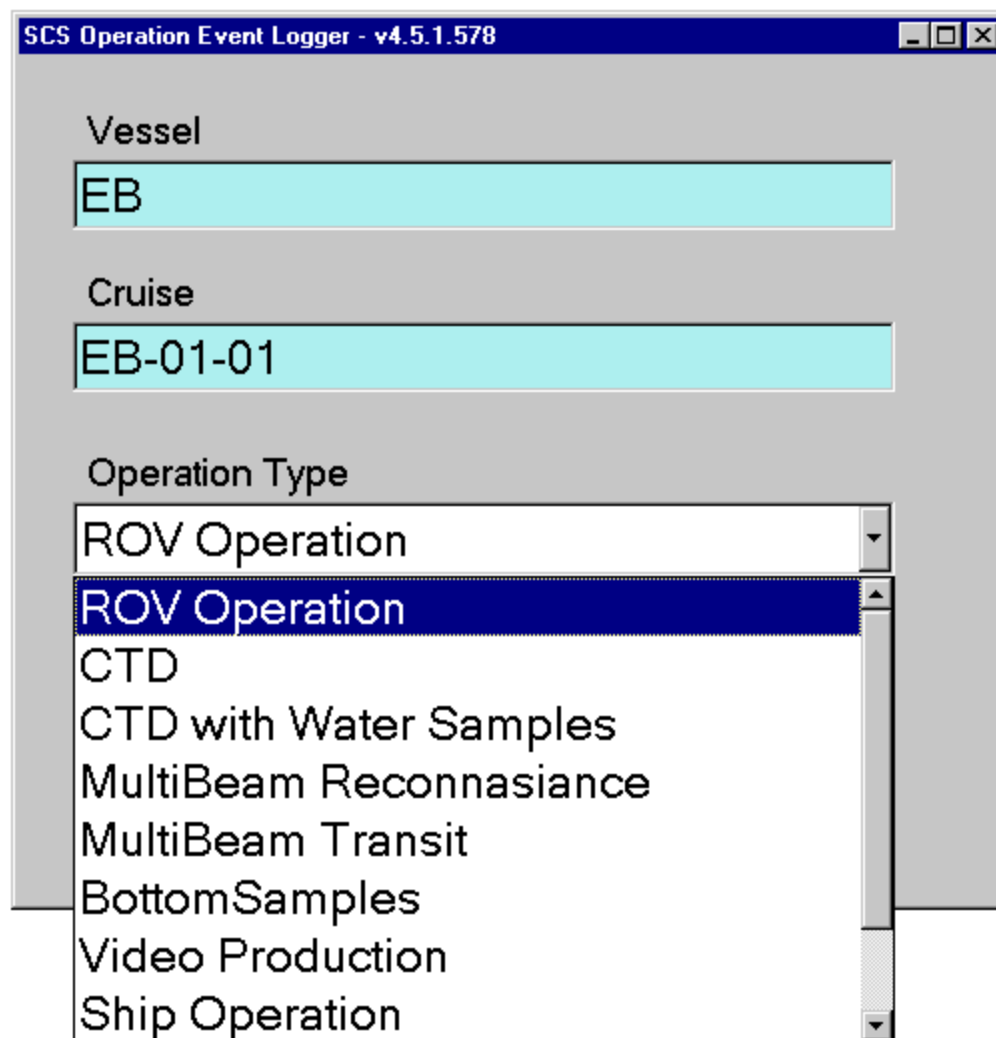
Logging Interval in seconds: 60

Buttons: Cancel, OK

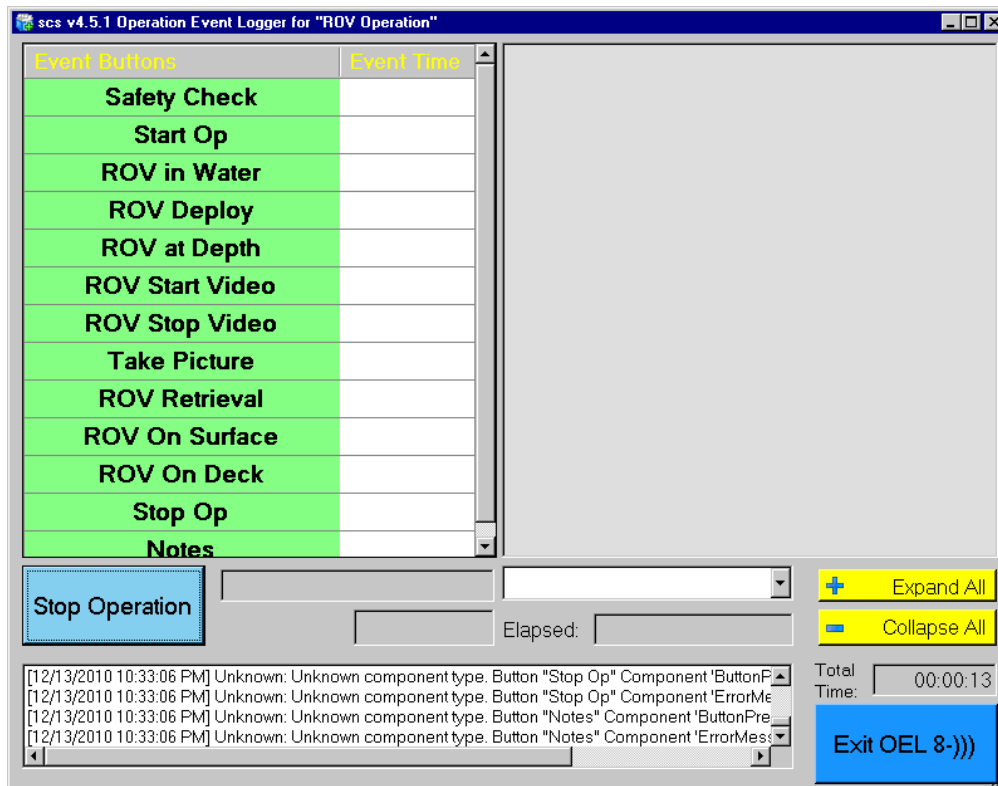
5.3 OEL Logger



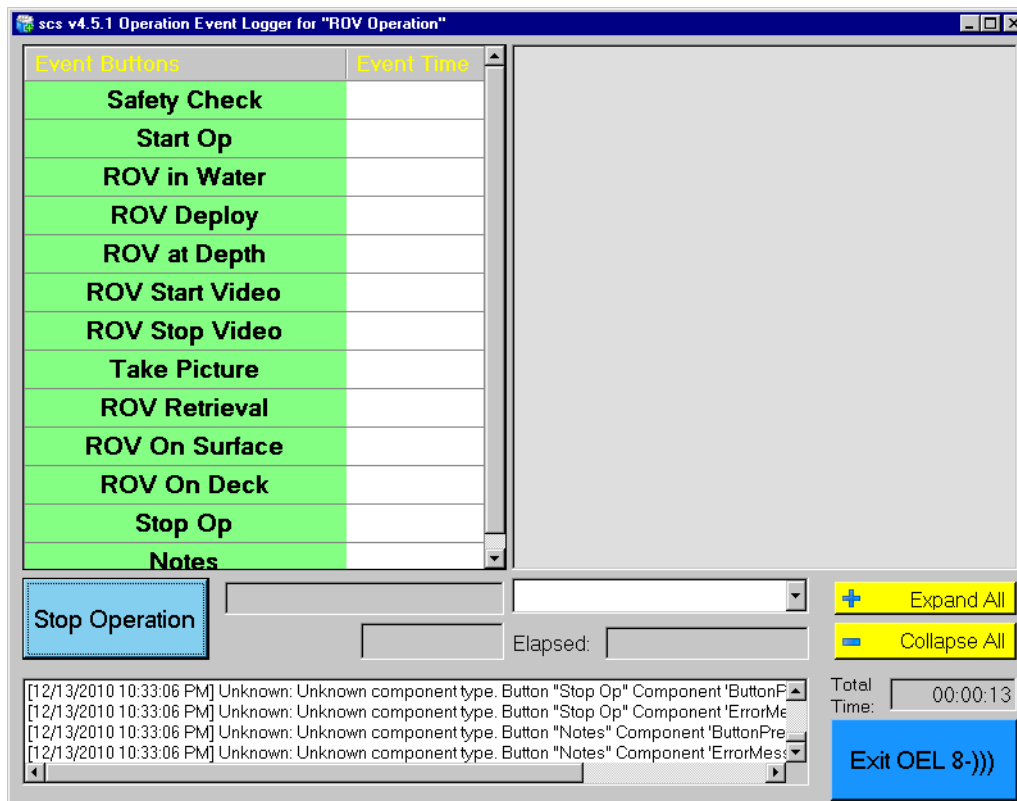
Opening screen



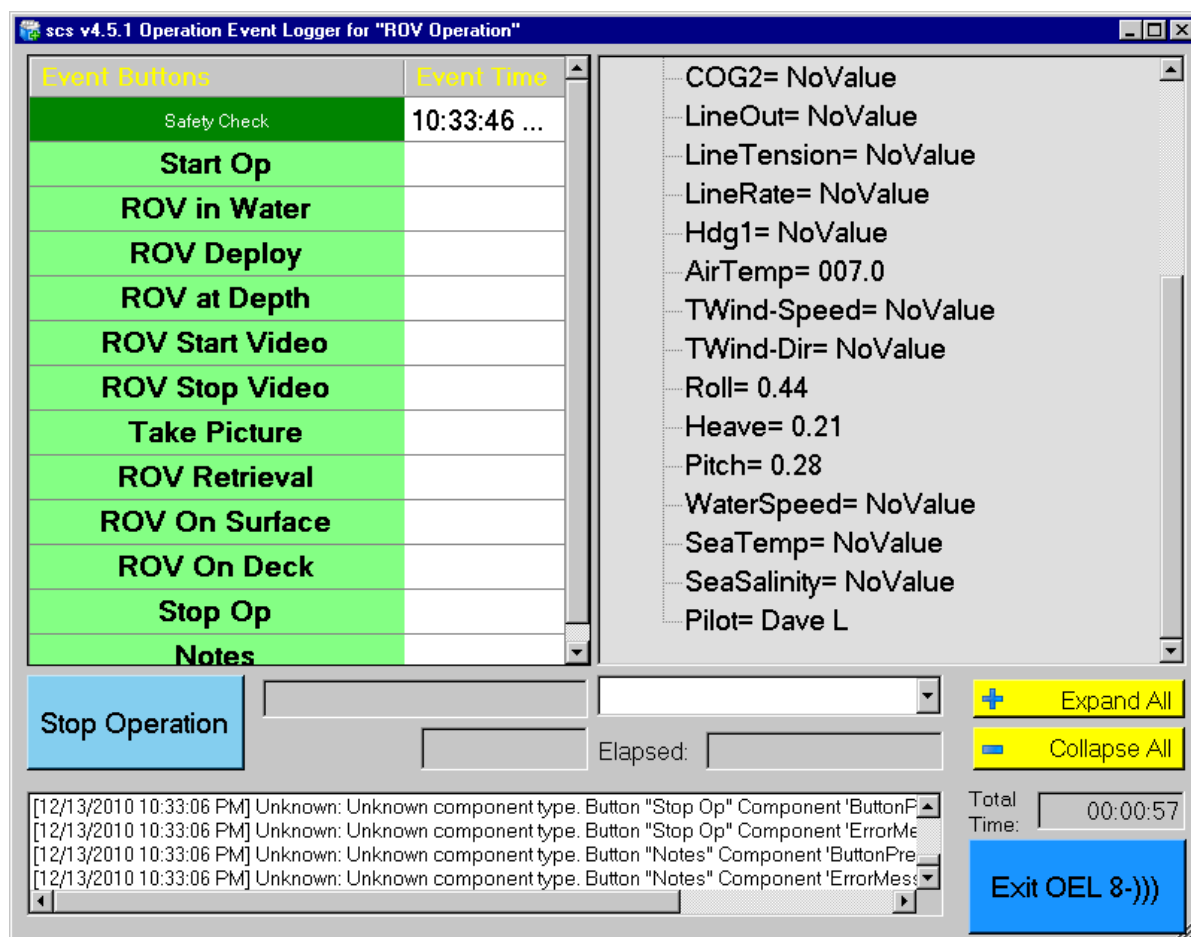
Select "ROV Operation", press "Start Operation"



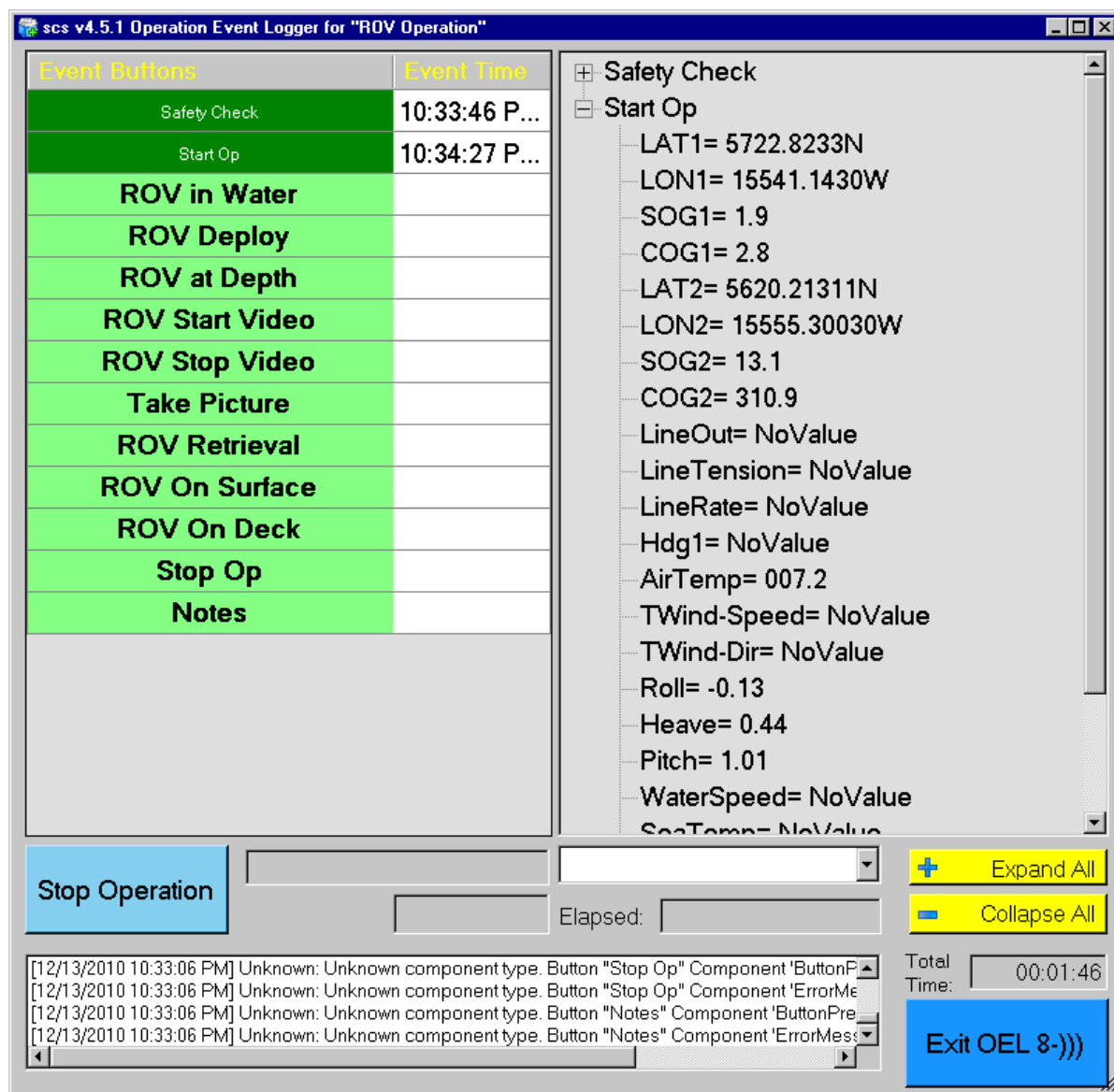
Press "Start Operation"



Press safety check button. Button background turns dark green indicating it has been pressed. Small font indicates it cannot be pressed again. Note tree view on right showing the values of all data components collected by the button press.



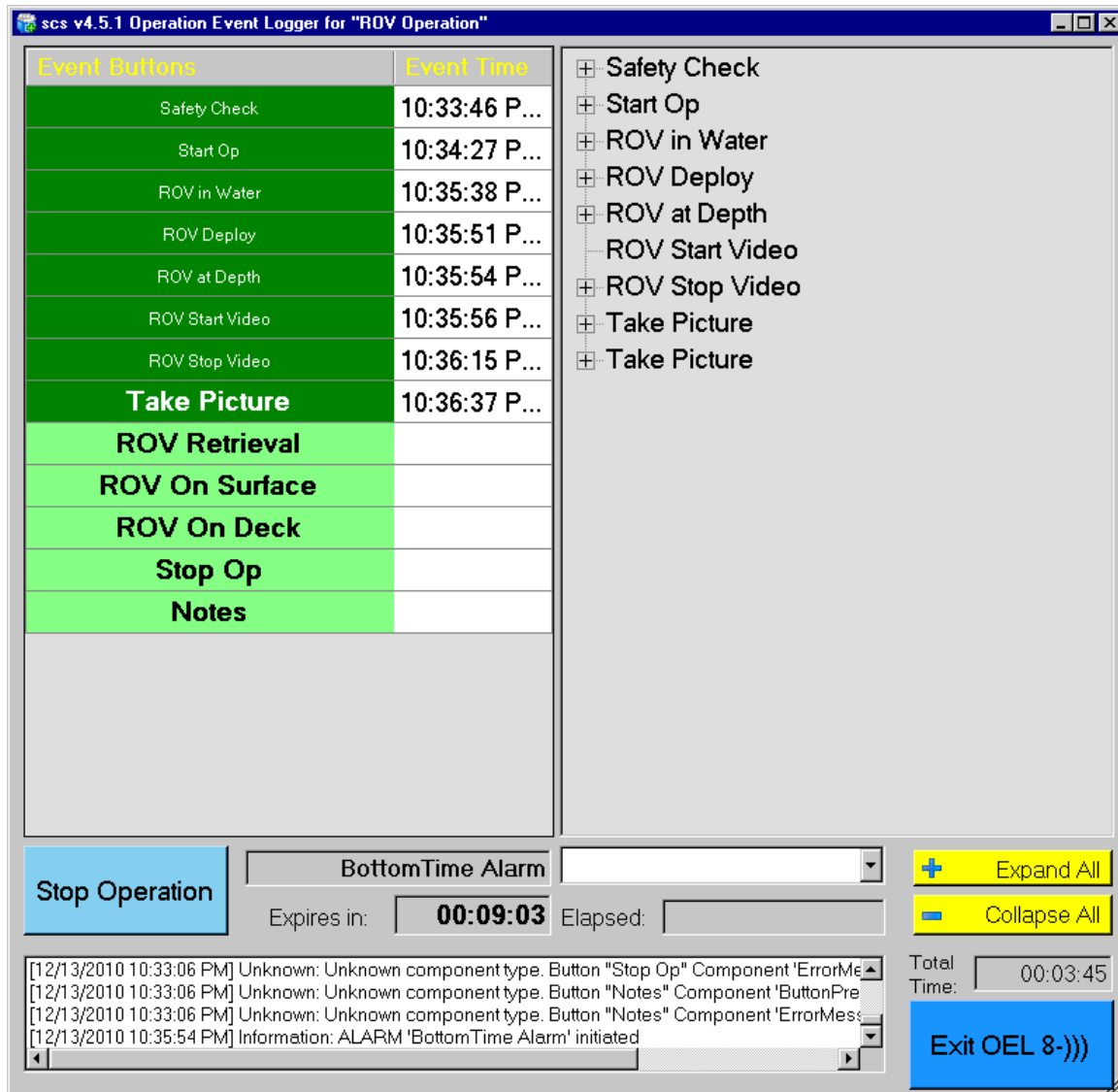
Press "Start Op". It cannot be pressed again. Note tree view on right. A new tree node is inserted for each button press.



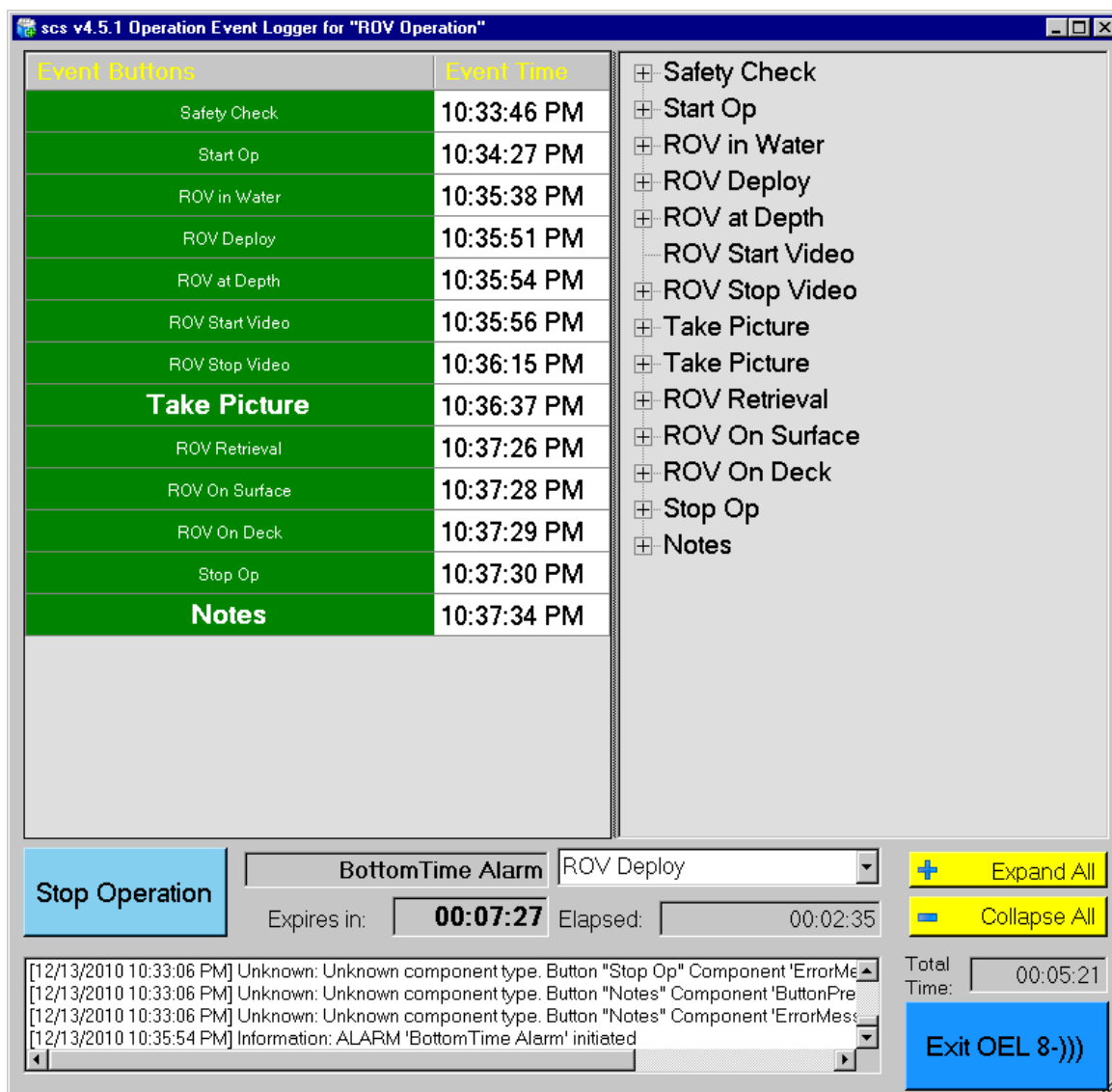
Press the next few buttons. Note that button order is not enforced.

Then press "Take Picture". Large font indicates it can be pressed multiple times.

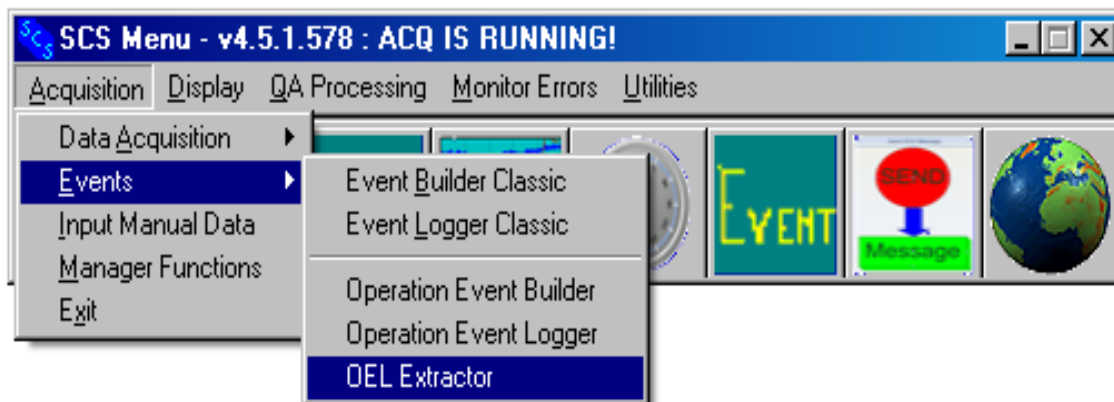
One of the buttons has started a monitor/ alarm (00:09:03)



Press a few other buttons. Tree continues to grow. Monitor continues to count down (00:07:27)



5.4 OEL Extractor

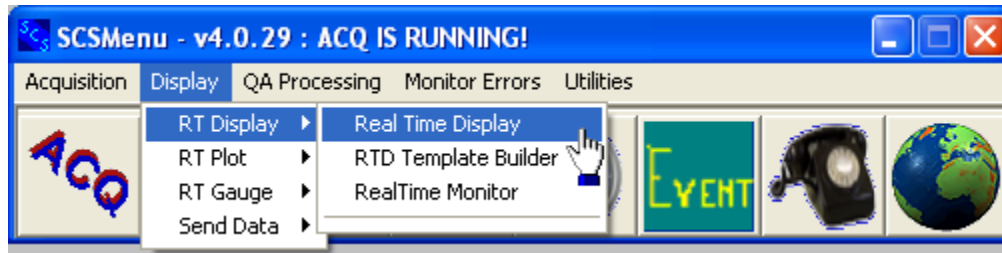


CHAPTER SIX

REAL-TIME DISPLAYS

6 REAL TIME DISPLAYS

6.1 Introduction



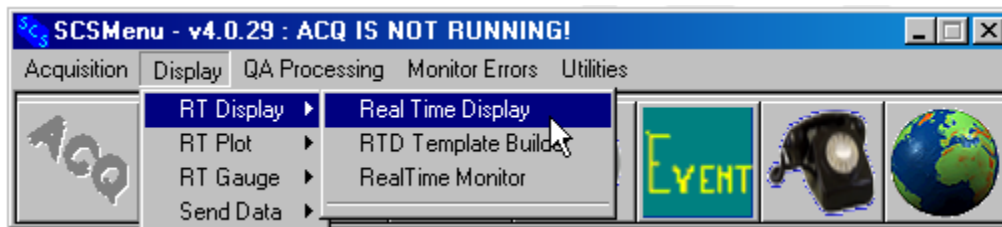
The programs invoked in the Display menu display sensor data in various text and graphic forms in real time. SCS provides these types of displays:

- The time series plot which allows the user to plot multiple SCS sensors against time.
- The XY plot which allows the user to plot multiple sensors against a sensor set as the X axis,
- The Track-line GIS program that allows the user to plot up to two navigable bodies on a latitude-longitude grid.
- Gauges, both circular and linear.
- Customized text displays

Also supported in the Display menu are text message outputs to serial and network ports that can include any combination of sensors.

All displays can be saved as *templates*, which can be recalled at a later time to recreate the exact look and feel of the display.

6.2 Real Time Display (Normal and Large Fonts)



Function

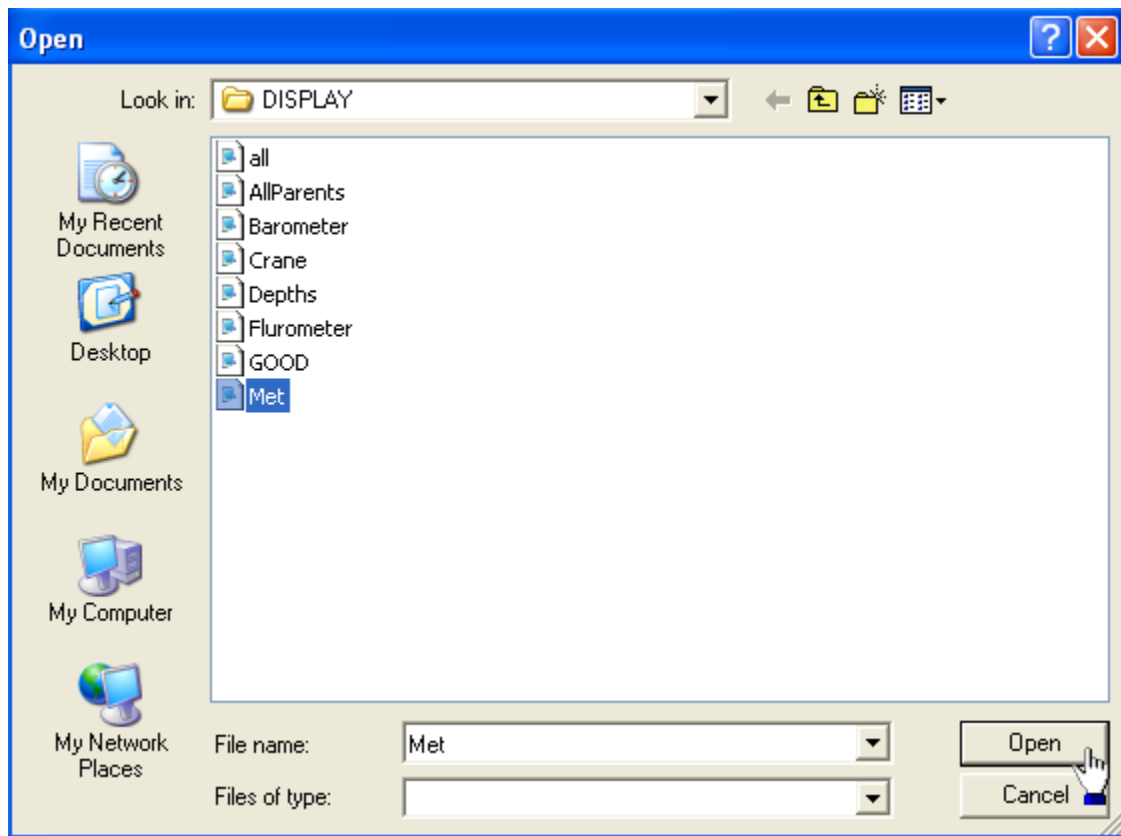
The primary function is to provide a real time sensor data text display from as it is being acquired in real time by ACQ.

This Real Time Display program provides a textual display of sensor data. The data is placed into windows on the screen - with sensor element name and units to the left of the data - that are set up by a display file (files with a .SEL extension located in the Templates/Display directory under the ship directory). These display files hold the sensor names which point to the information to be displayed in the window and are custom built by the user with the RTD TEMPLATE BUILDER program (see next section).

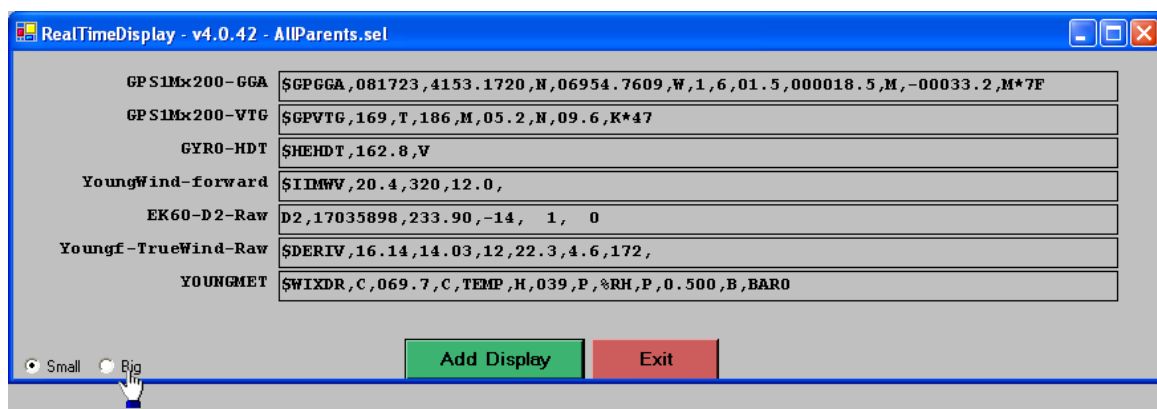
You can run as many RTD's as you want on the same screen.

Procedure

When the program is launched all of the available Display Files are listed, and the currently selected file is shown in the Selected File box. Select (highlight) the desired file and hit the OK button.



This will pop up a new window that will display the appropriate sensor data. Here is a sample Real Time Display of several parent sensors in the small font.



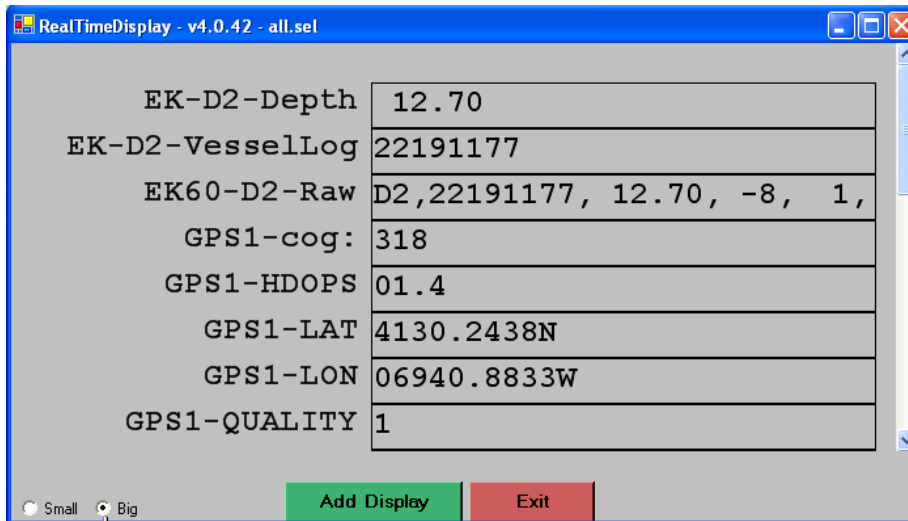
Options

6.2.1 Add Display Button

Simply brings up another instance of the original application, allowing the user to add multiple displays without needing to go back to the original SCS Menu.

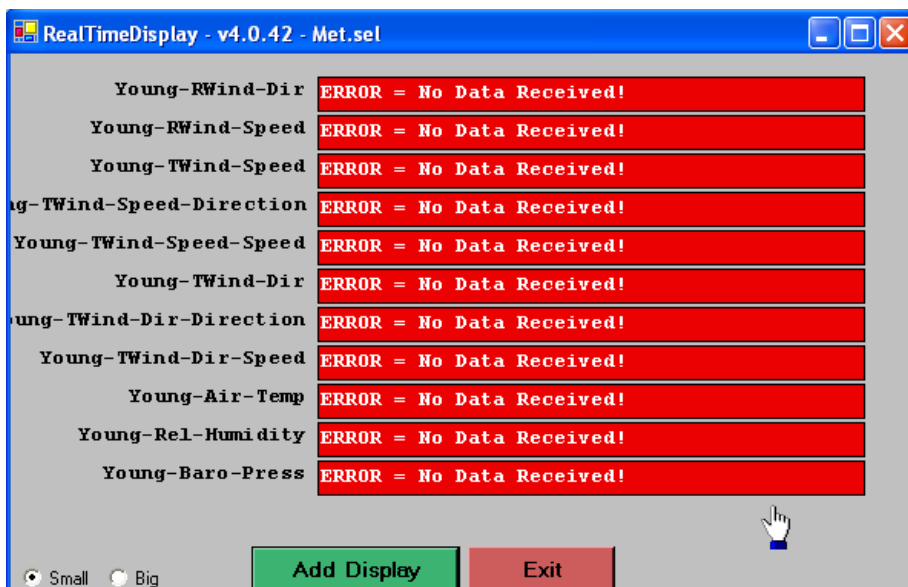
6.2.2 Small / Big Buttons

The user may toggle the font from normal to large font for readability. The font is not selectable. When toggling the font the user may have to adjust the window to make all data visible. Here is a sample display in the big font.



Error Checking

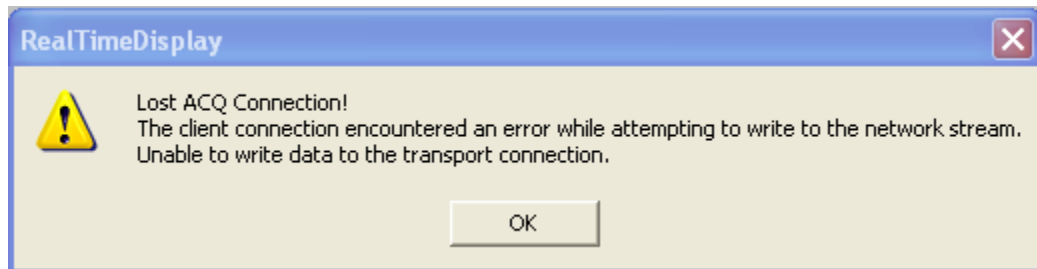
When the app starts up all text boxes will have red background indicating no data has been received.



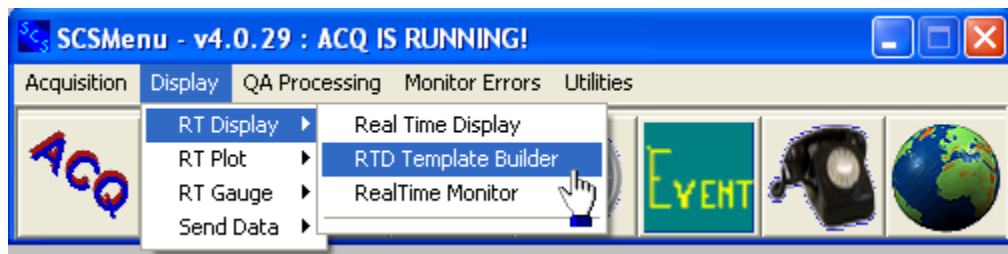
When the first message is received the background will change to gray. After that, the app monitors the data being received from ACQ. If it detects a data field or message is more than ten minutes old the values field will background will turn to yellow, and the sensor value is replaced by a message indicating how long the outage has been going on.

NOTE: If data does not appear in the window, ensure that the sensor is functioning.

Also, if there is no data coming into that specific data display, and error message will be displayed informing the user that the connection to ACQ has been lost.

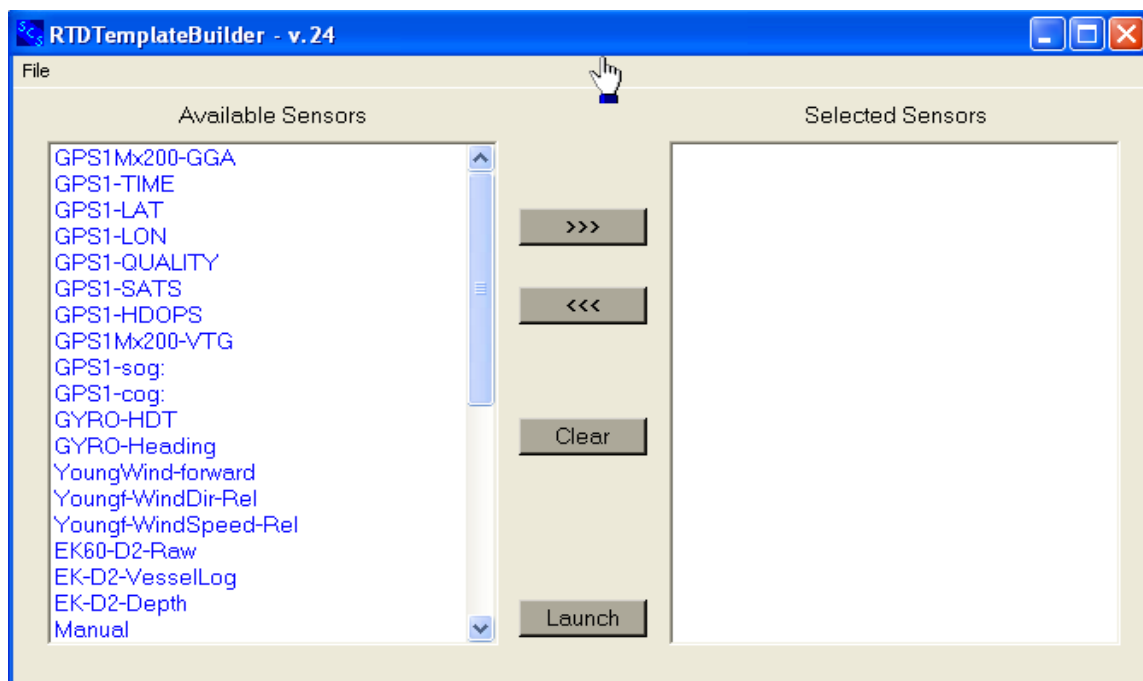


6.3 Real Time Display Template Builder



Function

This program Builds a real time display template RTD file (*.SEL) for use with the Real Time Display program. The file created will contain the sensor names of the data the user has selected to be displayed. This program creates a .SEL file in the ShipName\Templates\Display directory. Here is a sample main window.



Main Menu Bar

The main window has one Menu, “File”. Under that are several selections, as follows:

6.3.1 New

This clears out all of the sensors within the Selected Sensors list and allows the user to either create a new RTD template file.

6.3.2 Open

This displays a list of all the previously saved RTD files (.SEL) within a file selection box. All of these files are located in the ShipName\Template\Display directory and have the extension of SEL.

6.3.3 Save

This allows the user to save any changes that have been made in the RTD file. The user will be prompted to save - if changes have been made - upon exiting the program or when opening or creating another RTD file.

6.3.4 Save As

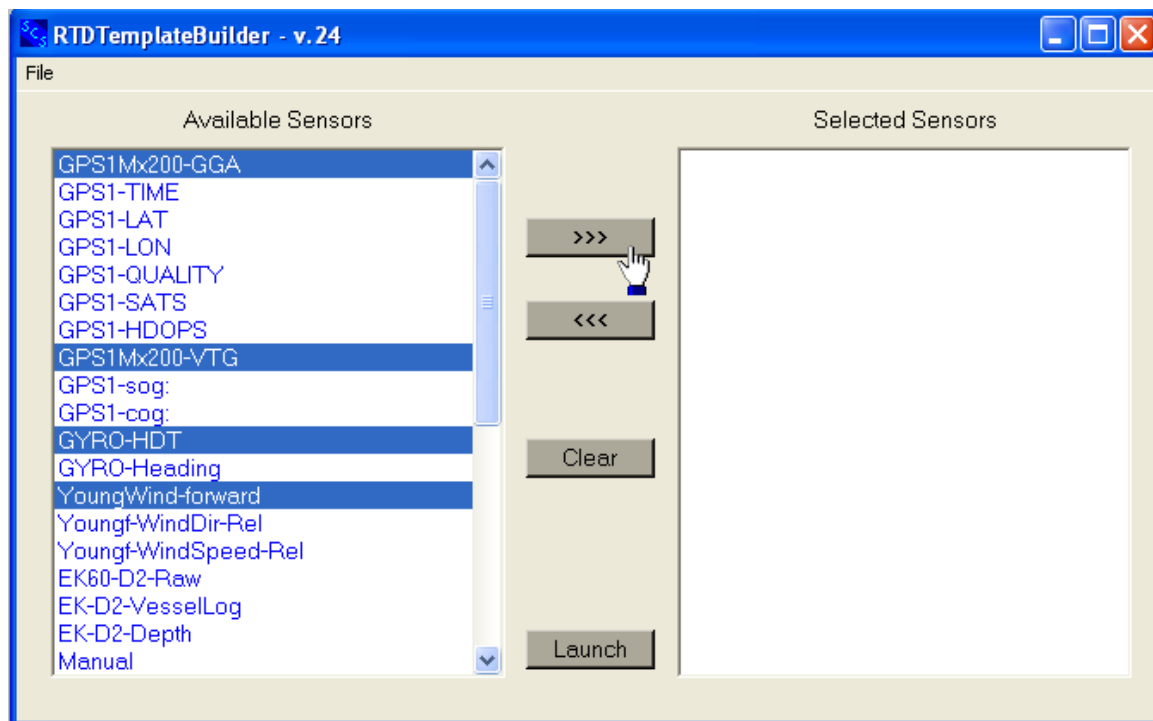
This allows the user to save any changes that have been made in the RTD file to another RTD file. A box will appear where the user is prompted to type in the name of the new RTD file this RTD. The RTD file will be saved as the new file name to the SHIP \Template\Display directory.

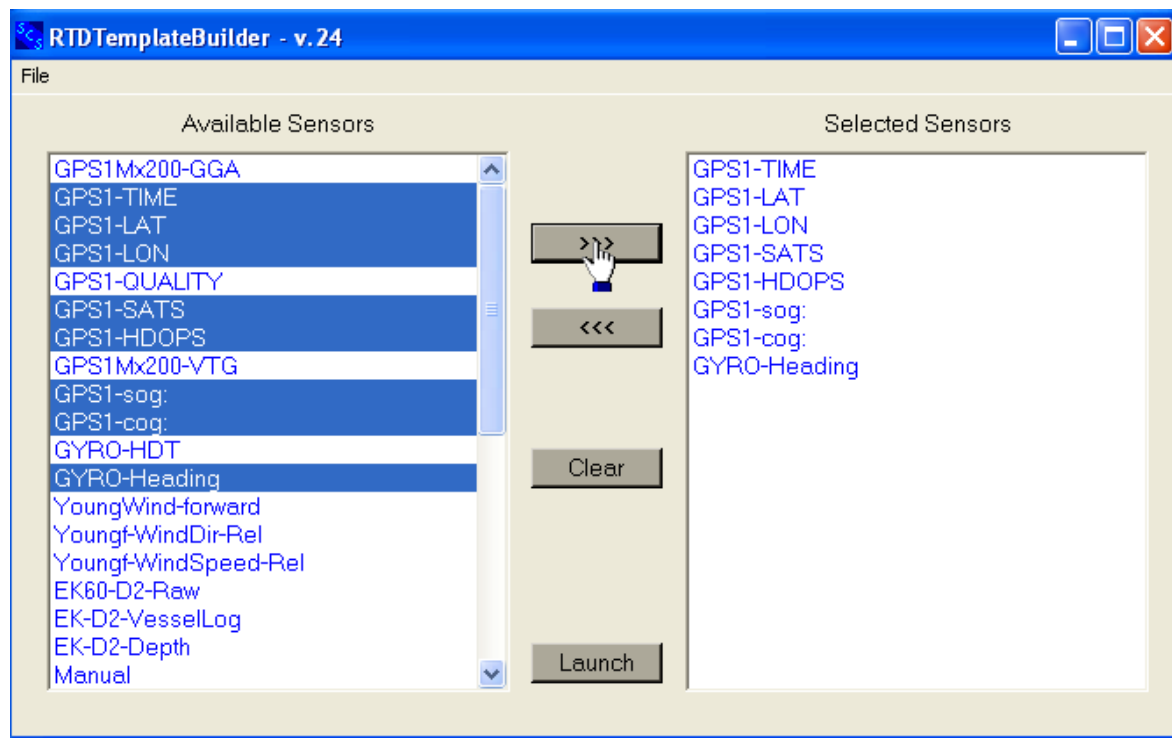
6.3.5 Exit

This exits the user out of the RTD Template Builder program. If changes were made to the current RTD file, the user will be prompted to save the file before exiting.

Available Sensors List

Displays which sensors are currently available to include in the selected RTD file, but are not yet included. If sensors need to be added, select the desired sensor(s) from this list and press the *Right Arrow Button*. The selected sensors will then be placed on the Selected Sensor list.





Selected Sensors List

Displays which sensors are currently included in the selected RTD file. If sensors need to be removed, select the desired sensor(s) from this list and press the *Left Arrow Button*. The selected sensor(s) will then be placed on the Available Sensor list.

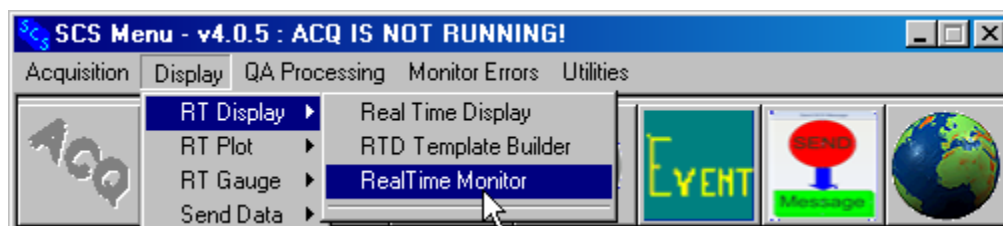
Clear Button

Removes all sensors from the Selected Sensors List and puts them back in the Available Sensors List.

Launch Button

This launches the Real Time Display directly from the builder as a convenience. This is the same as launching the Real Time Display from the SCS Menu Bar.

6.4 REAL TIME MONITOR



Introduction

The Real Time Monitor app generates a text display similar in nature to the Real Time Display in the previous section. It can be launched from the Display menu in the SCS Menu app, or from the main ACQ window (see chapter 3).

A default RTM display appears below. It is customizable but in different ways than the RTD, mainly in that template files cannot be created and saved.

You launch the RTM display quickly from the ACQ main window.

Real Time Monitor Display Menu

The selections from the Display menu are:

6.4.1 Sensors

You can choose to show only parent sensors, only child sensors, or both together.

6.4.2 Columns

You can select what kind of data to display about each sensor:

- Sensor data, entire message for parents, value for children.
- ACQ message timestamp as it appears in the RAW file.
- Sensor units from the sensor configuration file
- Error Status (red LED: no data ever received, green LED: at least one message received).
- Device, from the sensor configuration file.

6.4.3 Sorting

You can sort sensor names alphabetically, forward or reverse. Select “no sort” to display the sensors in the order they appear in the sensor configuration file.

6.4.4 Display Sensor Details.exe

Display Sensor Details application usually is launched from the SCSMenu “Utilities” menu. This gives a summary listing of the sensor’s parameter values. See Chapter Eight - Utilities for more detail.

6.4.5 Freeze Display

Select this to freeze the display temporarily. The display remains frozen until you select it again.

6.4.6 Status LEDs

In the main display window below, a red LED indicates that no data has ever been received from the sensor since the application started. A green LED indicates that at some since the application started at least one message has been received, but there is no indication of how long ago the last message arrived.

[illegible]

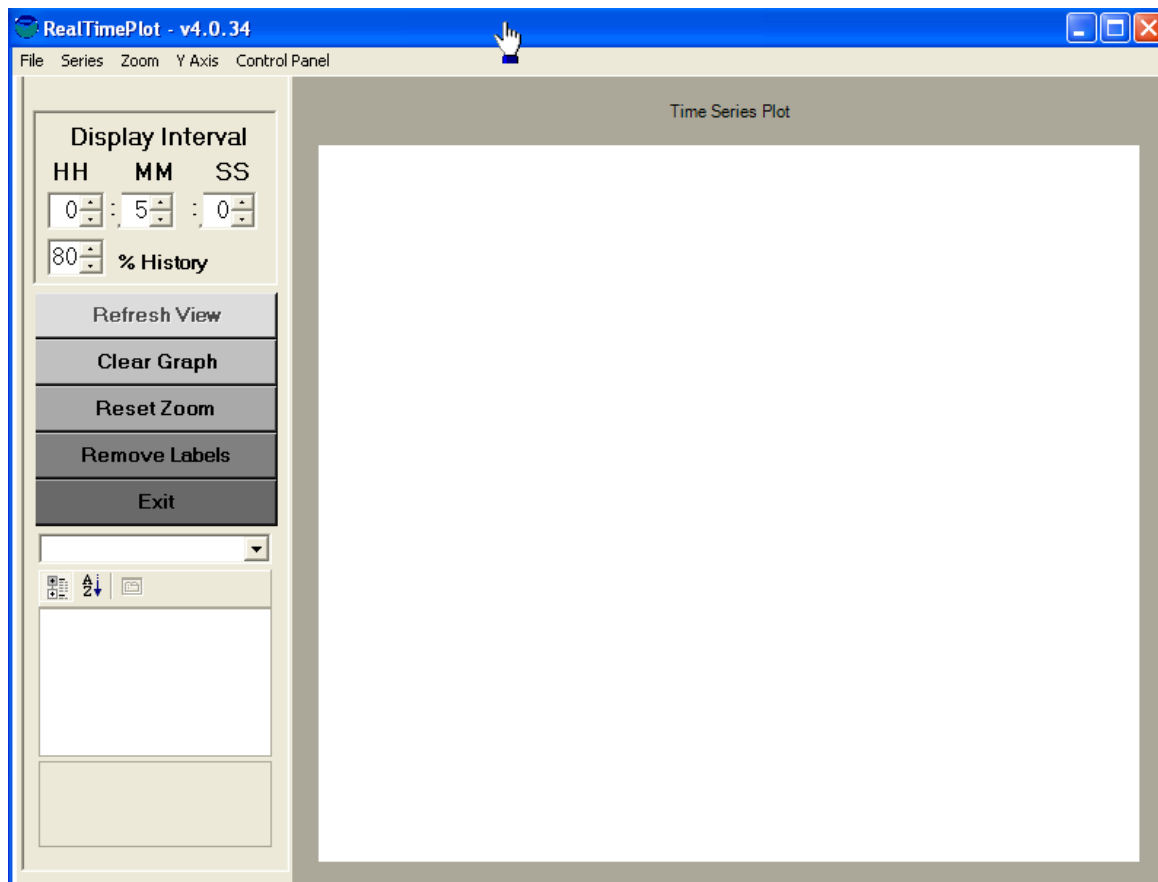
6.5 Real Time Time Series Plot



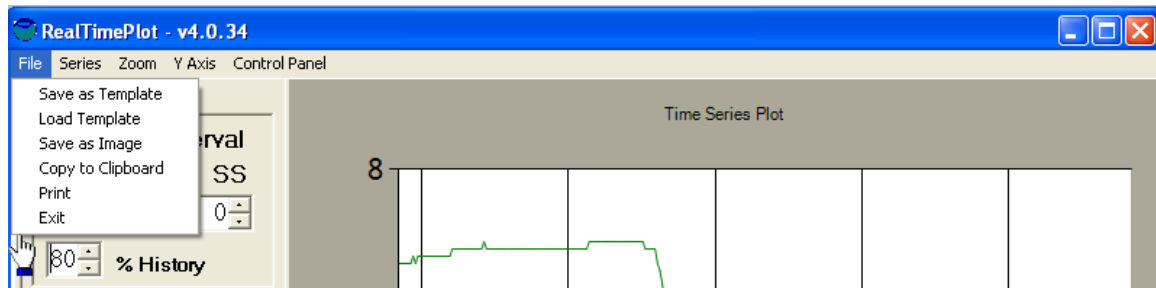
Function

This program displays as many as four real time sensor data time series in graphical format. These graphs may be easily manipulated by the user through such functions as zooming, scrolling, setting graph attributes and getting values of specific data points. The plot program automatically adjusts the amount of data displayed and the update rate of the plot based on the amount of data in the plot. The program provides an option for a dual axis plot where each time series may be associated with either axis.

When the program is launched a blank main window appears. It shows a Main Menu across the top of the window, a control panel on the left, and a blank drawing area in the body of the window. Here is a sample:



The File Menu

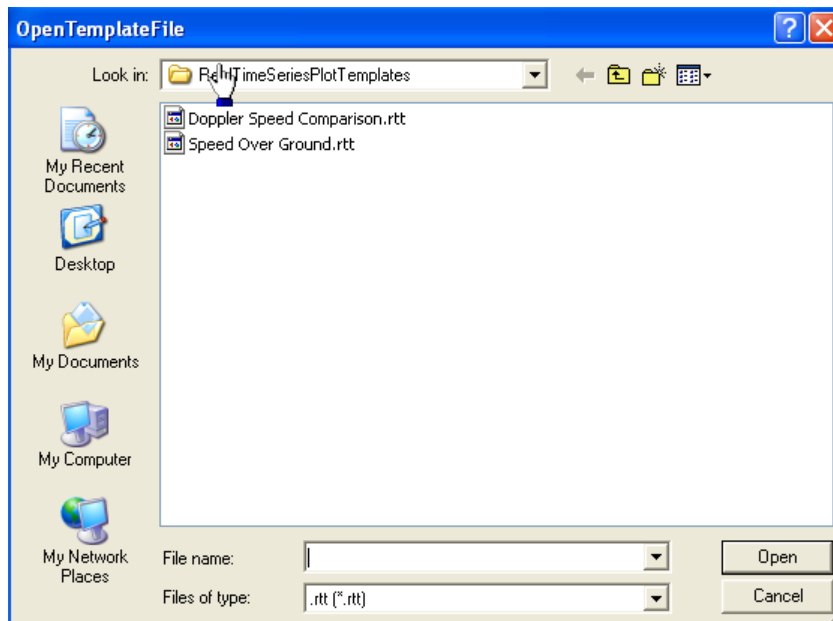


6.5.1 Save As Template

This allows you to save the current graph settings to a template file.. Later, you can recreate the windows of commonly used graphs without having to reconfigure by hand. A box pops up, prompting the user to specify a file name for the current settings (enter the file name without an extension or directory name). All files are placed into the subdirectory ShipName\Template\RealTimeSeriesPlotTemplates under the SHIP logical directory. Note that all configurable settings will be saved, including sensor names, line types and colors, screen colors, update rates, etc...

6.5.2 Load Template

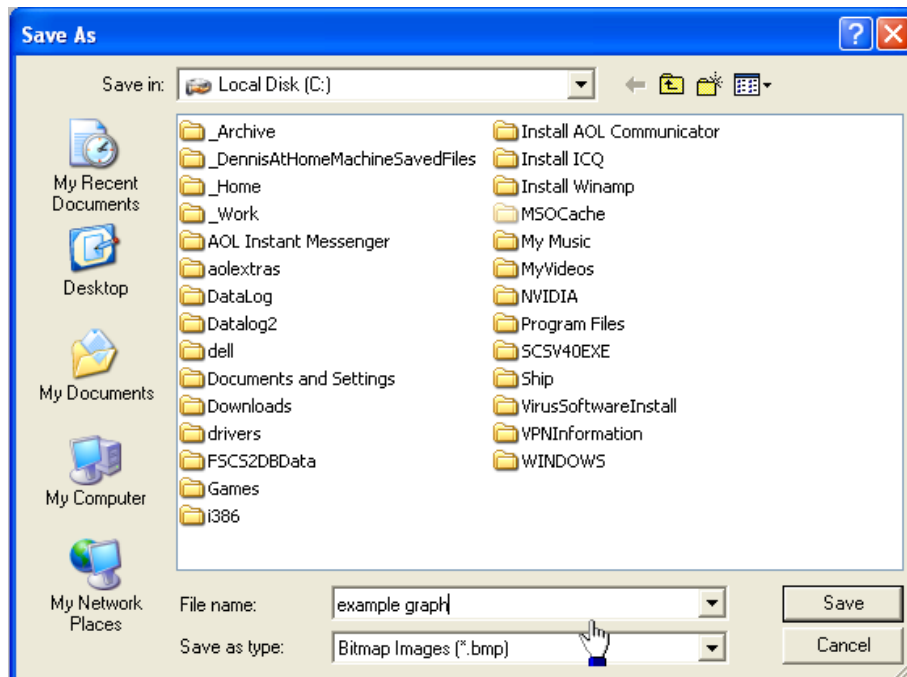
This allows the user to recall any previously saved graph settings. A list of all saved template files will be given (SHIP/Template/RealTimeSeriesPlotTemplates under the SHIP logical directory).



Select the desired settings file and hit the OPEN button. The graph will then start plotting the sensors that were specified in the settings file.

6.5.3 Save As Image

Use this to copy the present graph to a file as a standard bitmap graphics image.



6.5.4 Copy to Clipboard

Use this to copy the present graph to the clipboard for inclusion in other standard windows applications like Word and PowerPoint.

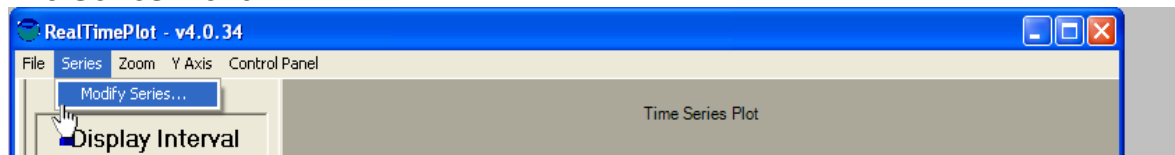
6.5.5 Print

Located under the 'FILE' menu, this allows the user to print the graph. The standard Windows print and properties dialog is displayed. Note that the default orientation is Portrait.

6.5.6 Exit

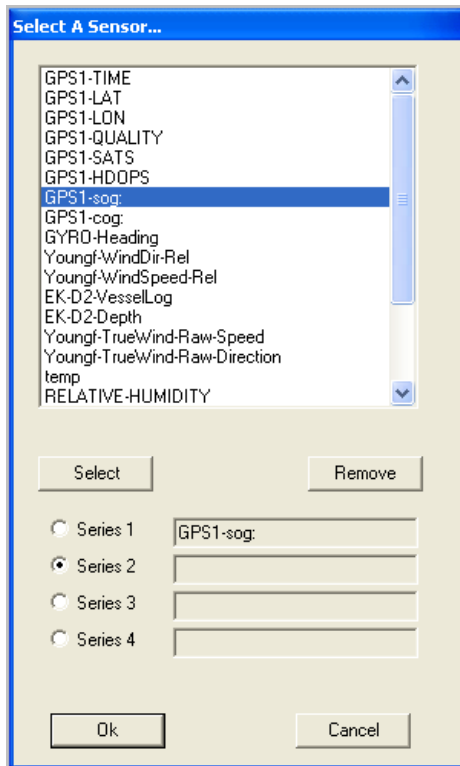
Close the RT Time Series Plotting program.

The *Series* Menu



6.5.7 Modify Series

This brings up the following dialog. You then select the sensors to plot. You also use this to change or add a sensor time series to the plot. As many as four sensors can be displayed.



The Dialog presents a list of child sensors to plot. Select a sensor and click on the SELECT button, or simply double-click the sensor. The radio buttons control which time series a sensor is plotted on, but in the end this has no practical consequence. The series radio button is automatically advanced for convenience in case multiple series are being configured.

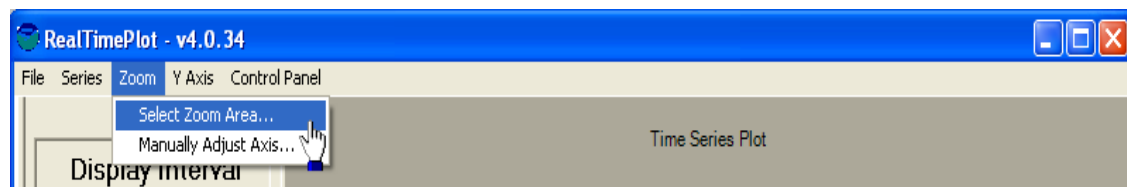
The REMOVE button is used to remove a sensor from a series.

Once you are satisfied with the configuration press the OK button to start the plot.

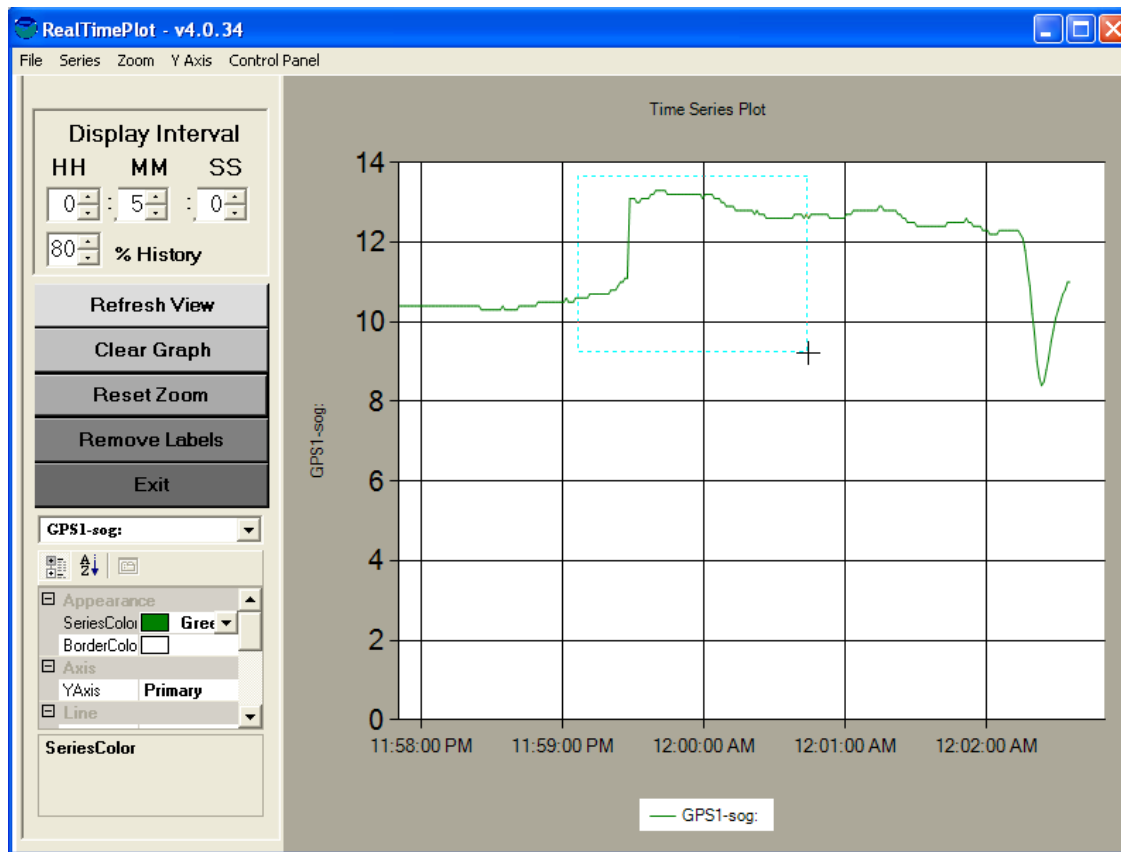
The Zoom Menu

Once the plot has been developed for some period of time you may be interested in zooming the plot to examine specific features.

6.5.8 Select Zoom Area...

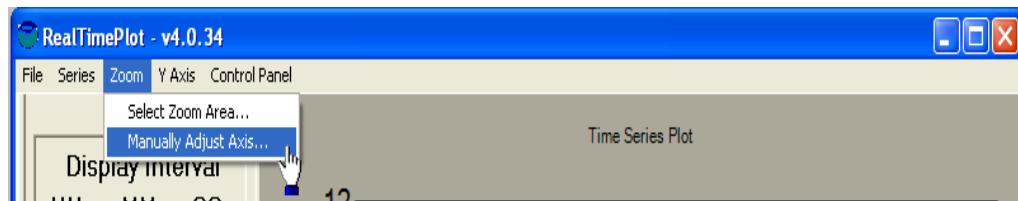


Click on Select Zoom Area, then select the zoom area on the plot by holding the left mouse button down and dragging a box over it, as in the sample screen here:

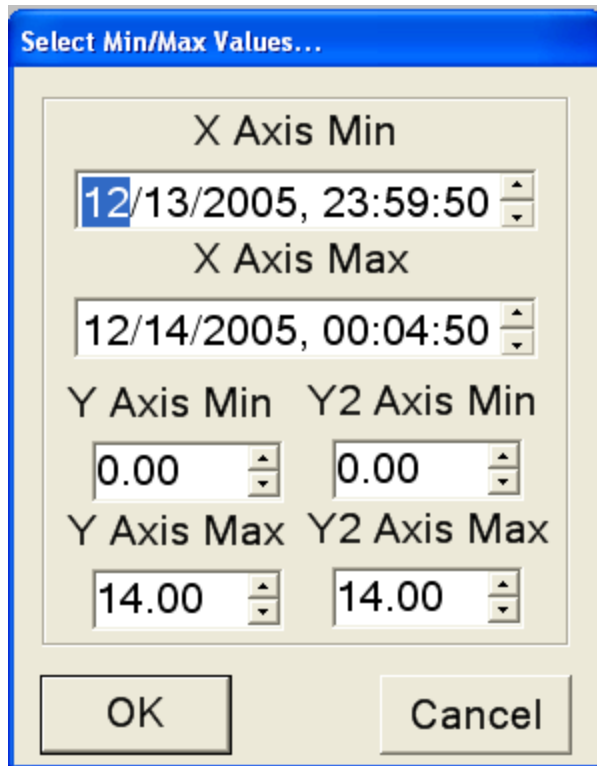


To restore the plot to its full extent use the RESET ZOOM button on the control panel described below.

6.5.9 Manually Adjust Axis...



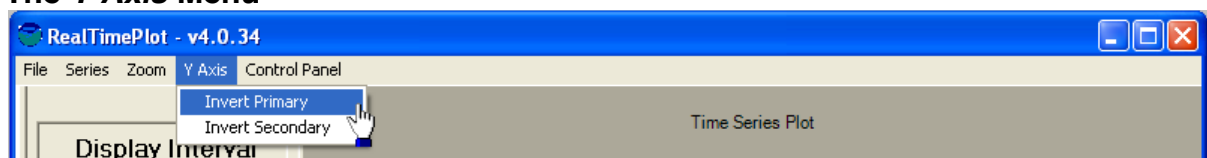
The plot may also be zoomed or sized manually by adjusting the Axis to the time period of interest.



6.5.10 Zooming Back Out

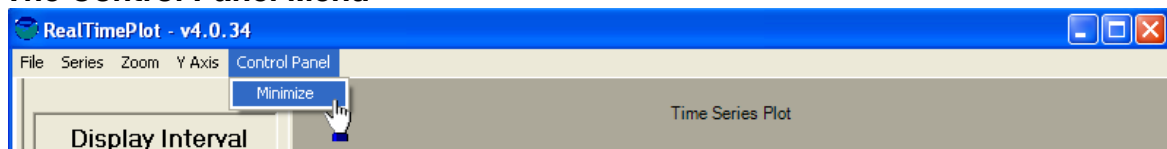
To restore the plot to its full extent use the RESET ZOOM button on the control panel (See Plot Control Panel Discussion for details).

The Y Axis Menu



Some plots (for example, of depth data) are best viewed with an inverted Y axis . This menu is used to toggle the inversion of the primary or secondary Y axis.

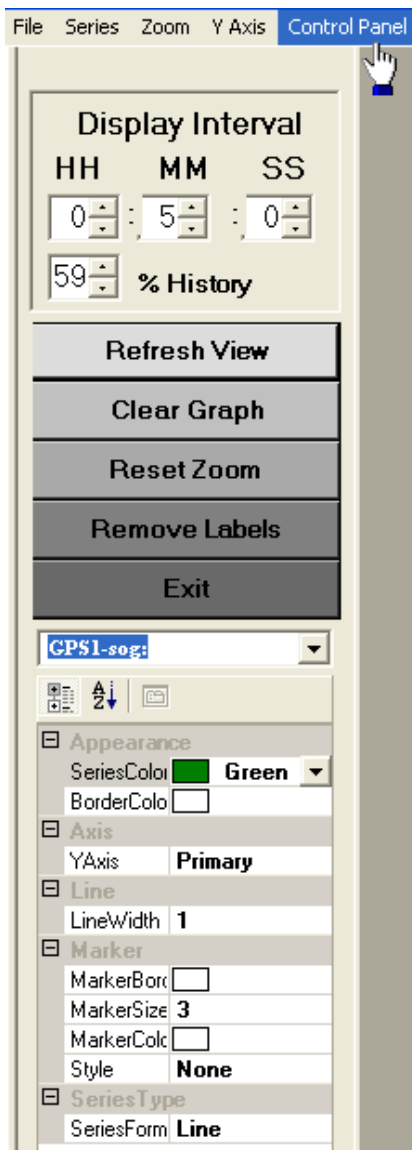
The Control Panel Menu



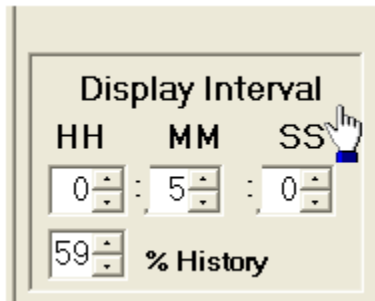
All SCS graphs include a common control panel which is used to modify colors or line thickness. This control panel may be minimized to maximize the plotting area, or re-activated by the control panel menu bar as show above. The Graph Control Panel is described in the following sections

Graph Control Panel

The Graph Control Panel is normally located to the left off the plotting area on all SCS plots. This interface used to modify properties of the plot in both data and appearance. Here is a picture of the control panel:



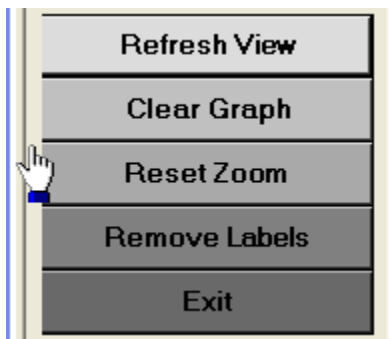
6.5.11 Graph Control Panel Display Interval



The display interval is used to define the length of the time covered by the X axis. The default window is 5 minutes but this may be expanded to incorporate 24 hours or more of data. The user simply types the values for each field or uses the up/down arrows to adjust. The % HISTORY is used to adjust the amount of blank space that is left on the right side of the plot after scrolling. This is the open area of the plot that will be graphed in real time. Normally this value is 80% implying that 20% of the graph is active on the right to accept new points.

NOTE: You must click on the Refresh View button to apply any changes you make.

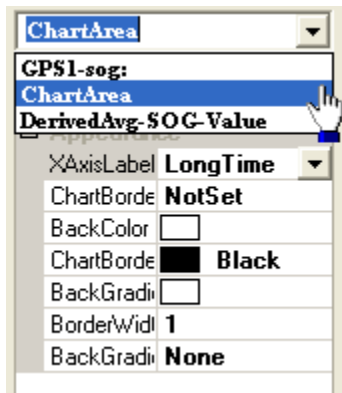
6.5.12 Graph Control Panel Buttons



These buttons perform direct actions on the plot and applications.

- **REFRESH VIEW:** Used to repaint the plot after the user has modified properties or the time axis. If the user does not refresh view the properties will not be applied until the next scroll action occurs.
- **CLEAR GRAPH:** This button allows the user to clear the graph and all internally stored data points.
- **RESET ZOOM:** This button is used to reset the plot to full extent after a set of zooming actions.
- **REMOVE LABELS:** This button is used to remove get data point labels from the graph. The user may inquire about any point by right clicking on the time series feature of interest.
- **EXIT:** This button is used to exit the application.

6.5.13 Graph Control Panel Chart Area Properties

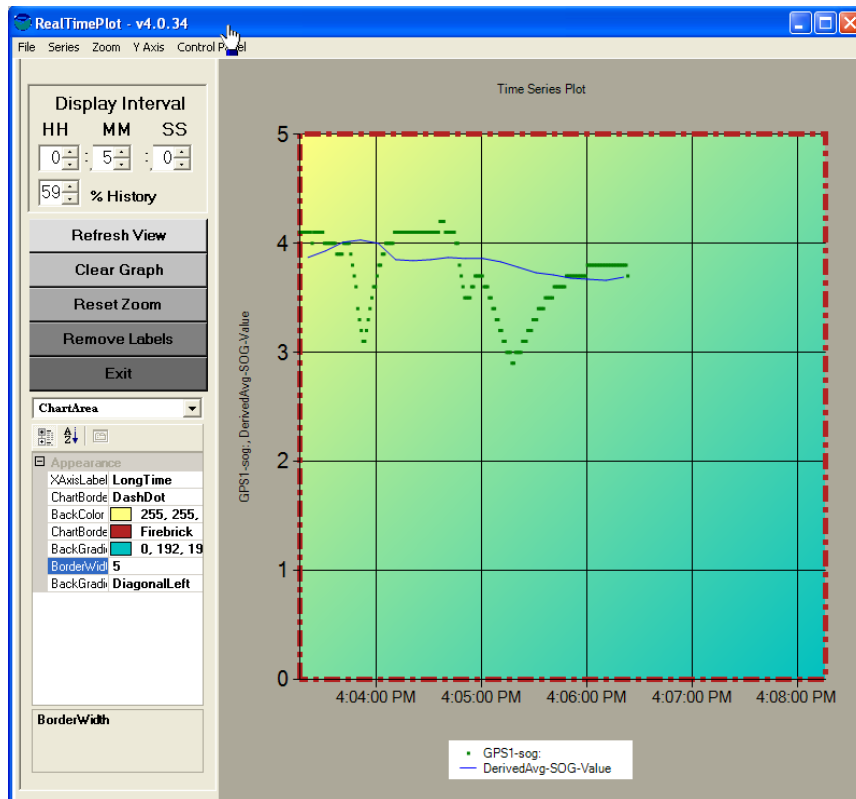


The pull down list in the control panel is your means of selecting a component of the graph to which you want to make changes. All-Time series are in the list plus an entry called Chart Area.

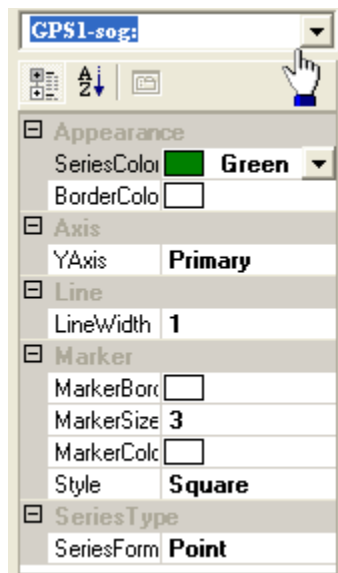
The following Properties may be adjusted by the user by selecting Chart area in the pull down selection box.

- X AXISLABEL: set the time format of the X time axis
- CHARTBORDER: toggle if the plot will have a border
- BACKCOLOR: Set the background color of the plot
- CHART BORDER COLOR: Set the border background color
- BACKGROUND GRADIENT ENABLED: Toggles if the background color is a gradient
- BORDER WIDTH: Sets the pixel width of the border
- BACKGROUND GRADIENT: Sets the color scheme of the gradient.

A sample plot follows that shows some of these features.



6.5.14 Graph Control Panel Sensor Time Series Properties



The user selects the name of the SCS sensor time series to modify and then is presented a set of properties to change. These properties are broken down into the following categories:

- APPEARANCE
 - ✓ SERIES COLOR: modify the color of the time series
 - ✓ BORDER COLOR: modify the background color
- AXIS
 - ✓ Y AXIS: set the time series to be part of the primary or secondary Y axis. The chart program is a dual axis program and a particular time series may be assigned to either series.
- LINE WIDTH
 - ✓ WIDTH: The pixel width of the line
- MARKER
 - ✓ MARKER BORDER: If a point has a border if plotting in points
 - ✓ MARKER SIZE: the size of a point on the plot, if plotting in points
 - ✓ MARKER COLOR: the color of the point, if plotting in points
- STYLE: the Style of the point, if plotting in points
- SERIES TYPE
 - ✓ SERIES FORMAT: a selection of points or line style for the time series.

Graph Behavior

6.5.15 Scroll Bar Behavior

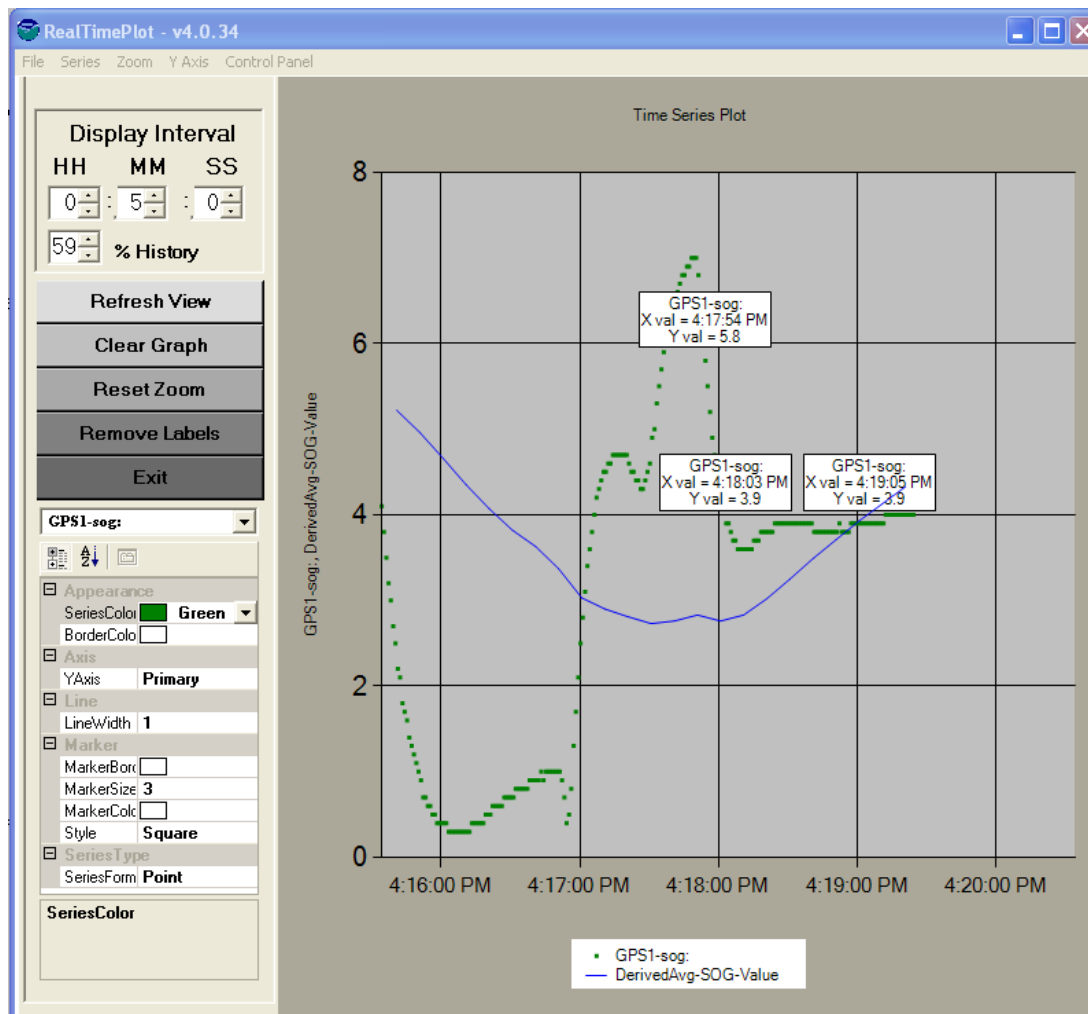
The graph will scroll to the left each time the plot reaches the right margin. The graph will be re-plotted to maintain the display window time interval (See Plot Control Panel Discussion for details) set by the user and will remove data points from the plot as required. All of the data collected in the plot is still available to be viewed based on the actions of the user to manipulate the display.

6.5.16 Data Point Filter

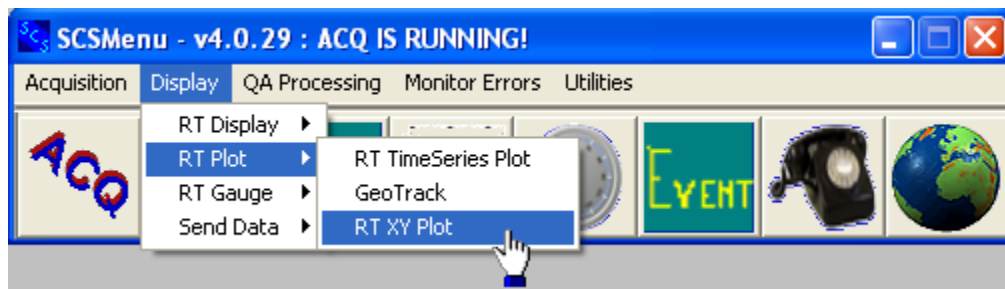
The graph will plot all data points in the current graph that will be visible to the user. This ensures that the plot reflects the data collected while maintaining efficiency by not over plotting duplicate pixels.

6.5.17 Show a Data Point

The user may right click over any data point or time series feature and the plot will display the value of the closest data point, as shown below.



6.6 Real Time XY Plot



Function

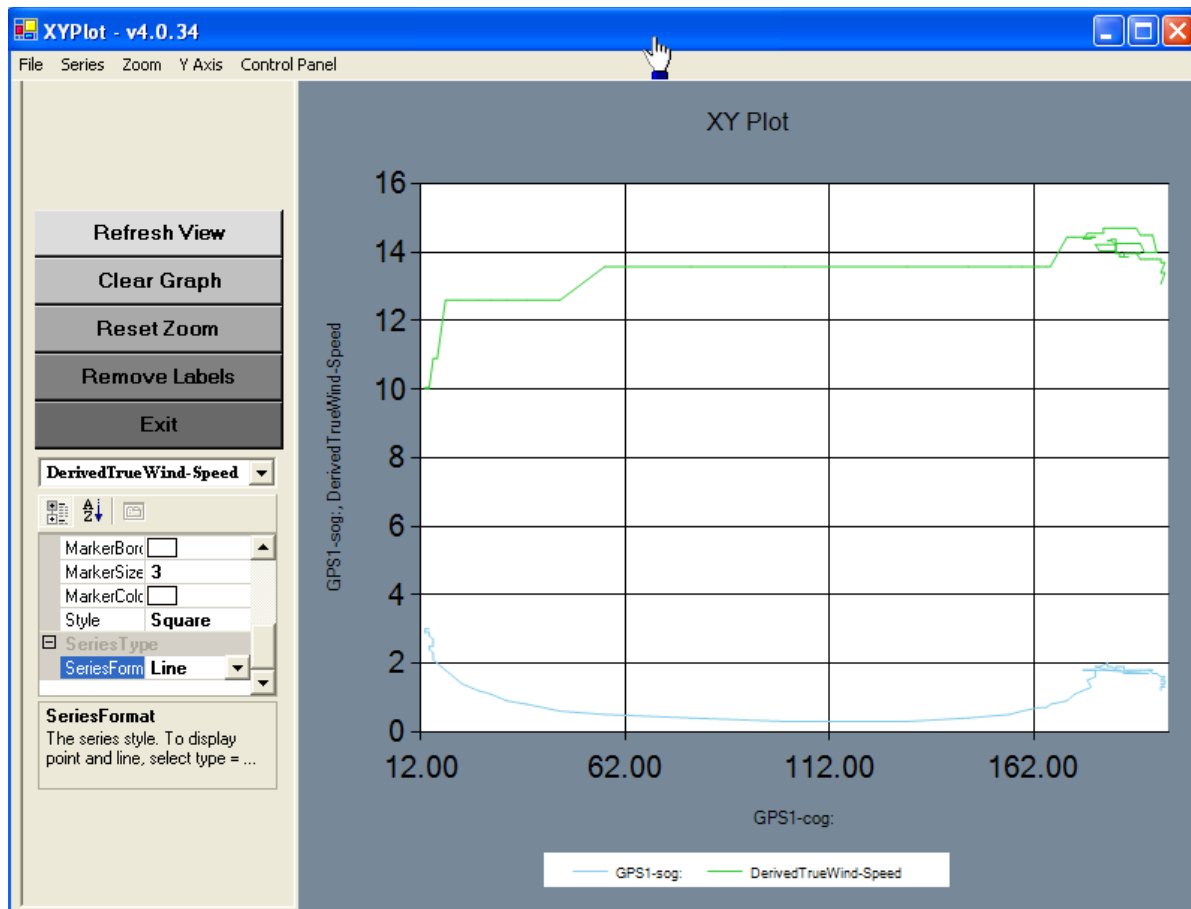
An XY plot shows a control sensor on the X axis and one or more dependent sensors on the Y axis. An example would be conductivity and temperature as a function of depth.

The user interface for this program is nearly identical as the time series program with the exception that you must distinguish between the sensor for the X axis and the four Y axis sensors.

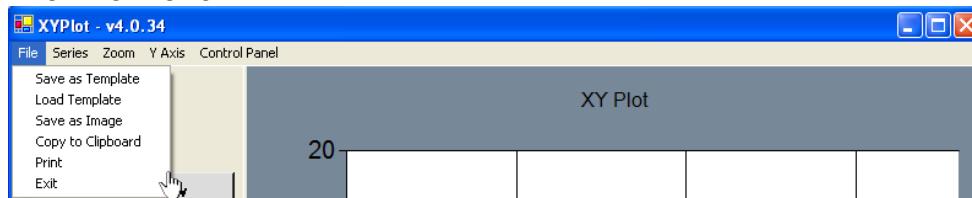
The plot itself is typically best rendered as a point plot since the data may not always move from the left to right. However in applications where sensors are tracked relative to depth or altitude line plotting is appropriate.

The Main Window

When the program is activated it shows a functions menu across the top of the window, a control panel on the left, and a blank drawing area in the body of the window. Here is a sample main window.



The File Menu

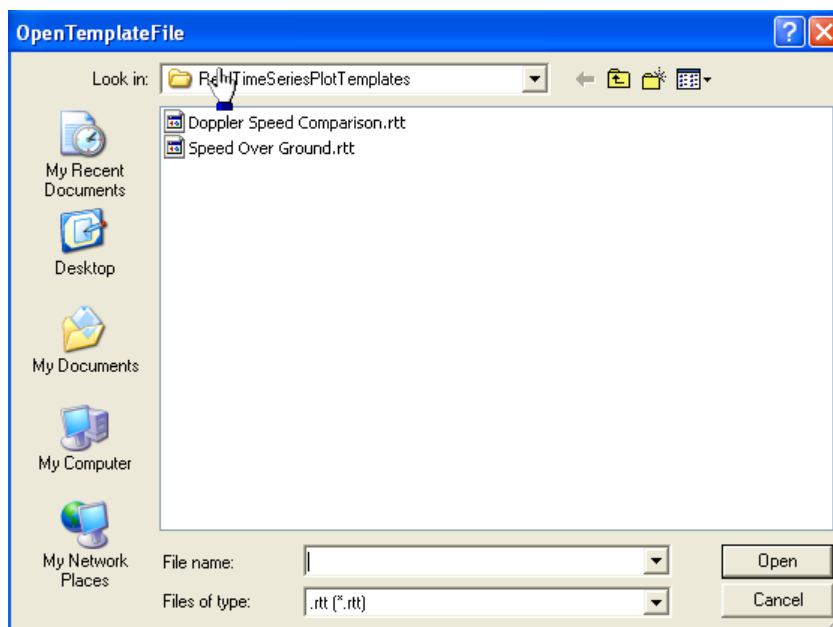


6.6.1 Save As Template

This allows you to save the current graph settings to a template file.. Later, you can recreate the windows of commonly used graphs without having to reconfigure by hand. A box pops up, prompting the user to specify a file name for the current settings (enter the file name without an extension or directory name). All files are placed into the subdirectory ShipName\Template\ XYPlotTemplates under the SHIP logical directory. Note that all configurable settings will be saved, including sensor names, line types and colors, screen colors, update rates, etc...

6.6.2 Load Template

This allows the user to recall any previously saved graph settings. A list of all saved template files will be given (SHIP/TemplateXYPlotTemplates under the SHIP logical directory).



Select the desired settings file and hit the OPEN button. The graph will then start plotting the sensors that were specified in the settings file.

6.6.3 Save As Image

Use this to copy the present graph to a file as a standard bitmap graphics image.

6.6.4 Copy to Clipboard

Use this to copy the present graph to the clipboard for inclusion in other standard windows applications like Word and PowerPoint.

6.6.5 Print

Located under the 'FILE' menu, this allows the user to print the graph. The standard Windows print and properties dialog is displayed. Note that the default orientation is Portrait.

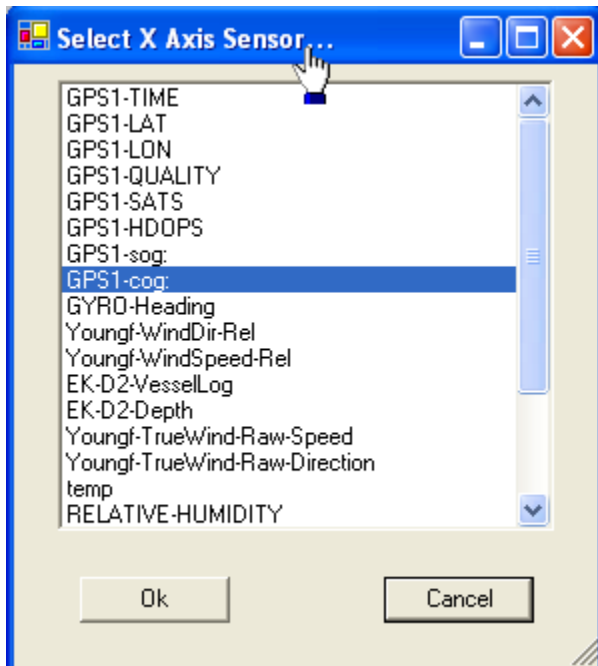
6.6.6 Exit

Closes the RT Time Series Plot program.

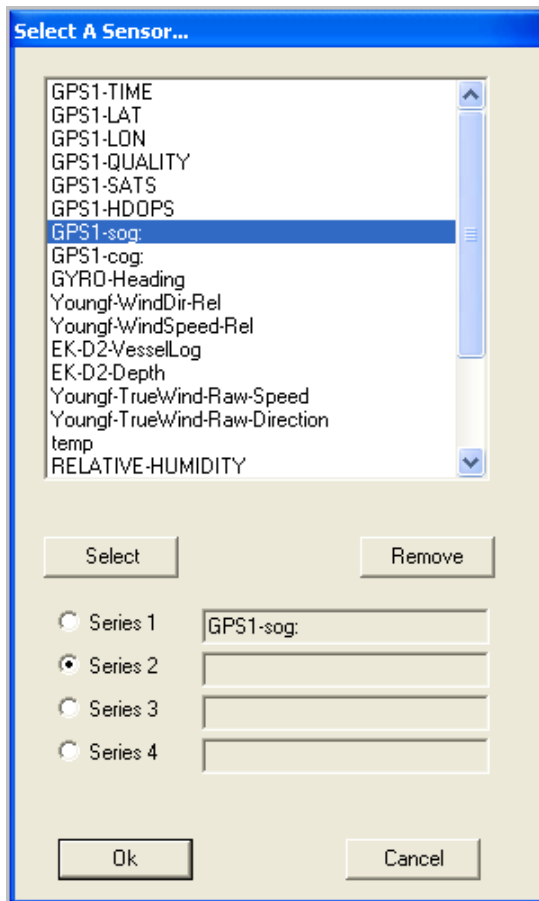
The *Series* Menu



Select the sensor to plot as the X axis using the Select X Axis Sensor. This menu is used exclusively for adding or changing the X Axis sensor.



Use Modify Series... to select the sensors to plot on the Y axis versus the X axis using the Series – Modify Series menu. This menu is also used for changing a sensor to the plot on the Y axis.

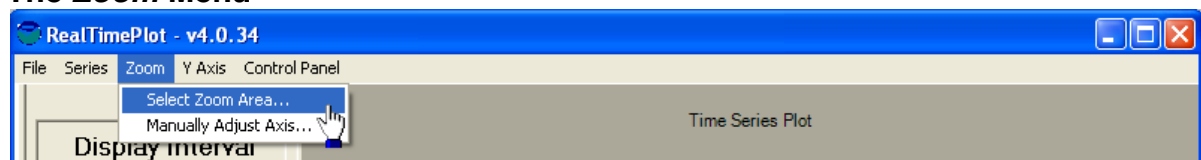


The user is presented with a list of child sensors to plot. The user selects a sensor and hits the SELECT button or simply double-clicks the sensor. The user may also select which time series the user wants to use for plotting the sensor. The series radio button is automatically advanced for convenience in case multiple series are being configured.

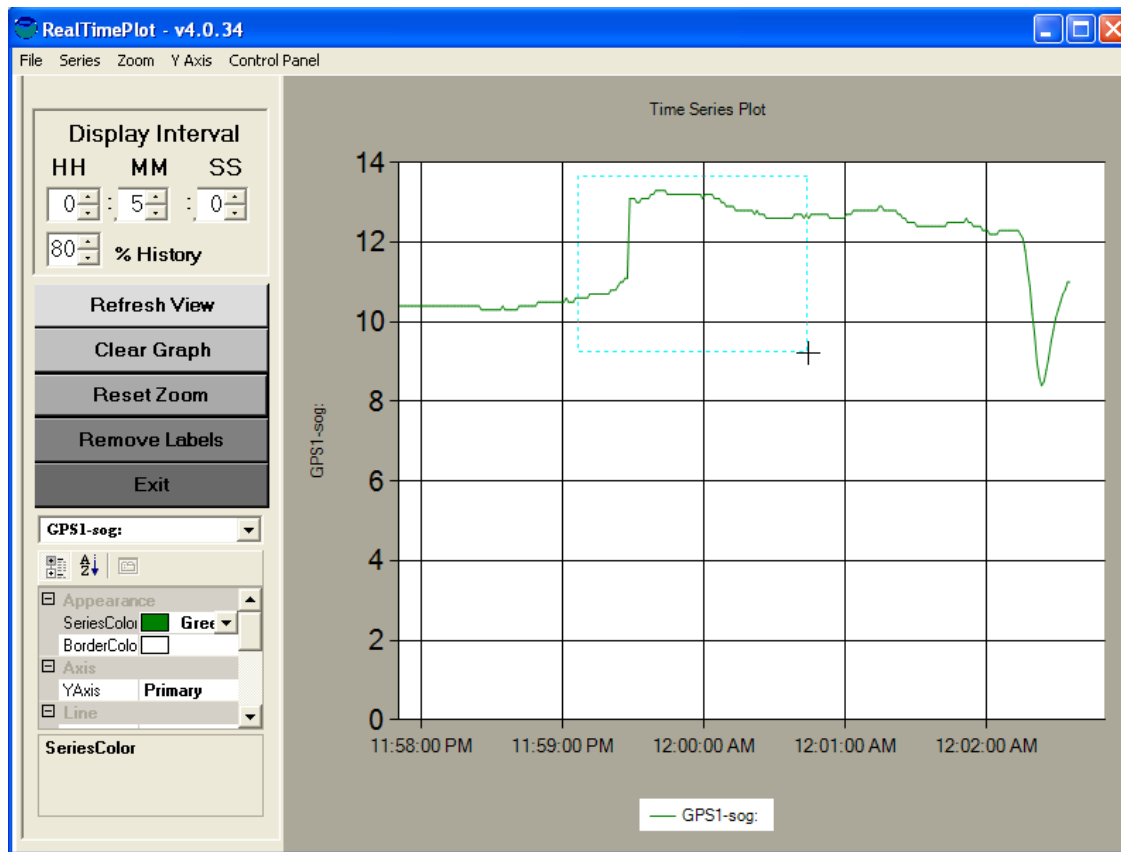
The REMOVE button is used to remove a sensor from a series.

Once the user is satisfied with the time configuration he should press the OK button to start the plot.

The Zoom Menu



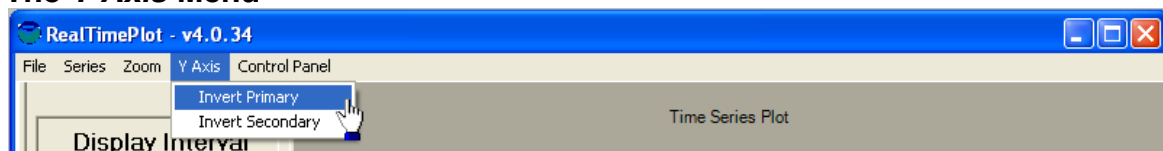
Once the plot has been developed for some period of time the user may be interested in zooming the plot to examine specific features. The user may zoom in on the plot by either drawing a zoom area on the plot by holding the left mouse button down or dragging a box over the interested feature.



To restore the plot to its full extent use the RESET ZOOM button on the control panel described below.

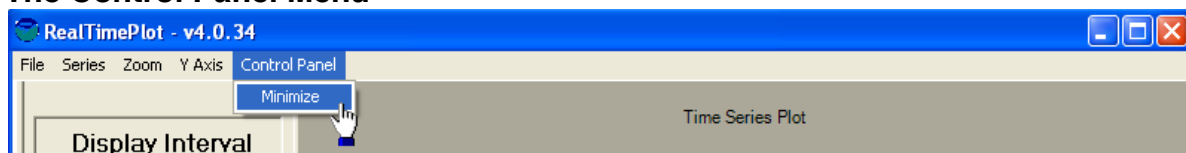
To restore the plot to its full extent use the RESET ZOOM button on the control panel (See Plot Control Panel Discussion for details).

The Y Axis Menu



Some plots are best viewed with an inverted Y axis such as plotting sensors versus depth or altitude. This menu is used to toggle the inversion of the primary or secondary Y axis.

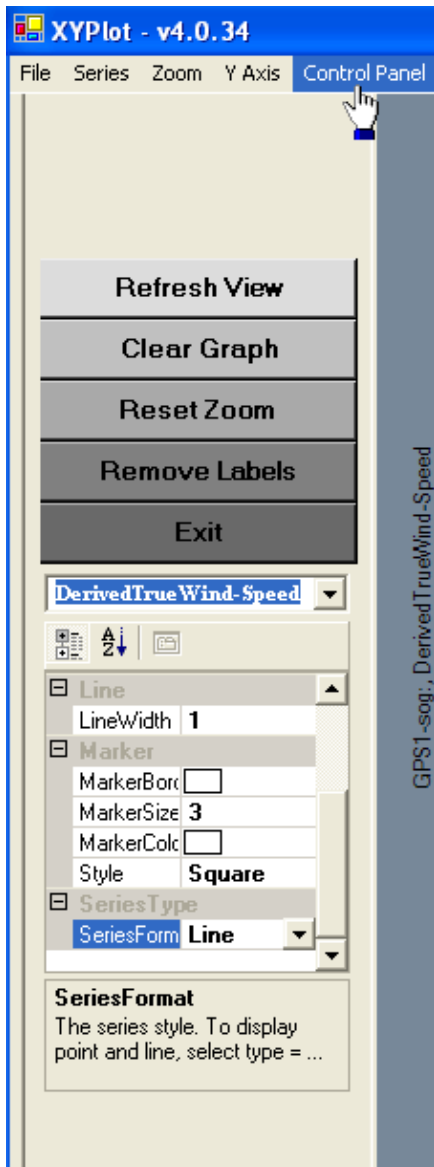
The Control Panel Menu



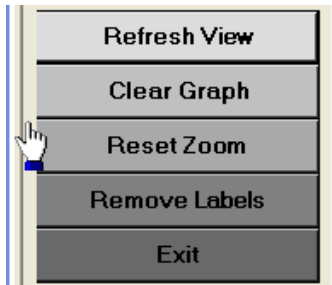
The standard view of all SCS graphs includes a control panel for easily modifying the view of the plots. This control panel may be minimized or activated by the control panel menu bar as show above. The Graph Control Panel is described in the following section.

The Graph Control Panel

The Graph Control Panel is normally located to the left off the plotting area on all SCS plots. This is the user interface panel used to modify properties of the plot in both data and appearance.



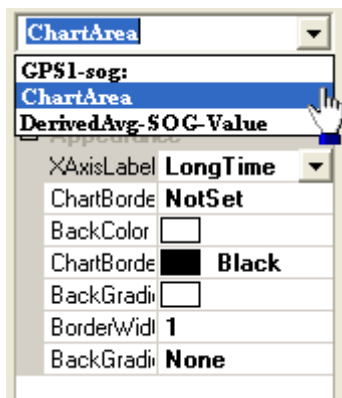
6.6.7 Graph Control Panel Buttons



These buttons perform direct actions on the plot and applications.

- **REFRESH VIEW:** Used to repaint the plot after the user has modified properties or the time axis. If the user does not refresh view the properties will not be applied until the next scroll action occurs.
- **CLEAR GRAPH:** This button allows the user to clear the graph and all internally stored data points.
- **RESET ZOOM:** This button is used to reset the plot to full extent after a set of zooming actions.
- **REMOVE LABELS:** This button is used to remove get data point labels from the graph. The user may inquire about any point by right clicking on the time series feature of interest.
- **EXIT:** This button is used to exit the application.

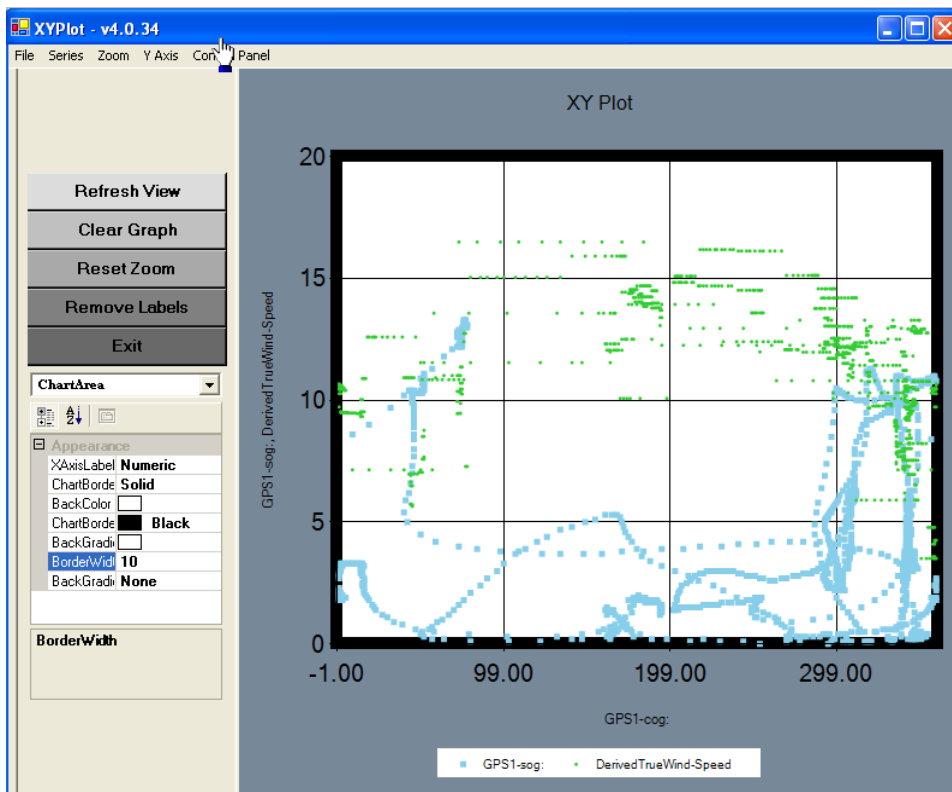
6.6.8 Graph Control Panel Chart Area Properties



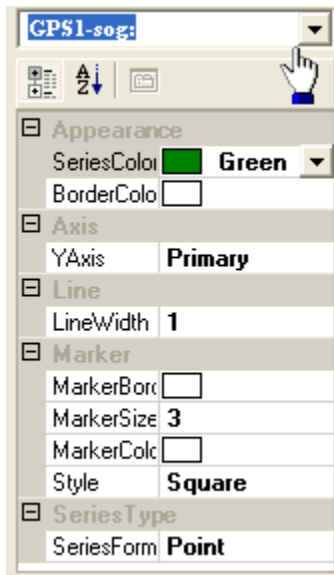
The following Properties may be adjusted by the user by selecting Chart area in the pull down selection box.

- X AXISLABEL: set the time format of the X time axis
- CHARTBORDER: toggle if the plot will have a border
- BACKCOLOR: Set the background color of the plot
- CHART BORDER COLOR: Set the border background color
- BACKGROUND GRADIENT ENABLED: Toggles if the background color is a gradient
- BORDER WIDTH: Sets the pixel width of the border
- BACKGROUND GRADIENT: Sets the color scheme of the gradient.

An Example plot follows:



6.6.9 Graph Control Panel Sensor Time Series Properties



The user selects the name of the SCS sensor time series to modify and then is presented a set of properties to change. These properties are broken down into the following categories:

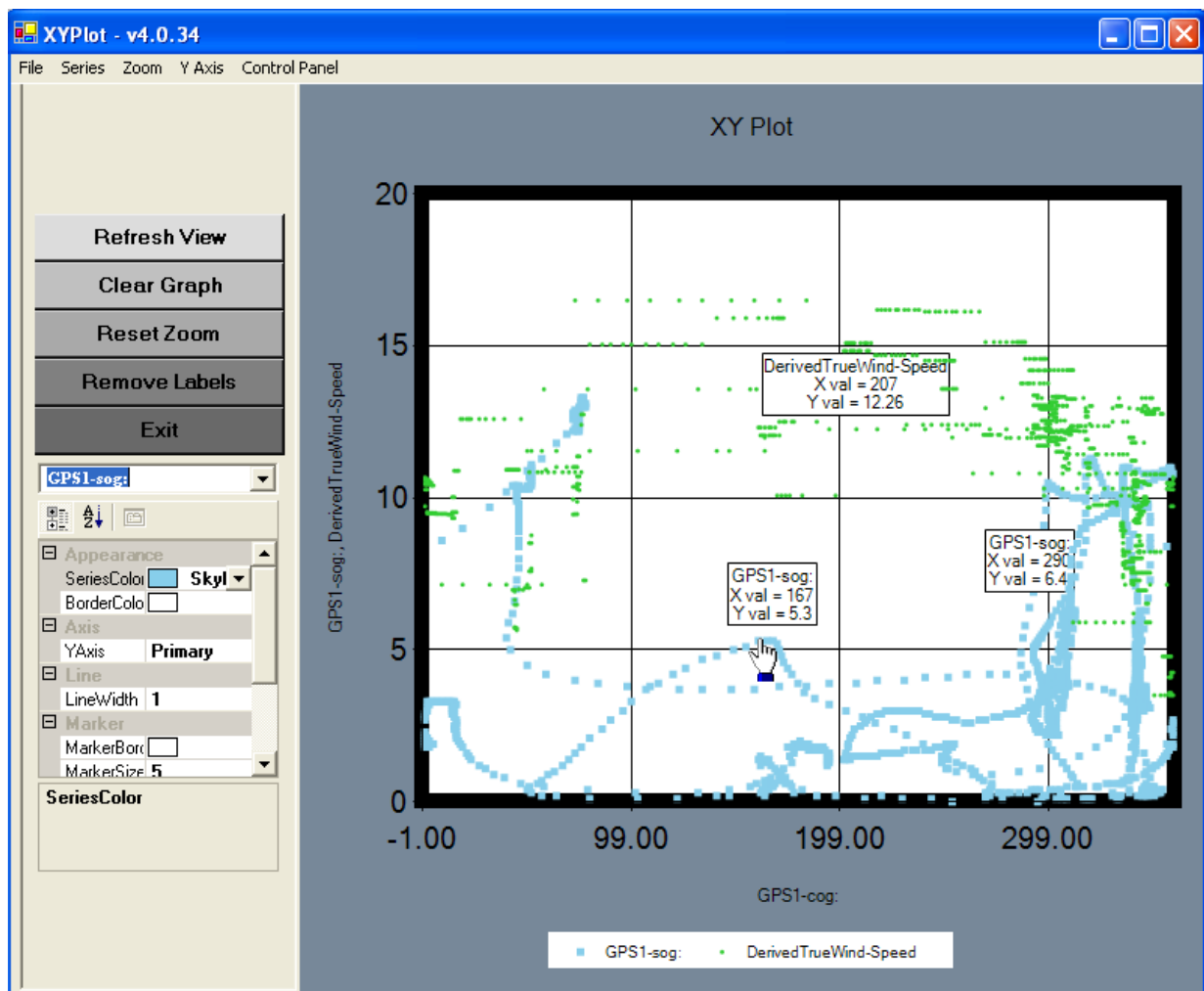
- APPEARANCE
 - ✓ SERIES COLOR: modify the color of the time series
 - ✓ BORDER COLOR: modify the background color
- AXIS
 - ✓ Y AXIS: set the time series to be part of the primary or secondary Y axis. The chart program is a dual axis program and a particular time series may be assigned to either series.
- LINE WIDTH
 - ✓ WIDTH: The pixel width of the line
- MARKER
 - ✓ MARKER BORDER: If a point has a border, if plotting in points
 - ✓ MARKER SIZE: the size of a point on the plot, if plotting in points
 - ✓ MARKER COLOR: the color of the point, if plotting in points
 - ✓ STYLE: the Style of the point, if plotting in points
- SERIES TYPE
 - ✓ SERIES FORMAT: a selection of points or line style for the time series.

Graph Behavior

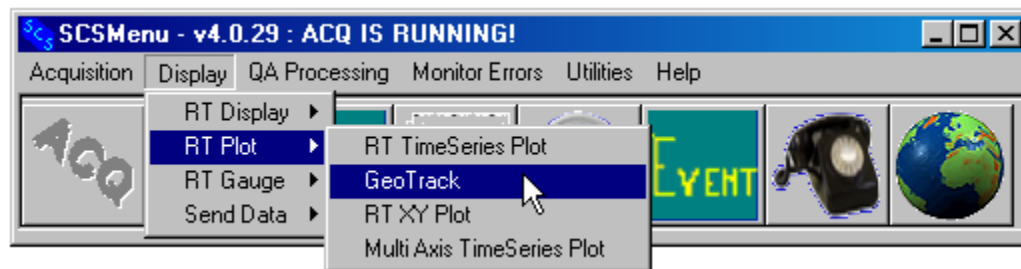
The graph will plot all data points in the current graph that will be visible to the user. This ensures that the plot reflects the data collected while maintaining efficiency by not over plotting duplicate pixels. XY plots can look inherently messy unless they are being applied to appropriate sensor values. They are excellent tools for plotting multiple sensors versus depth such as conductivity and temperature vs depth. They are not appropriate for comparing to normal sensors

6.6.10 Show a Data Point

The user may right click over any data point or time series feature and the plot will display the value of the closest data point.



6.7 REAL TIME TRACK LINE PLOT – “GEOTRACK”



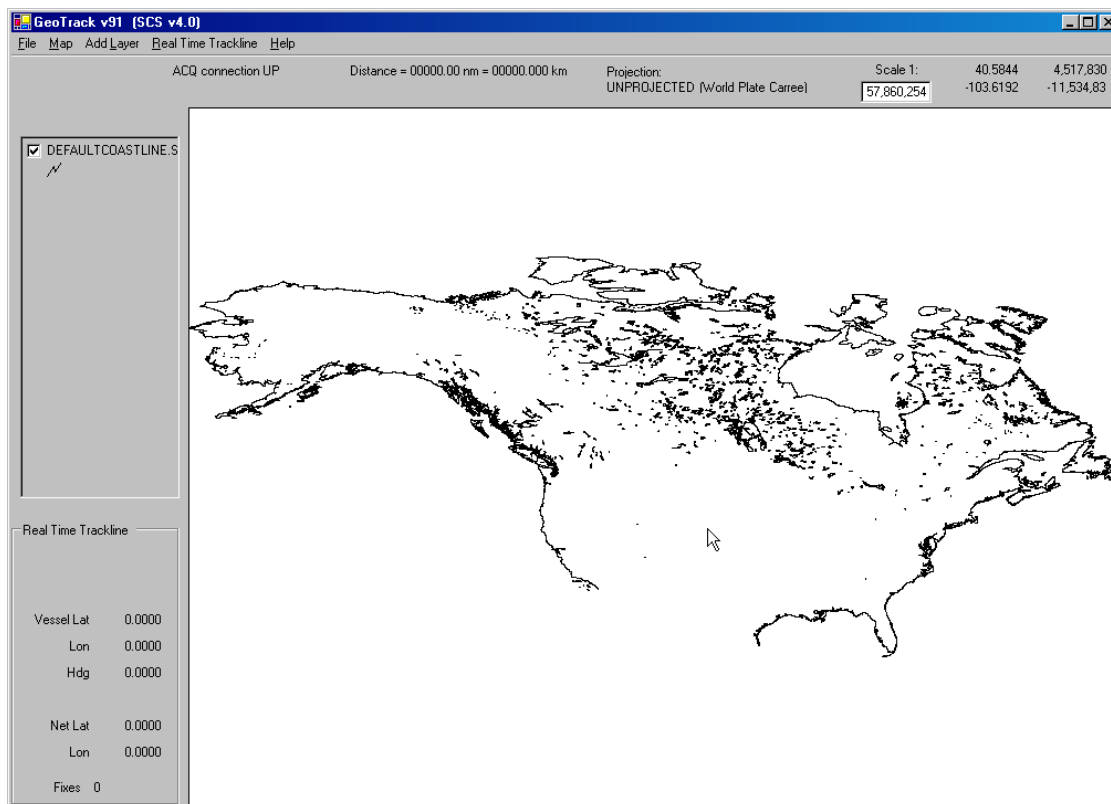
Introduction

Geotrack is a *Geographic Information System* (GIS) application. It is able to interact with commercial GIS products by exchanging data with them using files in various formats.

Geotrack was written using software provided by the Environmental Science Research Institute (ESRI), one of the world leaders in GIS products. Therefore, Geotrack can interact with ESRI products more easily than with other commercial GIS packages.

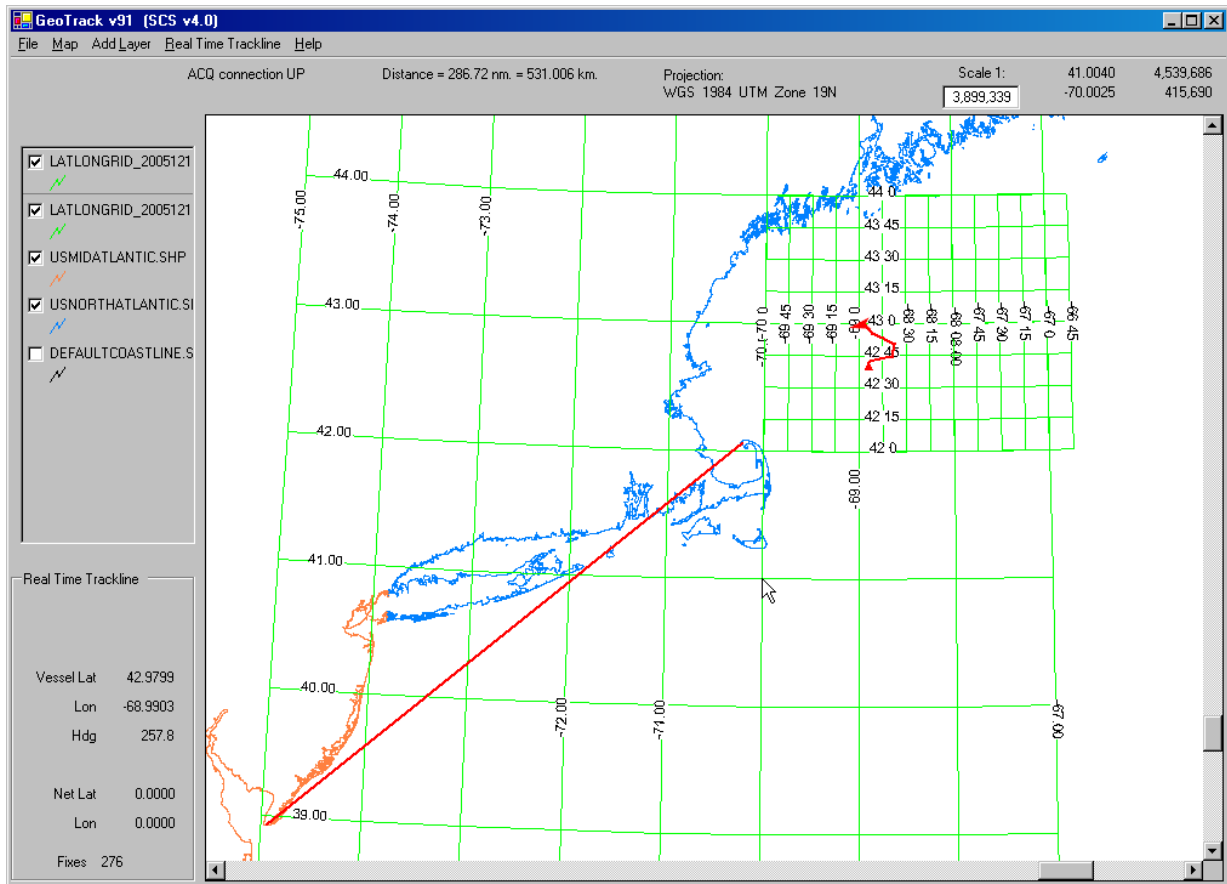
Starting Geotrack

You start Geotrack from the Display Menu -> RT Plot -> GeoTrack, or by clicking on the Globe icon in the toolbar. This will bring up a previously saved view or, if no view has been saved an initial default map of North America. The figure below shows the initial startup map.



The Geotrack Main Window: An Overview

Below is a sample map as it might appear during operations. Geotrack's main window is divided into sections, the largest being the *map*. To the left of the map, is the *Layer Legend*, showing summary information about each map layer. Below the layer legend is the real time position and heading of the ship. In the upper left corner of the window is the map scale. To the right of the scale are the Lat/Lon and equivalent XY coordinates of the mouse cursor. To the left of the scale is the current map *projection*. Each of these map components will be explained in the sections that follow.



6.7.1 Map Layers

A GIS map is made of one or more *layers*. Each layer has one component of the overall map. Think of a layer as a clear plastic sheet with some information on it. The map is then a stack of layers that overlay each other.

6.7.2 Layer Legend

The layer legend summarizes the content of each layer. The name of the layer is the name of the file that is the source of the layer's data. There is a check box for each layer that you use to control the layer's visibility. Each layer has a symbol that indicates both its color and the type of data being displayed. There are three types, point, line and polygon. Here all five layers are lines. Finally, the layers are stacked in the order that they are drawn on the map. Layers at the top are drawn over those underneath them, and will overwrite them. Refer to the section on the layer legend below for more detail.

6.7.3 Coastlines

Two layers show coastlines, in orange and blue. Geotrack comes with a library of world-wide coastlines.

6.7.4 World EEZ Boundaries

These are not shown in the diagram, but are available from the Add Layer -> Shape File menu.

6.7.5 Lat-Lon Grid

The two green layers are lat-lon grids, a wide area grid at one degree intervals, and a smaller grid in the working area at 15 minute intervals.

6.7.6 Real Time Trackline

The real time trackline is shown in red in the upper right part of the map. The ship is represented by the arrowhead. The orientation of the arrowhead reflects the heading of the vessel. The red line shows the history of the ship's movement, and is terminated by the red triangle that marks the start of the track line.

In the bottom left of the main window is the numeric readout of the ship's current position and heading. Geotrack can also display the position of a net or other towed body, but in this sample that feature is not being used.

6.7.7 Distance Measuring Tool

The long red straight line is from the distance measuring tool, which is being used to compute the distance between the line's endpoints, Cape May, NJ and Cape Cod, MA. The numeric readout is just above the map area, showing the distance in nautical miles (286.76) and kilometers.

Controlling the Map with the Mouse

Your mouse may or may not have a mouse wheel on it. Geotrack can work with or without a mouse wheel, but works much better with one. When Geotrack starts it tries to detect a mouse wheel. Geotrack can usually detect a mouse wheel, but may not, especially if you are connected to your computer through a KVM switch. If a mouse wheel is detected the Map-> Use Mouse Wheel menu item is checked; you can uncheck it to disable it and simulate use of a mouse with no wheel.

6.7.8 Panning and Zooming

With the Mouse Wheel, roll the mouse wheel to zoom in or out. Hold down the mouse wheel and move the mouse to pan. You can also zoom by left-clicking and holding, then dragging out a rectangle. Let the button go and you will zoom to that area.

Without the Mouse Wheel If a mouse has no detectable wheel, or if you have disabled the wheel, you must make a selection from the Map menu before doing any panning or zooming.

Select Map -> Zoom In. Then hold down the left mouse button and drag out a rectangle on the map. Let the button go and you will zoom to that area.

Select Map -> Zoom Out. Right-click on the map and you will zoom out and also center the map view on the selected location.

Select Map-> Pan. Hold down the left mouse button and move the mouse. The view will move with it.

The Map's Context Menu

Right-click on the map area to bring up the map's context menu, and select any of the choices

- Background color sets the color of the land and sea areas on the map.
- Recolor all layers sets all layers to a common color.
- Rotate lets you rotate the map from its usual north-up orientation.
- Full extent zooms the map all the way out so that the entire contents of all layers are visible.
- Measure Distance brings up the distance measuring tool to compute the length of a multi-segment line.

Using the Distance Measuring Tool

Use this tool to measure the distance between any two points, or to measure the length of a line with multiple line segments.

To invoke the tool, right-click on the map and select Measure Distance. Then, move the mouse to the starting point on the line and left click. Move the mouse to each waypoint and click at each one. Double-click at the end of the line to finish the operation. Read the distance at the top of the map.

Note the following about the distance tool:

- The distance will not be accurate using the default projection (“Unprojected”, or “World Plate Carre”).
- The cumulative distance to each way point is displayed during the operation, but is not accurate under any projection. Only the final result is accurate.
- Waypoint lines cannot be saved and will disappear as soon as the map display is changed in any way.

Controlling the Map with the Layer Legend.

Use the checkbox on each layer to control its visibility in the map. Drag layers up and down in the Legend to control which layers have drawing priority. Layers are drawn in order of their position on the stack, the bottom layer first, the top layer last. Each layer overwrites the layers that preceded it.

Right-click on a layer in the Layer Legend to control aspects of that layer:

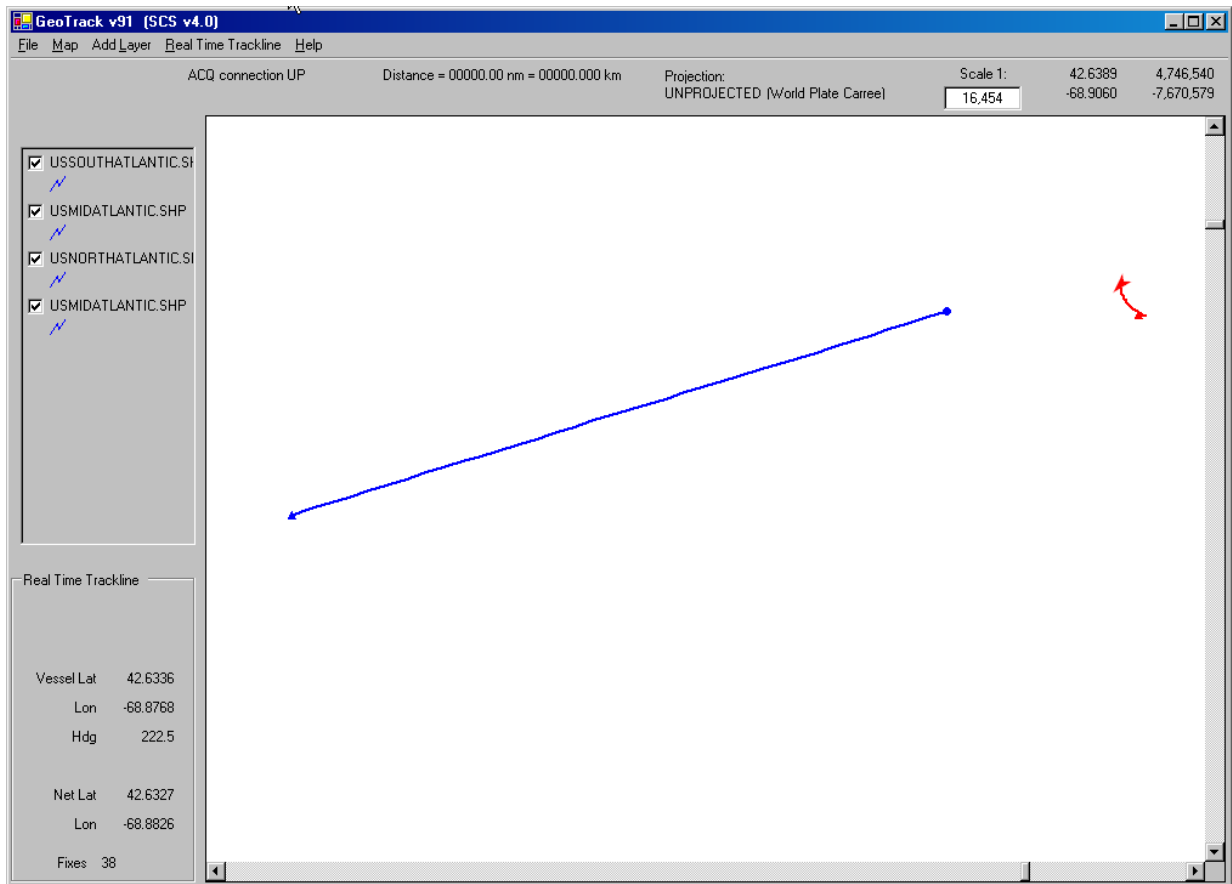
- Delete removes the layer from the map. The file holding that data is NOT deleted and can be brought back into the map at any time.
- Properties: not used.
- Background color: not used.
- Line color: changes the color of a line layer.

Working with Tracklines

Tracklines are layers that show the history of the ship's position over a period of time.

Geotrack offers simultaneous real time tracking of two objects. Geotrack refers to these as the ship and the net, although you could track anything that is reporting its position from a GPS receiver. You can track one or both positions.

Figure 2 shows the data displayed for two sample real time tracklines. In the lower left is the latest position and heading data as received from ACQ.



In this figure the ship data is in red. The ship itself is represented the arrowhead symbol, and its orientation is based on the heading data. The small red triangle indicates the position of the ship when the trackline was first enabled.

The “net” or other towed body data is shown in blue. The blue triangle is at the start of the trackline and the current position is at the blue dot. Heading data is not tracked for the “net”.

6.7.9 Real Time Trackline Properties

Before starting real time track line select Properties... from the Real Time Trackline menu. This brings up the window in figure 4.

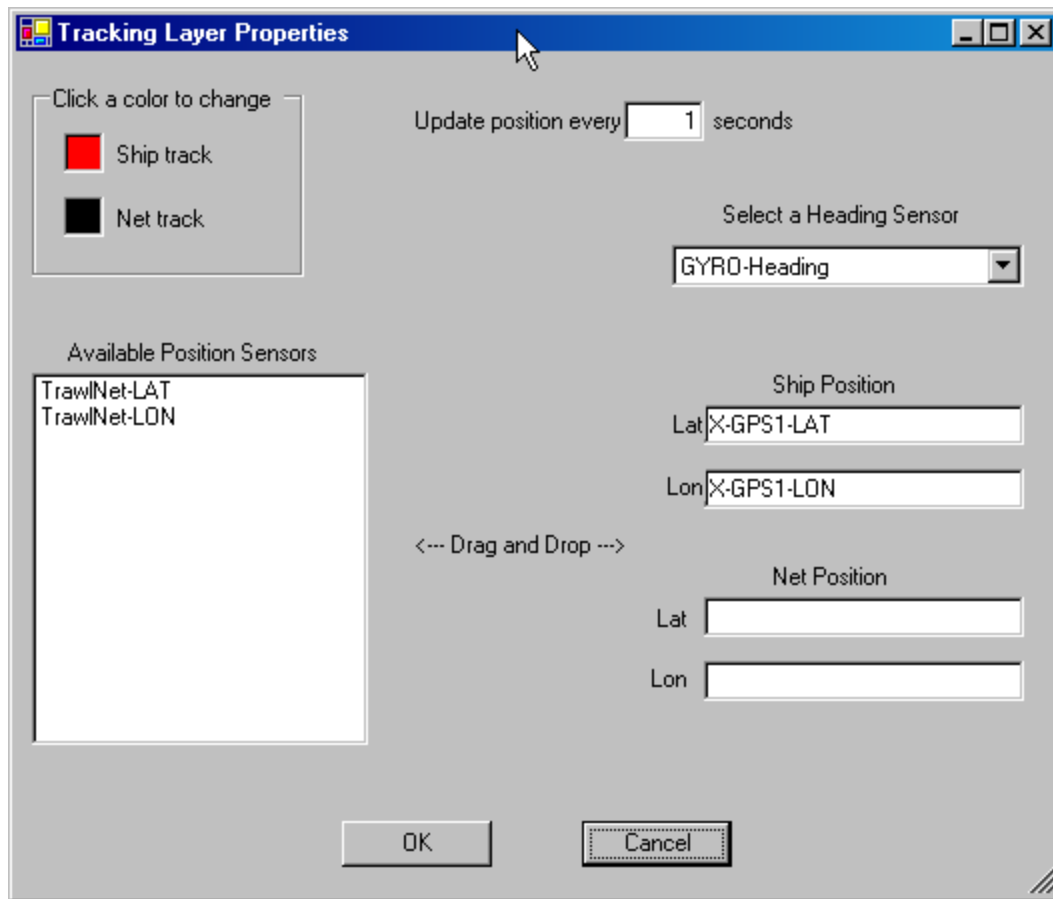


Figure 2 Trackline Properties

On the left is a list of position sensors (all sensors in the sensor configuration file having a type NAVSEC. Simply drag the desired sensor to the appropriate text box on the right. If an object's Lat and Lon boxes are both filled Geotrack will track the object. To stop tracking simply drag the lat/lon sensor from the text box back to the Available Position Sensors list.

Select the heading sensor from the drop down list. This will be used to orient the ship symbol during tracking operations.

Select the color of each trackline by clicking on the color box.

6.7.10 Starting and Stopping Real Time Trackline

To actually start tracklines (or stop them), select Enable from the Real Time Trackline menu. A check mark will appear whenever the tracking is enabled.

6.7.11 Changing Real Time Trackline Properties

You can change any aspect of the trackline's appearance during operations by simply going back to the Properties dialog and setting new colors, sensors or update rate. When you click on OK the new properties will take effect.

6.7.12 Saving Real Time Trackline Data

At any time you can save the current real time trackline by selecting Save to Shape File... from the Real Time Trackline menu. The file is written to the “Saved Trackline Data” subdirectory under the Datalog directory. The name of the file is simply the date and time of the first position recorded in the file. The trackline(s) will then be erased from the map and new tracklines will be started. After the trackline is saved you can then display it again using the Add Layer->Shape File... menu selection. See section for more information about Shape Files.

Shape Files

A *Shape File* is a commonly used file standard for different GIS products to exchange data. Shape files are central to the workings of Geotrack, as we shall see.

Geotrack can display any shape file you get from anyone else. The internet is filled with web sites that offer all sorts of geographic data you can download as Shape Files. You can also hand Shape Files created by Geotrack to visiting scientists to use in other GIS products back in their office

What people call a “shape file” is actually three files with an optional fourth. The files all have the same name. The three required files have extensions .DBF, .SHP and .SHX. If any of these files is missing the other two are useless.

The fourth (optional) file has extension .PRJ and is only used in displaying data in different map projections (“Mercator”, UTM, etc.) in more advanced GIS activities not covered here. Any shape file created by Geotrack will create the PRJ file, but not every shape file you get from outside sources will include a PRJ file. The PRJ file is never needed if you stay with the default Geotrack projection. But if you want to use a shape file with Mercator or any other projection the lack of a PRJ file will cause errors.

Shape Files and Map Layers

Shape Files are closely connected to Map Layers. Reading a shape file will generate an associated map layer, and the name of the layer in the legend comes from the name of the Shape File. Conversely, in creating a new map layer (Historical RAW navigation files, stations, lat-lon grids) and shape files constructed from RAW navigation data) Geotrack creates an associated shape file on disk.

There are three kinds of shapes that a shape file can contain: point, line and polygon. Any given shape file can contain only one of the three kinds; which kind is seen in the Label Legend by the symbol underneath the layer name.

The Adding Layers Menu

The selections in this menu all create map layers by reading in files of various sorts.

6.7.13 Add Shape File

Geotrack pops up a File Open dialog that starts you out in the Shape Files subdirectory of the GIS40 directory. Certain subdirectories (three coastlines) are created when you install SCS, you can add others yourself.

Browse to the desired directory and shape file, click on OK and Geotrack will read it in and create a new map layer.

6.7.14 Coastlines

SCS provides world coastline Shape Files in three levels of detail. A low resolution map of the world is displayed if you have not saved a view previously. It is not recommended for close-in working areas. The medium resolution map is at the same detail as SCS 3.3, and can be used for large regional maps. High resolution provides the best coastlines for use in every day work.

6.7.15 EEZ Boundaries

In the same folder as the coastline shape files is another folder for EEZ Boundaries. Two formats are available, un-shaded lines, and shaded polygons.

6.7.16 Add RAW Trackline

You can display the position of a ship or towed vehicle from any RAW file that contains NMEA \$GPGGA, \$GPGLL or \$GPRMC sentences. Browse to the desired RAW file and select it. The position data will be extracted and displayed as a new layer. Multiple file selections may be made and they will be concatenated into a single layer.

Geotrack asks you for the name of the layer to be created. Enter any name you wish.

The data is also saved as a Shape File which takes its name from the layer. The file is written to the “Saved Trackline Data” subdirectory under the Datalog directory. The file takes its name from the name of the RAW file. You can then hand the shape file to anyone for use in their own GIS product.

6.7.17 Add Stations

Station Files are collection of points of interest. There are several types, each displayed with a different symbol. To create or edit a station file you must select “Station Builder” from the SCS “QA Processing” menu, where you set the type, location and label for each point. Refer to chapter Six – Quality Assurance for more information on the Station Builder app.

Station types and the symbols that represent them are predefined and cannot be changed.

The following figures show the types and their symbols.



Figure 3 - Station Symbols

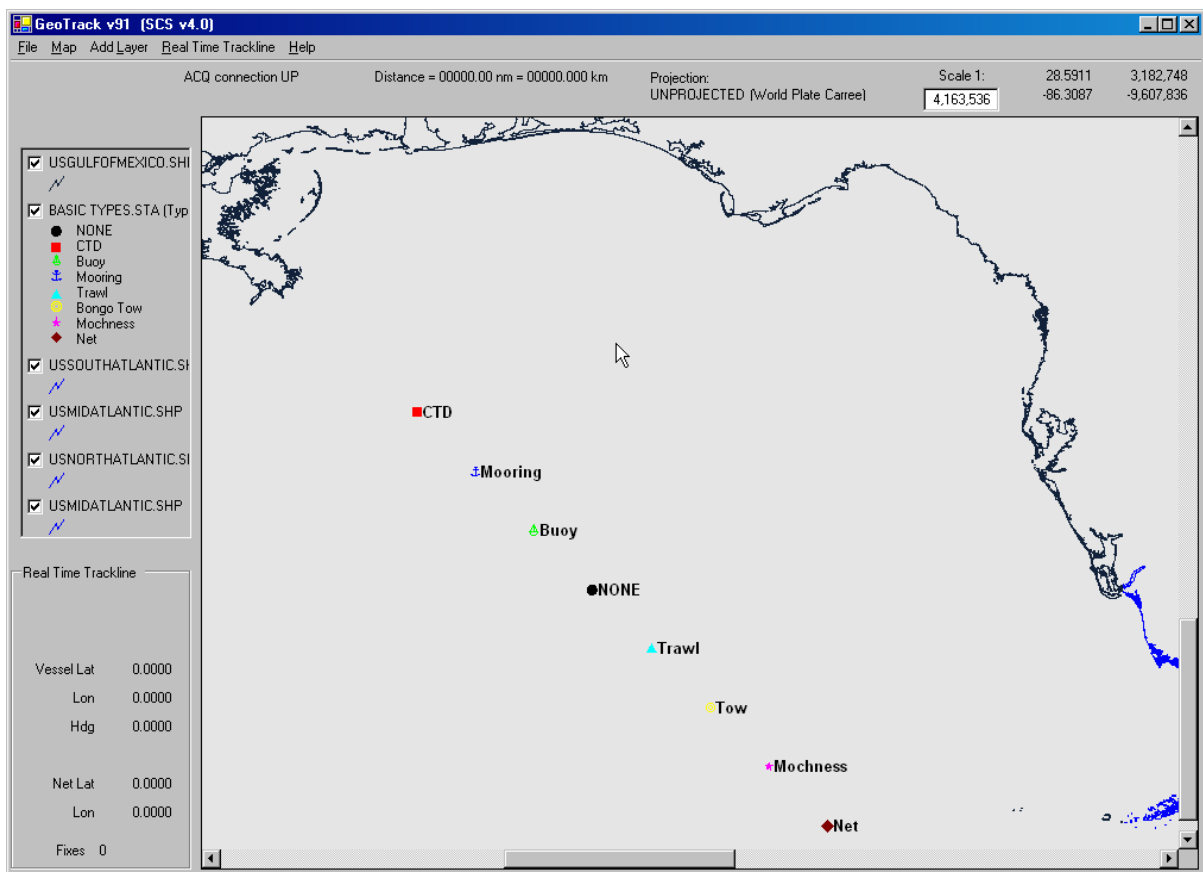


Figure 4 - Sample Station Display

To display a station file, select Stations... from GeoTrack's "Add Layer" menu. The stations in the file become another layer on the map, which is displayed in the Layer Legend. You can control layer visibility with the legend but nothing else.

6.7.18 Coverages

"Coverages" are another means of exchanging GIS data. Coverage is actually an entire directory tree with several files. It is proprietary to ESRI. GeoTrack can read Coverages and display them as layers, but cannot save layers to Coverages.

6.7.19 ESRI Grid

An “ESRI Grid” is another way to store GIS data. An ESRI Grid is actually several files in a directory tree. It is proprietary to ESRI. Geotrack can read and display ESRI Grids but cannot save layers as GRIDs.

6.7.20 Images

Images are photographic representations of GIS data and come in several public domain formats: TIFF, GIF, MPEG.

6.7.21 Add Lat-Lon Grid

Adds a grid of latitude and longitude lines to a map area you designate. There are two methods of creating a lat-lon grid. Each method produces a slightly different grid.

Method one is to select Lat-Lon Grid from the Add Layer Menu. Geotrack auto-computes the grid spacing based on the area of the current view, with the goal dividing the view into a dozen or so areas. Grid lines will be labeled with the degrees and minutes. You have no control over this behavior.

Method two is to hold down the control key and then drag out a rectangle as if you were trying to zoom in. This will bring up the window shown in Figure 5. Here you select the grid interval and the style of the gridline labels. Geotrack auto positions the labels so that they do not overwrite each other (this is scale-dependent), and you have no control over that.

In either method Geotrack creates another map layer and puts it at the top of the Layer Stack. From here it can be controlled as any other layer is. Refer to the above discussion of layers for more information.

Geotrack also creates a shape file to go along with the layer in the LatLonGrids subdirectory of the ScsGis directory. There it can be recalled through the Add Layer>Shape File menu or given to a visiting scientist.

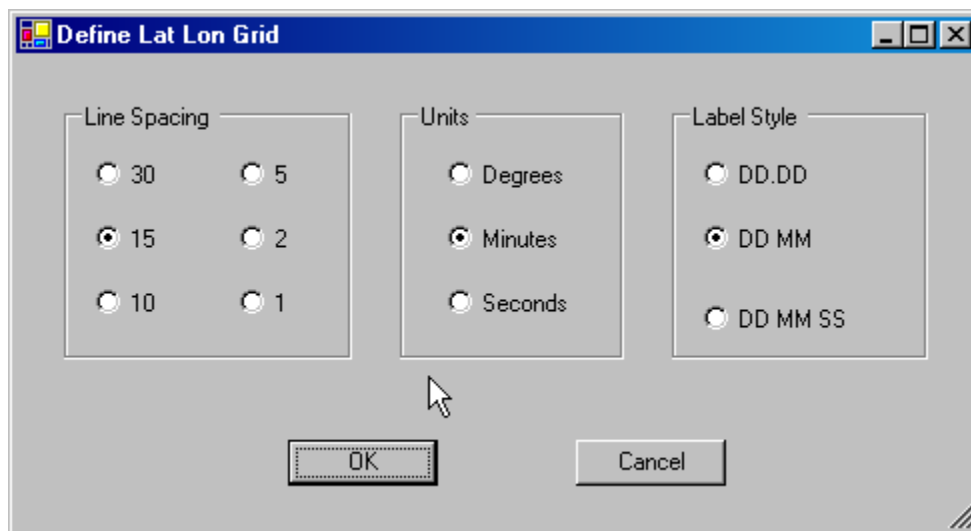


Figure 5 – Defining a Lat-Lon Grid

Saving and Restoring Views

A Geotrack “view file” is a means of quickly setting up certain map characteristics. In a view file you can save:

- Coastline files that have been loaded
- All items set in the Trackline properties window.
- Other map elements such as projection and rotation.

Then, you can open the view file at any time to change the map characteristics as you saved them.

To save a view file, select Save from the File menu and supply a name. The map characteristics listed above will be written to the file. Later, select Open from the File menu to restore those characteristics.

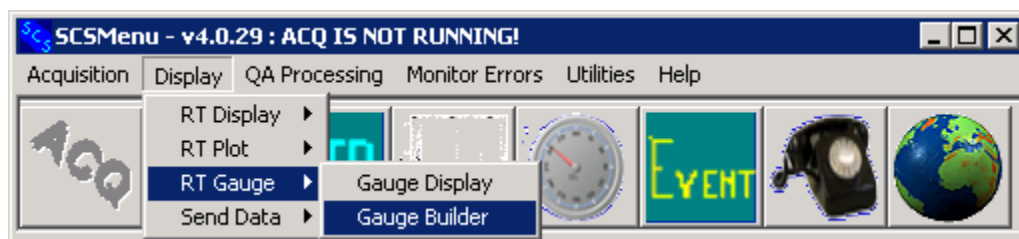
6.7.22 The Default View File

A view file can have any name, but the file named “Default” has a special use. When you start Geotrack it looks for that file, and if it exists Geotrack will read it in and construct the map according to it. If the map does not exist Geotrack will display the low resolution map of the world.

6.8 BUILDING AND USING REAL TIME GAUGE DISPLAYS

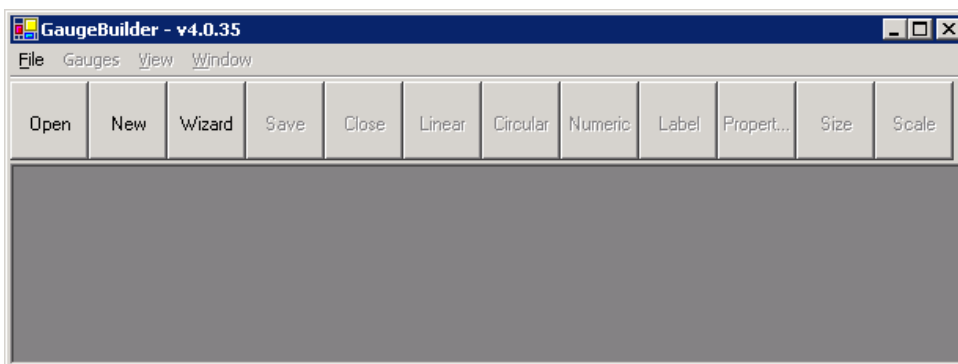
Gauge Builder

The Gauge Builder program allows for customization of gauges which are displayed using the *Gauge Display* program. These gauges can be used to visually represent incoming SCS sensor data and ranges in real time. They can take the form of a circular, numeric, or linear gauge or a combination of any of these.

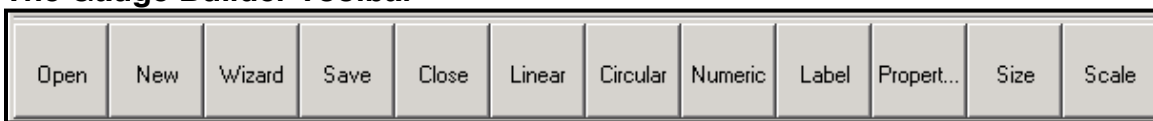


Using the Gauge Builder

Open up the main *SCS Menu* program and choose *Gauge Builder* from the *Real Time Gauge* menu under *Display*. You will be presented with an empty window in which you can start building a new gauge template, or open/modify an existing template file (*.gag). A blank display appears as shown here.

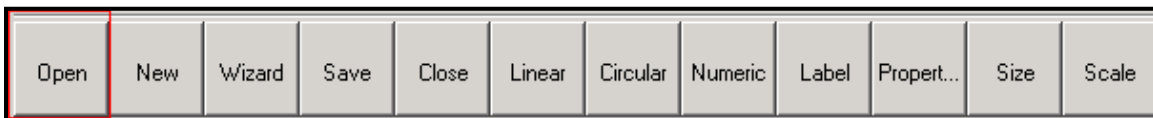


The Gauge Builder Toolbar

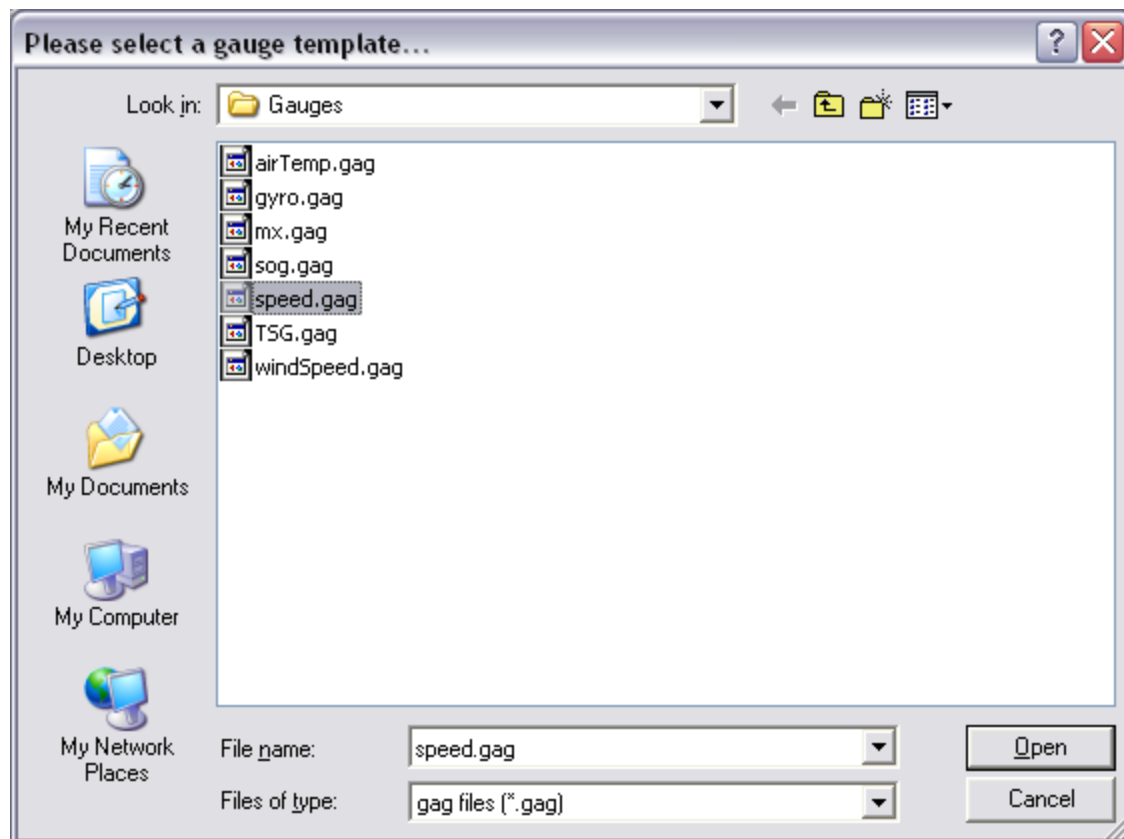


The main form has a toolbar which allows for easy access to the most common features of the gauge builder software. All of these buttons perform operations which can be accessed through the Main Menu or through the property grid for particular gauge objects.

6.8.1 Open Button

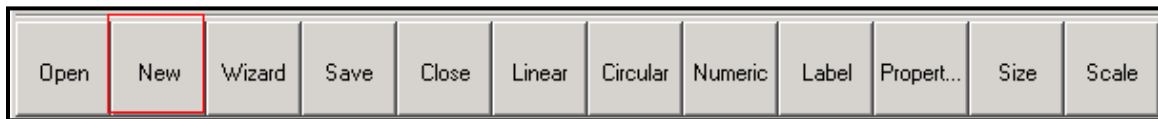


The open button is used to open a previously saved .gag template file. This file should be created and maintained using the gauge builder software.

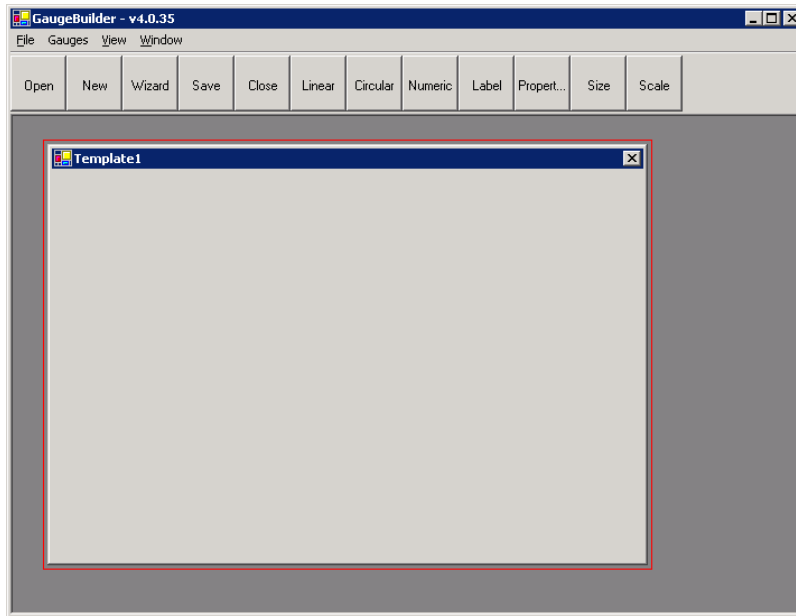


Select the appropriate file you want to open, and click *Open*, the preexisting template should load into the screen. It's possible to have multiple templates open at the same time.

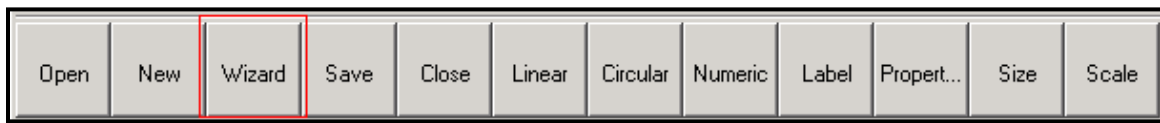
6.8.2 New



The New button is used to create a new blank template file. This file will provide an area where you can place new gauges of multiple types. Once you have a template area you will gain access to the rest of the toolbar (if they are grayed out). The template will look like a simple grey block until you add a gauge of some type... You can click and drag gauges/objects to move them around the template. You can access specific properties through the *Properties* box.



6.8.3 Wizard

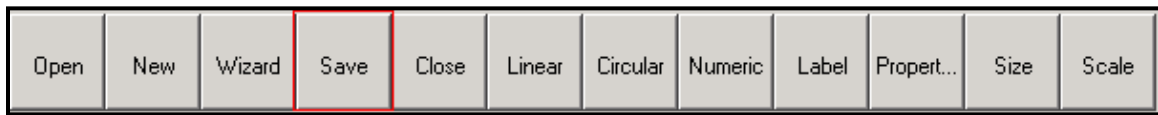


The wizard button is similar to the *New* button except that it brings up a wizard which allows for easier customization and visualization of a gauge. At this time, the wizard will only provide a single gauge on a template. The wizard can provide quite deep customization, however that is not necessary. A basic gauge will work just as well, but may not be as aesthetically pleasing.



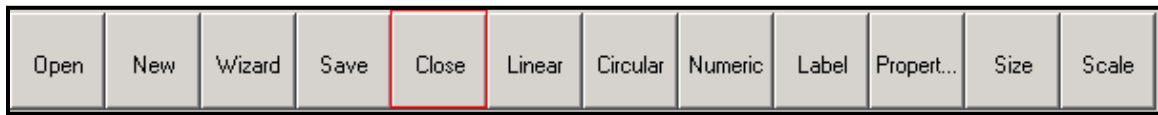
The gauge wizard starts off much as the figure above with the *Gauge Type Groups* set to *Current*. Please change that to the type of gauge you wish to build (Circular, Linear, or Numeric). You may also choose a default *Appearance Style* or you can manually change properties to mimic one of these styles. If you choose a style, you can still customize the appearance; this just sets a base to build from. There are a variety of different properties and attributes that you can set using the wizard. Please see the Gauge Wizard Demo for a quick introduction to some of these...

6.8.4 Save



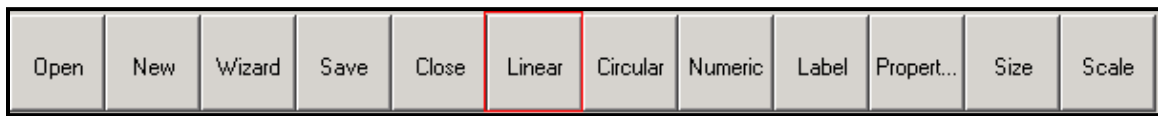
The save button saves an open template to its *.gag file. This file will be opened by Gauge Display for usage or by Gauge Builder for editing.

6.8.5 Close



The close button closes an open template. Make sure that you save any open templates that you want to edit or use at a later date before closing!

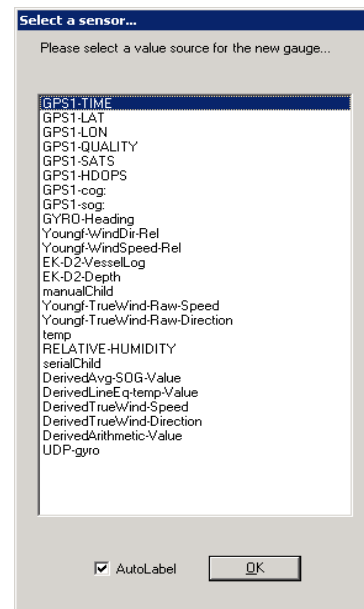
6.8.6 Linear

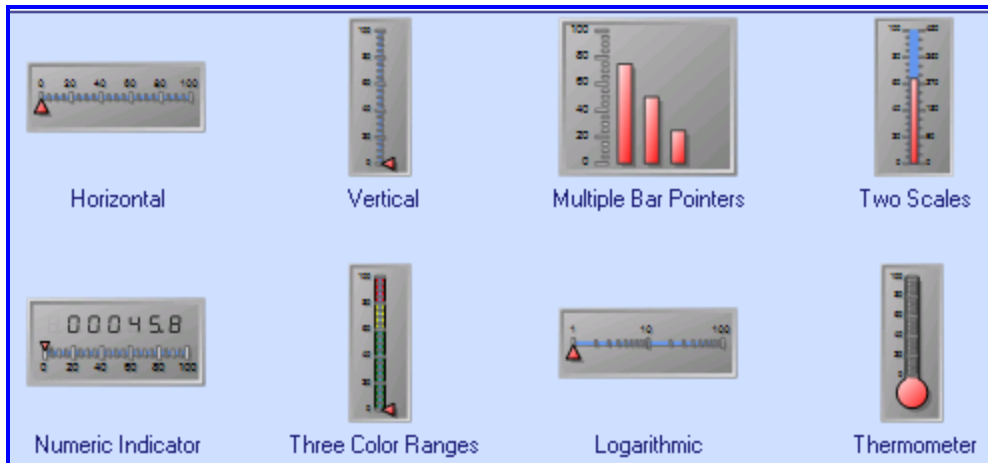


The Linear button inserts a gauge of the 'linear' type into the (preexisting) active template.

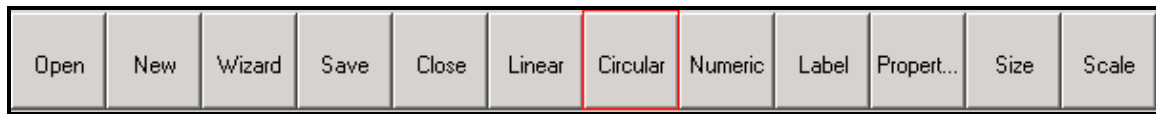
When you click on any of the gauge buttons, you will be prompted to select the sensor value you want to display with it. This will come up in a window before the gauge is actually displayed. Select one of the sensors and hit 'OK'. A label (sensor name) will be automatically attached to the new gauge. If you do not want a label automatically added, simply uncheck the 'AutoLabel' box next to the 'OK' button.

A linear gauge is a simple straight gauge. It can be represented as vertical or horizontal. It can look like bar charts, thermometers, numerical indicators, etc. Below are some examples (taken from the wizard) of linear gauges and their capabilities.

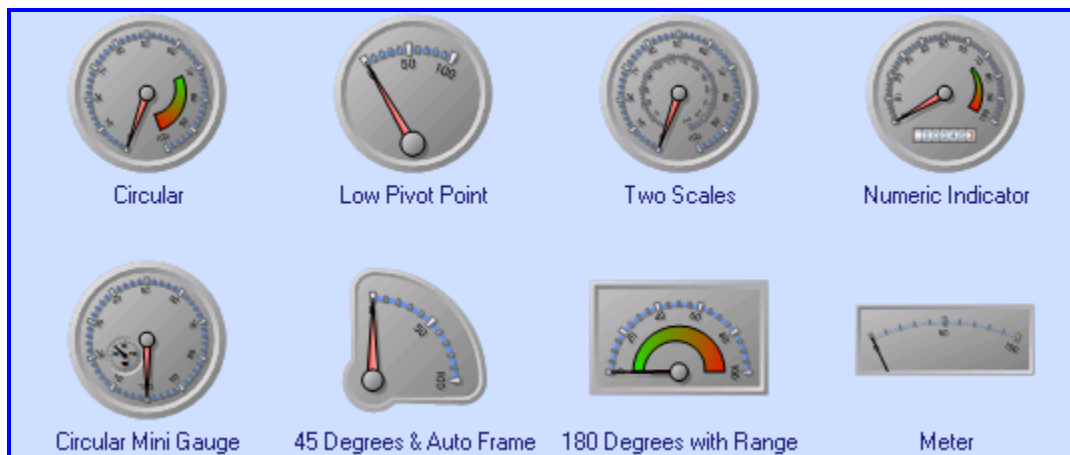




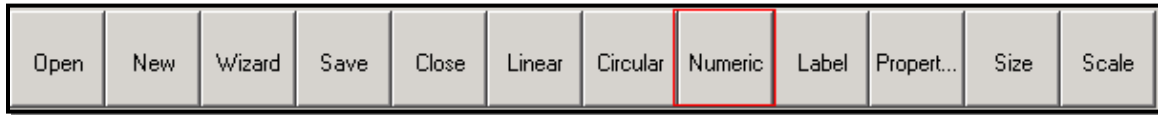
6.8.7 Circular



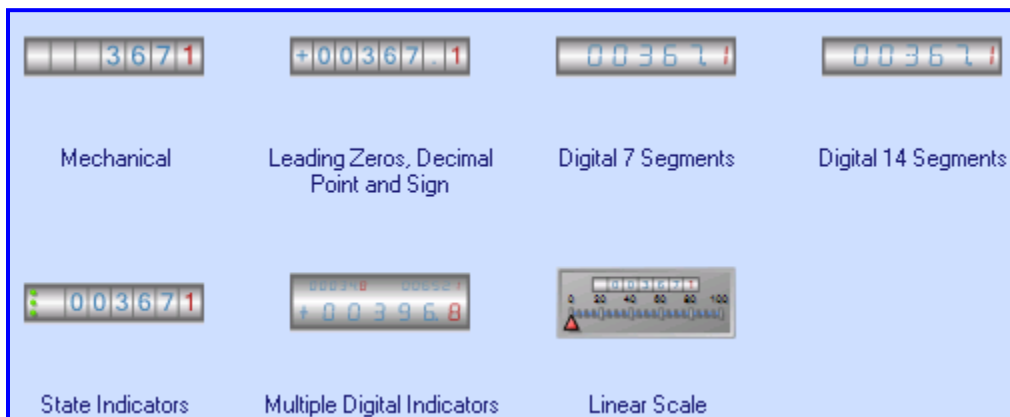
The Circular button inserts a gauge of the 'circular' type into the (preexisting) active template. There are a variety of different circular gauges that can be used. Below are some examples (taken from the wizard) of possible types of Circular Gauges.



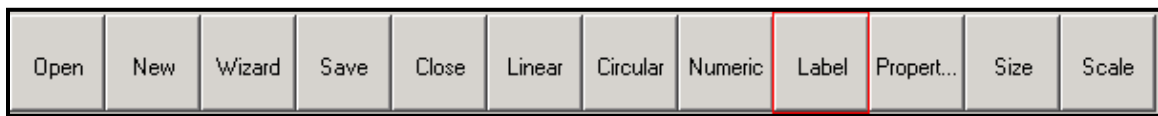
6.8.8 Numeric



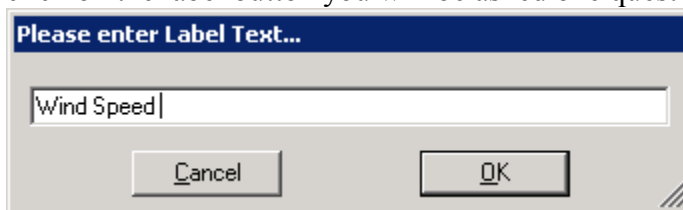
The Numeric button inserts a gauge of the 'numeric' type into the (preexisting) active template. Numeric Gauges are not as visually flexible as the circular and linear gauges since they simply display numeric data. However, there are still a variety of choices you can make and they can be the best way to display data. A good use could be as an additional feedback method. For example: put a numeric indicator inside of a circular gauge to get the visual indicator through the pointer and an exact value indication through the numeric gauge. Below are some examples (taken from the wizard) of possible numeric gauge configurations.



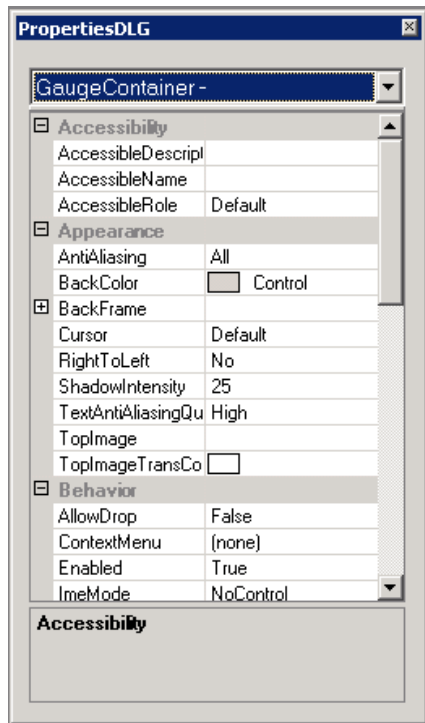
6.8.9 Label



The Label button inserts a simple label onto the template. This can be used to document what gauges or pointers are representing, or provide a title for the template, etc. When you click on the label button you will be asked one question: what do you want the label to say?



6.8.10 Properties



The Properties button opens the properties toolbar box. This is the main interface to customizing your gauges and the template. It consists of all public and modifiable properties which can pertain to all objects on the template. You can click on a particular gauge/object on the open template and all its properties will be displayed in this box, or you can select the object from the dropdown list located at the top of the box ("GaugeContainer" is selected in the figure to the right). Once you've selected the correct object, its properties are available for you to change to suit your desires. This box is the ultimate way to modify and customize your gauge templates.

NOTE: Some properties (collections, lists, etc) will display a special modification icon when selected (a dropdown arrow, a browse button, etc). Click on the button/icon to edit these properties.

NOTE: The property box can become confusing and complicated if you try to get too detailed. This is because the properties box allows for extreme customization; however, a basic functional gauge can be created and modified very easily through the top level properties. Feel free to delve into the detailed properties, consult the 'Info box' at the bottom of the property box ("Accessibility" is displayed in the 'Info box' in the diagram to the right) for an explanation of the selected property. As you get more comfortable with the properties of the gauges, you will be able to create some amazing gauges. The [Gauge Wizard](#) is basically a graphical interface to set the properties displayed in this box.

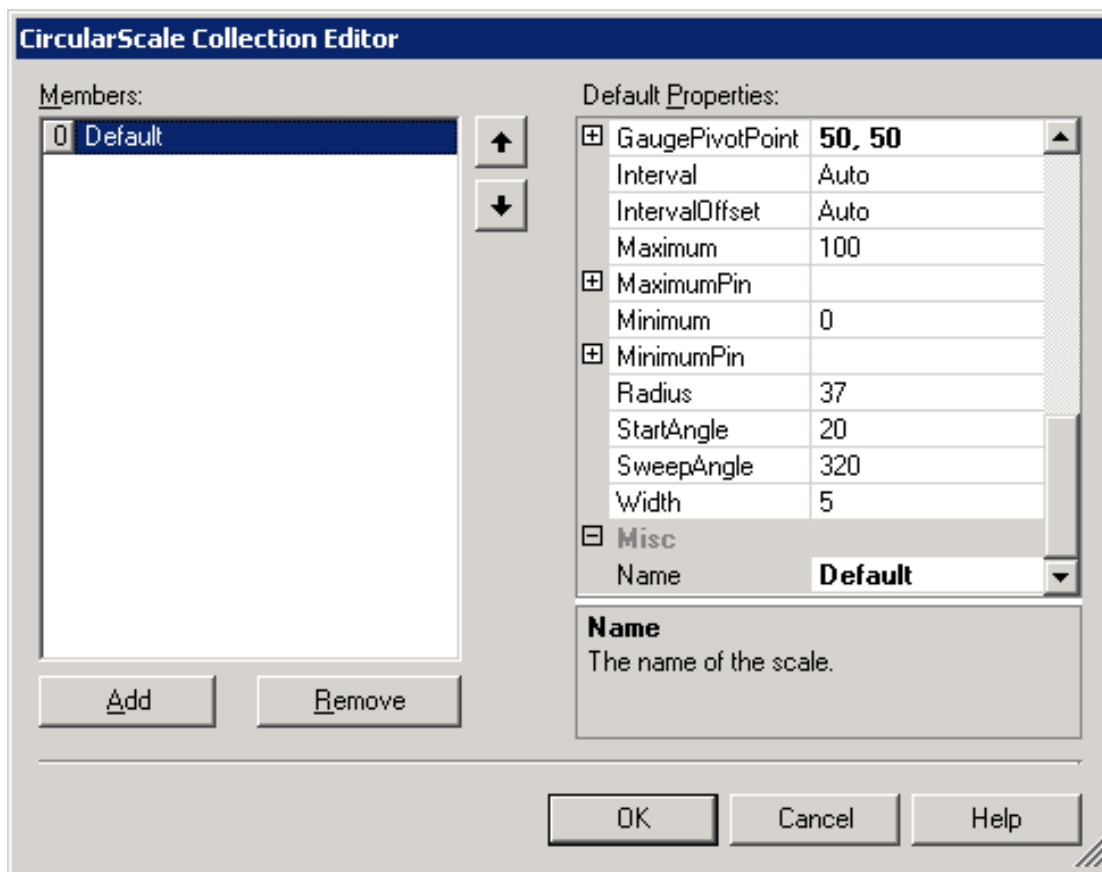
Some main properties to note:

This property gives you access to ‘Background’ for the gauge. You can change its shape, color, etc.

This collection holds all the scales for a gauge. You might want to use this to add alternate scales (i.e. one for kilometers and one for miles, etc). When you open a collection you will receive a custom property grid. This grid contains a list of items in the collection on the left, and all the properties for the selected item on the right (see below). The values found on the right can be modified in exactly the same manner as they are when referenced through the main property grid. Ensure that you have the correct member (in this case, Scale) selected before trying to modify the properties.

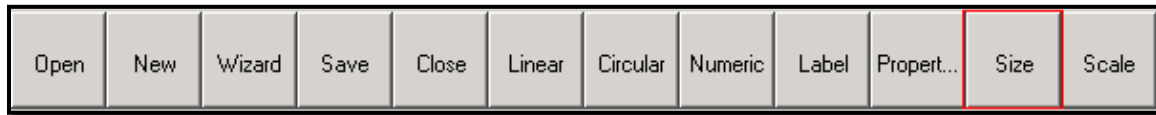
To add or remove scales from the collection simply click on the *Add* or (select the scale to remove FIRST) *Remove* buttons located under the list of objects on the left.

It is extremely helpful to enter a *Name* for each object you add to the collection or to the gauge. This helps in organization and tracking down bugs should something go wrong.

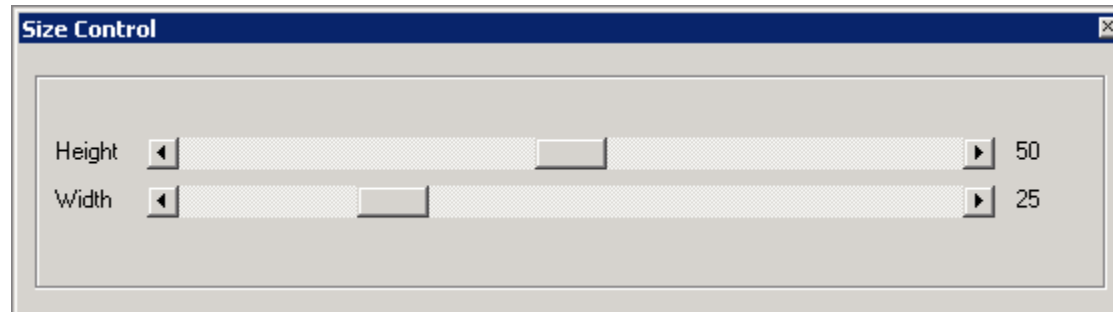


Inside the main *Pointers* Collection for a gauge, you’ll notice a property called *ValueSource* (in the Data Category). This is very important as it specifies how the pointer is supposed to act by defining which data is to be displayed. Whichever sensor value you select from the dropdown list is the sensor this pointer will point to when the gauge is run. Keep in mind, as with all collections, that there may be multiple pointers, ensure that you set the correct *ValueSource* for the appropriate pointer...

6.8.11 Size

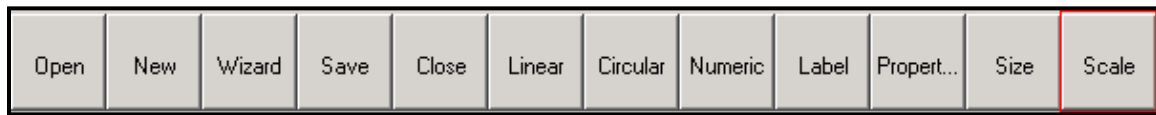


The *Size* button allows for modification of a selected gauge. Click on a gauge and the size dialog should automatically adjust to the current value for the size property.

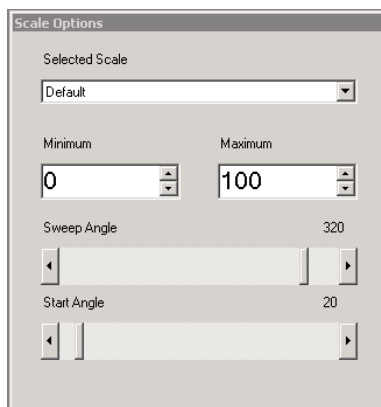


Slide the bar to the left/right to adjust the size. Remember that the size is based off a 0-100 scale and that the actual size may be dependent on both the height *and* the width (i.e. in order to make a circle larger, both the height and width must increase since it can only be as large as the smaller of the two, otherwise it'd be an ellipse!).

6.8.12 Scale



The *Scale* button presents a specialized box which can be used to modify a preexisting scale for the selected gauge. Simple click on the gauge you wish to modify (ensure the background color becomes faded to signal you've selected the correct gauge). Then click on the dropdown list on the Scale dialog to select the scale to modify. If your gauge only has a single scale then only one option will be presented in the dropdown list. Otherwise, select the scale you want to change. You can set the minimum and maximum values, adjust the sweep angle, and change the start angle. Play with it to see the effects...



Gauge Wizard Tutorial

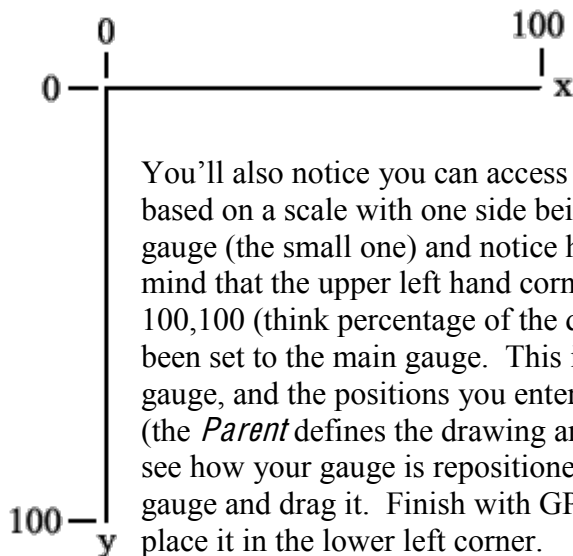
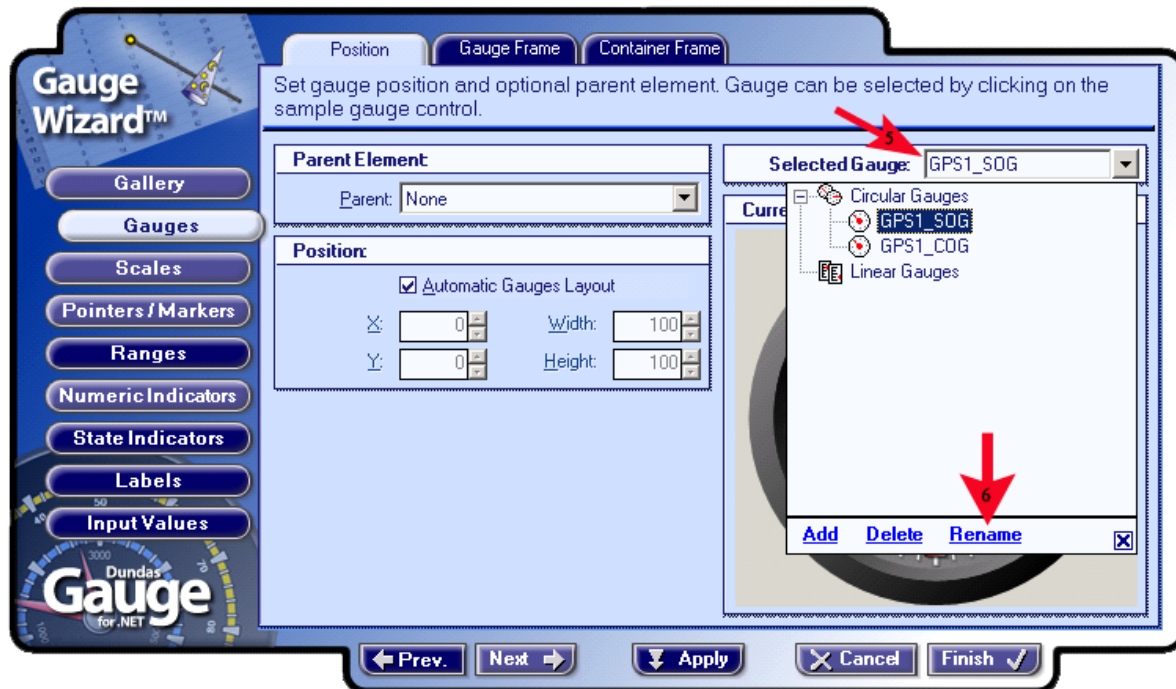
This demo hopes to provide a quick introduction to the [Gauge Builder Wizard](#). Please start Gauge Builder and click on the Wizard option in the toolbar or File menu. You should now have a blank template (in the background) and the Gauge Wizard interface with no data yet selected.

- 1) Let's set the *Gauge Type Groups* to *Circular Gauges*
- 2) Then choose the *Graphite* option for the *Appearance Style*.
- 3) Then click on the *Circular Mini Gauge* in the lower left hand corner
- 4) Once this is done, we can continue on to the next section. You can do this by clicking on the *Gauges* button presented in the linear row on the left.



Once you've continued on to the *Gauges* tab, you'll be able to see your selected gauge displayed on the right.

- 5) Select the drop down arrow next to the *Select Gauge* option. You'll see a list of all the gauges you've added to your template. Notice there are two listed under the *Circular Gauges* node. This is the main gauge and the little mini gauge which you can see in the lower left quadrant of the main gauge.
- 6) Choose the gauge named *Default* under the *Circular Gauges*. Rename it to GPS1_SOG. Change the *miniGauge* to GPS1_COG.

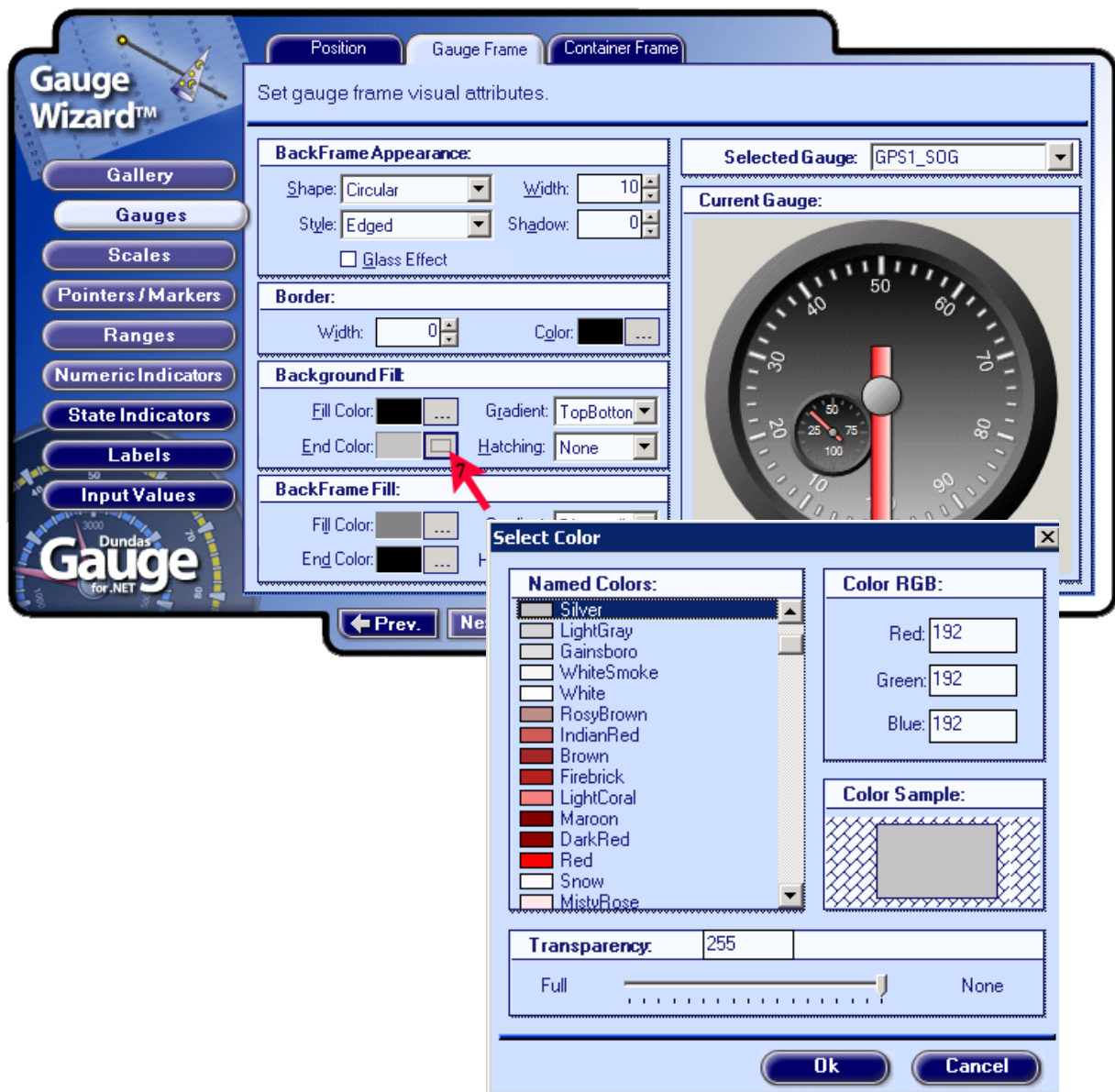


You'll also notice you can access a *Position* property. This property is relative and based on a scale with one side being 0, the other being 100. Select the GPS1_COG gauge (the small one) and notice how the values of the position change. Keep in mind that the upper left hand corner is 0,0 and the range only goes from 0,0 to 100,100 (think percentage of the drawing area). You also notice that the *Parent* has been set to the main gauge. This is because the mini gauge 'belongs' to the main gauge, and the positions you enter will be relative to the position of the main gauge (the *Parent* defines the drawing area for the position). Try playing around with it to see how your gauge is repositioned. You can enter manual values, or click on the gauge and drag it. Finish with GPS1_COG being set to X = 20, Y = 55. This will place it in the lower left corner.

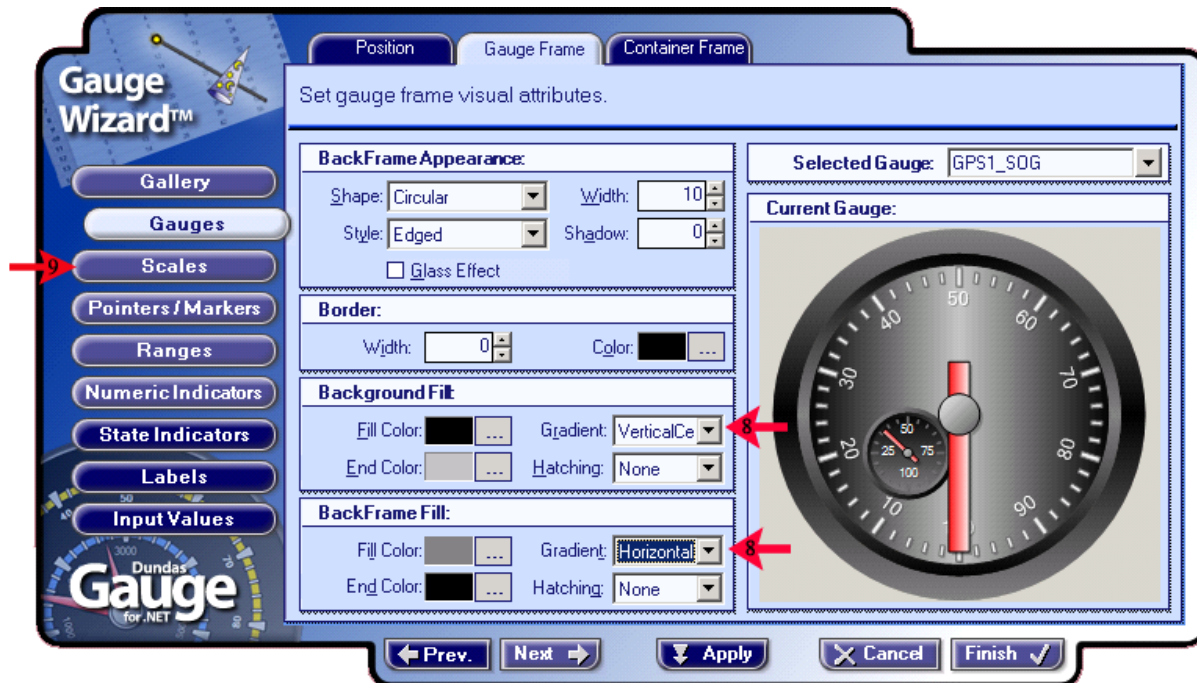
Once you've become familiar with the *Position* property, we'll continue to customize our new gauge. Click on the GSP1_SOG gauge, and shrink down the *Select Gauge* display so we can see what our changes look like. Then click on the *Gauge Frame* tab (located at the very top of the interface, between *Position* and *Container Frame*).

There are many different things we can change here. Personally, I like the default settings for the *Backframe Appearance* and the *Border*, so let's leave the default values and continue on with the Background Fill.

- 7) To add some more flare to our gauge, let's change the *End Color* of the *Background Fill* to silver. Click the little box next to *End Color*. Then choose *Silver* or enter the appropriate RGB values.

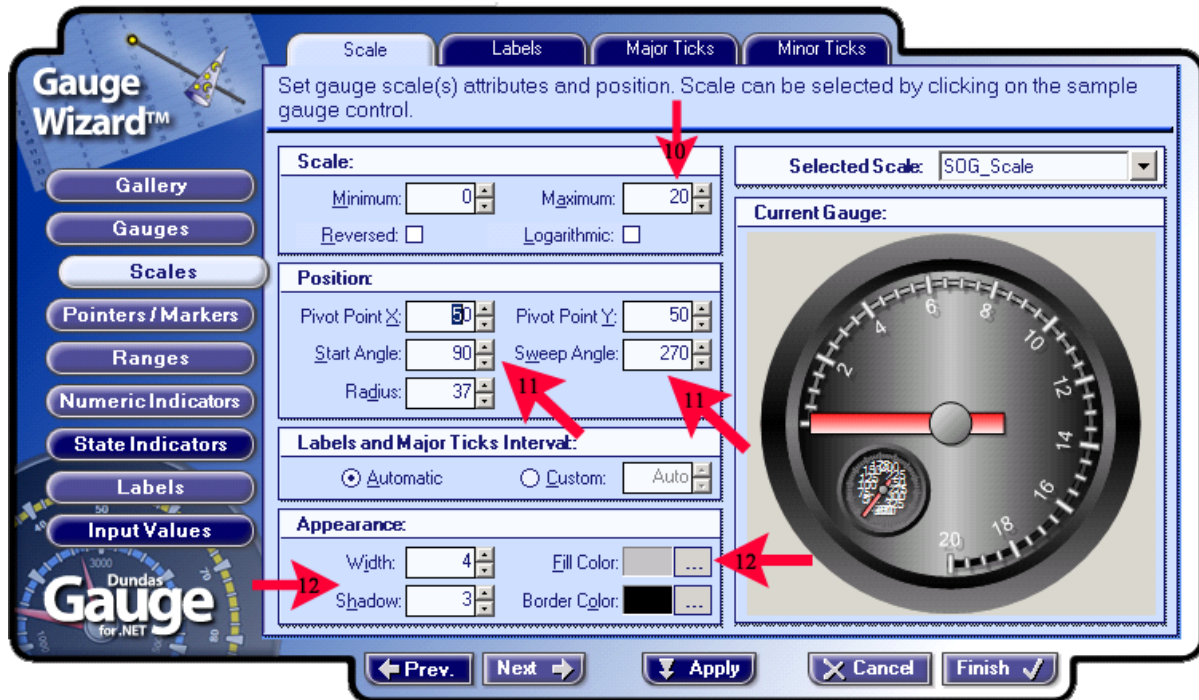


- 8) Then change the *Gradient* from *TopBottom* to *VerticalCenter*. Continue on and change the *Gradient* in the *BackFrameFill* from *TopBottom* to *HorizontalCenter* also.
- 9) When you've finished modifying the gradients, we'll continue on to *Scales* by clicking the *Scales* button.



The scales are the numeric indicators which allow a gauge to put its data in perspective. In our gauge, there are two scales: one for the main (large) gauge, and one for the small (mini) gauge. You can select different scales by clicking on them or by manually choosing (recommended) them from the *Selected Scale* dropdown list. Please rename your scales to *SOG_Scale* and *COG_Scale* respectively. This is done in the same manner as you renamed the *Circular Gauges* previously.

- 10) Set your *Scale* maximum to '20' for the *SOG_Scale*, and '360' for the *COG_Scale*.
- 11) Set the *Start Angle* and *Sweep Angle* to '90' and '270' for the *SOG_Scale*. This will make the scale start at 90° and finish at a 270° offset from the start.
- 12) Under *Appearance*: Set the *Width* to '4' and the *Shadow* to '3'. Also change the *Fill Color* to 'Silver.'



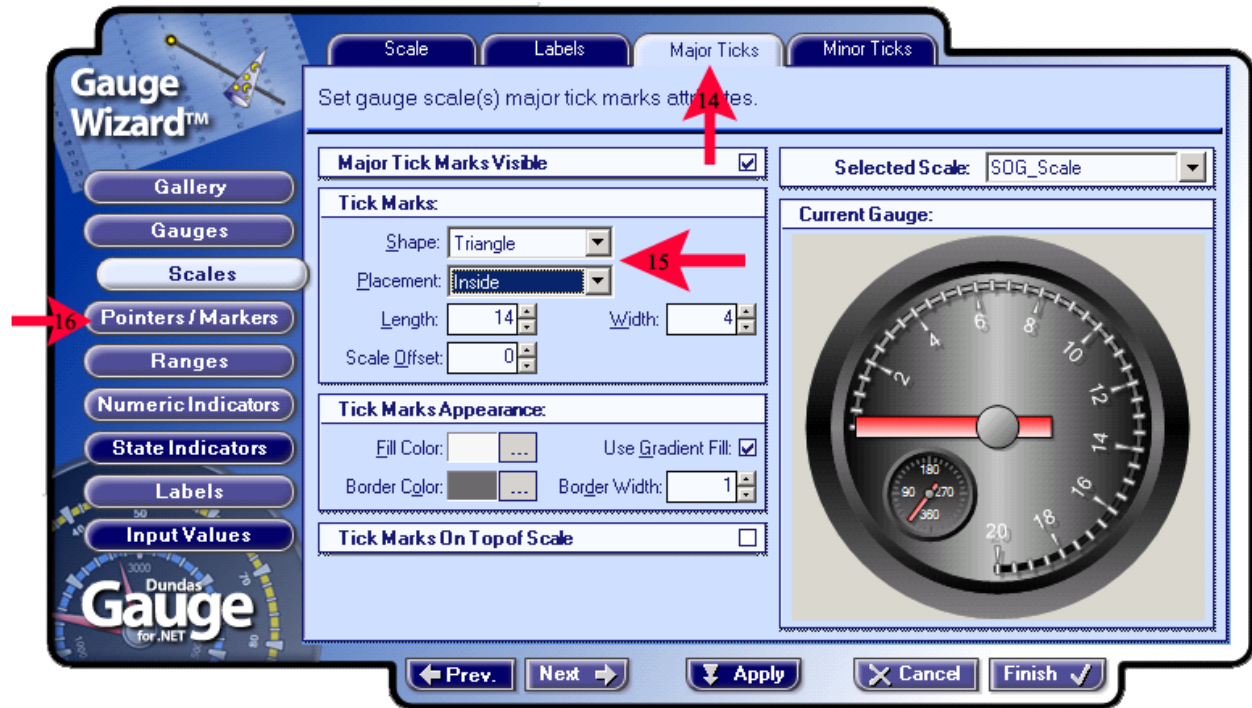
You'll notice that the COG scale is unreadable. There's too much information to force into such a small gauge. Let's modify the tick's interval so we can get some idea of where the pointer is actually pointing!

- 13) Simple select the *COG_Scale*, and change the interval to a Custom value – how bout '90'?

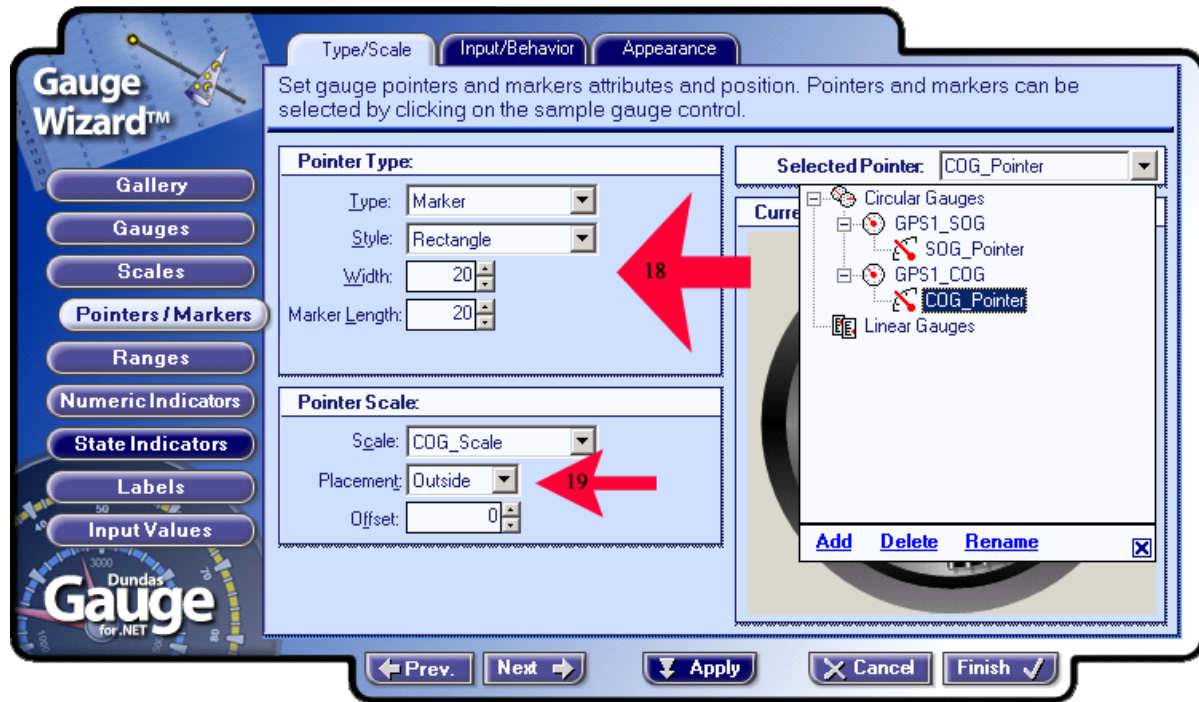


Once you've setup the initial scales, let's move on to modification of the ticks.

- 14) Click on the *Major Ticks* tab located at the top of the form.
- 15) Select the *SOG_Scale* and change the *Shape* and *Placement* to *Triangle* and *Inside*.
- 16) Let's continue on to *Pointers / Markers* by clicking on the toolbar.



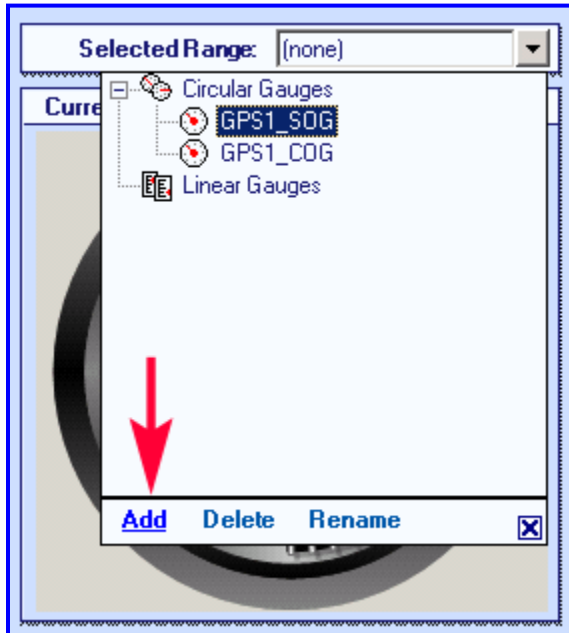
- 17) Open the *Selected Pointer* dropdown list and name the pointers *SOG_Pointer* and *COG_Pointer* the same as we renamed the gauges and scales.
- 18) Select the *COG_Pointer* and set the *Pointer Type* to *Marker*, set the style to *Rectangle*, with a *Width* and *Length* of '20.'
- 19) Set the *COG_Pointer Placement* to *Outside*.



20) Select the *SOG_Pointer* and set the *Pointer Type Style* to *NeedleStyle1*.



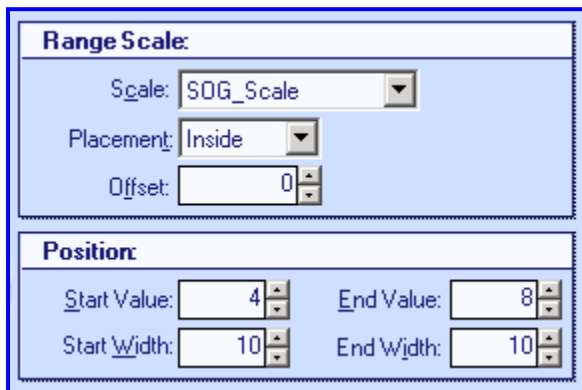
A screenshot of a configuration window titled "Pointer Type:". It contains two dropdown menus. The first, labeled "Type:", has "Needle" selected. The second, labeled "Style:", has "NeedleStyle1" selected.



A screenshot of a configuration window showing a list of gauges. The "Selected Range:" dropdown at the top is set to "[none]". Below it, a list of gauges is shown, including "Circular Gauges" and "Linear Gauges". Under "Circular Gauges", "GPS1_SOG" and "GPS1_COG" are listed. A red arrow points to the "Add" button at the bottom of the list. The "Add", "Delete", and "Rename" buttons are visible at the bottom of the dialog.

Feel free to play around and create your own personalized pointers. You might also want to look at the *Appearance* tab at the top of the form. This allows for even more customization of individual pointers. When you've finished, click on the *Ranges* button to continue to the next section.

There are no ranges on your gauge by default. This is because the gauge we selected in the first step didn't have one. However, you can add/remove all these objects to fully customize your gauge. In this step, we will have to add a new range to our gauge.



A screenshot of a configuration window with two sections. The top section, "Range Scale:", has a "Scale:" dropdown set to "SOG_Scale", a "Placement:" dropdown set to "Inside", and an "Offset:" spinner set to 0. The bottom section, "Position:", has four spinners: "Start Value:" set to 4, "End Value:" set to 8, "Start Width:" set to 10, and "End Width:" set to 10.

Select the drop down list of *Selected Ranges*. Note that only our two circular gauges are display, no ranges exist. Select the *GPS1_SOG* circular gauge and then click on *Add*.

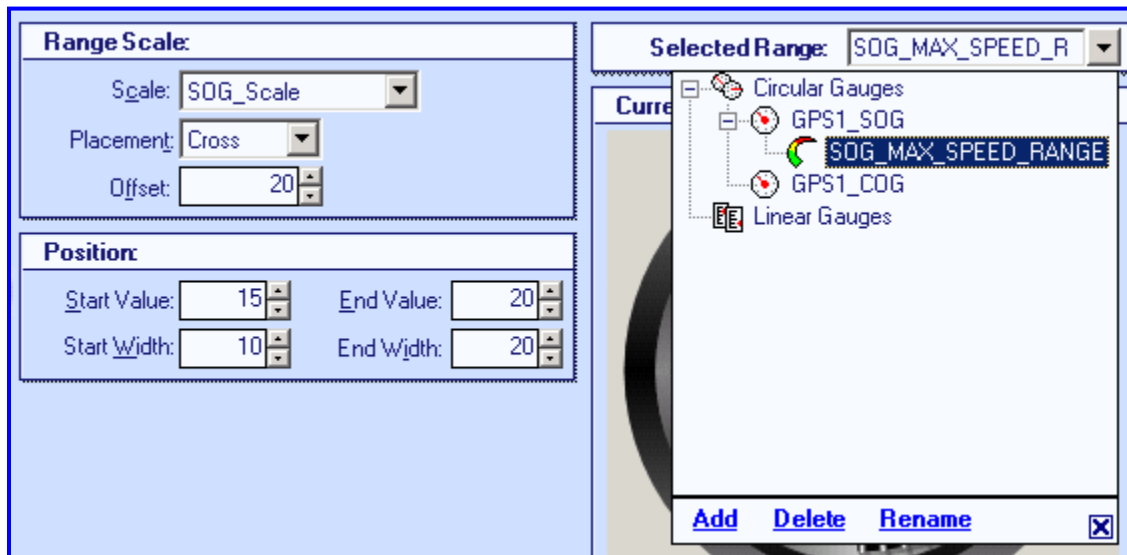
Rename the new *Range* to *SOG_MAX_SPEED_RANGE*

Set the *Placement* to *Cross*.

Set the *Offset* to '20'.

Set *Start/End* values to '15'/'20'

Set *Start/End* widths to '10'/'20'



Let's add an additional *Range*, this one can be used for ensuring we're at a proper speed when conducting Trawl operations. Click *Add* once again. Rename the new range *SOG_TRAWL_RANGE*.

Set its values as noted in the picture.

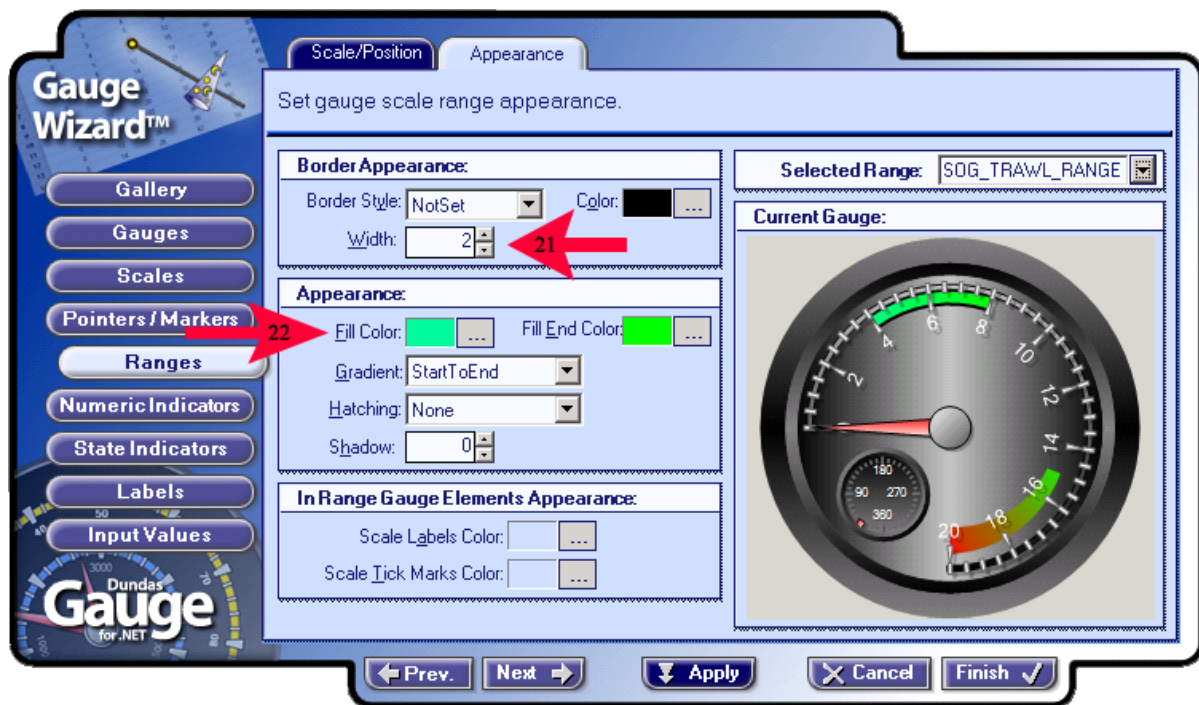
Now let's make their appearance more representative of their purpose...

Ensure that you have the *SOG_TRAWL_RANGE* selected.

21) Set the *Border Appearance Width* to '2'.

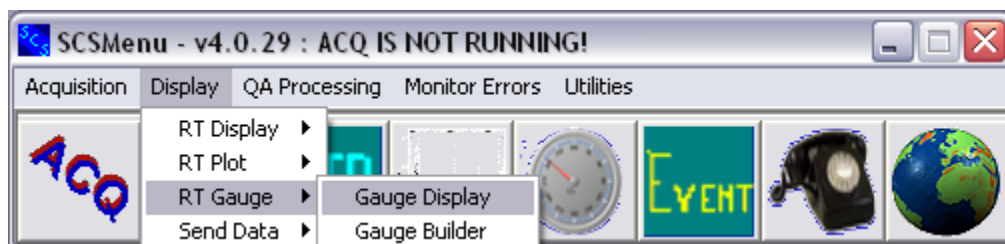
22) Change the *Fill Color* to RGB values (0, 250, 154) and the *Fill End Color* to RGB (0, 255, 0). The *Range* should be totally green at this point.

23) Select the *SOG_MAX_SPEED_RANGE* and set it's *Border Width* to '0'



Hit 'Finish' and you will be prompted to select sensor values for each of the gauges. Select the appropriate sensor for each pointer. After this, you should have a new template with your new gauge on it. Note: Labels will automatically be placed on the gauges, however, they are often hard to see and it is recommended that you modify the size/color/font of the labels as necessary.

6.9 Gauge Display



Function

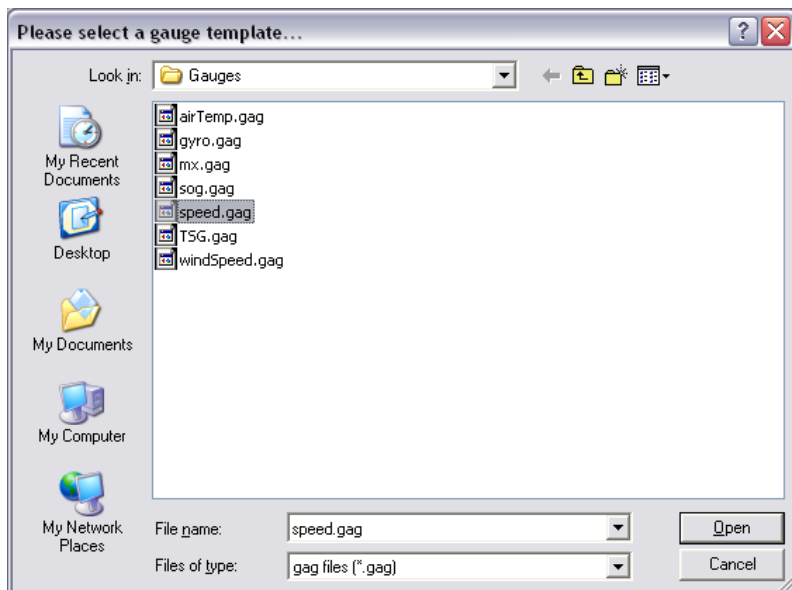
The Gauge Display program displays data that is being acquired or derived in real time from various sensors. The data is displayed in the format of a physical gauge on the computer screen. As well as displaying real time data, the gauge can record history, display minimum or maximum values, ranges, etc. Please consult the Gauge Builder software to get more details on the breadth of functionality provided by gauges.

The different characteristics of the gauge - sensor type and Id, history intervals, gauge shape, range, size, color, screen location, etc - are defined in the *.gag template files, which is built by the Gauge Builder.

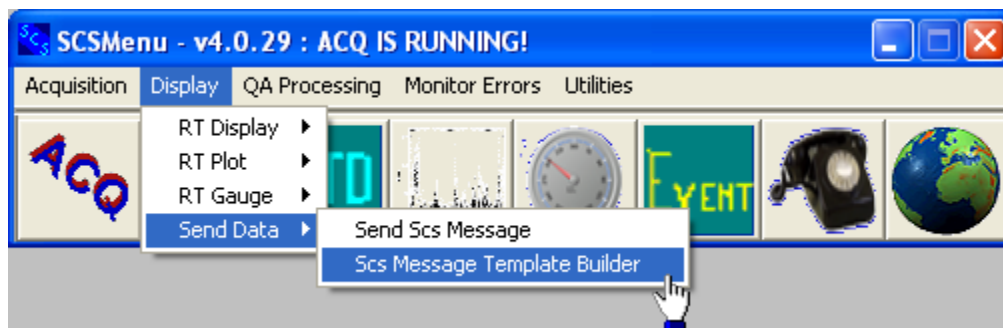
Procedure

When the 'Gauge Display' is selected, the “Gauge Selection template Files” dialog box prompts the user to choose a gauge template file. When a file is selected, click *Open* to display the gauge(s).

The gauge can take the form of either a bar or a dial. The gauge immediately begins displaying real-time data as specified in the *.gag template file.



6.10 SCS Message Template Builder



Function

SCS provides the ability to transmit collected sensor data in user defined formats over serial communication ports or network ports. The user defines the formatted message and data to be sent with the SCS MESSAGE BUILDER and then runs the SEND SCS MESSAGE program to start transmitting the message.

The template builder program builds an SCS Message that may be sent out to a serial communication port or a network TCP/IP port. The SCS Message may contain Alpha-Numeric Characters, Control Characters, SCS Sensor Information, and Time Stamps. You can also specify the rate at which the message will be sent, or set up a polling protocol so that will Send SCS Message will listen for a prompt to order a message be

NOTE: SCS 4.0 Template Builder applications are not backwards compatible with templates built under version 3.3 and before. All templates must be built from a fresh start.

[illegible]

SendSCSMessageBuilder - v4.0.34

File Help

- New
- Open...
- Save
- Save As...
- Exit

Builder Message Configuration

Component Type	Value

This allows the user to start building a new SCS message. This is the default condition when the program is launched.

This allows the user to open any previously saved Send Message template file. The standard Windows Open File dialog is used. Select the desired template file and hit the OPEN button. The builder will reflect the settings encapsulated in the file.

This allows the user to save the current message settings to a template file. A standard Windows Save/Save As dialog pops up, prompting the user to specify a file name for the current settings (enter the file name without an extension or directory name). All files are placed into the subdirectory Template\SENDSCS under the SCSSHIP logical directory.

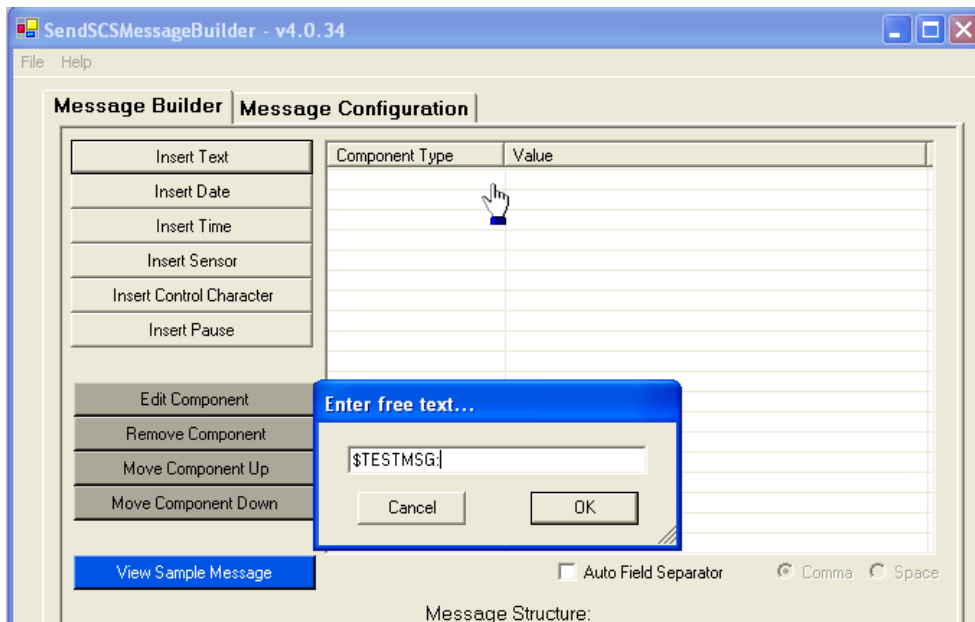
This exits the user from the SCS Message Builder program.

[illegible]

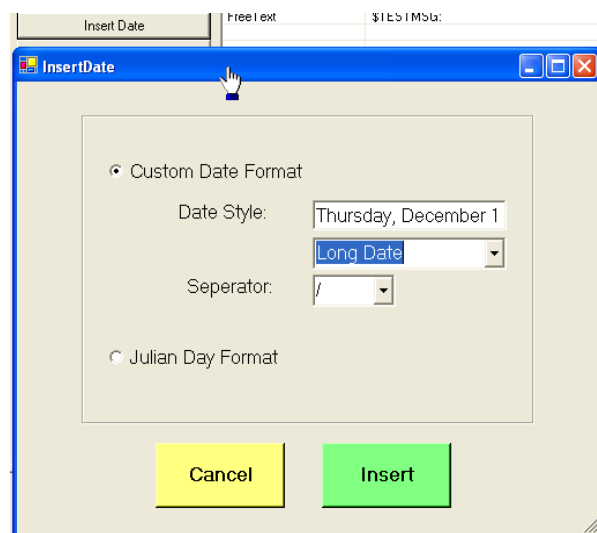
The Message builder TAB is used to build the message that will be output. The message is built by sequentially adding items to the message from left to right. Each item is called a component in this application. As components are added the message structure develops on the bottom of the window.

The Buttons on the left of the window describe the components that may be added to the message. The right of the window shows a grid where the user builds the components in sequence. A description of the Message builder component buttons follows:

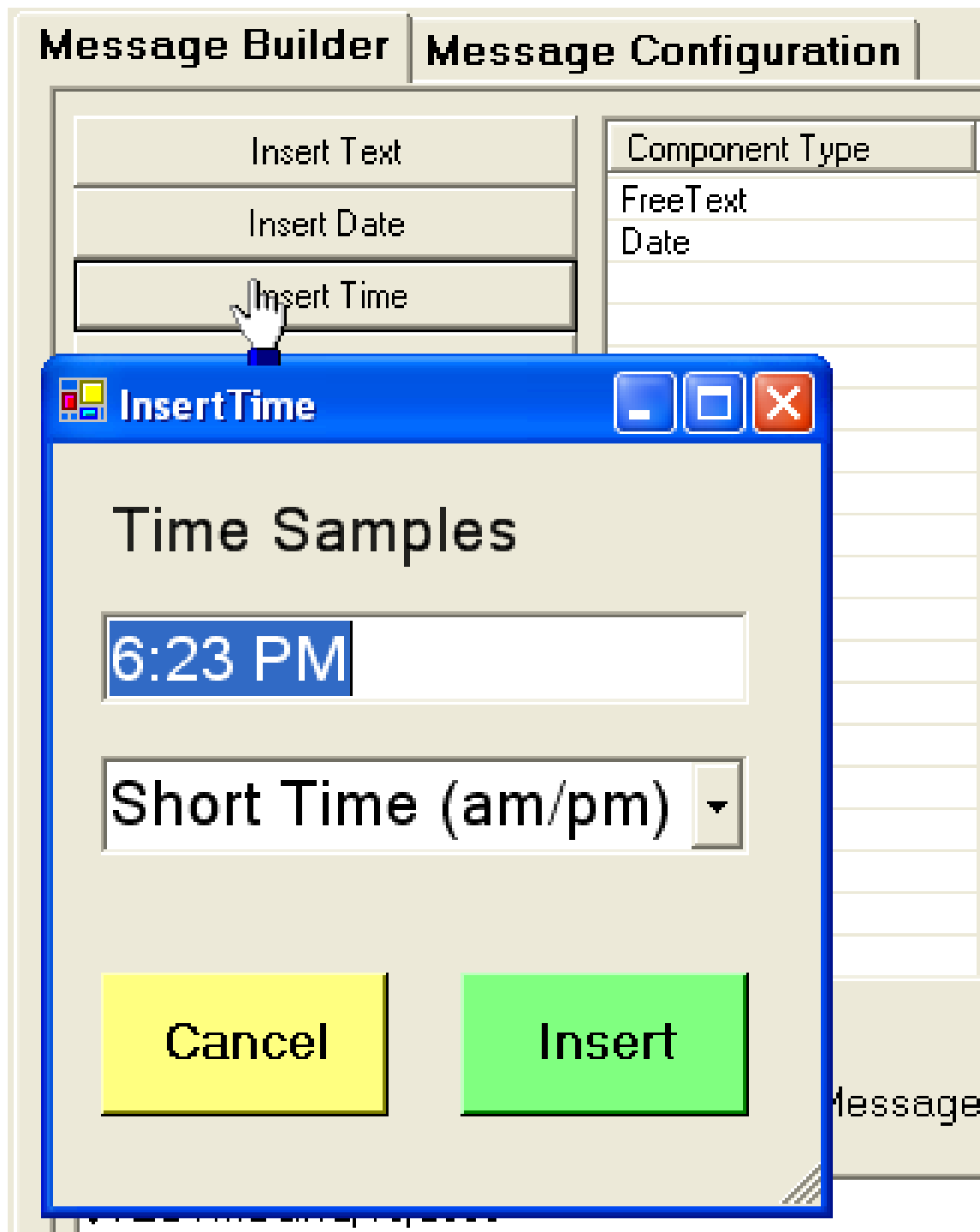
- **INSERT TEXT:** The user may enter any text or punctuation as a component



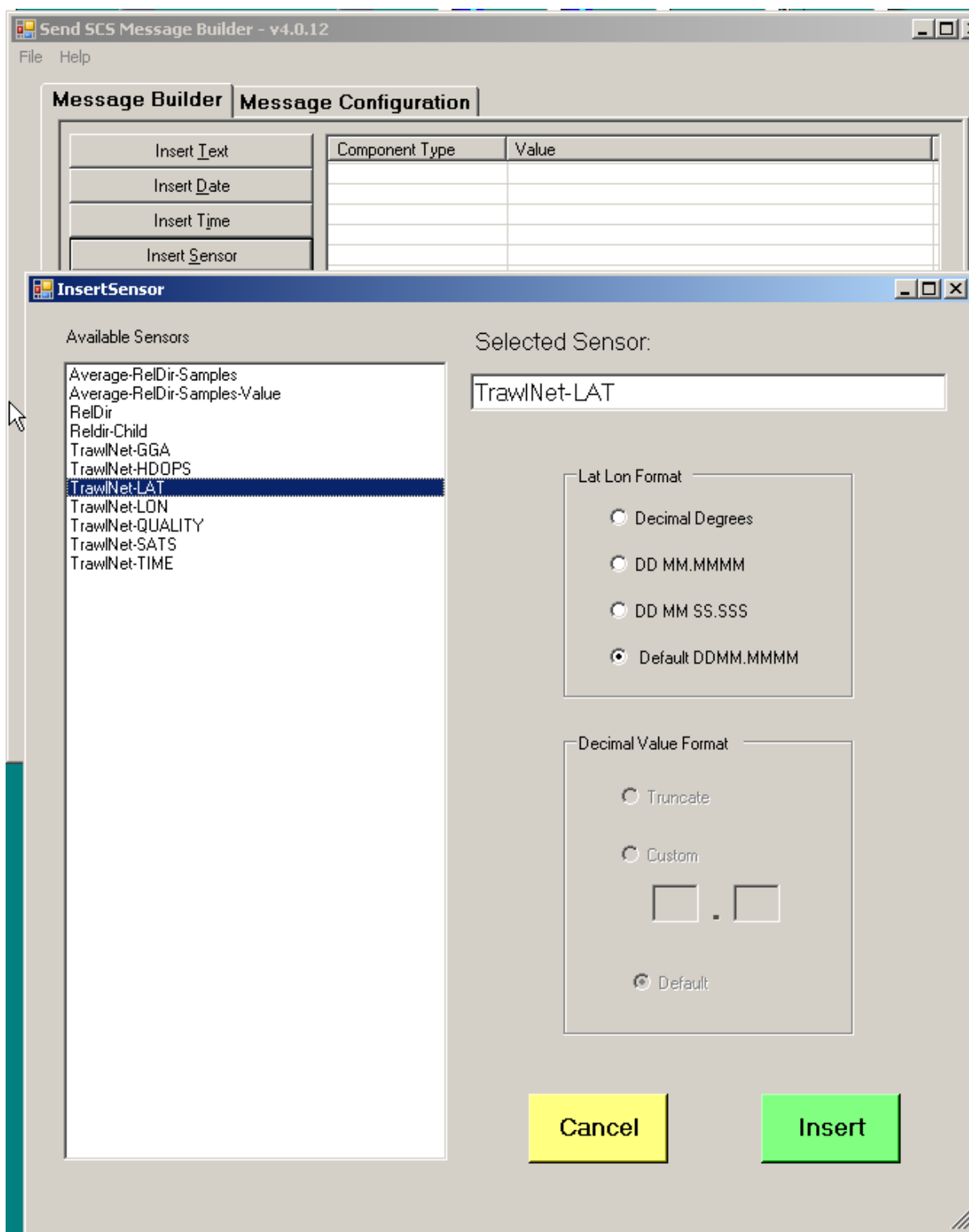
- **INSERT DATE:** The user may select from multiple formats for dates to be added as a component



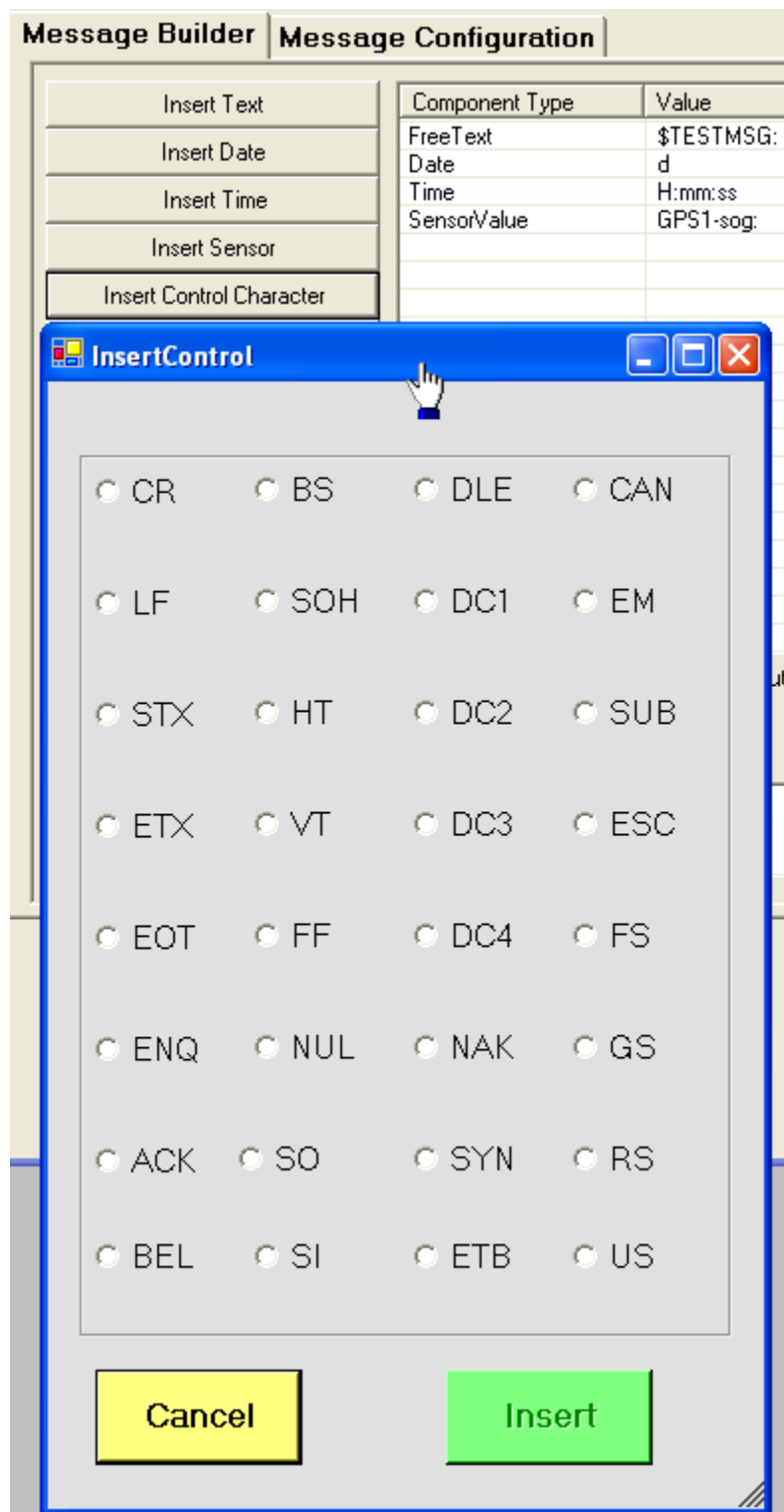
- INSERT TIME: The user may select from multiple formats for time to be added as a component



- **INSERT SENSOR:** the user may select an SCS Child Sensor whose value will be built into the message in real time. Sensors declared as NAVSEC decode types in CFE can be formatted in several different lat/lon formats.



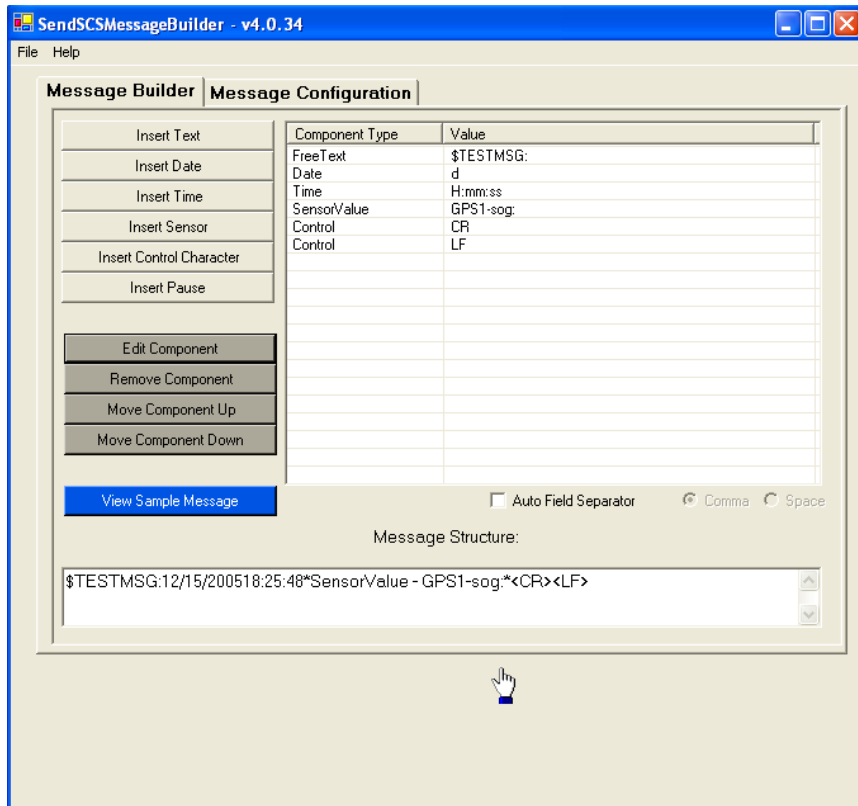
- **INSERT CONTROL CHARACTER:** The user may enter from a list of control characters



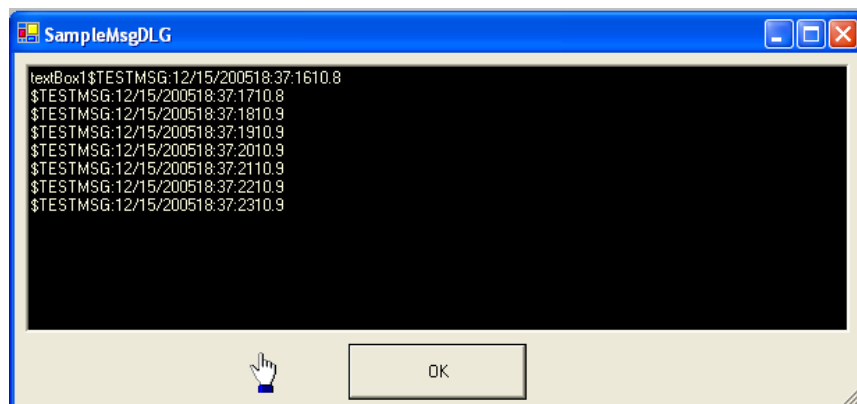
- **INSERT PAUSE:** the user may insert a delay in the transmission of part of the message. **NOTE:** Not supported in this release
- **Auto Field Separator:** the user may toggle this box to build messages with comma or space delimited components without building separate text components for punctuation.

Message Configuration	
Insert Text	
Insert Date	
Insert Time	
Insert Sensor	
Insert Control Character	
Insert Pause	
Edit Component	
Remove Component	
Move Component Up	
Move Component Down	
View Sample Message	<input checked="" type="checkbox"/> Auto Field Separator <input type="radio"/> Comma <input type="radio"/> Space
Message Structure: <pre>\$TESTMSG:12/15/200518:25:48*SensorValue - GPS1-sog:*<CR><LF></pre>	

When the users have built all of the required components the message format is complete. The following example is an example of a complete message format. Note the control characters at the end generally required by receiving programs.



- **EDIT COMPONENT:** the user selects the component to EDIT and then presses this button.
- **REMOVE COMPONENT:** the user selects the component to REMOVE and then presses this button.
- **MOVE COMPONENT UP:** the user selects the component to MOVE UP in the sequence and then presses this button.
- **MOVE COMPONENT DOWN:** the user selects the component to MOVE DOWN in the sequence and then presses this button.
- **VIEW SAMPLE MESSAGE:** The user may review the message in a dialog window in real-time



6.10.4.1.1 Message Configuration TAB Interface

The screenshot shows the 'Message Configuration' tab of the 'Send SCS Message Builder - v41.6.1' application. The window has a menu bar with 'File' and 'Help'. The main area is divided into several sections:

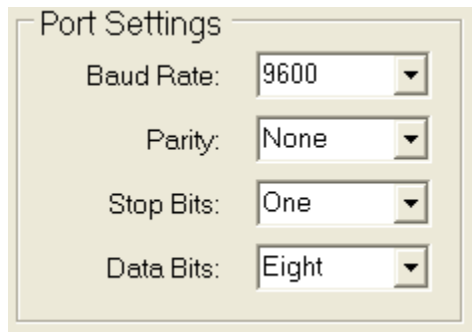
- Select Output Port:** Contains radio buttons for 'COM Port', 'TCP Socket', 'UDP Socket', 'UDP Broadcast' (selected), and 'UDP Multicast'. To the right are text boxes for 'Port Number' (containing '23457') and 'Peer IP / MCast Group'.
- Select Message Type:** Contains radio buttons for 'Updated' and 'Polled' (selected).
- Set Update Rate:** Contains three spin boxes for 'Hours' (0), 'Minutes' (0), and 'Seconds' (3).
- Port Settings:** Contains dropdown menus for 'Baud Rate' (9600), 'Parity' (None), 'Stop Bits' (One), and 'Data Bits' (Eight).
- Polled Command:** Contains a text box and a green 'Insert Control' button.

The message Configuration window is used to configure the physical characteristics of the output port where the message will be transmitted. The window is broken down into 5 parts. Only the appropriate selections are highlighted for both serial communication and network ports.

This close-up shows the 'Select Output Port' section. A mouse cursor is pointing at the 'COM Port' radio button, which is selected. Below it is the 'TCP Socket' radio button. To the right is a 'Port Number' text box containing 'COM1'.

The user specifies if the message will be transmitted via a Serial Com Port or a TCP socket. Based on this selection other setting will be highlighted. If COM is Selected the user types the com port of the form COMx where x is the number of the com port. If TCP is selected the user types the number of the socket port that network clients will connect.

NOTE: ports above 1500 are recommended so as not to interfere with well know ports and the ACQ port 505.



Port Settings

Baud Rate: 9600

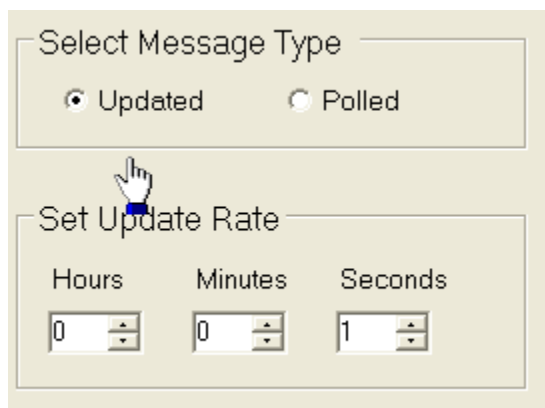
Parity: None

Stop Bits: One

Data Bits: Eight

This dialog box contains four dropdown menus for configuring serial communication settings. The Baud Rate is set to 9600, Parity to None, Stop Bits to One, and Data Bits to Eight.

If a COM port was specified the user sets the appropriate serial communication settings.



Select Message Type

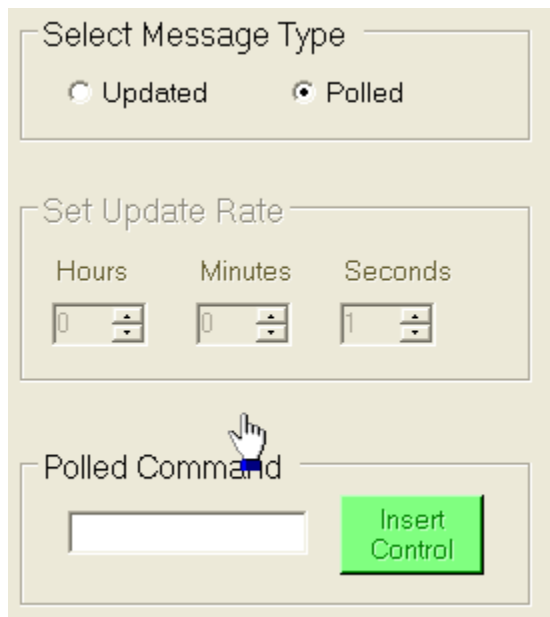
☒ Updated ☐ Polled

Set Update Rate

Hours: 0 Minutes: 0 Seconds: 1

This image shows two stacked dialog boxes. The top one, 'Select Message Type', has two radio buttons: 'Updated' (selected) and 'Polled'. The bottom one, 'Set Update Rate', has three spin boxes for 'Hours' (0), 'Minutes' (0), and 'Seconds' (1). A mouse cursor is pointing at the 'Set Update Rate' dialog box.

If updated messages are selected the user is required to set the output interval that the message will be sent.



Select Message Type

☐ Updated ☒ Polled

Set Update Rate

Hours: 0 Minutes: 0 Seconds: 1

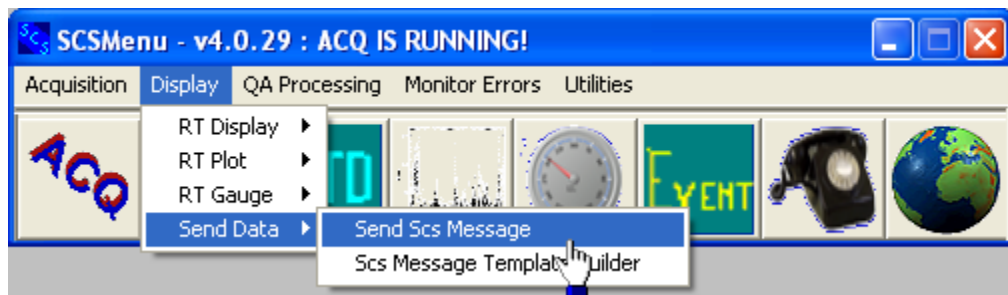
Polled Command

Insert Control

This image shows three stacked dialog boxes. The top one, 'Select Message Type', has two radio buttons: 'Updated' and 'Polled' (selected). The middle one, 'Set Update Rate', has three spin boxes for 'Hours' (0), 'Minutes' (0), and 'Seconds' (1). The bottom one, 'Polled Command', has a text input field and a green 'Insert Control' button. A mouse cursor is pointing at the 'Polled Command' dialog box.

If Polled messages are selected the user is required to set the Polling Command that the program will listen for prior to sending a message.

6.11 Send SCS Message Manager



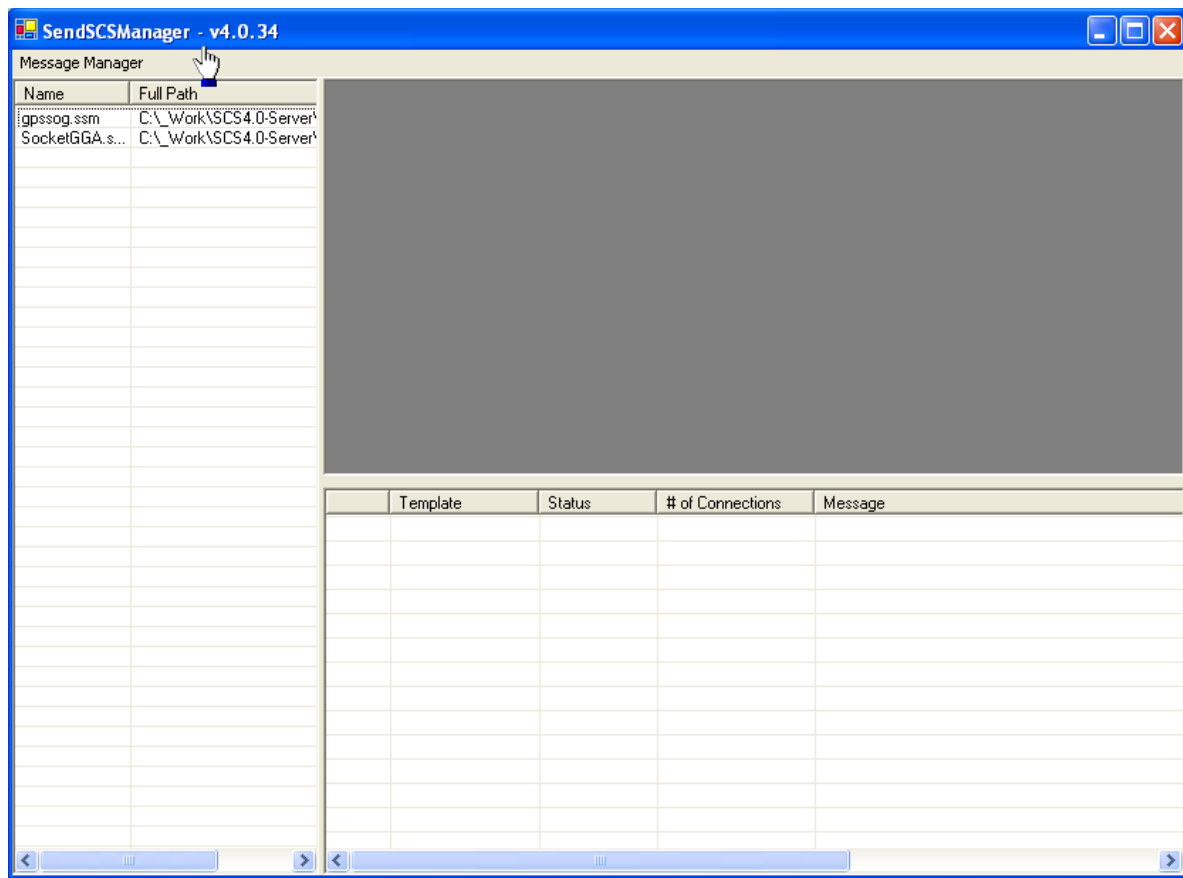
Function

This program controls the sending of all messages from a given computer. It does not control messages on other computers on the vessel. Messages are constructed using the Message Builder app (previous section).

The Message Manager lists the possible messages that may be sent, activates those messages when they are double clicked and provides a status window for all activated SCS messages.

The Message Manager GUI

Here is the Message Manager's main window immediately after the app is launched.



All template file names are displayed on the left side of the window. Until a message is activated the other two portions of the window are blank. Select and double click on the message to be transmitted and a Message Sender Window is activated in the upper right portion of the window. This window displays all of the characteristics of the message to be sent and prominently displays the START SENDING Button and the STOP SENDING Button.

Here is a sample message sender window:

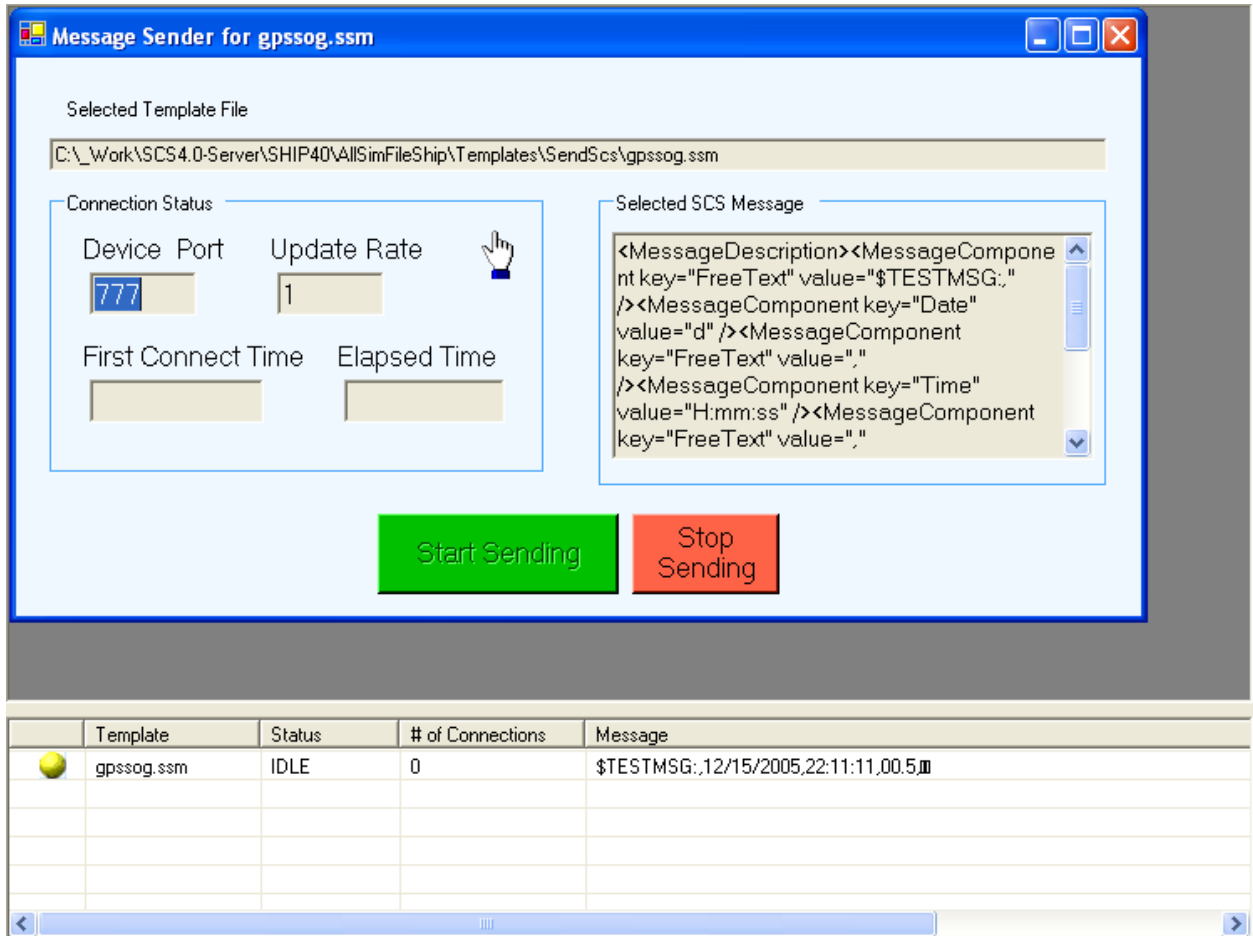
The screenshot shows a Windows-style application window titled "Message Sender for gpssog.ssm". The window has a light blue background and a blue title bar with standard Windows controls (minimize, maximize, close). The main content area is divided into several sections:

- Selected Template File:** A text box at the top containing the path "C:_Work\SCS4.0-Server\SHIP40\AllSimFileShip\Templates\SendScs\gpssog.ssm".
- Connection Status:** A section on the left with a blue border. It contains:
 - Labels "Device", "Port", and "Update Rate" above input fields.
 - The "Device" field contains "777" and the "Update Rate" field contains "1".
 - Labels "First Connect Time" and "Elapsed Time" above empty input fields.
 - A hand cursor icon is positioned over the "Update Rate" field.
- Selected SCS Message:** A section on the right with a blue border. It contains a text box with XML-like message data:

```
<MessageDescription><MessageCompone  
nt key="FreeText" value="$TESTMSG;"  
/><MessageComponent key="Date"  
value="d" /><MessageComponent  
key="FreeText" value=","  
/><MessageComponent key="Time"  
value="H:mm:ss" /><MessageComponent  
key="FreeText" value=","
```
- Buttons:** At the bottom center, there are two buttons: a green "Start Sending" button and a red "Stop Sending" button.

Press the *START SENDING Button* to start the message transmitting. Once the start button is pressed the status portion in the lower right updates to show that the message is being sent and provides real time status of the message.

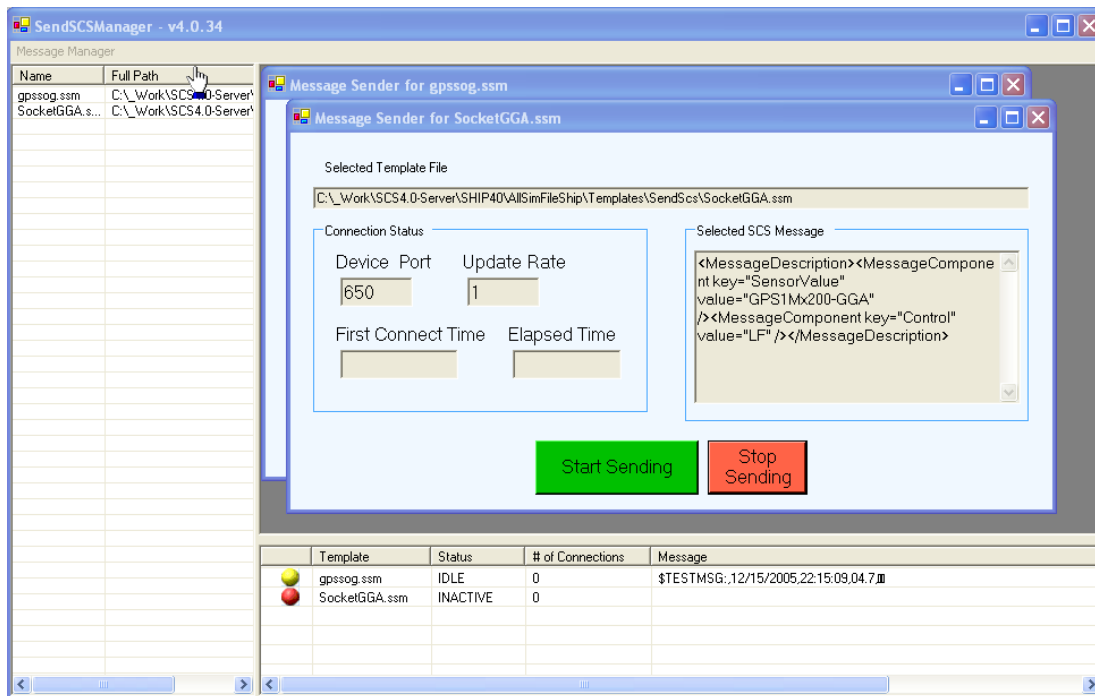
The figure below shows the status display for a message that has just been started.



The Icon in the first column changes color in the standard way; Red means “has never been sent”; Yellow means “has been sent at least once” ; The LED flashes to green every time a new message has been sent. The normal color of the LED should be yellow, with periodic flashes to green in accordance with the message’s assigned update rate.

The Status column normally shows IDLE (equivalent to a yellow LED), but whenever a message is sent the status flashes to ACTIVE and the LED changes to green for a quarter-second.

Additional messages may be activated at any time; new Message Sender windows will be activated and the message will be added to the list in the status window. Here is the display when two messages are being sent.



Transmission of messages may be tested as follows.

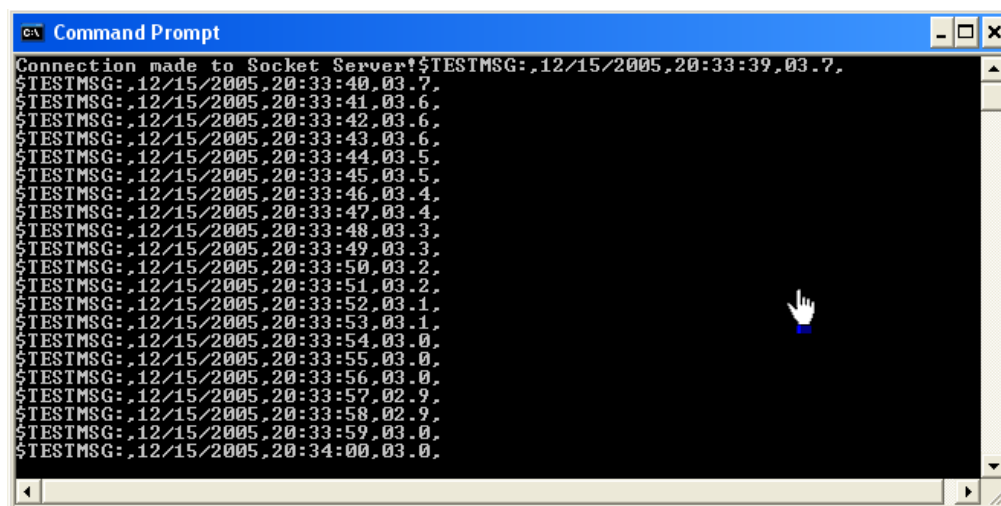
For serial messages the Hyperterm app or equivalent may be used to monitor the destination COM port.

For TCP socket messages, you can receive the message at any computer on your LAN. Open up a command prompt window (Start button on desktop, select “Run...” and enter “cmd” in the text box).

At the command prompt the command is “telnet”, followed by the IP address of the machine sending the message, followed by the port number being used. The three parts are separated by spaces. A sample command is

```
telnet 127.0.0.1 650
```

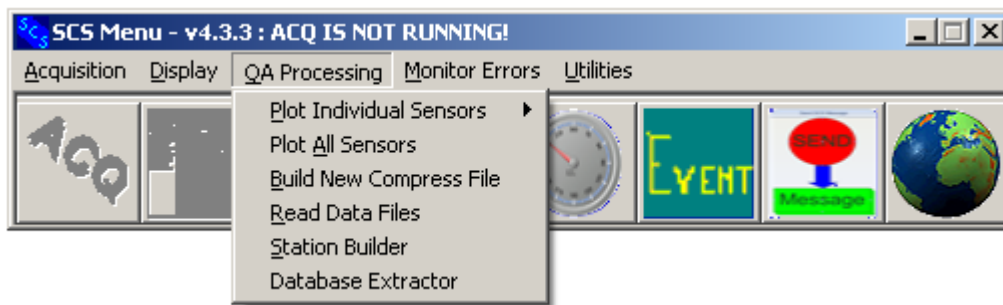
A sample output Message is shown below:



CHAPTER SEVEN

QUALITY ASSURANCE PROCESSING

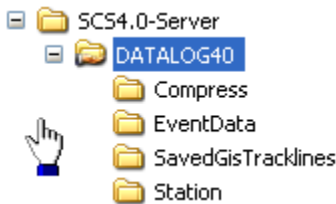
7 Quality Assurance Processing



This section discusses how to compress raw data files created during data acquisition into separate ASCII Comma (.ACO) data files and how to plot the Raw or Compressed data. It also discusses how to read RAW and compressed data files and build station files.

7.1 Files Used for QA Functions

All data files are found inside the DataLog folder for ease of management.



The DataLog folder is typically located on the hard disk partition that contains the most available free disk space. The SCS system manager may choose to do one of the following:

- 1) Share the entire DataLog directory (and all subsequent sub-directories) located on the primary Acquisition server. In this case, the system manager must be completely sure that all of these directories/files under DataLog are secure and cannot be changed and/or deleted without Administrator privilege.
- 2) The administrator may enable secondary logging on a remote machine in which case all raw data files will be written simultaneously to both locations. This alleviates the need for constant data backups
- 3) If the secondary logger is shared for all post plots and compress work then the primary data set located on the main SCS server will remain undisturbed.
- 4) Backups of primary and secondary systems should be done daily to ensure that in the event of a critical system failure the computer may be restored after the problem has been corrected.

Raw Data Files (*.RAW)

These files may be read and reviewed using the Read Data Files (PFE) program. Each data file is named for its parent sensor name with the ACQ initial time stamp added.

Bathy-Raw_19980106-020646.RAW	287KB	RAW File	1/5/98 6:57 PM
MX200-GPGGA_19980106-020646.RAW	225KB	RAW File	1/5/98 6:57 PM
MX200-GPVTG_19980106-020646.RAW	177KB	RAW File	1/5/98 6:57 PM
PCODE-GPGGA_19980106-020646.RAW	222KB	RAW File	1/5/98 6:57 PM
PCODE-VTG_19980106-020646.RAW	177KB	RAW File	1/5/98 6:57 PM
Raw_Fluorometer_19980106-020646.RAW	332KB	RAW File	1/5/98 6:57 PM
RAW-GYRO_19980106-020646.RAW	114KB	RAW File	1/5/98 6:57 PM
YOUNG-MET-RAW_19980106-020646.RAW	287KB	RAW File	1/5/98 6:57 PM
SEABIRD-SBE21_19980106-020646.RAW	4KB	RAW File	1/5/98 6:57 PM
Mast-TWind_Dir_19980106-020646.RAW	4KB	RAW File	1/5/98 6:57 PM
TSG_Unit_Temp_19980106-020646.RAW	5KB	RAW File	1/5/98 6:57 PM

Every time the Data Logger is restarted, new copies of the RAW data files are created with the appropriate time stamp (as designated by the start time). Each RAW file contains the exact sensor message string coming off the sensor, as well as a standard ASCII GMT time stamp (based on the internal PC clock).

Refer to chapter one for more details on RAW files.

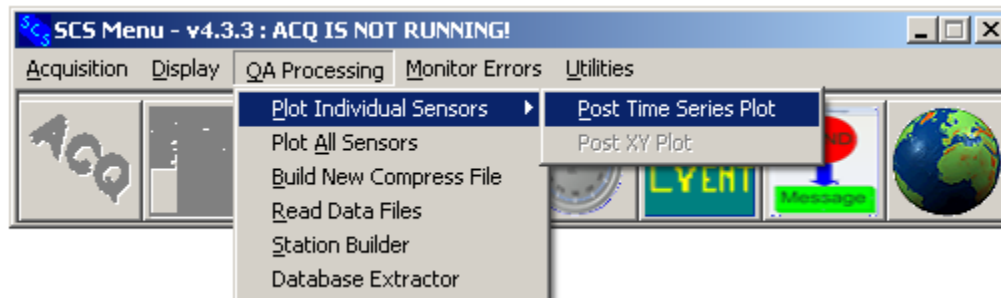
DeviceConfiguration.XML and NewSensor.XML Configuration Files

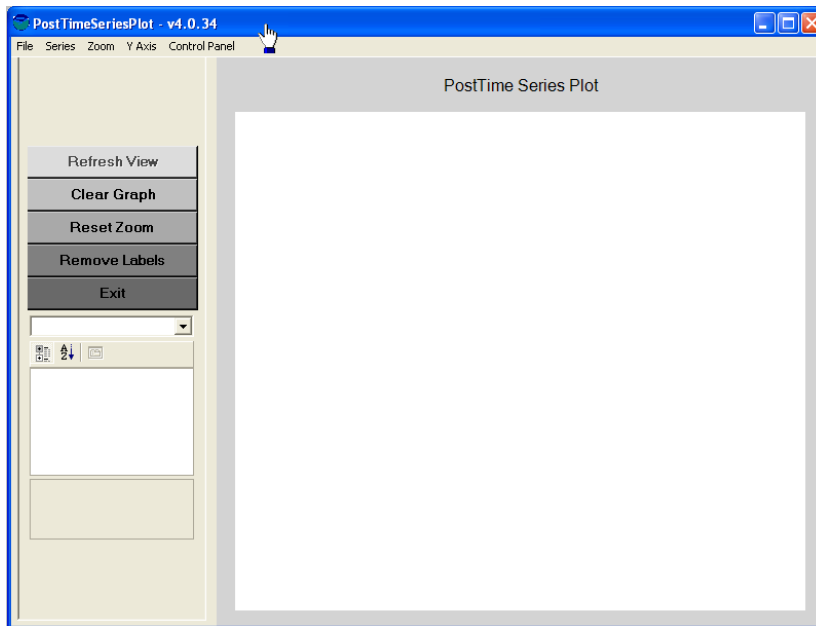
There is a corresponding XML file for every set of raw data files that were collected during an Acquisition session. These files represent in XML format the sensor configuration file and may be viewed with a standard browser.

Plot Template Files

These files are developed by the user using the plotting program to save the key characteristics of the plot so that the plot does not have to be created from scratch over and over. The plot template files generated for the post processing Time Series plot program are used in the automatic plotting of data for QA checks that takes place on a daily basis. These plot template saved settings files are found in the SHIP/Template/RealTimeSeriesPlot or the SHIP/Template/PostTimeSeriesPlot Templates directory under the SCSSHIP logical directory. The files are described in XML and it is not recommended that the user edit these files.

7.2 Post Time Series Plot





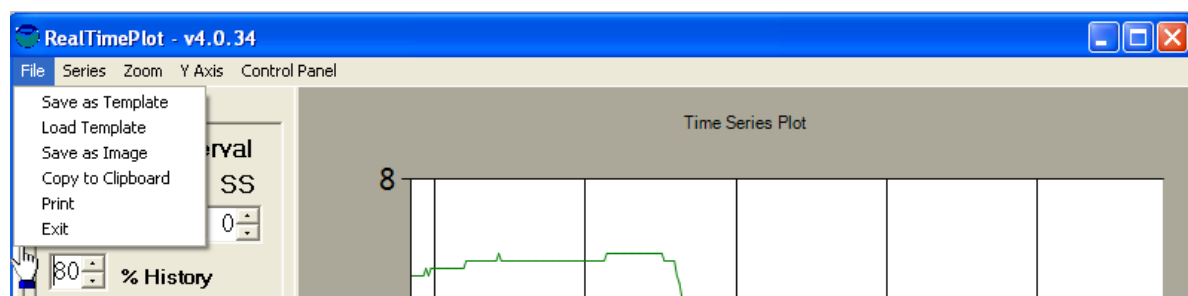
Function

This program allows the user to plot multiple sensor data time series in graphical format after the data has been collected. These graphs may be easily manipulated by the user through such functions as zooming, scrolling, setting graph attributes and getting values of specific data points. The plot program automatically adjusts the amount of data displayed and the update rate of the plot based on the amount of data in the plot. The program provides an option for a dual axis plot where each time series may be associated with either axis.

Interface Options

When the program is activated it shows a functions menu across the top of the window, a control panel on the left, and a blank drawing area in the body of the window.

7.2.1 File Menu Interface

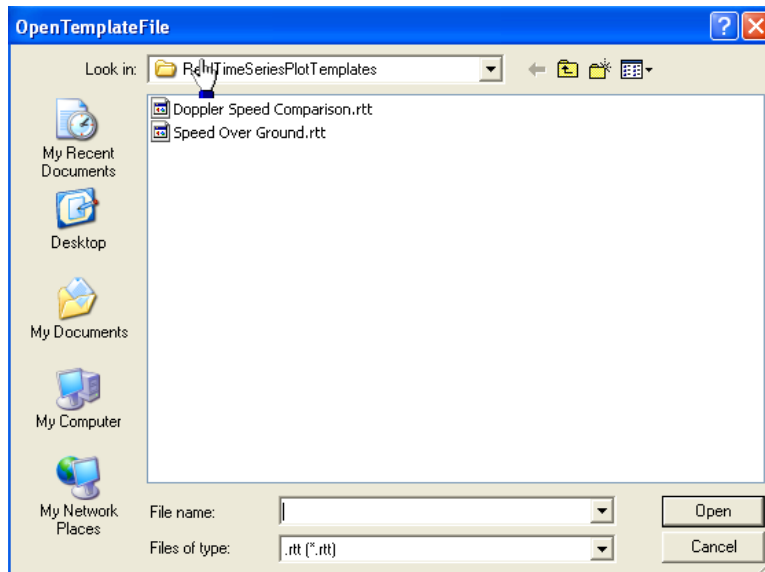


7.2.1.1.1 Save Current Settings

Located under the 'FILE' menu, this allows the user to save the current graph settings. This allows the user to recall the settings without having to reconfigure commonly used plots each time this program is run. A box pops up, prompting the user to specify a file name for the current settings (enter the file name without an extension or directory name). All files are placed into the subdirectory Templates/PostTimeSeriesPlotTemplates under the SCSSHIP logical directory. Note that all configurable settings will be saved, including sensor names, line types and colors, screen colors, update rates, etc...

7.2.1.1.2 Load Template

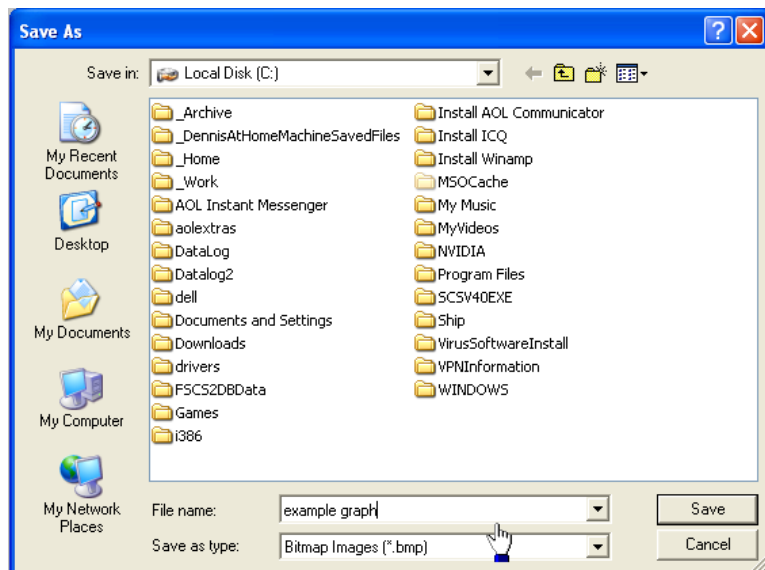
Located under the 'FILE' menu, this allows the user to recall any previously saved graph settings.



Once selected, a list of all of the saved settings files will be given (Templates/PostTimeSeriesPlotTemplates under the SCSSHIP logical directory). Select the desired settings file and hit the OPEN button. The graph will then start plotting the sensors that were specified in the settings file.

7.2.1.1.3 Save As Image

Located under the 'FILE' menu, this allows the user to copy the present graph to the clipboard or to a file as a standard bitmap graphics image.

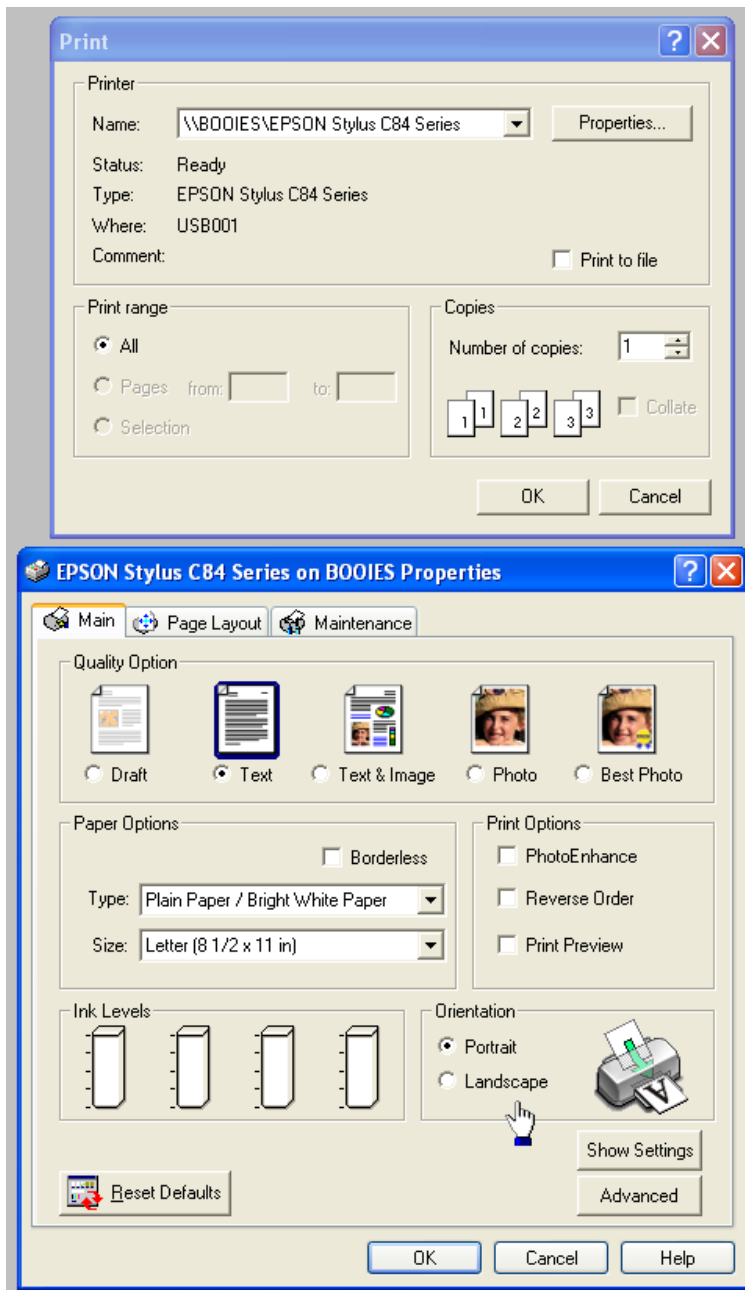


7.2.1.1.4 Copy to Clipboard

Located under the 'FILE' menu, this allows the user to copy the present graph to the clipboard for inclusion in other standard windows applications like MS Word and PowerPoint.

7.2.1.1.5 Print

Located under the 'FILE' menu, this allows the user to print the graph. The standard Windows print and properties dialog is displayed:

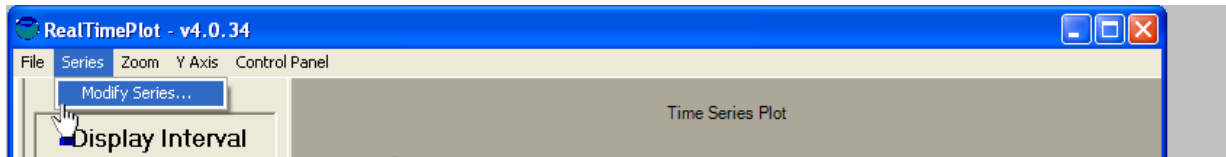


Note: that the default orientation is Portrait.

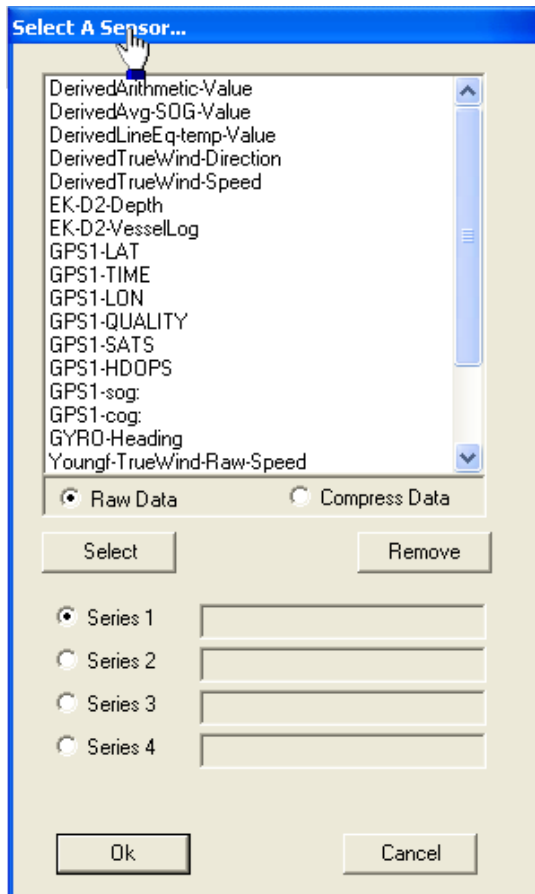
7.2.1.1.6 Exit

Located under the 'FILE' menu, this exits the user from the RT Time Series Plot program.

7.2.2 Series Menu Interface



The user selects the sensors to plot using the Series – Modify Series menu. This menu is also used for changing or adding a sensor time series to the plot.

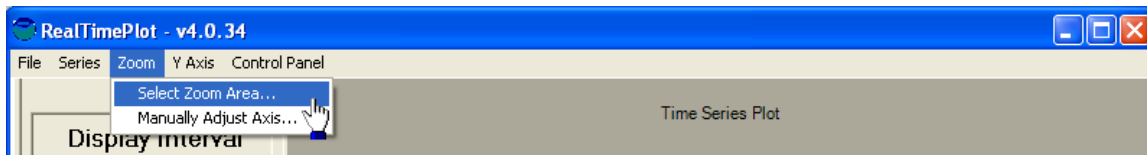


The user is presented with a list of child sensors to plot. The user selects a sensor and hits the SELECT button or simply double-clicks the sensor. The user may also select which time series the user wants to use for plotting the sensor. The series radio button is automatically advanced for convenience in case multiple series are being configured. The user also indicates whether the data should be plotted from the RAW sensor files or from the custom compress files created by the user.

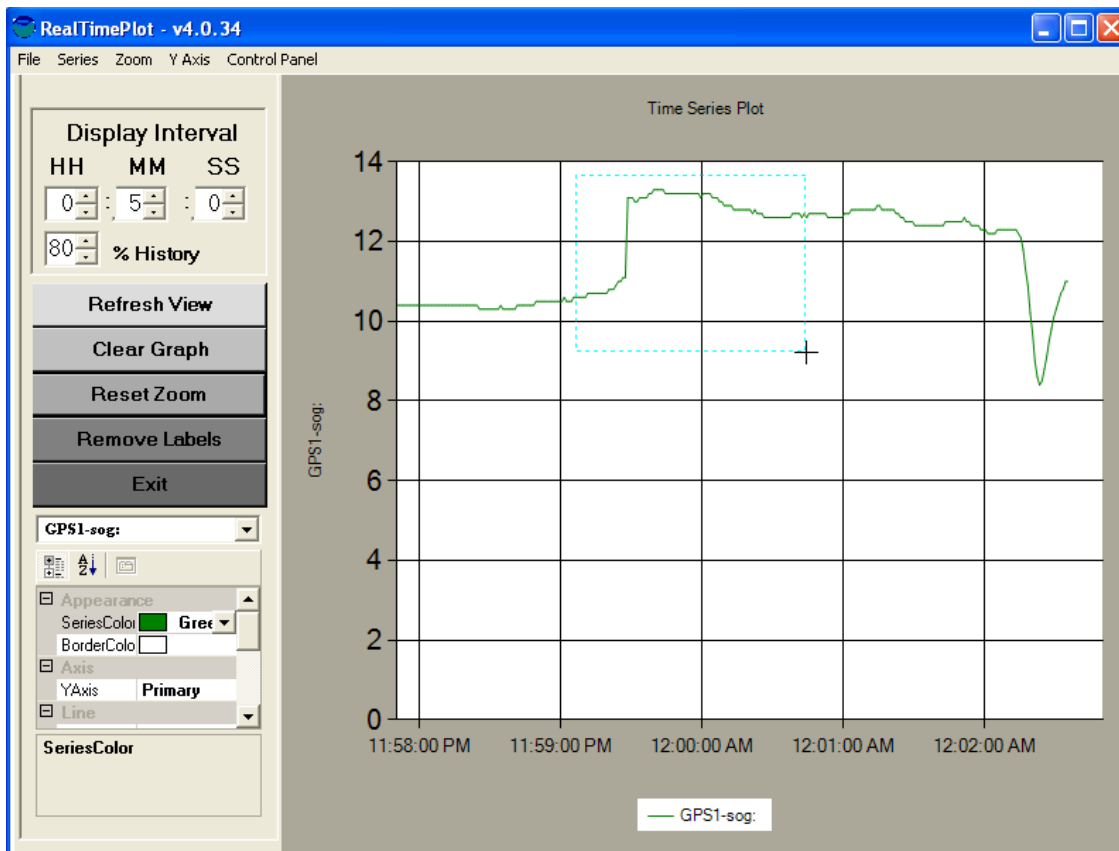
The REMOVE button is used to remove a sensor from a series.

Once you are satisfied with the time configuration press the OK button to start the plot.

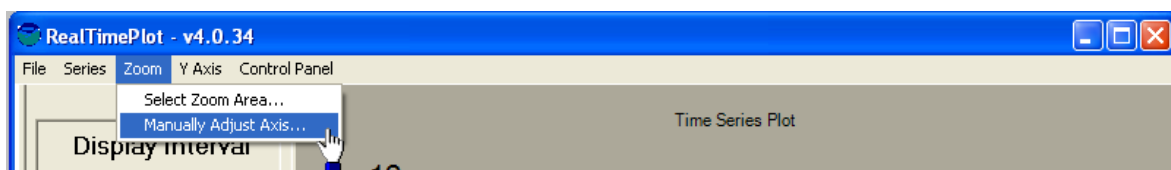
7.2.3 Zoom Menu Interface



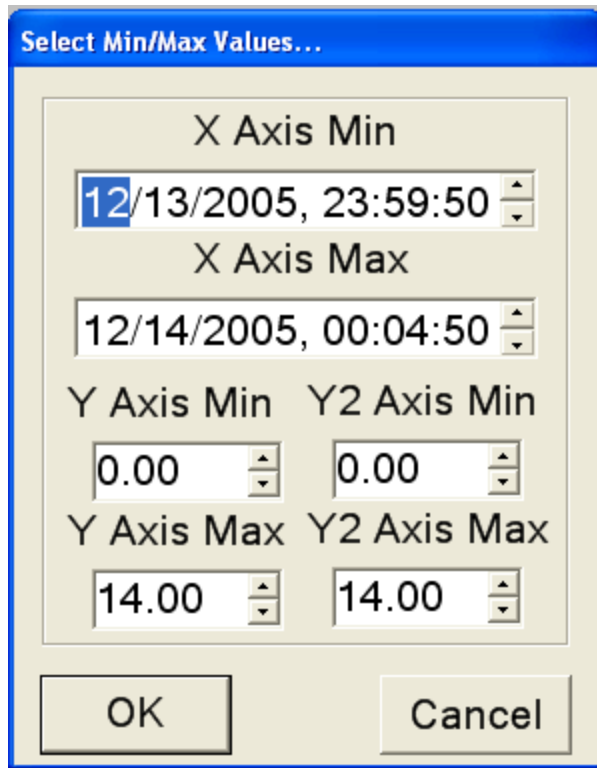
Once the plot has been developed for some period of time the user may be interested in zooming the plot to examine specific features. The user may zoom in on the plot by either drawing a zoom area on the plot by holding the left mouse button down or dragging a box over the interested feature.



Once zoomed in horizontal and vertical scroll bars appear to allow access to all parts of the original plot area. To restore the plot to its full extent use the RESET ZOOM button on the control panel described below.

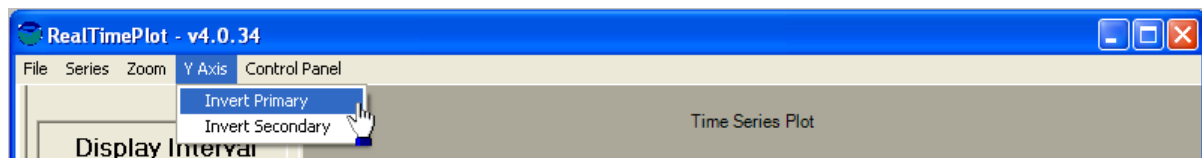


The plot may also be zoomed or sized manually by adjusting the Axis to the time period of interest.



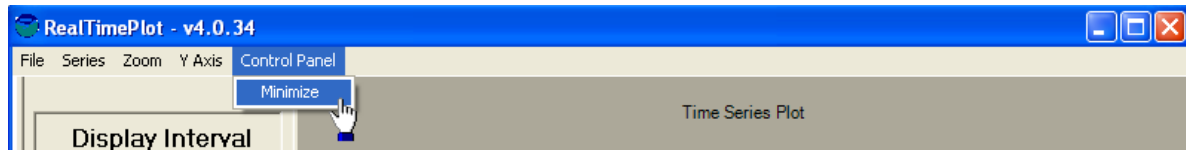
To restore the plot to its full extent use the RESET ZOOM button on the control panel (See Plot Control Panel Discussion for details).

7.2.4 Y Axis Menu Interface



Some plots are best viewed with an inverted Y axis such as plotting sensors versus depth or altitude. This menu is used to toggle the inversion of the primary or secondary Y axis.

7.2.5 Graph Control Panel Menu

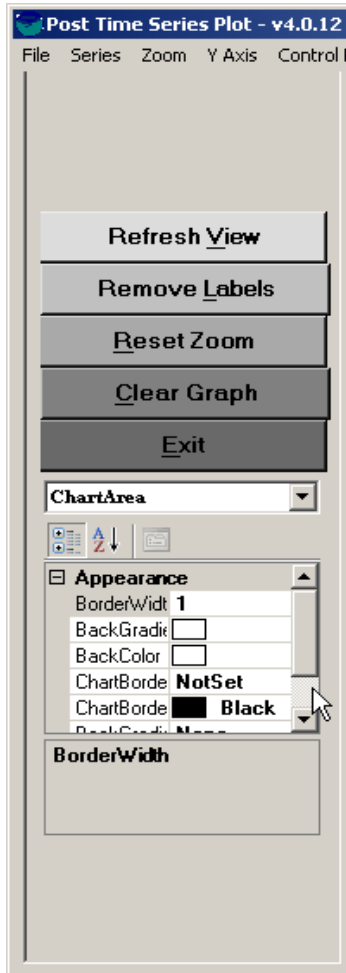


The standard view of all SCS graphs includes a control panel for easily modifying the view of the plots. This control panel may be minimized or activated by the control panel menu bar as shown above. The Graph Control Panel is described in detail below.

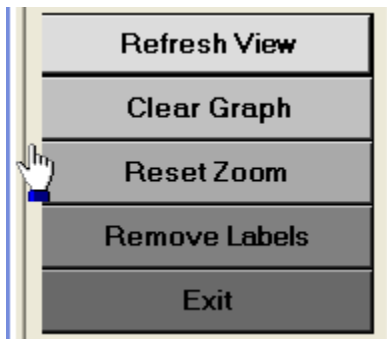
7.2.6 Graph Control Panel

The Graph Control Panel is normally located to the left of the plotting area on all SCS plots. This is the user interface panel used to modify properties of the plot in both data and appearance.

The control panel for post time series plots is almost identical to the one in the real time series plot. The only difference is that the post time panel has no control for display interval. Here is a sample graph control panel:



7.2.6.1.1 Graph Control Panel Buttons



These buttons perform direct actions on the plot and applications.

REFRESH VIEW: Used to repaint the plot after the user has modified properties or the time axis. If the user does not refresh view the properties will not be applied until the next scroll action occurs.

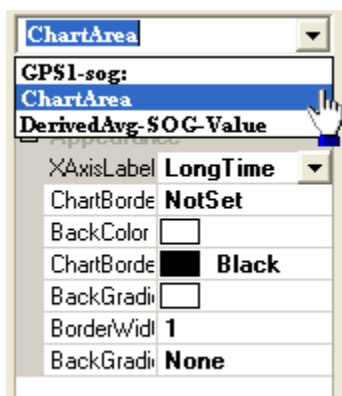
CLEAR GRAPH: This button allows the user to clear the graph and all internally stored data points.

RESET ZOOM: This button is used to reset the plot to full extent after a set of zooming actions.

REMOVE LABELS: This button is used to remove get data point labels from the graph. The user may inquire about any point by right clicking on the time series feature of interest.

EXIT: This button is used to exit the application.

7.2.6.1.2 Graph Control Panel Chart Area Properties

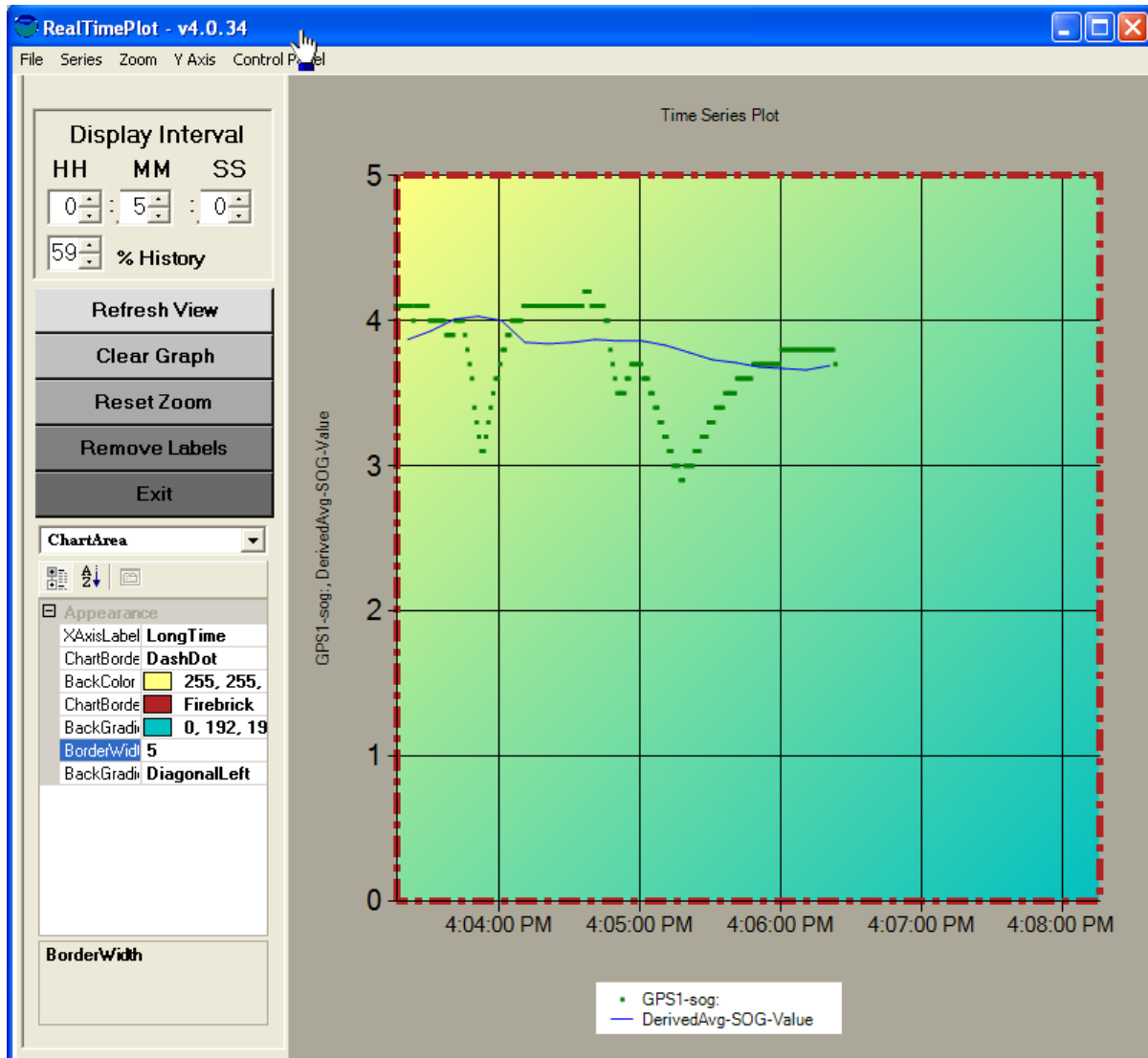


The following Properties may be adjusted by the user by selecting Chart area in the pull down selection box.

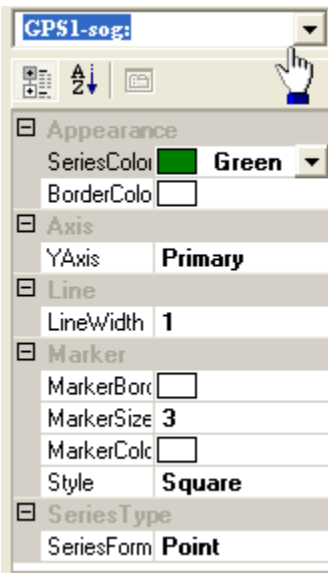
- **X AXISLABEL:** set the time format of the X time axis
- **CHARTBORDER:** toggle if the plot will have a border
- **BACKCOLOR:** Set the background color of the plot
- **CHART BORDER COLOR:** Set the border background color

- BACKGROUND GRADIENT ENABLED: Toggles if the background color is a gradient
- BORDER WIDTH: Sets the pixel width of the border
- BACKGROUND GRADIENT: Sets the color scheme of the gradient.

An Example plot follows:



7.2.6.1.3 Graph Control Panel Sensor Time Series Properties



The user selects the name of the SCS sensor time series to modify and then is presented a set of properties to change. These properties are broken down into the following categories:

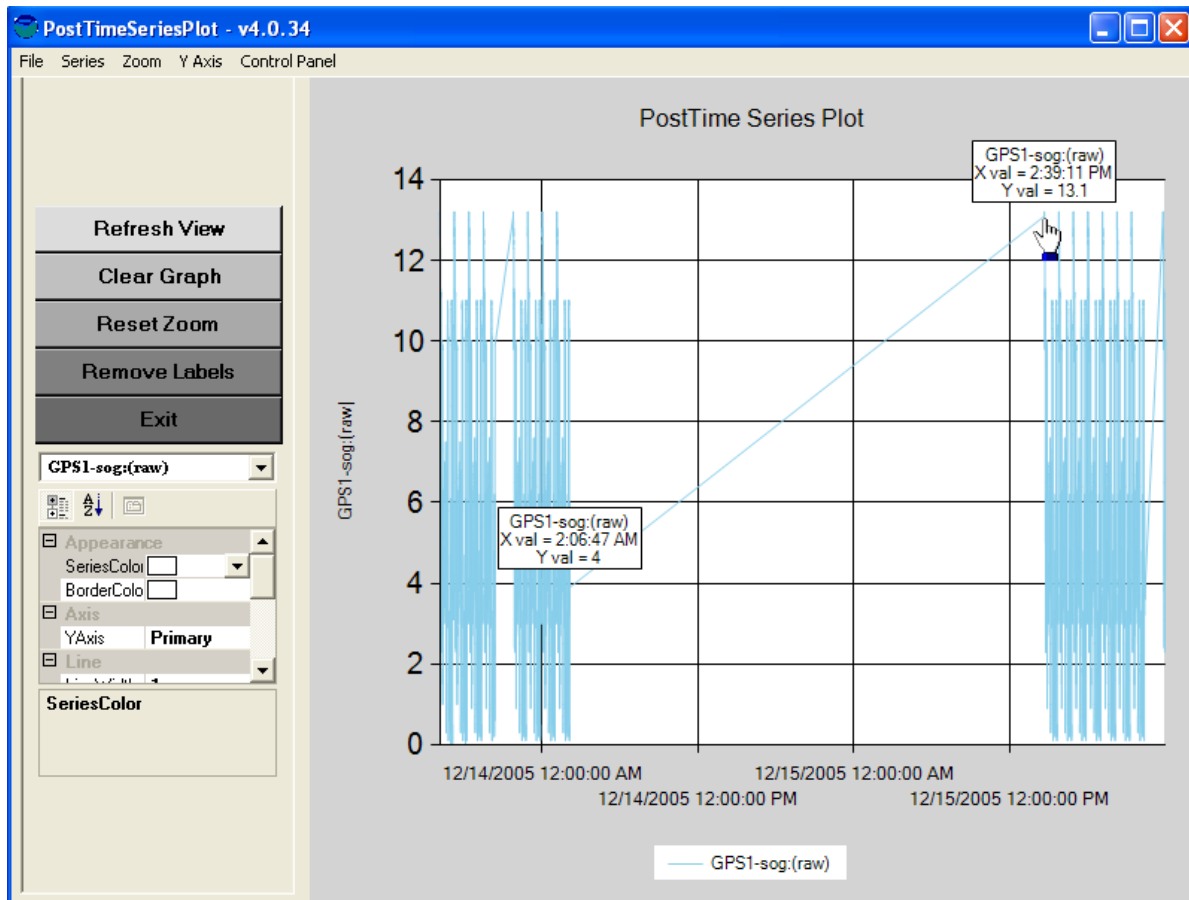
- APPEARANCE
 - ✓ SERIES COLOR: modify the color of the time series
 - ✓ BORDER COLOR: modify the background color
- AXIS
 - ✓ Y AXIS: set the time series to be part of the primary or secondary Y axis. The chart program is a dual axis program and a particular time series may be assigned to either series.
- LINE WIDTH
 - ✓ WIDTH: The pixel width of the line
- MARKER
 - ✓ MARKER BORDER: If a point has a border if plotting in points
 - ✓ MARKER SIZE: the size of a point on the plot, if plotting in points
 - ✓ MARKER COLOR: the color of the point, if plotting in points
 - ✓ STYLE: the Style of the point, if plotting in points
- SERIES TYPE
 - ✓ SERIES FORMAT: a selection of points or line style for the time series.

7.2.7 Graph Behavior

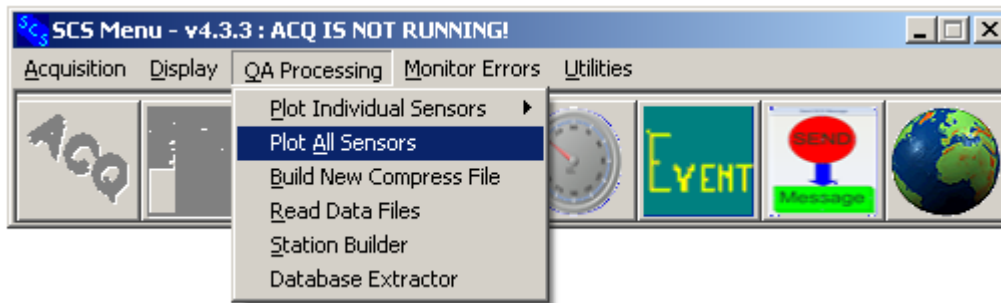
Scroll Bar Behavior – The graph will scroll to the left each time the plot reaches the right margin. The graph will be re-plotted to maintain the display window time interval (See Plot Control Panel Discussion for details) set by the user and will remove data points from the plot as required. All of the data collected in the plot is still available to be viewed based on the actions of the user to manipulate the display.

Data Point Filter – The graph will plot all data points in the current graph that will be visible to the user. This ensures that the plot reflects the data collected while maintaining efficiency by not over plotting duplicate pixels.

Show a Data Point – The user may right click over any data point or time series feature and the plot will display the value of the closest data point.



7.3 Plot All Sensors



Function

This app will use post processing plot templates, and systematically plot and print all data automatically.

SCS 3.3 the newspaper plot utility automatically sent the plots to the printer. SCS 4.1 generates post time series plots on the screen and then writes the plot in JPEG format to the JPG folder underneath your SCSLOG folder

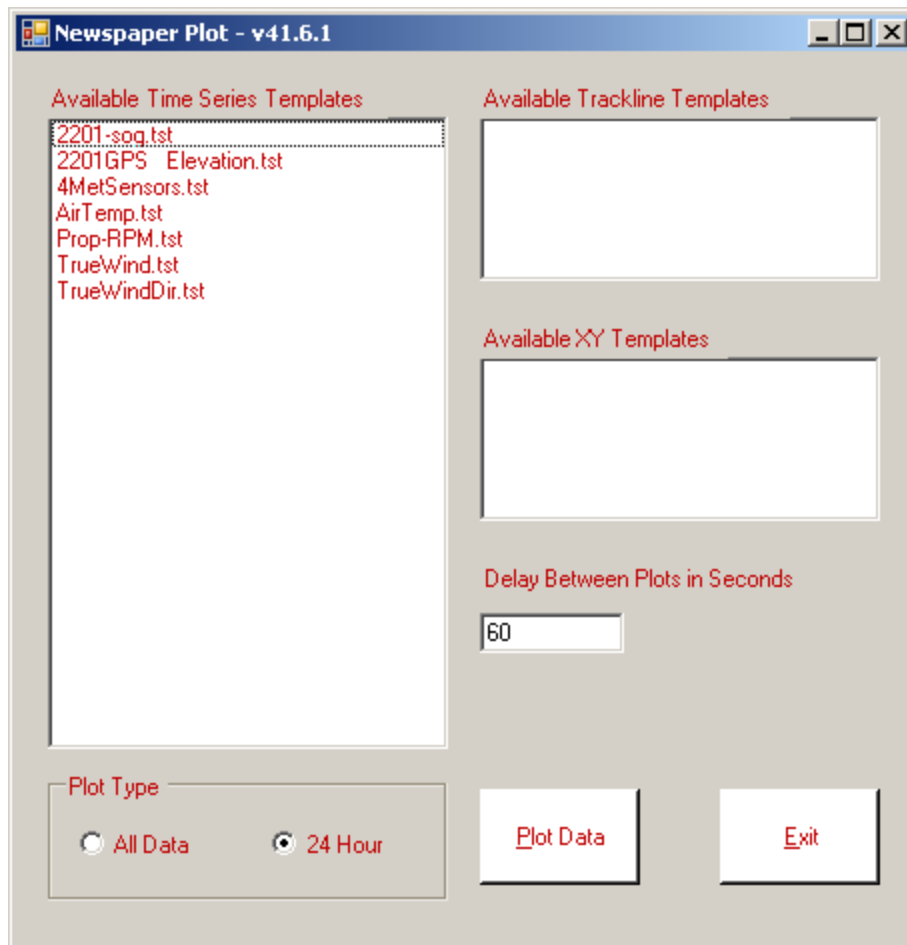
NOTE: This app should be run at least once a day (with the 24 hour option) during the cruise to ensure data quality.

Options

The application's main window is shown below. In the upper left are all post time series plot templates created beforehand by the PTSP template builder.

(NOTE: There are other windows for post track line plots and XY plots, but they are not implemented for this version of SCS.)

Select any and all templates required for newspaper plots and then click on Plot Data.



7.3.1 Plot Type

By default, the application will plot 24 hours worth of data. When you press the “Plot Data” button a calendar will pop up allowing you to select the day to plot. Plotting always starts at midnight of the selected day.

If the Comprehensive option is selected, all of the data since the beginning of the cruise will be plotted.

NOTE: If Comprehensive is selected, the program may take an extremely long time to complete, especially after several days into a cruise).

7.3.2 Plot Delay

You must set the delay factor to reflect the actual time to generate individual plots. If the delay factor is less than the actual generation time, the app will attempt to generate two (or more) at once, and may run out of memory and other resources, which will cause it to crash. Setting it too high is safer, although this will result in a longer time to generate the complete set of graphs.

Output

For either Plot Type, the output consists of graphs on the screen, generated sequentially.

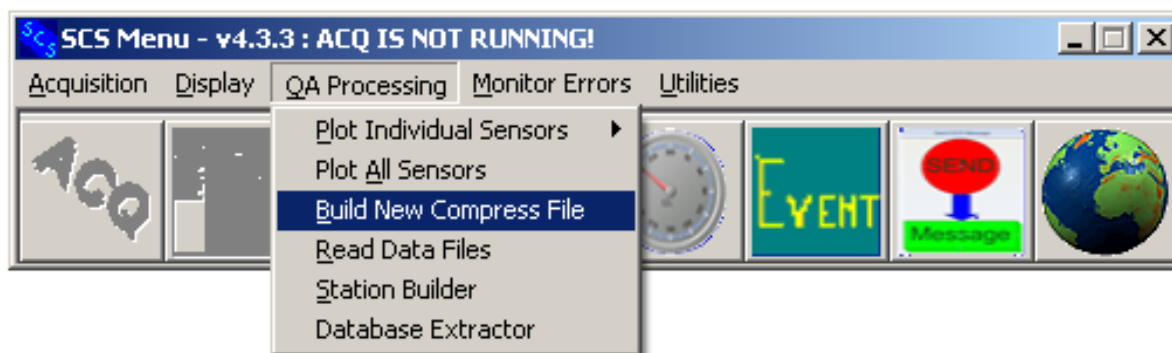
7.3.3 24 Hour Plots

24 Hour plots have a different look. There are no menus or controls to change the look of the graph. In addition to being displayed on the screen the graph is written to a file in JPEG format. This file is placed in a subdirectory called “Jpg” underneath your SCSLOG directory.

7.3.4 “All Data” Plots

The Newspaper plot generates graphs using the Plot Individual Sensors app, run on all selected templates. All data plots look exactly like individual sensor plots. Changes can be made to the plot properties and saved again to a template. JPEG files are not produced automatically, but can be generated from the graph window’s File menu.

7.4 BUILD NEW COMPRESS FILE (SCS Compactor)



Introduction

This utility creates compress files from the raw data files located in the SCSLog logical directory. Compress files contain comma delimited data values that can be imported into a variety of standard PC Applications such as EXCEL or MATLAB. Refer to the beginning of this chapter, and to chapter 1, for more detail on the Compress file format.

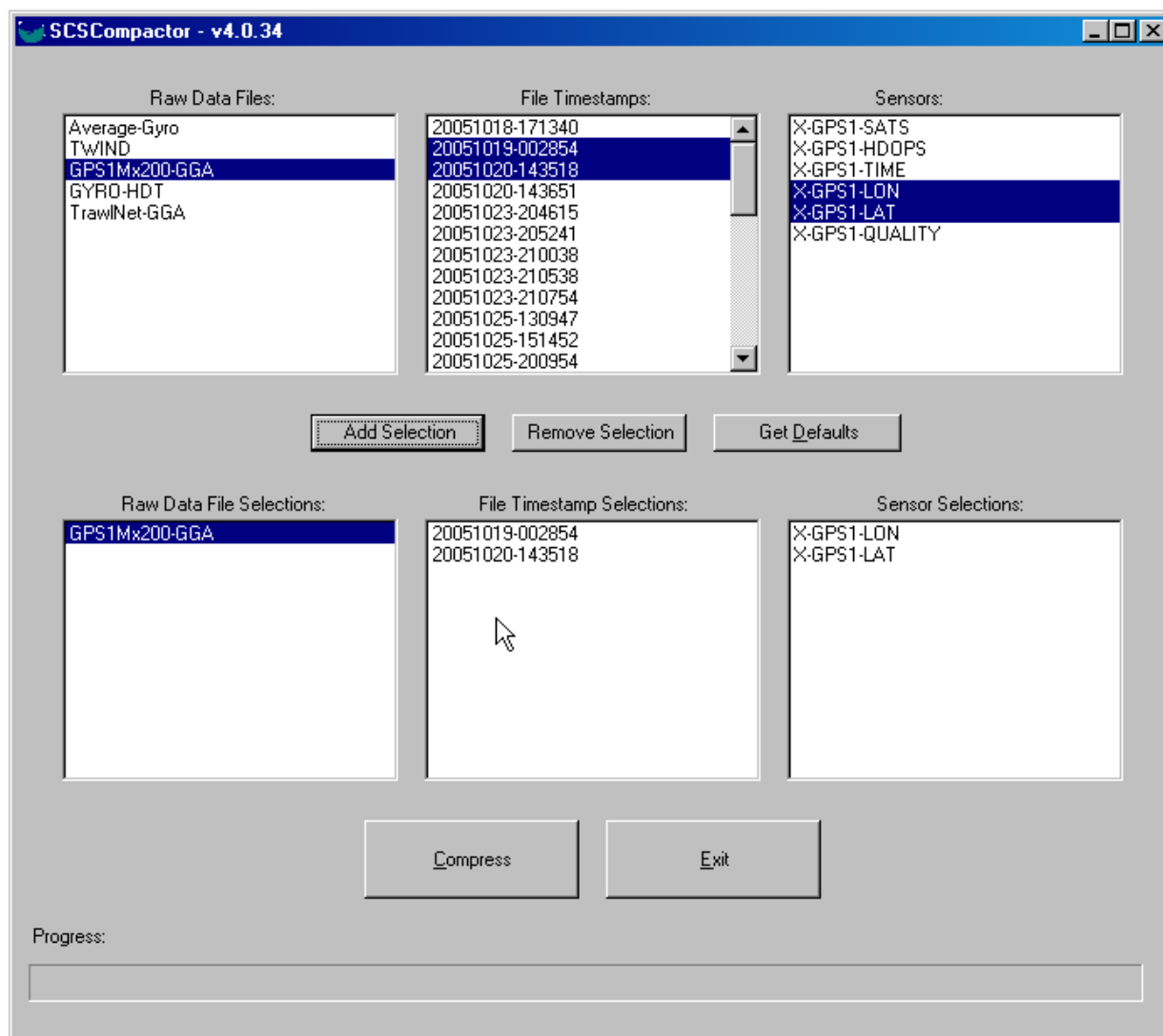
Compress files are a legacy from earlier versions of SCS (3.3c and before), whose plotting software required compress files to operate. In SCS 4.0, all applications are able to read data directly from RAW files and Compress files are no longer strictly necessary.

SCS 4.0 includes this function for backwards compatibility, to allow existing shoreside processing software to continue to analyze data without modification.

Previous releases of SCS required you to check the “Include in Compress” box in order to include a sensor value in the compress file. In SCS v4, checking that box is optional – even if left unchecked, a child or derived sensor value is eligible for inclusion whenever this utility is run.

The SCS Compactor Main Window

The main window for SCSCompactor is presented below.



Available Raw Data

The top three lists show the raw sensor data available for compacting. The lists are built from the sensors contained in the Sensor Configuration File, and from the contents of the data logging directory and its subdirectories.

On the left, “Raw Data Files” shows the names of each parent and derived sensor in the sensor configuration file that has at least one RAW file in its assigned data logging directory. Recall that by default a sensor’s RAW files are written to the SCSLOG directory, but you can assign a subdirectory when you configure the sensor.

When you select one of the Raw Data Files the left-hand list, “File Timestamps” changes to show the different timestamps for the selected sensor in its assigned logging folder, and the right-hand “Sensors” list shows all child sensors in the parent’s message. Logging folder and child sensors come out of the sensor configuration file.

Selecting Data for Compacting

Selecting data for compacting is a four step process

- Select a parent sensor from the Raw Data Files list
- Select one or more timestamps covering the time interval you want to compress over
- Select one or more names from the Sensors list whose data you want to appear in the Compress file.
- Click on the Add Selection Button. This adds a “compress job” to the selection lists on the bottom of the window.
- Repeat the above steps to add as many selections as you desire.

Selected Raw Data Lists

The bottom three lists in the main window show the data you have selected for compression. They present the same information as their analogous lists on the top.

You can select a Raw Data File and click on Remove Selection to unselect it.

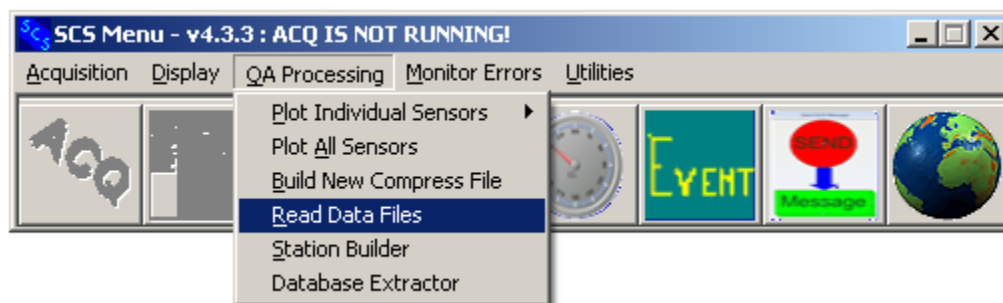
The Get Defaults Button

Click on this button and SCS Compactor scans the sensor configuration file, looking for child sensors whose “Include In Compress File” box is checked. The parent sensors of all such children are copied to the selection lists.

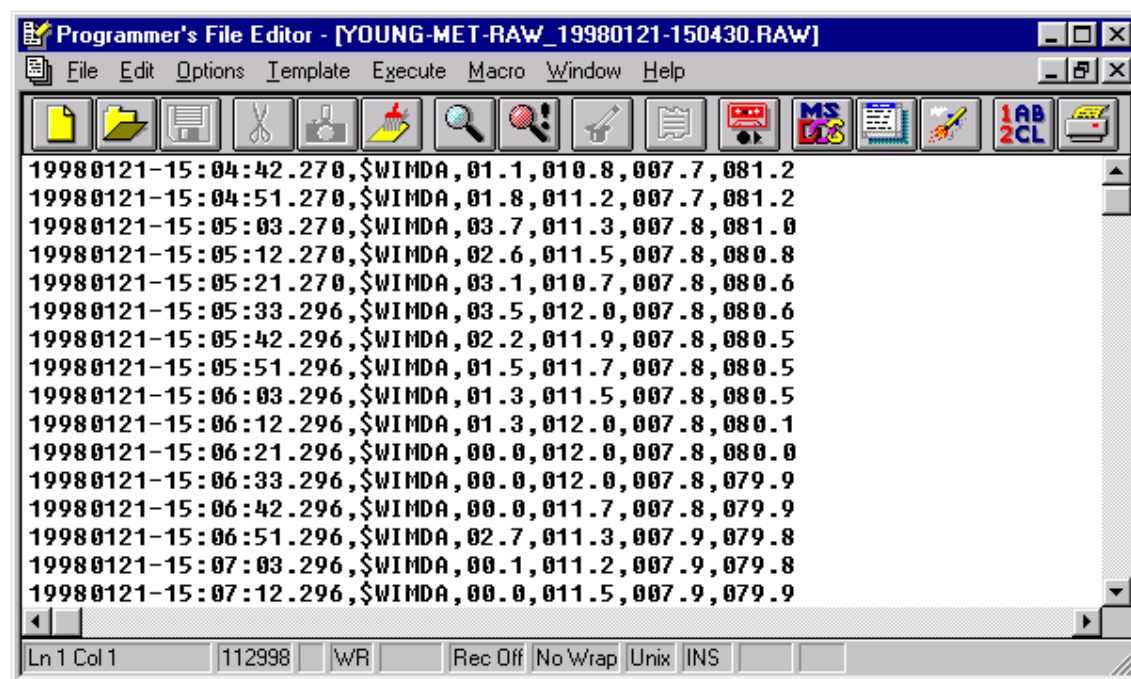
Performing the Compression

Click the Compress button and SCS Compactor goes to work on the selected items. For each sensor in the Raw Data File Selections list, the compactor reads through all the raw files whose timestamps are in the File Timestamp Selections list. For each message in those files, it pulls the child sensors in the Sensor Selections list and creates a line in the compress file.

7.5 Read Data Files



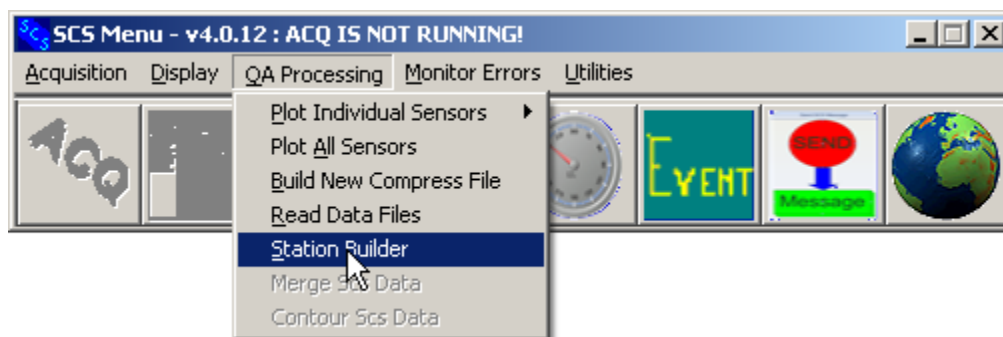
All data files in SCS are ASCII, and SCS includes a shareware program called Programmer's File Editor (PFE) that may be used to read any RAW, ACO, LAB, ELG file. PFE is launched when the Read Data Files menu selection is made. Here is a sample main window:



PFE is optimized for reading large data files very quickly and provides a number of options to act on the data. In the context of SDS we are using this program to provide a program to read and print individual data files. PFE's File menu also keeps track of most recent files for fast access to work in progress.

The PFE.EXE file is installed along with the other SCS apps. A full PFE distribution including help files is included on the Installation CD in the Supplementary Files directory.

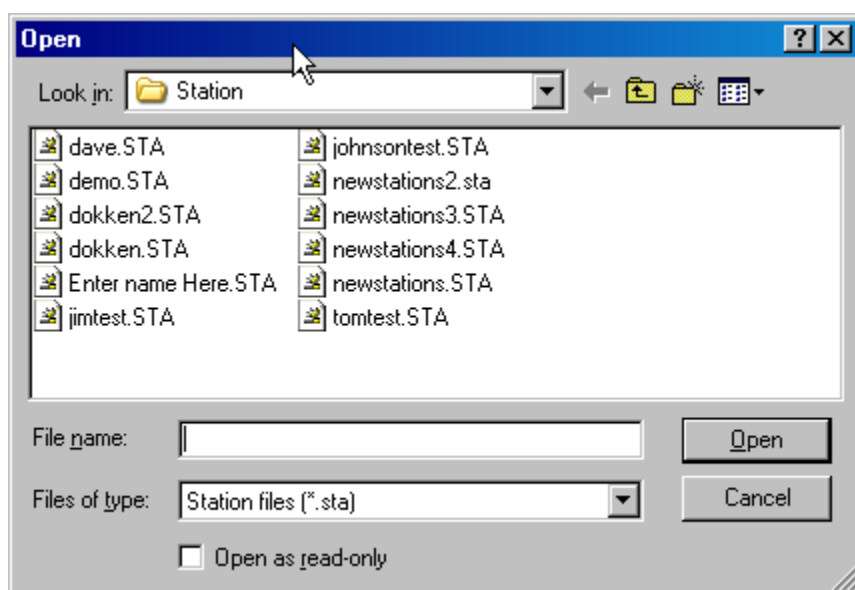
7.6 Build Station File



You use the Station Builder app to build a station file that contains latitude and longitude coordinates and station descriptors. These files are used by the Geotrack app to plot station symbols at the indicated lat/lon coordinates.

The File Menu

1. File/New – Enables the user to create a new station file from scratch.
2. File/Open – A standard Windows File Open dialog for you to select and open a station file that has been previously saved. The default directory is the Station subdirectory under the SCSLog logical directory.



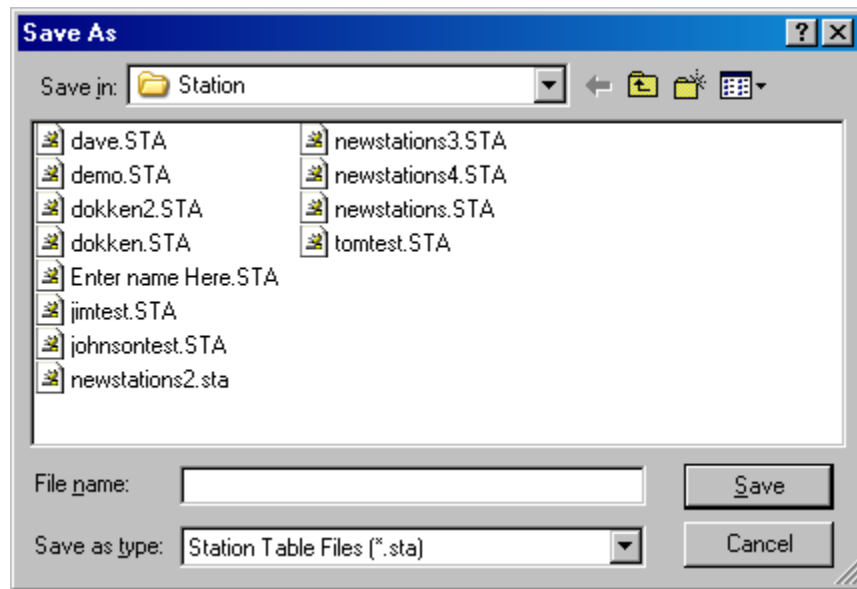
Select the desired station and hit the Open button.

3. (NOT IMPLEMENTED FOR SCS 4.0) File/Import...

A Standard Windows File Open dialog allows you to select and import a file in Navtrack format, which is an ASCII comma-separated file (*.CSV) used on the Bridge automated navigation system. In this file format the first three lines are considered the Header and are skipped during the import. Data begins on line four.

3. File/Save - Saves any changes that have been made to the currently opened station file.

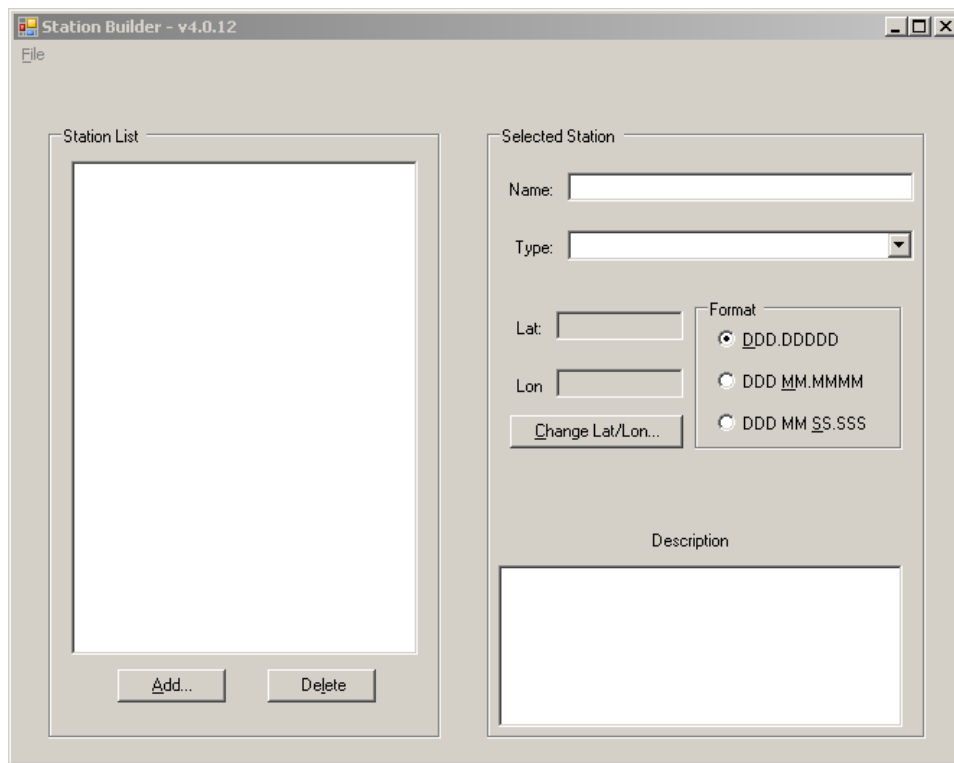
4. File/Save As... - Enables you to save the currently open station file as a different named file.



5. File/Exit - If changes have been made to the station file, you are prompted on whether or not the file should be saved first.

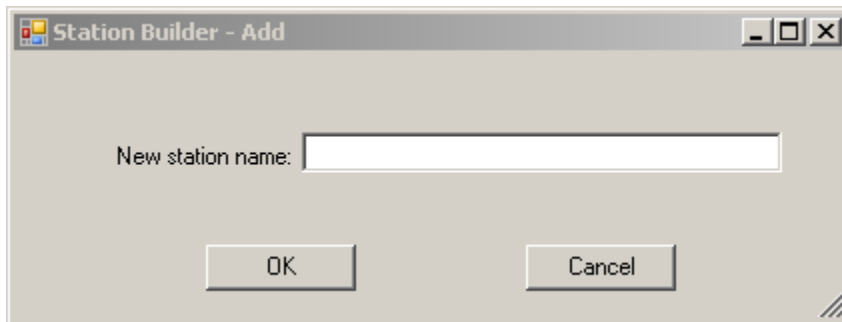
Station Builder Main Window Controls

The main window is show below.

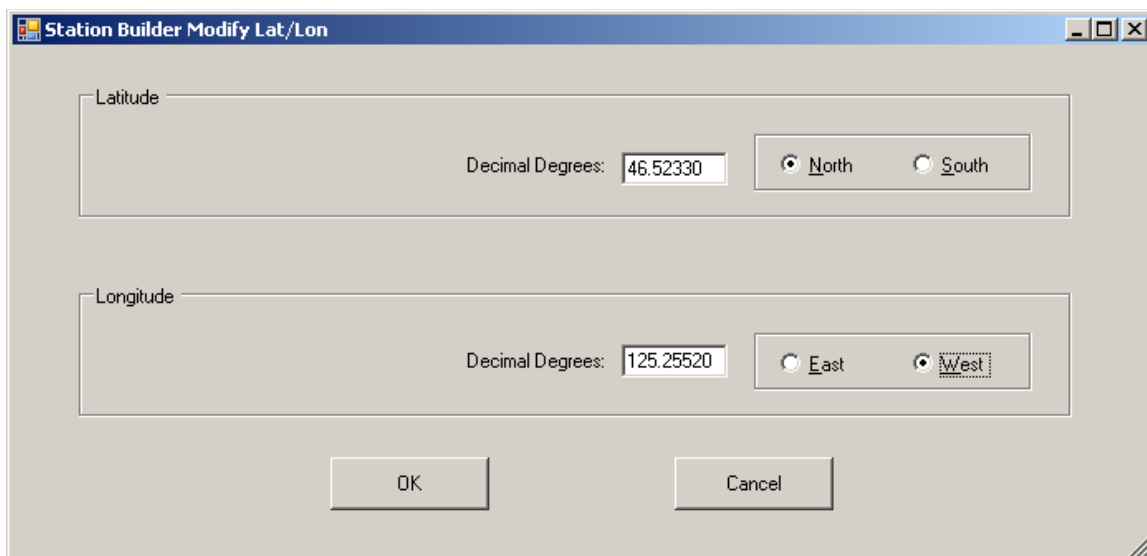


7.6.1 Add Button

Allows you to add a new station to the station list. A dialog pops up to ask for the new station's name.



Enter the station name and press OK. A new dialog will pop up for you to enter the latitude and longitude of the station.



When you enter this information and press OK the new station will appear on the station list, where you can select it to edit the other properties of the station.

Delete Button

Removes a single station (the station currently selected/highlighted) from the currently opened station file.

The Format Radio Buttons

These control the format in which you enter lat/lon data. Select the desired format; then when you press the Change Lat/Lon... button the appropriate dialog will pop up allowing you to enter the data in the chosen format.

Note that if you enter the data in one format then select another, the data is automatically converted to the new format.

Modify Button

Allows the user to modify the latitude/longitude coordinates of the selected station. One of three dialogs pops up, depending on which lat/lon format you have selected in the Station Builder main window. The user has 3 options in regards to how the positional data may be entered: Decimal Degrees, Degrees Minutes, or Degrees Minutes Seconds.

Decimal Degrees:

The dialog box titled "Station Builder Modify Lat/Lon" contains two sections for Latitude and Longitude. Each section has a text input field labeled "Decimal Degrees:" followed by a radio button for "North" and "South" for Latitude, and "East" and "West" for Longitude. The Latitude field contains the value "46.52330" and the Longitude field contains "125.25520". At the bottom are "OK" and "Cancel" buttons.

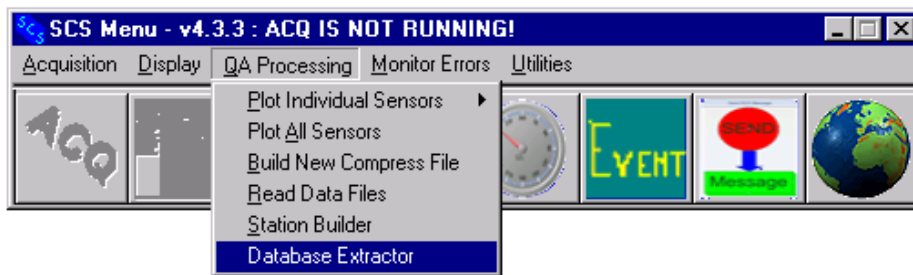
Degrees Minutes:

The dialog box titled "Station Builder Modify Lat/Lon" contains two sections for Latitude and Longitude. Each section has two spin button fields: "Deg:" and "Min:". The Latitude fields contain "46" and "31.3380" respectively. The Longitude fields contain "12" and "15.3120" respectively. Each section also has radio buttons for "North" and "South" for Latitude, and "East" and "West" for Longitude. At the bottom are "OK" and "Cancel" buttons.

Degrees Minutes Seconds:

The dialog box titled "Station Builder Modify Lat/Lon" contains two sections for Latitude and Longitude. Each section has three spin button fields: "Deg:", "Min:", and "Sec:". The Latitude fields contain "46", "31", and "23.880" respectively. The Longitude fields contain "12", "15", and "18.720" respectively. Each section also has radio buttons for "North" and "South" for Latitude, and "East" and "West" for Longitude. At the bottom are "OK" and "Cancel" buttons.

7.7 Database Extractor



Function

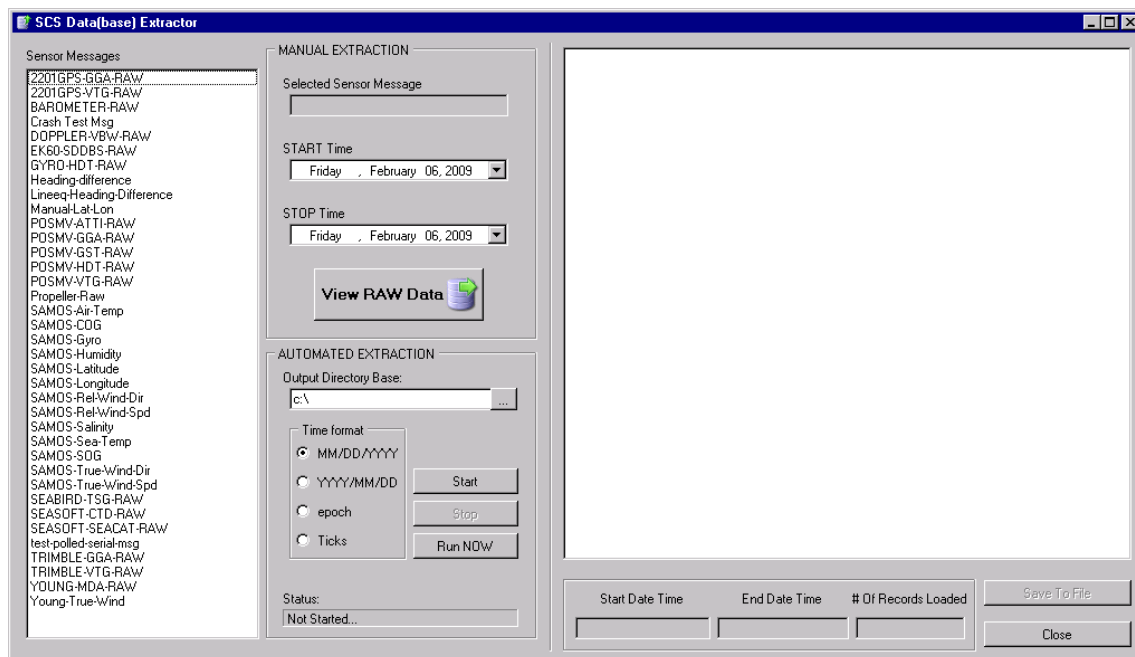
The SCS Database Extractor is currently (v 4.3.4) the sole method of retrieving raw messages from the SCS database, aside from using SQL commands in the SQL Manager Console. It will extract messages sensor by sensor and create files in the RAW file format (time tagged raw sensor messages) and store them in files named exactly like the original RAW files.

Since ACQ logs raw messages to RAW files whether or not database logging is enabled, this app will produce a product identical to the original RAW file. Hence its sole use is to recover from the database any RAW files that have been accidentally lost or damaged.

The Database Extractor is a technology demonstrator that illustrates the methods by which a SCS app can access the database for different purposes. In future releases of SCS all post processing software will access the database for data to plot or otherwise analyze.

Main Window GUI

Invoking the app shows the following GUI.



On the left is a list of all Sensor Messages in the database. All items have corresponding RAW files logged by ACQ in the DATALOG40 directory during acquisition.

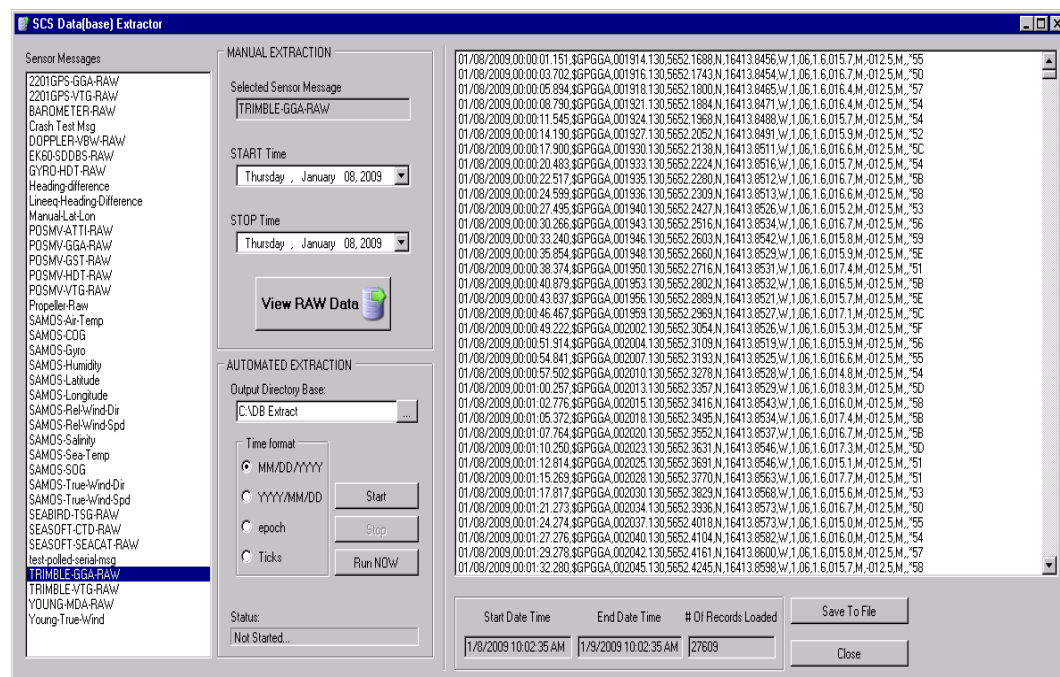
In the center you see two different modes that the app runs in, manual extraction and auto extraction. They are described in detail below.

NOTE: the two modes operate in completely different ways for different purposes. You have no control over this, and you cannot combine the two modes into any sort of hybrid operation.

Manual Extraction Mode

The top mode is Manual Extraction. In this mode you select one (and only one) Sensor Message, and then select the starting and ending time. In reality they are starting and ending days.

Click on View RAW data and for the days selected the app will extract the specified message for the specified day interval and display in the list box on the right. Note that the app works only in units of whole days. At the end of the extraction process the Start and End dates and the Number of Records text boxes are filled in. Here is a sample extraction of some GPS GGA messages.



At this point the data has not been saved to a file, it is just being displayed. Click on the Save to File button and the app will present you with a file open dialog where you can select the folder and file name you wish the displayed data to be saved to. The default file name is taken from the Start Date of the extraction process.

Auto Extraction Mode

To run an auto extraction, you first select the *Output Directory Tree*. Under this directory the app will create sub folders named after every day of extraction it performs.

Click on the *Start* button and the app goes to sleep. You leave the app running indefinitely. The app wakes up every 24 hours at midnight GMT to save the previous day's data in day files. This mode operates on ALL messages in the list.

Alternatively, you can click on the *Run NOW* button. This starts the operation immediately, but it extracts data for the 24 hours prior to *now*. In other words it does not work midnight to midnight as the other operations do.

CHAPTER EIGHT

MONITOR ERRORS

8 Monitor Errors



8.1 DataMon and Sensor Data Quality Assurance

Introduction

The SCS *DataMon* utility monitors sensor data for quality according to standards that you specify in the sensor configuration. DataMon performs the following types of *quality checks*:

1. *Synch Check*: DataMon examines each incoming Parent sensor message, and if a predetermined character string is not at the specified place in the message, DataMon will report an error. An example is "the first six characters of the message from NMEA Parent message from a GPS receiver must be \$GPGGA".
2. *Timeout Check*: If a sensor message is not logged within the specified number of seconds after the previous message, DataMon will report an error. Datamon defines a timeout to be more than three times the Log Rate between messages in the Log file.
3. *Range Check*: DataMon looks at each logged sensor measurement (either a sub sensor, derived sensor...), and if the value falls outside the specified range, DataMon will report an error.
4. *Delta Check*: DataMon compares each logged sensor measurement to the previous one, and if their difference falls outside the specified range, DataMon will report an error.
5. *Decode Check*: Datamon will attempt to verify that the data field XXX

DataMon runs primarily during data acquisition and works closely with Logger. It can also do the same quality checks on legacy data files in the *ScsLog* Directory in a stand-alone mode, independent of ACQ and Logger.

Incidents

When a sensor message or subsensor measurement fails any of DataMon's quality checks, DataMon generates and logs an *incident*. An incident can be one of two types:

- A *Long Duration* incident has separate start and stop times. Synch, Timeout and Range Check incidents are all long duration.
- In an *Instantaneous* incident the start and stop times are the same. A Delta Check incident is instantaneous.

8.1.1 The Lifetime of a Long Duration Incident

Consider a data field giving a temperature value. You have specified its *normal range* using the CFE-DB-DB Range Max and Range Min parameters, and then begin data acquisition operations. A long duration *Range Incident* begins its life when the temperature goes out of the normal range. DataMon *opens* a Range Incident, noting the *start time*. It may be some time before the sensor again reports a temperature back in range, and until this happens we say that the incident is *active*. While an incident is active DataMon keeps track of the temperature's maximum deviation out range, and the time this occurs. When the temperature does return to normal DataMon records the incident's *stop time*, and *closes* the incident. The incident is no longer active.

8.1.2 Long Duration Incidents: An Example

Let's look at a real example. Using CFE-DB-DB, you enable Range Check and enter the value 15.0 for Range Minimum and 33.2 for Range Maximum parameters. You also specify a Log Rate of 120 seconds. You then start up ACQ and Logger on February 4, 1999 at 12:34 GMT. The initial logged temperature at that time is 32.9, but by 15:00 it has reached 33.3. At that point DataMon generates a Range Check incident and records the incident's start time. The temperature continues to rise until it peaks at 33.7 degrees at 16:00. It then starts to decline, reaching 33.2 at 16:46, and continues to decline after that. Between 15:00 and 16:46 is 106 minutes, or 53 logged measurements.

When DataMon displays or prints this incident you will see the following information:

<u>Incident Type</u>	<u>Range</u>
Start Time	4 Feb 1999 15:00
Value Most In Error	32.9
Maximum Error	0.7
Time of Value	Most In Error 16:00
Stop Time	16:46
Duration	105 minutes

The other long duration incidents are handled in a similar manner. Some details change depending in the type of incident being logged.

8.1.3 Instantaneous Incidents

An instantaneous incident does not have a lifetime. It opens and closes at the same time, so in DataMon's display the start and stop times will be the same. By definition, an instantaneous incident can never be active. DataMon reports the Maximum Error and Value Most In Error, but they are based on the single sample logged at the incident's start time.

DataMon and Other SCS Software

This section explains DataMon's interactions with other SCS software applications.

8.1.4 Configuration File Editor

You specify the quality checks for each sensor in the CFE-DB Application. Using CFE-DB, you enable or disable checks, and specify the parameters needed to carry them out.

<u>Type of Check</u>	<u>Applicable Parameters</u>
Range	Range Minimum, Range Maximum
Delta	Delta Minimum, Delta Maximum
Synch	Synch String, Synch Position
Timeout	Timeout

The values you assign to these parameters determine how stringent or loose those checks are. Refer to the Appendix 1, Sensor Parameter Dictionary for complete definitions of these parameters.

8.1.5 SCS Logger

DataMon monitors data that has been logged to disk in *.RAW data files, which are stored in the SCSLog directory. DataMon has no connection with except to verify ACQ is running. DataMon monitors data written to *.RAW files. This has several important consequences:

- You can be assured that the data passing the quality checks has already been successfully written to disk and thus cannot be changed by any of the SCS real time software.
- Since Logger does not log all messages received from a sensor (see the Log Rate parameter) DataMon cannot monitor every message sent by a sensor – only those logged to a *.RAW file.

8.1.6 SCS ACQ

When DataMon starts it reads the timestamp's Sensor Configuration File, SENSOR_19990506-1234.SCF. This timestamp is just an example – ACQ creates these SCF copies every time it starts in the SCSLog directory. Note that this is not the SENSOR.SCF file that you usually work with.

As mentioned above, DataMon gets its data from logged data files and does not use any of the other SCS Data services. However, when you start DataMon from the SCS menu, DataMon checks to see if ACQ is running or not, and this affects the options available to you (see section 1.5.1.1, the Startup Dialog, for more information.)

Timestamps

When ACQ starts, acquisition for all sensors starts at the same time.

This common time is called the *timestamp*, which is composed of the date and time (GMT), for example if acquisition begins at 21:09:45 GMT on May 6, 1999 the time stamp looks like this: 19990506-210945. We often refer to a group of RAW files simply by their timestamp.

When you start ACQ all RAW files created have the same timestamp included as part of their name. At the same time ACQ also makes a time-stamped copy of the operating

SENSOR.SCF file in the ScsLog directory. The timestamps of these SCF files appear in the DataMon startup dialog.

8.1.7 The Startup Dialog

DataMon first determines if ACQ is running, then it presents a startup dialog similar to Figure DM-1.

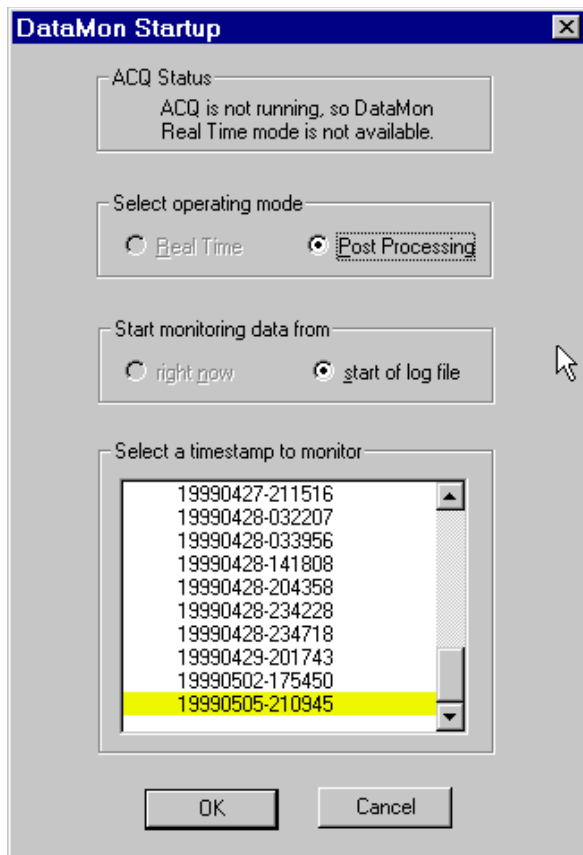


Figure DM-6. DataMon Startup Dialog

8.1.7.1.1 Starting Point

Datamon will run from the beginning of the current RAW file to the end of the logged data, then will update in real time.

8.1.7.1.2 Timestamp to Monitor

At the bottom of the Startup Dialog is the list of available timestamps to monitor. The timestamps are from the files currently in the SCSLog directory.

Each time you run DataMon on a timestamp in Post Processing Mode, you will overwrite the previous incident archive. DataMon will ask your permission before deleting the previous version.

8.1.8 After Startup: The Summary Window

When DataMon's Startup Dialog is complete it displays its Summary Window. A typical Summary Window is shown in figure DM-2.

8.1.8.1.1 Description

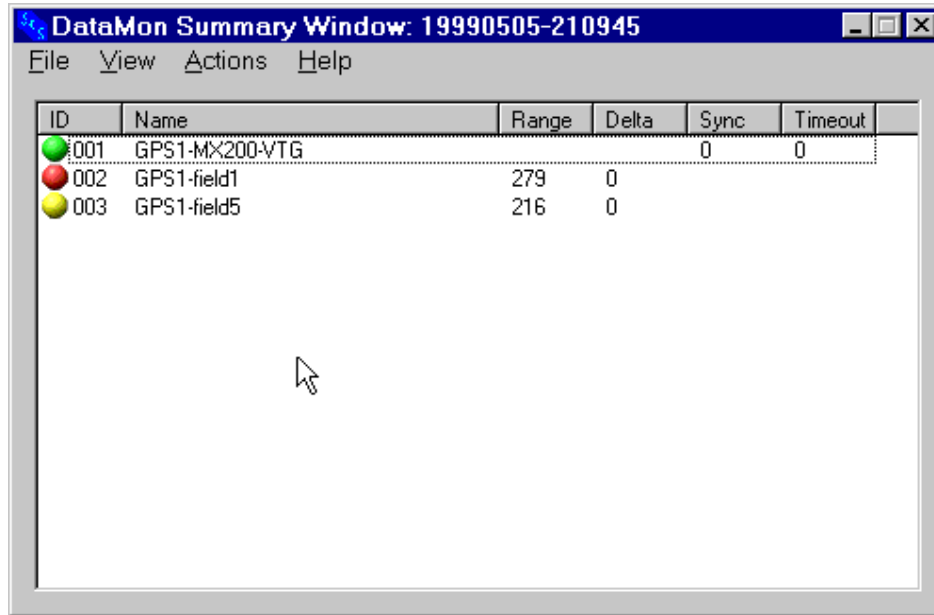


Figure DM-2. DataMon Summary Window

The title bar of the window shows the timestamp being monitored.

In the window is a list of sensors you have selected for display (this is described below). In the first column is the sensor ID and an LED icon whose color indicates the current status of the sensor:

Red Has an active incident.

Yellow Has had at least one incident, but none is active now.

Orange Has a current error and it's message also has an error.

Green Has never had an incident.

Column two of the list has the Sensor Name. Columns three through six show the total count of incidents of each type for the sensor. A column is blank if that check is not applicable to the sensor.

When DataMon starts it displays by default only the sensors that have had at least one incident.

8.1.9 Clicking on Sensors

The sensors in the list respond to various types of single- or double-clicks.

You may select one or more sensors from the Summary Window list in preparation for follow-up actions (to be described below). The standard Microsoft Windows selection methods are available:

- Left mouse click to select a single sensor
- Right-click to open a context menu.

Selecting a sensor does not by itself do anything - only the follow-up action does.

Double clicking on a sensor is the same as invoking the View | Details menu selection. See below for a description of this action.

Summary Window Menus

8.1.10 The File Menu

- File | Exit: Exits DataMon.

8.1.11 The Sensor Context Menu

- Is Enabled: Enables/Disables the sensor and all of its child sensors
- View Details: View the past instantaneous errors (up to the log limit) for the sensor.
- View Summary: View the past continuous errors (up to the log limit) and running counts of all the errors.

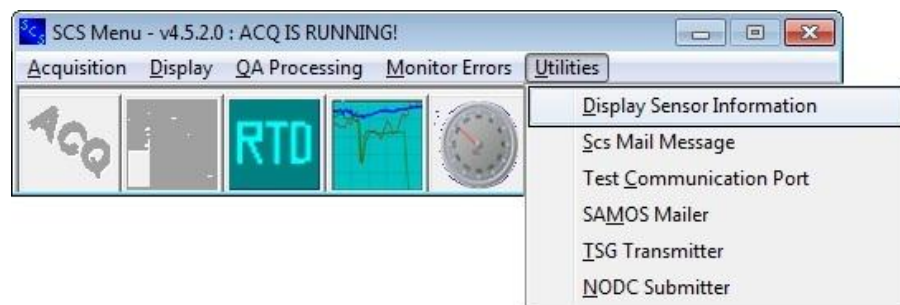
CHAPTER NINE

UTILITIES

9 SCS Utilities

This chapter provides some basic SCS utility functions that are primarily meant to be used by the System Manager.

9.1 DISPLAY SENSOR INFORMATION

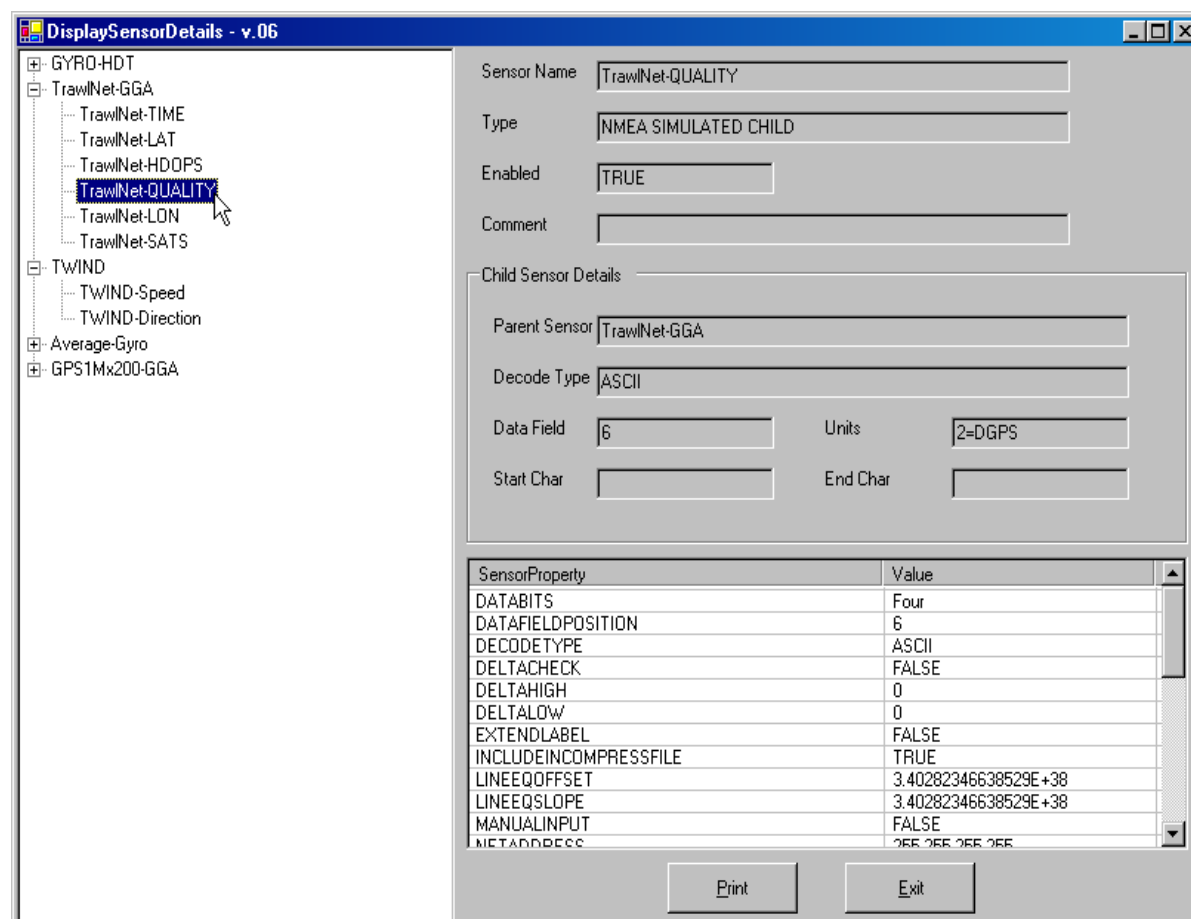


Function

This utility program provides the System Administrator an alternative to CFE-DB to view the sensor configuration file. For end users, it is the only way to view the configuration file, as CFE-DB requires manager privileges to launch.

The Main Window

Here is a sample main window



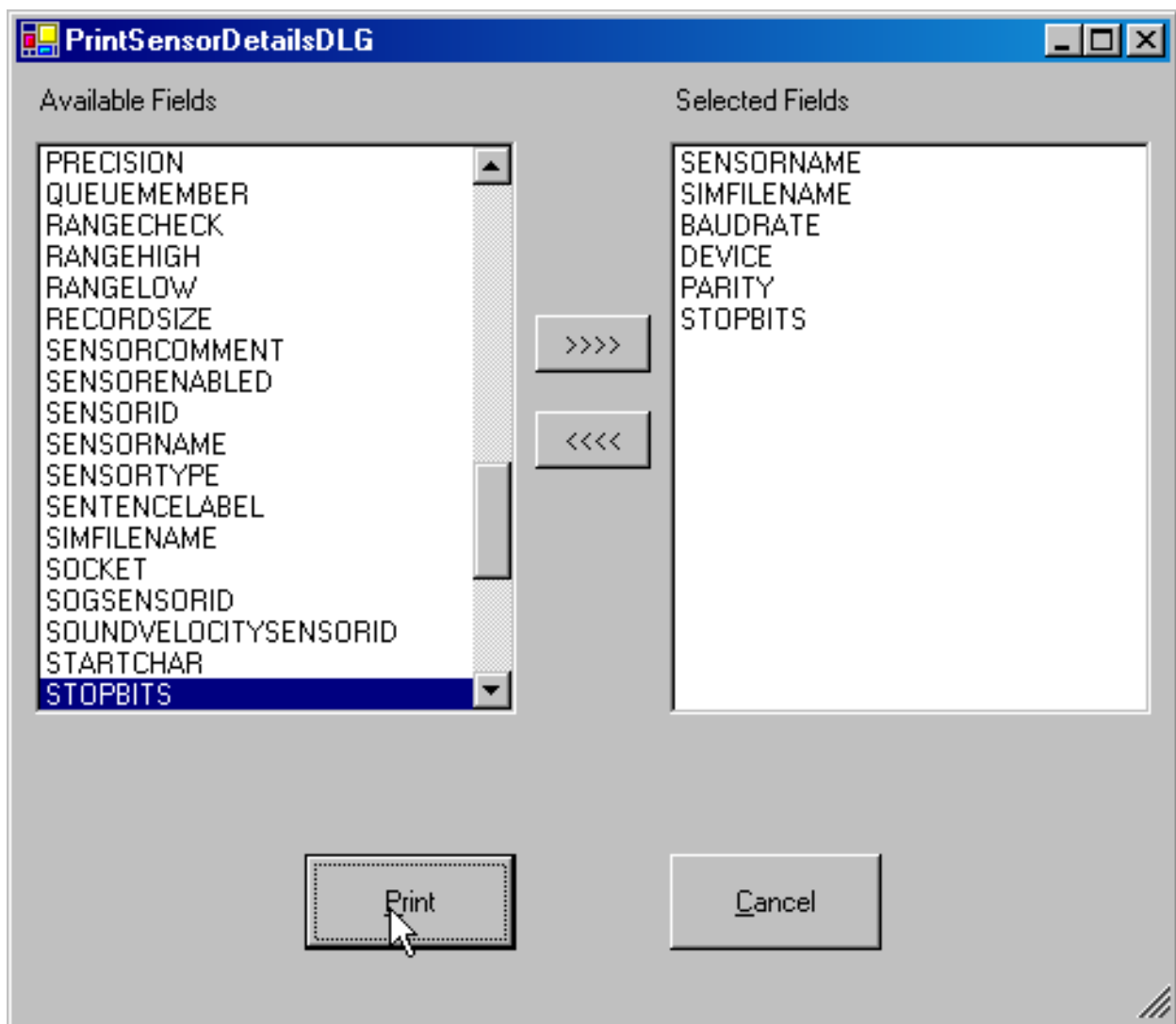
On the left side is a tree structure with parent sensors at level one of the tree and all associated children shown at level two. Selecting a sensor on the tree causes the information for that sensor to be displayed on the right side of the window.

All parameters relevant to the selected sensor are displayed. Those parameters that are not defined for the sensor are not displayed. For example, Baud Rate is not defined for child sensors.

Refer to Appendix 1, The Sensor Parameter Dictionary, for a complete description of sensor parameters.

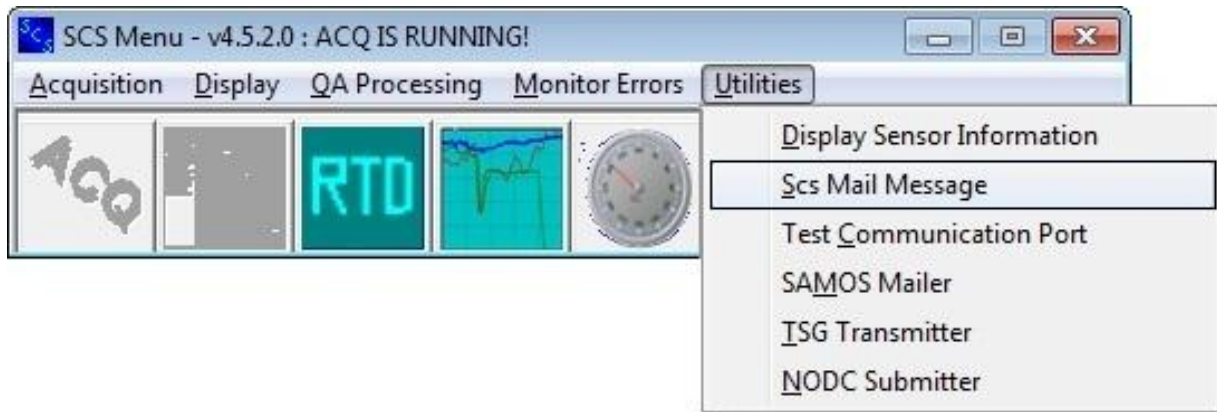
The Print Button

Press the print button and the Print Sensor Details dialog pops up as shown here.



Select the parameters you want to print out and press the dialog's Print button. A standard Windows Printer Selection dialog pops up to let you direct the report to the desired printer.

9.2 SCSMAILER



Function

This utility program provides the ability of taking snapshots of the SCS real time data and mailing these snapshots to users aboard the ship or to any email address ashore. The facility receiving the mail messages may then process the data as they wish.

One common use of this program is to send periodic reports to NOAA's ShipTracker web site, which in turn displays locations and movements of all SCS vessels.

Using SCSMailer

9.2.1 Virus Scanners

SCS Mailer uses TCP Port 25. This port may have been blocked by a virus scanner, firewall or other security software. If so the port must be unblocked prior to using SCS Mailer.

9.2.2 Select Sensors to Be Mailed

SCSMailer uses Real time Display templates created by the RTD Template Builder found in the SCSMenu Display Menu. Simply select the desired sensors as if you were going to generate a real time display and save them to a template. SCSMailer reads these templates and generates email for the sensors it finds inside.

9.2.3 Launch SCSMailer

The program is launched from the SCS menu Utilities Bar. When launched the following main window is presented.

SCSMailer - v4.0.5

Configuration

Mail Server: fleethub.onco.noaa.gov

Display File: MailDisplay.Sel

Send Interval: 3600

Destination Address: tom.stepka@noaa.gov

Ship Name: delaware

Status

Last Message Time: 12/27/2005, 7:19:35 PM

Next Message Time: 12/27/2005, 8:19:34 PM

Send Count Down:

X-GPS1-TIME X-GPS1-LAT X-GPS1-LON,HHMMSS,DEGMIN,DEGMIN,202803,4256.9179N,

Start Stop

9.2.4 Enter Configuration Details

The top portion of the main window contains the information needed to configure the mailer to do its job. The fields are

- Mail Server. The fully qualified network name of the destination mail server.
- Display File. The name of the template file in the RTD template directory having the list of sensors whose values the Mailer will send off.
- Send Interval. The time in seconds between consecutive emails to be generated. Enter 3600 to send mail every hour, or 86400 to send it once a day.
- Destination Address. The destination email address.
- Ship Name. The name of the ship sending the data. This name is built into the “sender” field of the message, as shown later.

9.2.5 Sending Email

Click on the Start button to begin real time operations. At the appropriate interval SCSMailer will query ACQ for the latest values of the sensors listed in the designated Display template. It composes a message in a specific format, which is illustrated in the sample message here:

```
PCODE-TIME, PCODE-LAT, PCODE-LON,  
HHMMSS, DEGMIN, DEGMIN,  
202803, 4256.9179N, 07022.5829W,
```

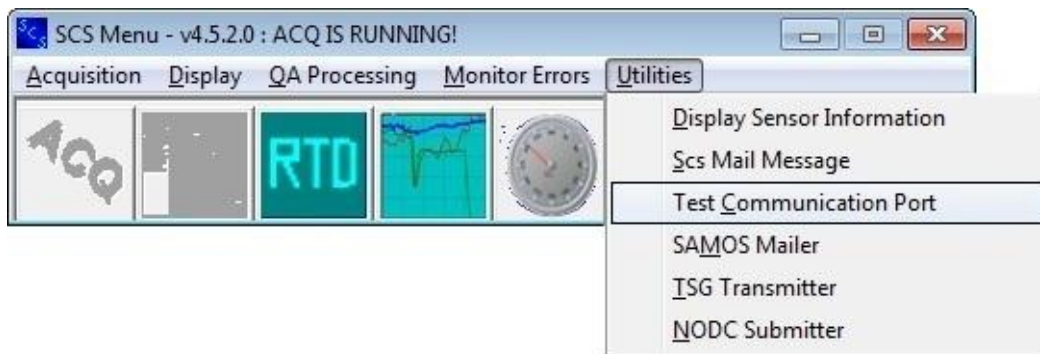
Each message has three lines. The first has the names of the sensors being transmitted, the second has the sensor's units field in the Sensor Configuration File, and the third has the sensor values. The lists on each line are comma separated values, and the last entry on each line always has a terminating comma. This format is fixed and there is no way to customize it.

As SCSMailer runs, the lower part of the main window show the status: the time of the last message, of the next message and the time to go until the next message. Each message is placed in the text box as it is mailed.

Stop Program

The user may stop SCSMailer at any time by pressing the STOP button.

9.3 Test Communications Port



Introduction

This menu item *used to* open the Hyperterm app located in the SCS Exe folder. It is now assumed that Hyperterm has been installed in Windows and is available through the Windows Start Button.

This menu item *now opens* a new SCS app called the Serial Port Diagnostic (SPD). It is a supplement to Hyperterm to cover a couple of deficiencies in that app.

Hyperterm works wonders in conditioning serial ports, but for configuring SCS sensors it leaves two things to be desired, especially when it displays serial parent messages.

1. It does not display control characters in plain text.
2. It is difficult to count how many characters there are in the message, or at what character position a given data field may start within the message.

Using the Serial Port Diagnostic

The use of this app is simple and straightforward. When you start it, simply set the COM port parameters on the left, then press the start button. Messages will appear as they come in the matrix on the right.

The figure below illustrates the display of a serial parent barometer sensor.

Note: that the terminating Line Feed character is displayed in plain text. All other control characters are displayed in a similar manner.

Note: that the column positions are displayed, allowing you to easily set the Start Char and Stop Char parameters for this parent sensor using CFE-DB.

The screenshot shows a window titled "COM9 9600-Eight-None-One". On the left, there are configuration settings for a serial port:

- COM Port: COM9
- Baud Rate: 9600
- Data Bits: Eight
- Parity: None
- Stop Bits: One
- Flow Control: None

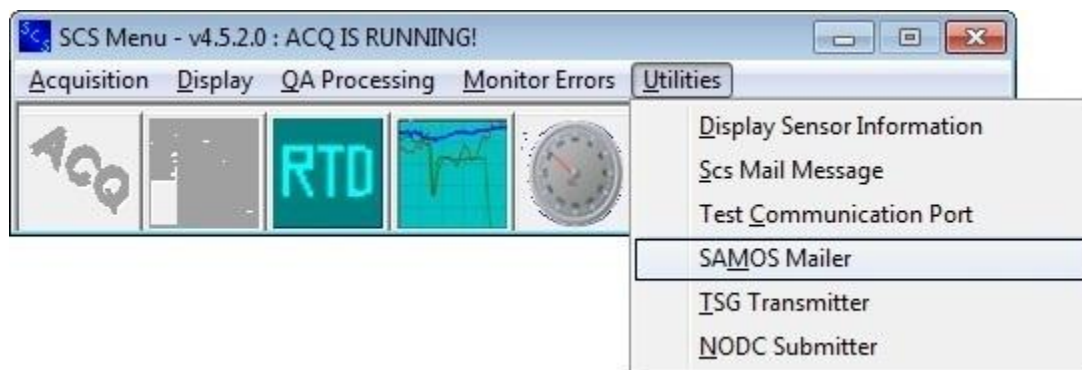
At the bottom left are "Start" and "Stop" buttons. On the right is a data matrix with 8 columns labeled 1 through 8. The matrix contains the following data:

	1	2	3	4	5	6	7	8
1	1	0	2	0	.	9	8	LF
2	1	0	2	0	.	9	5	LF
3	1	0	2	1	.	0	3	LF
4	1	0	2	1	.	0	1	LF
5	1	0	2	0	.	9	8	LF
6	1	0	2	0	.	9	8	LF
7	1	0	2	1	.	0	5	LF
8	1	0	2	1	.	0	0	LF
9	1	0	2	0	.	9	8	LF
10	1	0	2	0	.	9	8	LF
11	1	0	2	0	.	9	7	LF
12	1	0	2	1	.	0	5	LF
13	1	0	2	1	.	0	4	LF
14	1	0	2	0	.	9	5	LF
15	1	0	2	0	.	9	7	LF
16	1	0	2	1	.	0	0	LF

COM Port Polling

A new feature for SCS v4.3.4 is available by right-clicking anywhere on the message display. The app will listen on all com ports in the Sensor Configuration file and cycle through the combinations of baud, parity, stop bits etc. until ASCII text is found. It will then display the results.

9.4 SAMOS Mailer



Introduction

The SCS *Event Log Mailer* app takes a SCS Event Logger ELG data file, converts Event Log ELG files to another format, compresses it to a ZIP file, and emails the data to a remote client site. The app performs this function automatically every 24 hours at 0001 GMT.

The initial functional requirement for this app was developed in conjunction with the SAMOS project in the COAPS office of Florida State University. Hence the app is also known as the *SAMOS Mailer*. Please refer to the Appendix 3 for directions for NOAA ships to send data to the SAMOS project.

Future versions of SCS this app will be more flexible to allow for generation of mailings in other formats to other organizations on other schedules.

Setting up the SCS Configuration File

A few new variables need to be added to the SCS Configuration File *SCS40.CFG* before running Event Log Mailer. Note the specific references to SAMOS.

SAMOS_EMAILSYSTEMNAME is the name of the ship's mail server. Typically this parameter has the same value as *SHIP_EMAILSYSTEMNAME* in the configuration file.

SAMOS_MAIL_BODY defines a short text string you would like to include in the body of the emails.

SAMOS_MAIL_FROM is the 'from' address which appears in the email.

SAMOS_MAIL_TO is the email address to send the data files to.

SAMOS_MAIL_SUBJECT is a string which you want to appear in the subject line.

**** Example settings appear as below. They are specific to a SAMOS setup on the NOAA Ship Oscar Dyson.**

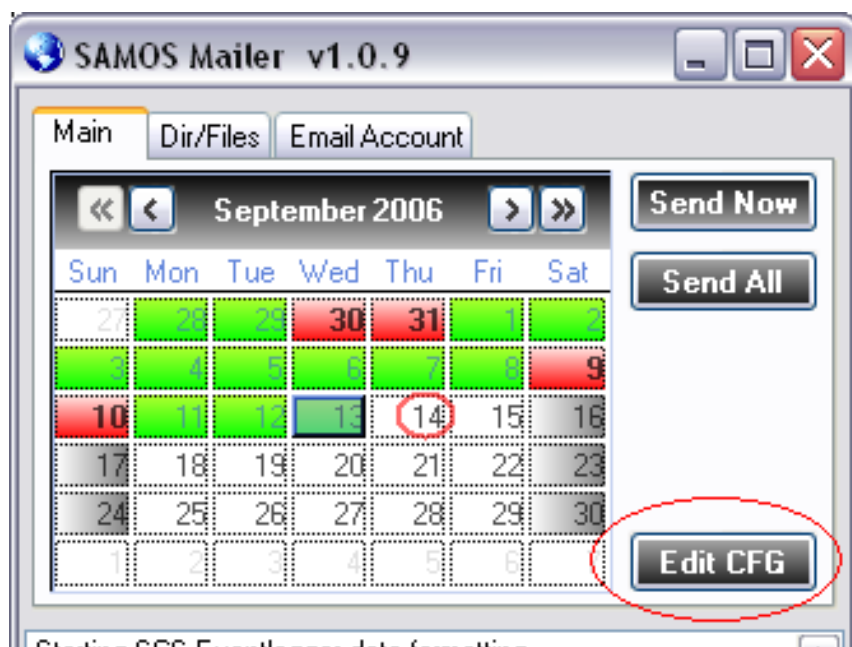
SAMOS_EMAILSYSTEMNAME=mail.nems.noaa.gov;

SAMOS_MAIL_BODY=SAMOS data from the NOAA Ship OSCAR DYSON (WTER);

SAMOS_MAIL_FROM=chiefst.oscar.dyson@noaa.gov;

SAMOS_MAIL_TO=samos_data@coaps.fsu.edu;

SAMOS_MAIL_SUBJECT=SAMOS Mail From the NOAA Ship Oscar Dyson (WTER);



Note: The configuration file can be edited from the Event Log Mailer by clicking on the Edit CFG button on the main tab.

Setting up the Sensor Configuration File

For submittals to the SAMOS program certain Derived Average Sensors must be added to the sensor config file. These sensors are described in Appendix 3. For use with other programs you may use other types of sensors in the sensor config file.

Setting up the Event Logger Template File

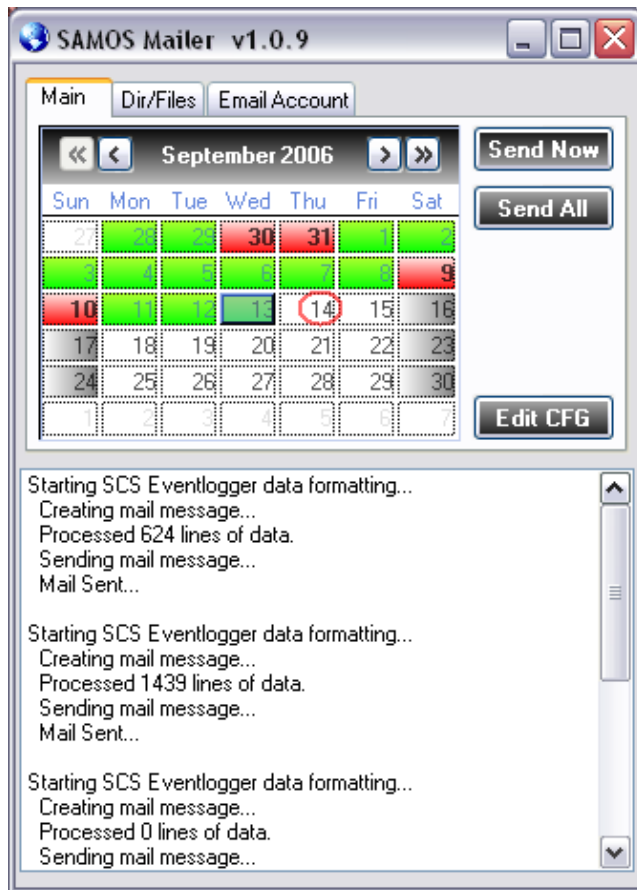
The Event Log Mailer requires a specific event template to be run using the SCS Event Logger client software. This event creates a MetaDataSensorDescription_NNN.TXT (NNN refers to the index number of the file, e.g. 001). This file is vital to the proper operation of the Event Log Mailer. The name of the template must be SAMOS.TPL. Information on the exact spec for the template is found in Appendix 3.

Running the Event Logger

For The event logger for SAMOS template should be started with ACQ and run continuously, uninterrupted from then on out. Event logger will create data files which will span multiple days; SAMOS Mailer will parse these files to send data snapshots (24 hour periods) to the email address defined in the SCS configuration file (see above).

Running the SAMOS Mailer

The mailer app is run independently of the Event Logger. It may be brought up or down as needed during the cruise. The main window comes up like as shown below. The “Main” tab consists of a calendar, a few action buttons, and a log display.



9.4.1 Calendar

The calendar displays the current month with a red circle around the current day. All prior dates in which data was sent are set to green (see Sept 8th). If a date was missed, it is marked red (see Sept 9th).

You also use the calendar to navigate to previous dates using the arrow buttons displayed next to the month at the top of the calendar. You can select a prior date and using the 'Send' buttons as described below to email the data for that date.

9.4.2 The Action Buttons

NOTE: The default operation of this app is to send an email of the previous day's data every day at 0001 GMT.

In addition, the two Send buttons allow for manual interaction with the SAMOS Mailer software. They are located on the Main screen tab; their results may be displayed in the log window.

NOTE: Use of these buttons does not affect the default operation of the app. Mail will always be sent at 0001 GMT.



The *Send Now* button sends data for the currently selected date on the calendar. It only sends the 24 hours of data for that date. This button is useful if data that was sent was not received, or if the data was corrupted for some reason. Also this button comes into play if data was never sent (red days on the calendar) to begin with.

In searching for past data, the Mailer searches through all versions of the file SAMOS-OBS_NNN.ELG (“NNN” here represents the SAMOS event’s index number).



The *Send All* button sends ALL available data to the designated email address. The data in SAMOS-OBS_NNN.ELG is wrapped and sent, for *all* index numbers of this file.

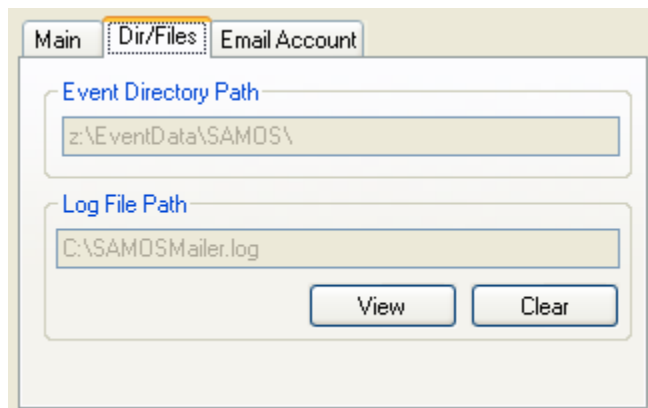
NOTE: This may take some time to process and it may create an email which exceeds your server’s size quota/limit.

As with the *Send Now* button, *Send All* does not impede the normal ‘send cycle’ (the previous day’s data will still be emailed at 0001 GMT, regardless of whether you click this button).



This button opens up the SCS Configuration file editor. Unlike editing this file with the SCS Configuration File Editor app, this button provides that changes made to the file will be reflected in the SAMOS Mailer software immediately. A restart of the Mailer app is not necessary.

9.4.3 Dir/Files tab



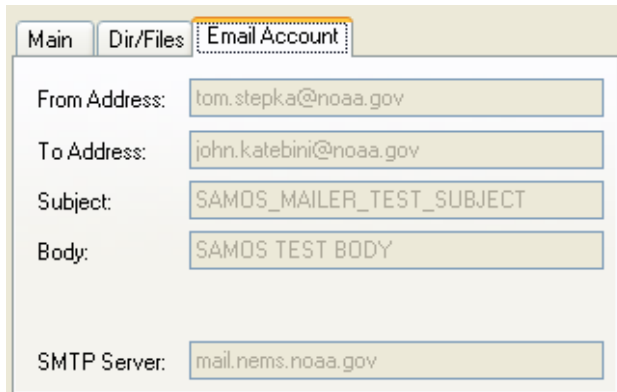
This tab displays information pulled from the SCS Configuration file. Namely the location of the SAMOS event logger data repository and the path to the log file.

Click on ‘View’ in order to view the complete log file. Click on ‘Clear’ to empty the log and start a fresh one.

Note: This is for display purposes only. To change values, please edit the SCS Configuration file as described above.

9.4.4 Email Account

The Mailer app generates email using the SMTP mail protocol.

A screenshot of the 'Email Account' tab in a software interface. The tab is highlighted with an orange border. Below the tab are five text input fields. The first field is labeled 'From Address:' and contains 'tom.stepka@noaa.gov'. The second is labeled 'To Address:' and contains 'john.katebini@noaa.gov'. The third is labeled 'Subject:' and contains 'SAMOS_MAILER_TEST_SUBJECT'. The fourth is labeled 'Body:' and contains 'SAMOS TEST BODY'. The fifth is labeled 'SMTP Server:' and contains 'mail.nems.noaa.gov'.

This tab displays information pulled from the SCS Configuration file; namely the settings regarding SMTP and mail information. The server, to/from email addresses, a subject and body are available for preview.

Note: This is for display purposes only. To change values, please edit the SCS Configuration file as described above.

9.4.5 System Tray



SAMOS Mailer resides mainly in the system tray. Since the software should run continuously and behind the scenes, you can always find its status and quick functions available via its icon of the globe. If the globe is blue then all is well, if however the globe is red, then a problem has been encountered within the last 24 hours and you might want to open the main interface or log to see if it is an issue.

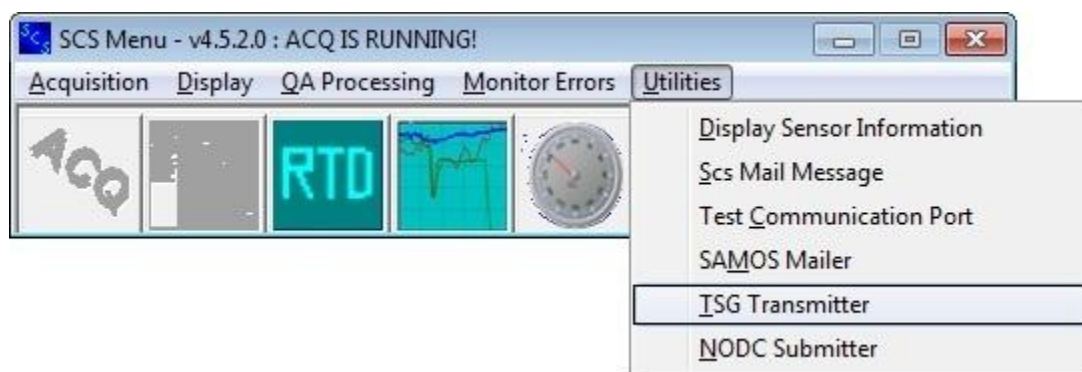


Right click on the icon to access common functions.

The action buttons which send data are available, as well as access and control of the log. Clicking 'Exit' will exit SAMOS Mailer and stop the 'send cycle' for good until the software is restarted.

Note: When you click on the red X on the main interface, or attempt to close SAMOS Mailer in the normal ways, it will not close, simply minimize back to its system tray icon...

9.5 TSG Transmitter



Introduction

The TSG Transmitter app is very similar in both form and function to the SAMOS Mailer app discussed in the sections above. It was added to the SCS suite to provide daily TSG data to the NOAA Atlantic Oceanographic and Meteorological Laboratory (AOML) in Miami, Florida.

Like SAMOS Mailer, this app works with a specialized Event Log. It wakes up once per day at 0001 GMT, reformats the previous day's data contained in the Event Logger's ELG file, wraps it up in a ZIP file, and mails it to the AOML data center.

Differences in the formatting of the data in the ZIP files mean that two separate mailer apps are required. Future versions of SCS this app will be more flexible to allow for generation of mailings in other formats to other organizations on other schedules.

Appendix 4 discusses the procedures for NOAA ships to send data to the AMOL/TSG project.

Setting up the SCS Configuration File

A few new variables need to be added to the SCS Configuration File *SCS40.CFG* before running Event Log Mailer. Note the specific references to TSG project.

- *TSG_EMAILSYSTEMNAME* is the host name or IP address of the ship's mail server. Typically this parameter has the same value as *SHIP_EMAILSYSTEMNAME* in this configuration file.
- *TSG_MAIL_BODY* defines a short text string you would like to include in the body of the emails. There are no specific requirements for the contents of this string.
- *TSG_MAIL_FROM* is the 'from' address which appears in the email. Typically this is the email address of the SCS administrator.
- *TSG_MAIL_TO* is the email address to send the data files to.
- *TSG_MAIL_SUBJECT* is a string which you want to appear in the subject line. This should include the name of the ship and its radio call letters.

Example settings appear below. They are specific to a TSG Transmitter setup on the NOAA Ship Oscar Dyson. – change them to match your ship.

Note the AOML address to send the data to. This is the only value that you must use.

TSG_EMAILSYSTEMNAME=mail.nems.noaa.gov;

TSG_MAIL_BODY=TSG data from the NOAA Ship OSCAR DYSON (WTER);

TSG_MAIL_FROM=chiefst.oscar.dyson@noaa.gov;

TSG_MAIL_TO=aoml.tsgdata@noaa.gov;

TSG_MAIL_SUBJECT=TSG Mail From the NOAA Ship Oscar Dyson (WTER);

Note: The configuration file can be edited from the Event Log Mailer by clicking on the Edit CFG button on the main tab.

Setting up the Sensor Configuration File

The sensors supplying to the TSG event logger have no special characteristics. Unlike SAMOS sensors, they are not averaged.

Setting up the Event Logger Template File

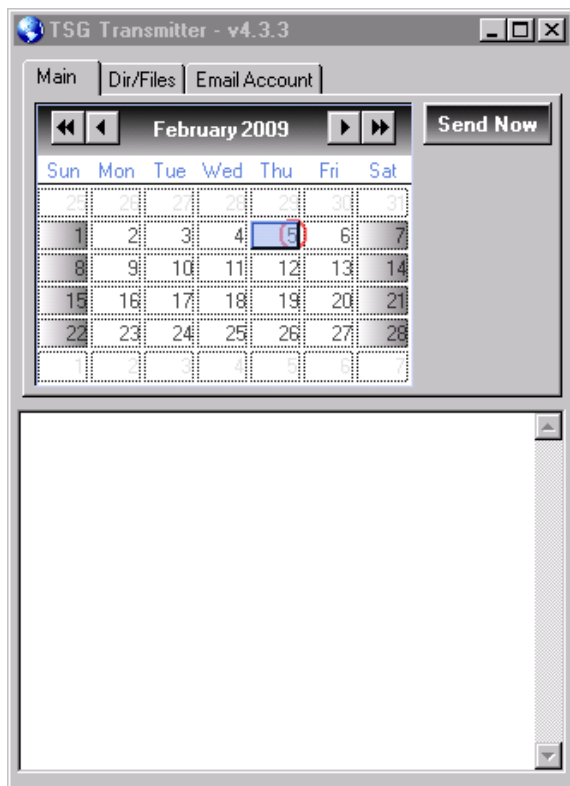
The TSG Transmitter requires a specific event template to be run using the SCS Event Logger client software. This event creates a file called MetaDataSensorDescription_NNN.TXT (NNN refers to the index number of the file, e.g. 001). This file is vital to the proper operation of the TSG Mailer. The name of the template must be TsgTransmitter.TPL. Information on the exact spec for the template is found in Appendix 3.

Running the Event Logger

For The event logger for TSG template should be started with ACQ and run continuously, uninterrupted from then on out. Event logger will create data files which will span multiple days; TSG Transmitter will parse these files to send data snapshots (24 hour periods) to the email address defined in the SCS configuration file (see above).

Running the TSG Transmitter

The mailer app is run independently of the Event Logger. It may be brought up or down as needed during the cruise. The main window comes up like as shown below. The “Main” tab consists of a calendar, a few action buttons, and a log display.



9.5.1 Calendar

The calendar displays the current month with a red circle around the current day. All prior dates in which data was sent are set to. If a date was missed, it is marked red (these colors are not illustrated on the above figure).

You also use the calendar to navigate to previous dates using the arrow buttons displayed next to the month at the top of the calendar. You can select a prior date and using the 'Send' buttons as described below to email the data for that date.

9.5.2 The Action Buttons

NOTE: The default operation of this app is to send an email of the previous day's data every day at 0001 GMT. This happens automatically. No buttons need to be pushed to make this happen.

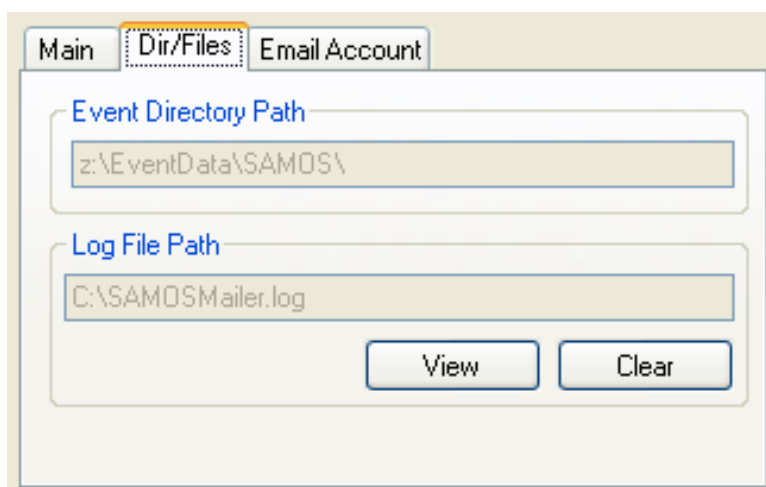
In addition, the Send Now button allows for manual interaction with the SAMOS Mailer software. It is located on the Main screen tab; their results may be displayed in the log window.



The *Send Now* button sends data for the currently selected date on the calendar. It only sends the 24 hours of data for that date. This button is useful if data that was sent was not received, or if the data was corrupted for some reason. Also this button comes into play if data was never sent (red days on the calendar) to begin with.

In searching for past data, the Mailer searches through all versions of the file TSG-OBS_NNN.ELG ("NNN" here represents the TSG event's index number).

9.5.3 Dir/Files tab



The screenshot shows a window with three tabs: 'Main', 'Dir/Files' (selected), and 'Email Account'. Under the 'Dir/Files' tab, there are two text input fields. The first is labeled 'Event Directory Path' and contains the text 'z:\EventData\SAMOS\''. The second is labeled 'Log File Path' and contains the text 'C:\SAMOSMailer.log'. Below the 'Log File Path' field, there are two buttons: 'View' and 'Clear'.

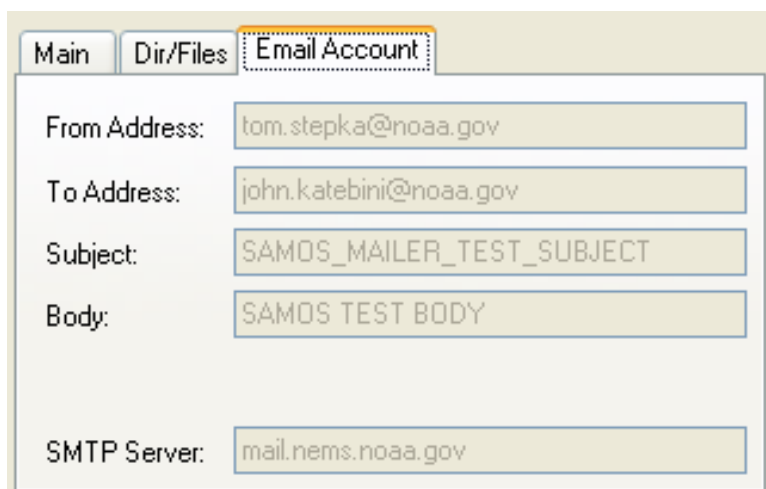
This tab displays information pulled from the SCS Configuration file. Namely the location of the TSG event logger data repository and the path to the log file.

Click on 'View' in order to view the complete log file. Click on 'Clear' to empty the log and start a fresh one.

Note: This is for display purposes only. To change values, please edit the SCS Configuration with the SCS Configuration File Editor.

9.5.4 Email Account

The TSG Transmitter app generates email using the SMTP mail protocol.

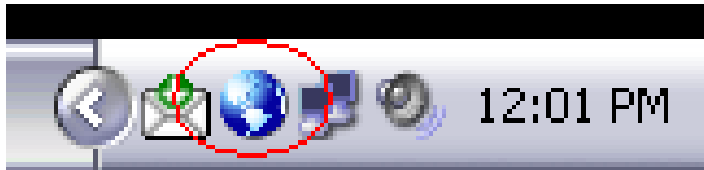


The screenshot shows a window with three tabs: 'Main', 'Dir/Files', and 'Email Account' (selected). Under the 'Email Account' tab, there are five text input fields. The first is labeled 'From Address:' and contains 'tom.stepka@noaa.gov'. The second is labeled 'To Address:' and contains 'john.katebini@noaa.gov'. The third is labeled 'Subject:' and contains 'SAMOS_MAILER_TEST_SUBJECT'. The fourth is labeled 'Body:' and contains 'SAMOS TEST BODY'. The fifth is labeled 'SMTP Server:' and contains 'mail.nems.noaa.gov'.

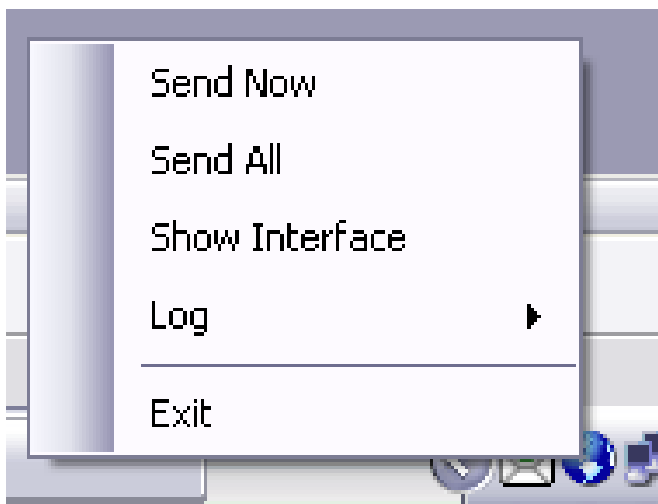
This tab displays information pulled from the SCS Configuration file; namely the settings regarding SMTP and mail information. The server, to/from email addresses, a subject and body are available for preview.

Note: This is for display purposes only. To change values, please edit them with the SCS Configuration File Editor app.

9.5.5 System Tray



TSG Transmitter app resides mainly in the system tray. Since the software should run continuously and behind the scenes, you can always find its status and quick functions available via its icon of the. If the globe is blue then all is well, if however the globe is red, then a problem has been encountered within the last 24 hours and you might want to open the main interface or log to see if it is an issue.



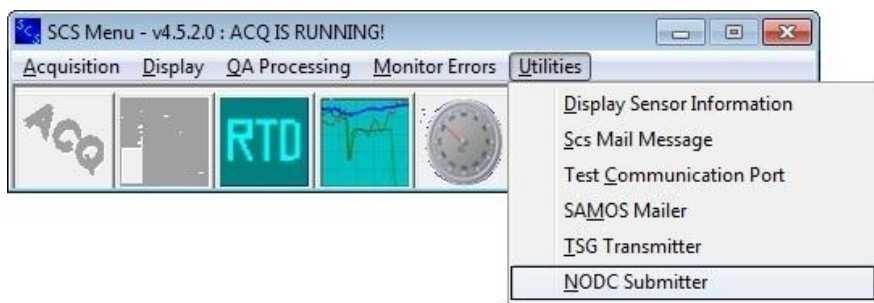
Right click on the icon to access common functions.

The action buttons which send data are available, as well as access and control of the log.

Clicking 'Exit' will exit TSG Transmitter and stop the 'send cycle' for good until the software is restarted.

Note: When you click on the red X on the main interface, or attempt to close TSG Transmitter in the normal ways, it will not close, simply minimize back to its system tray icon...

9.6 NODC Submission Application



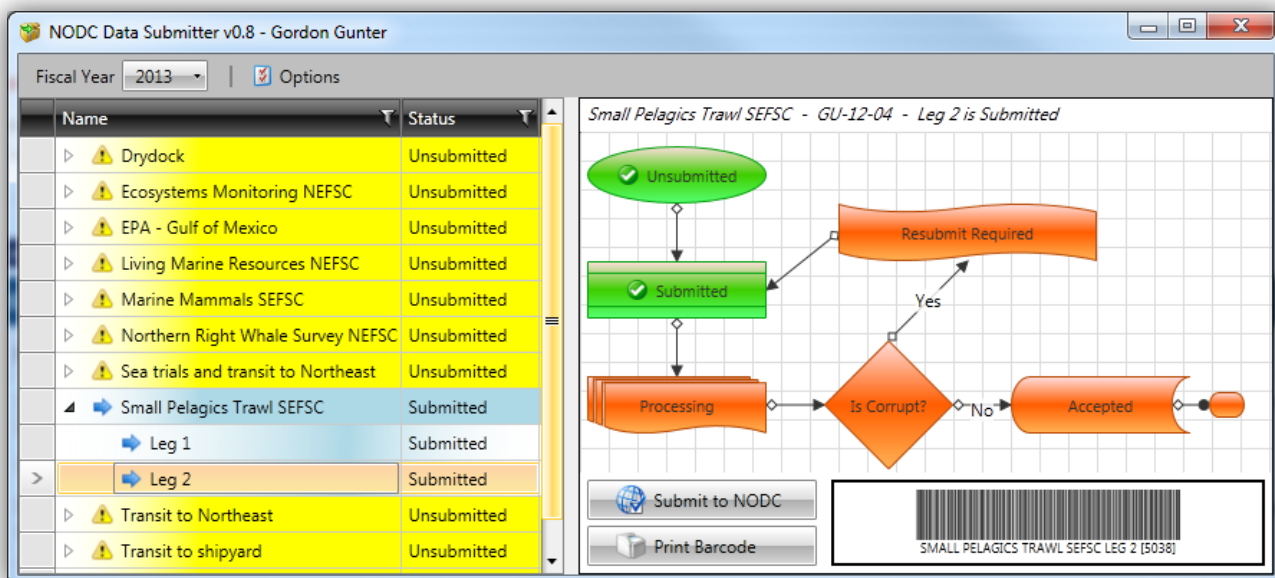
Operating Instruction

The NODC Submitter software is an attempt to make collection and submission of SCS data to the national archive less of a manual process. The software displays a list of cruise legs for your ship and will automatically find all data files (in RAW data folder), zip them up with the appropriate metadata and ftp them over to NODC. In addition the new software allows for the ship to get feedback from NODC regarding their submission. The feedback consists of the current status of the submission and [when published] a public link to a webpage where they can access that data from that point forward.

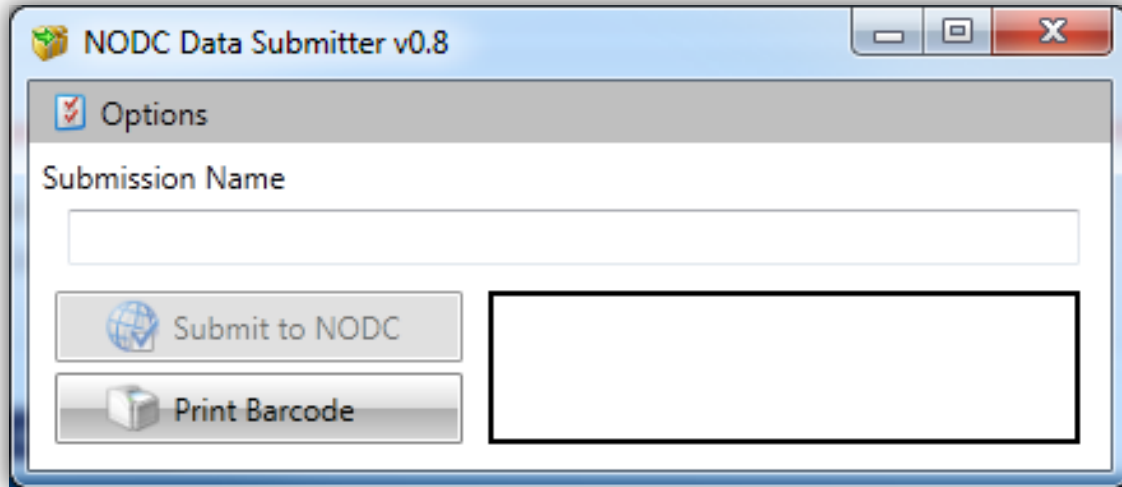


Program Interface

Large NOAA vessel (default):

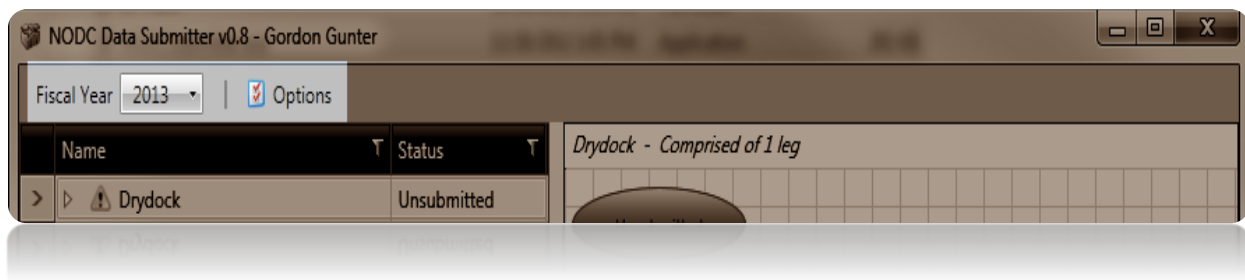


Other Vessel:

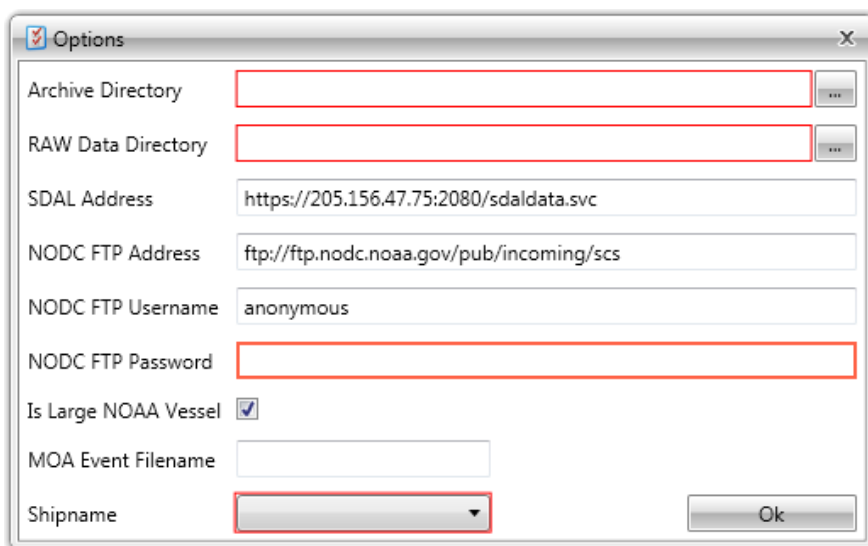


Settings

The settings are accessed via the menu at the top of the program interface.



- The first item is a dropdown box which allows you to change the fiscal year (Large NOAA Vessel only)
- The second item brings up a more detailed settings/options display.



Archive Directory

Once a cruise leg is selected and submitted the entire submission is wrapped up in a zip file. The Archive Directory is the directory to archive/store this zip file after it's been submitted. This way you can keep a copy of all your submissions, in a compressed format, on a local drive or data store.

RAW Data Directory

In order to submit SCS data the software has to have knowledge of where you log your RAW data. The RAW Data Directory is the directory all SCS data is logged to (RAW Data Directory as specified in the SCS40.cfg file by the SCSLOG variable).

SDAL Address

In order to have a list of cruises and cruise legs the SCS NODC Submitter utilizes the ship schedules as reported by SDALs. A web-service has been deployed to provide this software with limited access to the SDALs database. The default address of the service is:
<https://205.156.47.75:2080/sdaldata.svc>

Note: Do not change this value unless directed to do so by an SCS/SDAL administrator.

NODC FTP Address

This is the address of the FTP site used by NODC for entities to submit their data. All SCS data submissions to NODC should be ftp'd to: <ftp://ftp.nodc.noaa.gov/pub/incoming/scs>

Note: Do not change this value unless directed to do so by an SCS/NODC administrator.

NODC FTP Username

Access to the NODC FTP server is more or less anonymous. The authentication requires the username to be anonymous and the password to be the submitter's email address.

The default value for this item is: *anonymous*

Note: Do not change this value unless directed to do so by an SCS/NODC administrator.

NODC FTP Password

Access to the NODC FTP server is more or less anonymous. The authentication requires the username to be anonymous and the password to be the submitter's email address.

Please enter your email address as the *NODC FTP* Password.

Is Large NOAA Vessel

If checked then the software will attempt to connect to the SDALs web-service in order to get the ship's schedule and other metadata.

If unchecked then the software will allow the user to submit their data but will require manual entry of cruise information and dates. If your ship appears in the Shipname dropdown then leave this checked. If your ship does not appear in the dropdown then uncheck this option.

MOA Event Filename

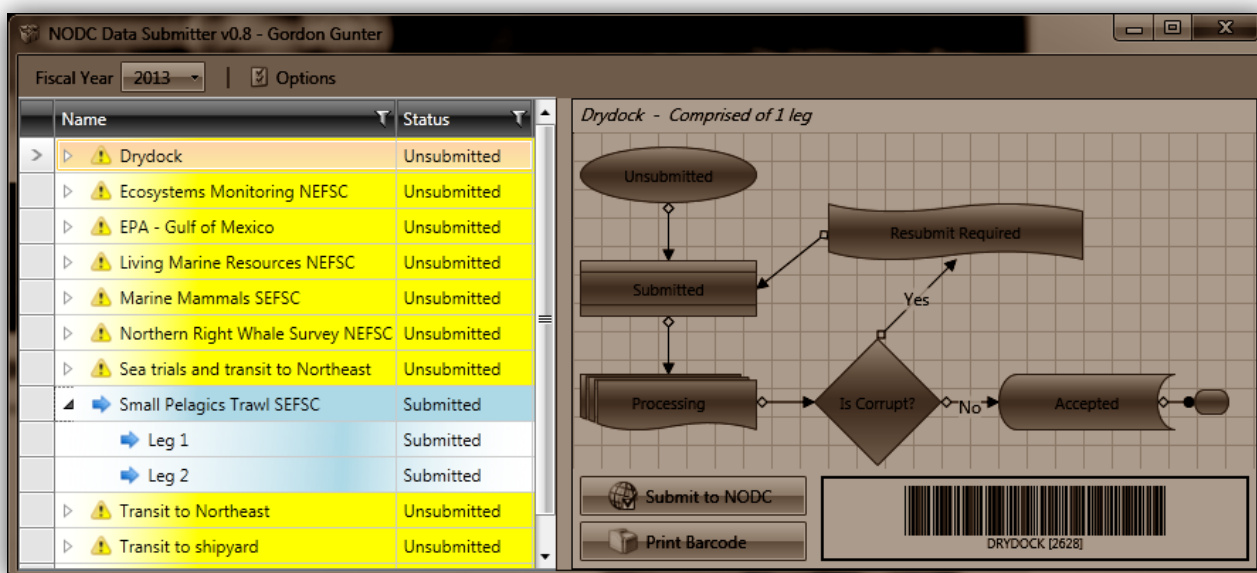
If your ship is running a MOA (Marine Operations Abstract / Log) SCS event, then additional metadata can be retrieved from the event data and submitted with your RAW dataset.

If you run this event please enter the name of the event file (eg MOA.tpl). Do not enter the full path (directory), just the filename (and *don't forget the .tpl extension*)!

Shipname

If you're one of the large NOAA vessels, please select your ship's name from this dropdown. If not, then please uncheck the Is Large NOAA Vessel option.

Cruise / Cruise Leg Selection

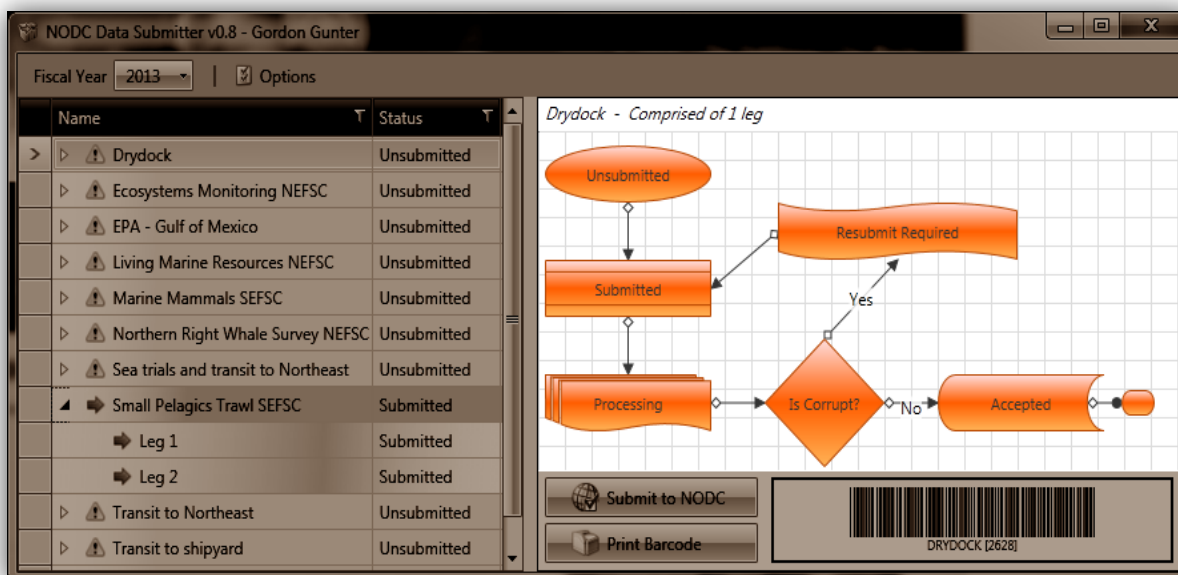


Once you've selected the appropriate Fiscal Year and Shipname the list of all Cruises (a.k.a. Projects) and Cruise Legs (a.k.a. Periods) are pulled from the SDALS system and displayed in a tree format on the left hand side of the primary screen.






If the schedule is incorrect, please notify your OPS officer who will contact SDALS support on the shore to correct the issues.

Helpful Tip: Hover the mouse cursor over a Cruise to get the cruise ID.

Submission Status



When a cruise, or cruise leg, is selected then the current status of the submission is displayed to the right. Each cruise leg is in one of the following states:

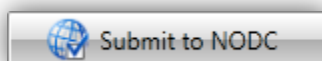
-  **Unsubmitted** This dataset has not been sent to NODC yet.
-  **Submitted** This dataset has been sent to NODC and is awaiting processing.
-  **Processing** This dataset has been received by NODC and is being processed, but is not published yet.
-  **Corrupt** NODC has encountered issues while processing the dataset, please resubmit.
-  **Accepted / Published** This dataset has been processed, published and fully Accepted by NODC. Consult SDALS for a link to its public landing page.

The status of a cruise is determined by the 'sum' of its cruise legs. The cruise is only as good as its worst leg.

For instance, a cruise isn't marked as Accepted/Published until all of its legs are marked as Accepted/Published. If a cruise has 3 legs, 2 of which are Accepted/Published and 1 of which is Unsubmitted, then the cruise isn't marked as Accepted/Published until the 3rd and final cruise leg moves all the way from Unsubmitted to Accepted/Published also.

Submission Screen

Once you've selected a cruise leg, and filled out the required configuration variables in the Options screen, the Submit to NODC button will become enabled.



Clicking the Submit button brings up the submission screen.

Submit to NODC

You have selected to submit **Leg 2 (ID# 5038) of the Small Pelagics Trawl SEFSC cruise.**
This leg has already been submitted.

You may choose to resend the archived file again (recommended method, unless NODC tells you otherwise) or regenerate a new package to send them. If the package is not in the archive (never archived, deleted from archive, etc) then the package cannot be resubmitted and a new one will be generated for you.

Select local archiving action:

Do not archive

Copy to local archive

Move to local archive

Please verify date range(s) to send:

Leg 2 — Start: 11/1/2012 End: 11/20/2012

Please verify submitter information (required by NODC):

First Name

Last Name

E-mail Address

Submit Data

Cancel

Operation Status

Archive Action

A dialog box titled "Select local archiving action:" with three buttons: "Do not archive", "Copy to local archive", and "Move to local archive". The "Copy to local archive" button is highlighted in yellow.

Select local archiving action:

Your dataset submission consists of multiple files, both RAW data and metadata. These files are compressed into a single zip file in a temp location. After the FTP submission is complete, the zip file is deleted.

If you choose to archive then prior to the cleanup/deletion phase the zip file is saved to the *directory you specified in your settings*. This is a good way to keep a copy of each cruise's data for your own storage and reference and even to give to the science party so they have a copy of the SCS data for the cruise, too.

If you choose not to archive, then the deletion proceeds as planned and no local record of the dataset is created by the software. It is recommended you then manually move your RAW data files to a save local location (disk, network, DVD, etc).

Note: Move to local archive is not available at this time.

Cruise Leg(s) Date Range

A dialog box titled "Please verify date range(s) to send:" with two rows for cruise legs. Each row has a label (Leg 1, Leg 2), a start date field, and an end date field, each with a calendar icon.

Please verify date range(s) to send:

Leg 1 — Start: 10/10/2012 End: 10/29/2012

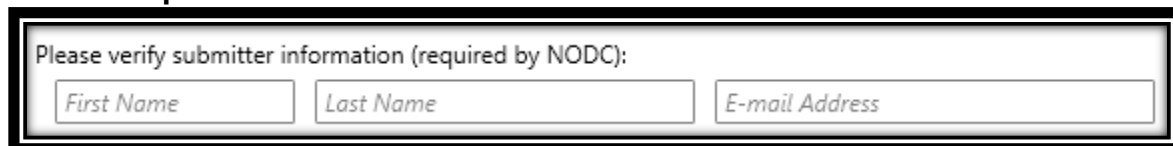
Leg 2 — Start: 11/1/2012 End: 11/20/2012

All cruise legs and their start/end dates will appear in this section. If you are submitting a single leg then only one leg will appear in the list. If you selected a cruise then each leg will appear in this section.

The cruise leg start/end dates are pulled from SDALs. If they are incorrect (eg you started a day late or came back a day early, etc) then adjust them so they accurately represent your leg.

The software will go to the directory you specified as the *RAW Data Directory*, gather files created in these date ranges and add them to the zip file to be submitted to NODC.

NODC Requested Data

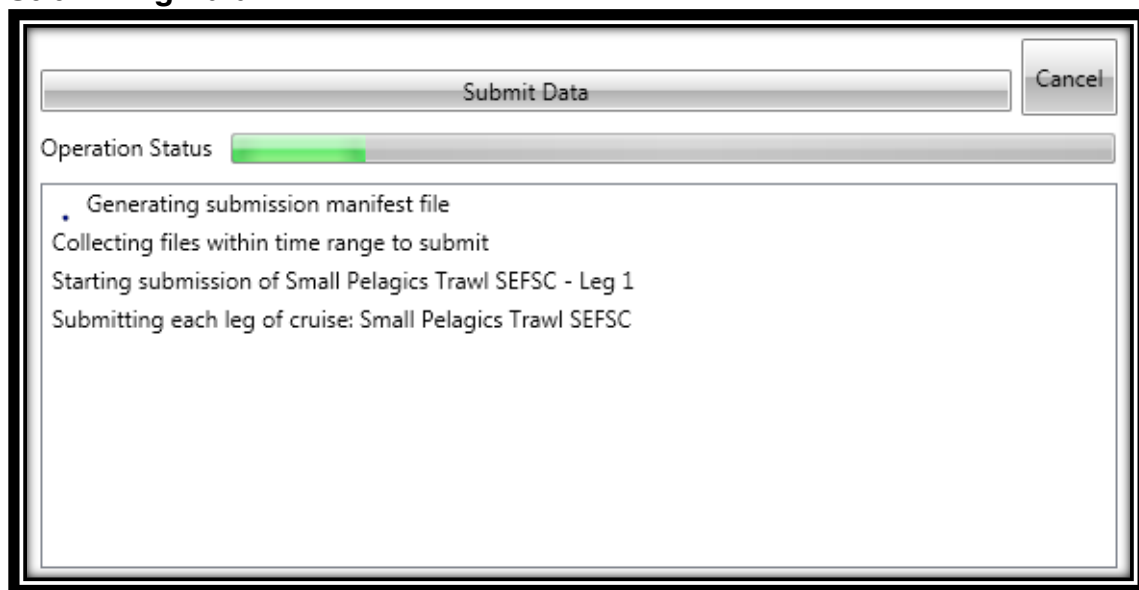
A dialog box titled "Please verify submitter information (required by NODC):" with three text input fields: "First Name", "Last Name", and "E-mail Address".

Please verify submitter information (required by NODC):

NODC requests that the submitter's name (First, Last) and contact email address are supplied with each submission. This is independent of the *email address/password* you supplied in the *Options* page.

Review this information and make sure it correctly identifies you.

Submitting Data



Once you've filled out the information above the Submit Data button will become enabled.

Clicking the Submit Data will iterate through each leg, find all files, zip them up, FTP them over to NODC and update the submission status' in SDALs.

The box below the Submit Data button will contain a running log of what the software is currently doing and report any errors it encounters.

*** Note: This operation can be quite lengthy, it's recommended you do this on a dedicated connection and not over VSAT if possible.*

Quick Start / Recommended SOP

1. Fill out the Options for your ship.
2. At the end of a cruise leg start the NODC submission software and select the leg in the list.
3. Click the Submit to NODC button.
4. Verify that Copy to Local Archive is selected
5. Verify the leg start/end dates.
6. Verify the submitter information.
7. Click Submit Data
8. Monitor prior submissions and ensure they are moving forward towards the accepted status.
9. Once a submission has reached the accepted status, you can delete those RAW files from your local *LOG40* directory. You can now get this data from either your local archive or the NODC archive.

Note: Once a cruise has been accepted it's a good idea to burn the local copy/zip of the cruise leg from your archive to a DVD. You can select the cruise leg from the list and click the 'Print Barcode' button on the main screen to have a scan-able barcode to stick on the DVD. Or print it to a file and include it with the zip on the DVD.

APPENDIX ONE

SENSOR PARAMETER DICTIONARY

10 APPENDIX ONE – The Sensor Parameter Dictionary

This Appendix provides an *alphabetical* listing of all sensor parameters controlled by the Configuration File Editor (CFE-DB) program.

Most of these parameters are used by ACQ only, but some are used by other SCS apps, and these are mentioned in the Remarks section of each parameter entry.

10.1 Format of Entries in This Dictionary

Entries in this dictionary appear in the following format:

X.X.X Parameter Name

Description: A thumbnail description of the parameter.

Defined for: Objects (Physical Device, Sensor Device, Message, Data Field) for which the parameter is defined. (See Chapter 3 - Acquisition for descriptions of these objects).

Type: Nature of the parameter value:

Text: A string of characters (usually a name or other descriptor).

Number: A numeric value.

Base sensor: The ID (number) and name of another sensor in the file.

List: A list of predetermined values from which you select one.

Check box: The parameter value is displayed as a check box, which you either check or leave empty.

Range: Range of acceptable values for the parameter. Different information is given, depending on the type of the parameter. Note: Consult the Interactions section for other restrictions on the range.

Text: The minimum and maximum string length and restrictions on the content are given.

Number: The minimum and maximum allowable values, possibly restricted to integer (whole number) values

List: Selections in the list, with explanations.

Check box: For a condition: True or False, for a feature: Enabled or Disabled

Interactions: In addition to the limits described in “Range”, a parameter’s value may depend on the values of other parameters of the same sensor, of the same parameter of other sensors in the configuration file. Any such interactions are described here.

Remarks: Additional explanations and restrictions as needed.

See also: References to other parameters in the dictionary relating to this one.

10.2 SENSOR PARAMETERS

Average Interval

Description: The length of time to collect base data samples before averaging them.

Defined for: Sensor Devices of Derived Average type.

Type: Number

Range: One to 86400, whole number only.

Interactions: None

Remarks: This parameter is activated by selecting the “By Interval” radio button in the Sensor Device definition screen. Average will be computed on the N most recent samples, regardless of how long it took to collect those samples.

See also: Average Type, Number of Samples.

Average Type

Description: The type of averaging to be performed on the base sensor’s data.

Defined for: Sensor Devices of Derived Average type

Type: List

Range: Polar: Used for sensors whose output is in the range 0 to 360, such as gyros and COG, and Wind Direction

Arithmetic : Used for all other sensors.

Interactions: None.

Remarks: During acquisition each average value is logged with the time of the earliest of the samples used to compute the average.

See also: Number of Samples, Average Interval

Baud Rate

Description: The speed (in bits per second) of a serial connection through a communications port.

Defined for: COM type sensor devices

Type: Number

Range: 110 – 115000, whole numbers only, or “Default”

Interactions: None

Remarks: You may make a selection from the drop down list, or you may enter your own value into the edit box. “Default” instructs ACQ to accept the pre-existing baud rate for the port.

See also: Parity, Stop Bits, Device, and Data Bits

COM Port

Description: The COM port through which sensor messages will flow to ACQ

Defined for: Sensor Devices of COM or Polled COM type.

Type: Drop down list

Range: 0 – 99

Interactions:

Remarks:

See also: Device Type

Code Translation

Description: A table of data field values and their translations. It is used to translate coded data field values into a string meaningful to users. The translated string replaces the original code in the RAW file and in all data displays.

Defined for: Data Fields of NMEA Messages

Type: Strings

Range: Not applicable

Interactions: None

Remarks: See example in Chapter Three - Acquisition

See also: None

Command Prompt

Description: Character string needed to poll a sensor for another message

Defined for: Polled Serial Messages

Type: Text

Range: Nineteen characters or fewer. The decimal ASCII value of each character must be in the range one to 127.

Interactions: None.

Remarks: To include a control character in the string, enclose its decimal ASCII value in angle brackets. Example: Linefeed is entered as "<10>"

See also: None.

Comment

Description: A free form text field to use as you wishes. Use this to make notes or documentation about the sensor. This field is ignored by all SCS software. However, for Polled Serial Parent sensors, it can also be used to signal to ACQ that this sensor is queued. See Chapter Three for details.

Defined for: All objects

Type: Text

Range: No restrictions

Interactions: None.

Remarks: None.

See also: None.

Data Bits

Description: Number of data bits in each character received through a serial communications port.

Defined for: Sensor Device Object

Type: List

Range: Seven, Eight

Interactions: None.

Remarks: There is no guarantee that these will be supported by your hardware.

See also: Device, Baud Rate, Parity, and Stop Bits

Data Field Category

Description: The nature of the measurement that a data field value represents.

Defined for: Data Fields

Type: List of categories

Range: Indeterminate

Interactions: With Base Data Fields in Messages of Derived Sensors.

Remarks: The list of base sensors displayed in each Base Data Field List is restricted to sensors of the appropriate type of Data Field Category. For example, in the Derived True Wind message, only sensors that are declared of category “Heading” will be listed as available sensors in the “Gyro Base Data Field” drop down list. It is the responsibility of the user to ensure all data fields are appropriately categorized.

Data Field Position

Description: Position of the data field of a child sensor within the parent’s message.

Defined for: Data Fields of NMEA messages. Data Field Position is reserved for future use in Data Fields of Serial and Polled Serial Messages. You must use the By Char Position option.

Type: Number

Range: One or more, whole numbers only

Interactions: Value must be less than or equal to the Record Size of the parent sensor, minus the length of the parent’s Sentence Label, all divided by two.

Remarks: Data Field Position is counted as the number of comas preceding the data field. As of SCS 4.3.4 only NAVSEC decode types have data field positions that are handled. For NAVSEC data fields, the Data Field Position locates the quadrant of the lat or lon data field. Valid values are “N”, “S”, “E” or “W”. For NAVSEC data fields, the quadrant is added to the value field as a suffix, forming a single NAVSEC value.

See also: Units Field Position, Decode Type

Decode Type

Description:	Specifies how a measurement is to be interpreted after it has been extracted from a message.
Defined for:	Data Fields
Type:	List
Range:	NAVSEC. An ASCII text string in the form DDDMM.MMMM ^Q , used to specify latitudes and longitudes by GPS receivers.
ASCII:	An ASCII character string representing a numeric value that will be parsed and used for derived calculations, plot displays, gauge displays and summary meta item calculation by the Event Logger.
Real*4:	An eight-byte binary value to be interpreted as a double precision floating point number
Real*8:	A four-byte binary value to be interpreted as a single precision floating point number
Int*4:	A four-byte binary value to be interpreted as an integer (whole number).
Interactions:	None
Remarks:	Real*4, Real*8 and Int*4 are reserved for future use. NAVSEC is used to specify a GPS latitude or longitude value. ASCII is used to indicate any other number format besides NAVSEC.
See also:	Data Field Position, Units Field Position

Delimiters

Description:	A list of valid delimiters of data fields
Defined for:	Messages of NMEA, Serial and Manual types
Type:	None
Range:	None
Interactions:	None
Remarks:	Not yet implemented.
See Also:	None

Delta Check

Description:	Enables or disables “delta checking”
Defined for:	Data Fields
Type:	Check box
Range:	Enabled or Disabled.
Interactions:	None.

- Remarks:
1. A delta check compares consecutive sensor values and flags changes that are too big. Delta checks are performed by the “*DataMon*” real time quality assurance tool.
 2. When Delta Check is disabled the Delta High and Delta Low edit boxes are disabled any you cannot change or enter any values.

See also: Delta High and Delta Low

Delta High

Description: The largest allowable difference from one sensor value to the next.

Defined for: Data Fields

Type: Number

Range: No restrictions

- Interactions:
1. Must be greater than Delta Low, taking sign into account. For example, -3 is less than -2, although its magnitude is bigger.
 2. Delta High is disabled if Delta Check is disabled.

- Remarks:
1. Subtract the previous value from the next value. The difference must be less than Delta High, taking signs into account.
 2. The DataMon real time quality checking software performs delta checks
 3. The following examples illustrate the effects of various values of Delta High and Delta Low.

Previous Value	Delta Low	Delta High	Acceptable Next Value
6.0	2.0	3.0	8.0 to 11.0
6.0	-2.0	4.0	4.0 to 10.0
6.0	-7.0	-5.0	-1.0 to 1.0

See also: Delta Check and Delta Low

Delta Low

Description: The smallest allowable difference from one sensor value to the next.

Defined for: Data Fields

Type: Number

Range: No restrictions.

- Interactions:
1. Must be less than Delta High, taking sign into account. (See Delta High)
 2. Delta Low is disabled if Delta Check is disabled.

Remarks: Subtract the previous value from the current value. The difference must be greater than Delta Low. Refer to remarks for Delta High for examples.

See also: Delta Check and Delta High

Device Order

Description: The position of an Sensor Device object in the Sensor Device List

Defined for: Sensor Devices of all types

Type: Integer

Range: One, up to the number of Sensor Devices in the Sensor Devices List

Interactions: None

Remarks: You can change a Sensor Device's order through drag and drop or cut/copy/past operations.

See also: None

Device Type

Description: The type of Sensor Device being configured.

Defined for: All Sensor Device objects

Type: Pull Down List

Range: COM, Derived Arithmetic, Derived Average, Derived Linear Equation, Derived True Wind, Manual, Network, Polled COM

Interactions: None

Remarks: Establishes the nature of the source of the sensor messages.

See also:

Eligible Base Data Field

Description:

Defined for: Data Fields

Type: Check Box

Range: True or False

Interactions: ????

Remarks:

See also:

Enabled

Description: Determines whether SCS software will ignore a Sensor Device in the Sensor Configuration File.

Defined for: All Sensor Device Objects

Type: Check box

Range: Enabled or Disabled.

Interactions: Disabling a Sensor Device is subject to the same sorts of restrictions as deleting a Sensor Device:

You cannot disable a Sensor Device if any data field under it is the base of any derived sensors.

You cannot enable a derived sensor if its base sensor is disabled.

Remarks: “Enabled” applies only Sensor Devices. When a Sensor Device is disabled/enabled, all messages data fields for that Device are also disabled/enabled.

There is no way to enable or disable message objects individually.

All disabled devices and messages are excluded from writing to DeviceConfiguraron.XML file during the Export/Publish function.

See also: None.

Ending Char Position

Description: The location of the last character of a data field within a message

Defined for: Data Fields of Serial and Polled Serial Messages

Type: Number

Range: One or more, whole numbers only

Interactions: Must be greater than or equal to Starting Char Position for the same message, and less than Record Length. See Starting Char Position entry for other restrictions.

Remarks: None.

See also: Starting Char Position, Record Length.

Extend Label

Description: Enables the use of label extenders, a means of specifying that the content and meaning of one data field depends on the content of another field.

Defined for: Messages of NMEA type.

Range: Enabled or Disabled

Interactions: None.

Remarks: Refer to Chapter 3 - Acquisition for details on using this feature.

See also: Label Extenders

Installed To

Description: Specifies the connection between a Sensor Device and a Physical Device

Defined for: Sensor Devices of all types

Interactions: None

Remarks: See Chapter Three Acquisition for details on the process of installing a Sensor Device

See also: None

IP Address

Description: The IP address of the Physical Device from which ACQ will receive data.

Defined for: Sensor Device of Network type

Range: Four integers, each in the range 0-255.

Interactions: None.

Remarks: ACQ opens a connection to the Physical Device on the specified Port Number in order to receive messages over the LAN. Refer to *Chapter 3 - Acquisition for Details* on the use of manual sensors. User must ensure there are no conflicts in IP addresses on the LAN.

See also: Port Number

Is Template

Description:

Defined for: Sensor Devices, all types

Type: Check Box

Range: True or False

Interactions:

Remarks:

See also:

Label Extenders

Description: Refer to *Chapter 3 - Acquisition for Details* on using this feature.

Defined for: Messages of NMEA type.

Type: Text

Range:

Interactions:

Remarks:

See also: Extend Label

Log Sub Folder

- Description: Directory in which raw message data files (*.RAW) are stored.
- Defined for: All Message types (NMEA, Serial and Derived)
- Type: Text
- Range: You may leave Logging Folder blank. If not blank, the value must be recognizable to Windows NT as a directory name, and must not contain multiple levels of subdirectories. Certain characters having special meaning to the operating system are not permitted.
- Interactions: None.
- Remarks: 1. If Log Sub Folder is blank ACQ will put the raw data files (*.RAW) for the message into the SCSLog directory. If not blank ACQ will create (if necessary) a subfolder under the SCSLog directory and ACQ will put the RAW file there.
2. Refer to section 1.5 of Section One, SCS System Overview in the **SCS User's Guide** for further discussion of RAW files.
- See also: None.

Logging Rate

- Description: Time interval (in seconds) between messages logged to *.RAW files.
- Defined for: All Message types (NMEA, Serial and Derived)
- Type: Number
- Range: Normally, one or greater, whole numbers only
- Special case: -1 instructs SCS to log *a//* messages
- Interactions: None.
- Remarks: ACQ logs the first message or value it receives after Log Rate seconds have passed since the last logged message or value.
- Derived sensor values are computed every Log Rate seconds.

Name

- Description: The descriptive name you give to an object.
- Defined for: All Objects: Sensor Device, Message, Data Field, Physical Device
- Type: Text
- Range: One to 64 characters in length. Underscore is not allowed. Underscore is not allowed. Only characters that are legal in a Windows file name are allowed.
- Interactions: Must not be the same as any other enabled object in the configuration. Comparisons among objects for uniqueness is NON case sensitive.
- Remarks: CFE-DB will convert embedded spaces to hyphens (" - ").
- See also: Enabled.

Network Protocol

Description: The network protocol used to establish a connection between ACQ and a Physical Sensor.

Defined for: Sensor Devices of Network type.

Type: String (Drop Down List)

Range: TCP, UDP Broadcast, UDP Multicast

Interactions: None

Remarks: None

See also: IP Address, Port Number

Number Of Samples

Description: The number of measurements of the base sensor to be averaged.

Defined for: Sensor Devices of Derived Average type.

Type: Number

Range: One or greater, whole numbers only

Interactions: None

Remarks: This parameter is activated by selecting the “By Number of Samples” radio button in the Sensor Device definition screen. The average of the most recent Number Of Samples measurements is computed, regardless of the time span over which those samples are collected. Compare to Average Interval.

See also: Average Interval

Offset

Description: The “b” in the equation $y = mx + b$

Defined for: Derived Line Equation

Type: Number

Range: No restrictions

Interactions: None.

Remarks: Refer to the description of the Derived Line Equation sensor type in Chapter Three, Acquisition for further information.

See also: Slope

Operator

Description: The arithmetic operation used to combine two sensor values.

Defined for: Sensor Devices of Derived Arithmetic type

Type: String (Drop Down List)

Range: Plus, Minus, Times, Divided by

Interactions: None

Remarks: None

See also: None

Parity

Description: Parity of a serial connection through a com port.

Defined for: Sensor Device Objects

Type: List

Range: Odd, Even, Space, None, Mark

Interactions: None.

Remarks: None.

See also: Baud Rate, Device, Stop Bits, and Data Bits

Precision

Description: The number of places behind the decimal point of a measurement shown in displays and plots, and logged to disk.

Defined for: Data Fields of Derived Messages

Type: List

Range: One to eight. “Default” means “display as output from ACQ’s calculation”; this may be as many as 15

Interactions: None.

Remarks: None.

See also: None.

Port Number

Description: The TCP port number that ACQ will listen to for incoming manual or network sensor data.

Defined for: Sensor Devices of Manual or Network types

Type: Number

Range: Integer only.

Interactions:

Remarks: Take care that there are no conflicts between inbound manual and network messages, and outbound SCS messages using the same port numbers. SCS has no way to ensure that there are no conflicts.

See also: IP Address

Queue Order (NOT YET IMPLEMENTED)

Description: Order in which a set of queued polled serial sensors is queried for messages.

Defined for: Messages of Polled Serial type

Type: Integer

Range: Greater than 0

Interactions: None.

Remarks: Not yet implemented. Physical Devices are polled in the order in which their messages appear under the Sensor Device.

See also: None.

Range Check

Description: Enables or Disables Range checking by the DataMon quality assurance tool.

Defined for: Data Fields

Type: Check box

Range: Enabled or Disabled.

Interactions: None.

Remarks: Controls the applicability of the Range High and Range Low parameters. When Range Check is disabled the Range High and Range Low edit boxes are disabled any you cannot change or enter any values. When Enabled, each new sensor value is compared to Range High and Range Low, and is flagged if it is out of range.

See also: Range High, Range Low

Range Low

Description: Lower limit of the acceptable range of a measurement.

Defined for: Data Fields

Type: Number

Range: No restrictions

Interactions: Must be less than Range High for the same sensor. May be disabled by Range Check parameter.

Remarks: See remarks under Range Check

See also: Range Check, Range High

Record Size

Description: Your estimate of the maximum *actual* length of a sensor's message, in bytes.

Defined for: Messages of Serial and Polled Serial types.

Type: Number

Range: 1 to 10000, whole numbers only

Interactions: Synch Position + the length of Synch String must not exceed the Record Size.

Remarks: *Important:* ACQ detects the end of a sensor message by the Termination Character, or counting Record Size characters, *whichever comes first*. If your Message definition also includes a Termination Character your Record Size

estimate should always be well on the high side so that the Term Char is detected before Record Size characters have been received.

See also: Termination Char

Range High

Description: Upper limit of the acceptable range of a measurement.

Defined for: Data Fields

Type: Number

Range: No restrictions

Interactions: Must be greater than Range Low for the same sensor. May be disabled by Range Check parameter.

Remarks: See remarks under Range Check

See also: Range Check, Range Low

Sentence Label

Description: The identifier of a NMEA -183 formatted message.

Defined for: Messages of NMEA, Manual or Derived types.

Type: Text

Range: The minimum length is one character. The maximum length is 24 characters.

Interactions:

- Remarks:
1. When reading NMEA messages the Sentence Label is *case sensitive* in ACQ.
 2. ACQ writes Derived and Manual messages in a NMEA format, including the Sentence Label specified for the message.
 3. Sentence Labels must be unique between all Messages defined under a single Sensor Device.
 4. CFE-DB converts space characters to underscores ('-')
 5. The NMEA-183 standard requires the first character of the label to be "\$". If this is not the case, CFE-DB gives a warning and accepts the label.

See also: None.

Simulation File

Description: The name of a RAW file to supply data to ACQ.

Defined for: Sensor Device of type Simulated

Type: Text

Range: Any legal Windows file name. Name only, no extension or path.

Interactions: Used

Remarks: File must be in the AcqSimulatorData subdirectory of the Ship directory and have a .RAW extension

See also:

Slope

Description: The “m” in the equation $y = mx + b$

Defined for: Sensor Device of Derived Line Equation type.

Type: Number

Range: No restrictions.

Interactions: None.

Remarks: Refer to the description of the Derived Line Equation sensor type in Chapter Three, Acquisition for further information.

See also: Offset

Starting Char Position

Description: Location of the starting character of a data field within a message

Defined for: Messages of Serial or Polled Serial type

Type: Number

Range: One or more, whole numbers only

Interactions: 1. Starting Char Position must be less than or equal to Ending Char Position.
2. Within a group of child sensors having the same parent, the data fields defined for each sub-sensor by the Start/End Character pairs must not overlap each other.

Remarks: None.

See also: Ending Char Position, Record Length

Stop Bits

Description: The number of stop bits in a serial connection through a com port.

Defined for: Sensor Device objects

Type: List

Range: 1, 2

Interactions: None.

Remarks: None.

See also: None.

String Position

Description: The location of the start of the Synch String within a Message

Defined for: Messages of all types

Type: Number

Range: One or greater, whole numbers only

Interactions: 1. String Position + the length of Synch String must not exceed the Record Size.
2. String Position is inoperative if Synch Check is disabled.

Remarks: 1. A Value of N indicates the first character of the Synch String is found at the Nth character position of the message.
2. Synch checks are performed by the DataMon real time quality assurance tool.

See also: Synch Check, Synch String

Synch Check

Description: Determines if a synchronization test will be performed on incoming sensor messages.

Defined for: Messages of NMEA, Serial and Manual types

Type: Check box

Range: Enabled, Disabled

Interactions: If Synch Check is disabled, String Position and Synch String are inoperative and their edit boxes are disabled.

Remarks: Synch Checks are performed by DataMon real time quality assurance tool.
DataMon attempts to locate a Synch String in every message starting at the position given by String Position.

See also: String Position, Synch String

Synch String

Description: A synchronization string, which must be found in a sensor's message.

Defined for: Messages of NMEA, Serial and Manual types

Type: Text

Range: No restrictions

Interactions: 1. String Position + the length of Synch String must not exceed the Record Size.
2. Synch String is inoperative if Synch Check is disabled.

Remarks: Used by the DataMon real time quality assurance tool.

See also: Synch Check, String Position

Termination Char

Description: The character that terminates a sensor message

Defined for: Messages of NMEA and Serial Type

Type: Drop Down List

Range: Normally one to 31. Special selections:

“None” do not search for a termination character. Rely on Record Size to terminate message.

“Any” any control character terminates the message.

Interactions: None.

Remarks: ACQ detects the end of a sensor message by the Termination Character, or counting Record Size characters, *whichever comes first*.

See also: Record Size.

Units

Description: The units in which a sensor sends its data.

Defined for: Data fields

Type: Text

Range: One to 32 characters long, or blank

Interactions: None

Remarks: CFE-DB converts blank input to a single space character. No effort is made to interpret Units or use it in any calculations; it is merely copied to real time displays.

See also: None.

Timeout

Description: Number of seconds to pass since the previous message before ACQ considers a sensor’s data to be “stale”

Defined for: Messages of all types (NMEA, Serial, Derived, Manual)

Type: Number

Range: 1 or greater, whole numbers only, units of seconds. (-1 indicates “legacy mode”; see remarks).

Interactions: None.

Remarks: Effect on ACQ: Logging stops; Derived values based on stale data stops. Effect on Scs Send Message: Stale data is not sent out, it is replaced by blanks. Usage by Event Logger: stale data is not logged in ELG files; no value is. Legacy Mode: Indicated by timeout = -1. In this case timeouts are not called no matter what. ACQ and client apps all use the most recently acquired value no matter how old it is.

Units Field Position

Description: Location in a NMEA parent’s message of the field specifying a data value’s unit of measurement.

Defined for: Data Fields with NAVSEC Decode Type.

Type: Number

Range: One or more, specified as the number of comas preceding the units field in the parent sensor's message. Whole numbers only.

Interactions:

1. Not defined if Decode Type is not NAVSEC.
2. Value must be less than or equal to the length of the parent message, minus the length of the parent sentence label, all divided by two.
3. Units Field Position cannot be the same as Data Field Position.

Remarks: None.

See also: Data Field Position, Decode Type

Update Rate

Description: For polled serial Sensor Devices, the time interval (in seconds) between transmissions of the Output Prompt string. For derived Sensor Devices, the interval between calculations of derived values.

Defined for: Sensor Devices of Derived or Polled Serial types

Type: Number

Range: 1 to 86400 seconds, whole numbers only

Interactions: None.

Remarks: 86400 seconds = 24 hours.

See also: Output Prompt

APPENDIX TWO

INSTALLING SCS WEB SERVICES

11 APPENDIX TWO: Installing SCS Web Services

Introduction

IIS 7.5 is the Web server role in Windows Server® 2008 R2 and the Web server in Windows® 7. Windows Server 2008 R2 operating system has all the IIS features needed to support the hosting of Web content in production environments. Windows 7 and Windows Vista also include IIS features, but the available features depend on the operating system versions.

The Web server was redesigned in IIS 7.x to enable you to customize a server by adding or removing modules to meet your specific needs. Modules are individual features that the server uses to process requests. For example, IIS 7.0 and IIS 7.5 use authentication modules to authenticate client credentials and use cache modules to manage cache activity. Both versions of IIS also provide:

- A new management interface.
- The ability to share tasks with Web site owners.

Before You Begin

Ensure that you have administrative user rights on the computer on which you plan to install IIS 7. Note that by default, you do not have administrative user rights if you are logged on as a user other than as the built-in administrator, even if you were added to the local Administrators group on the computer (this is a new security feature in Windows Server 2008 called Local User Administrator).

Log on either to the built-in administrator account, or explicitly invoke applications as the built-in administrator by using the run as command-line tool.

Note: You can run `runas /user:administrator cmd.exe` so that every application you run from that command line will be elevated, eliminating the need to use the `runas` syntax from that command line.

If you are logged on to an account other than the built-in local administrator account, you may see the following security alert dialog box.

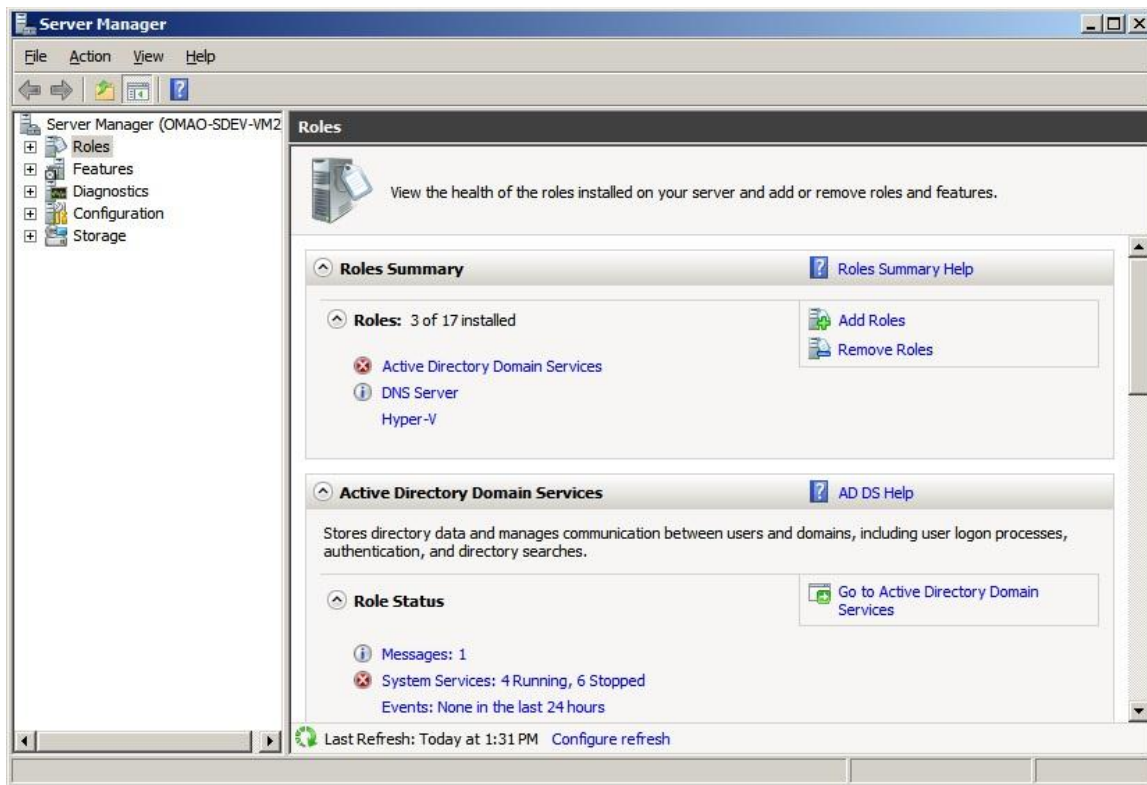


11.1 Install IIS 7.5 on Windows Server 2008 R2

IIS is one of the Windows Server® server roles. IIS 7 can be installed through the graphical user interface (GUI) by using the new Server Manager interface after the Windows Server operating system is installed.

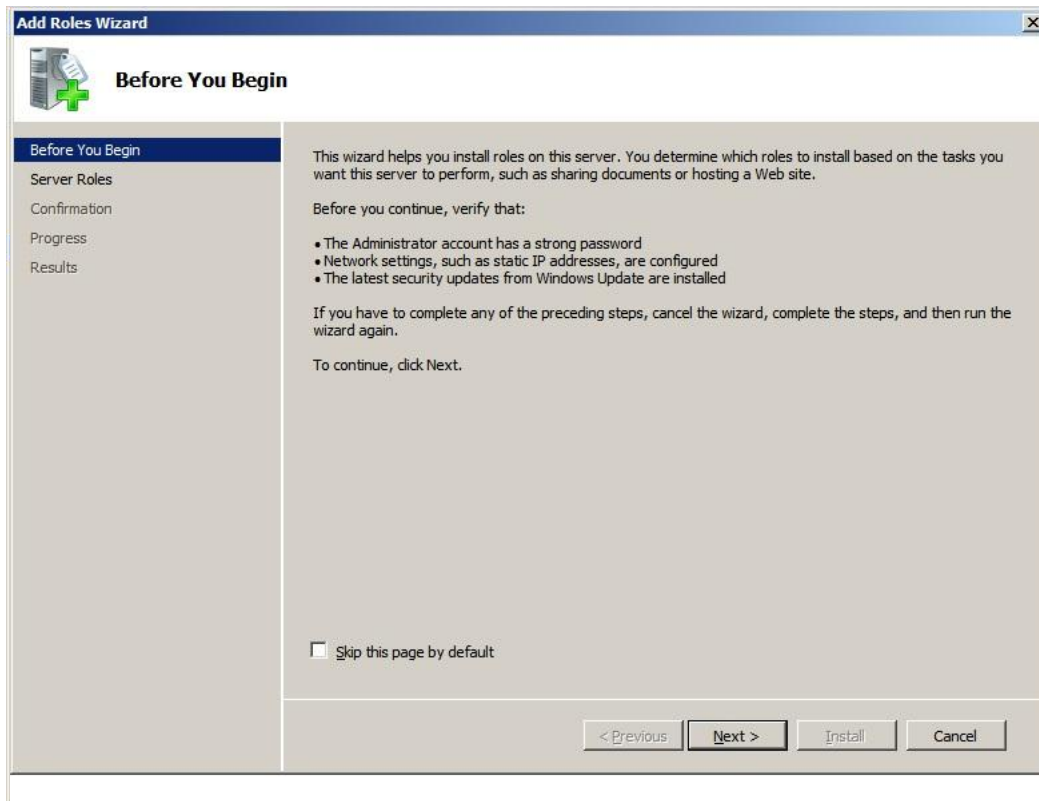
Server Manager provides a single dashboard to install or uninstall server roles and features. Server Manager also gives an overview of all currently installed roles and features. When IIS 7 is chosen from the Server Manager, the basic components and services needed for IIS are automatically selected.

1. Click Start -> All Programs -> Administrative Tools -> Server Manager.

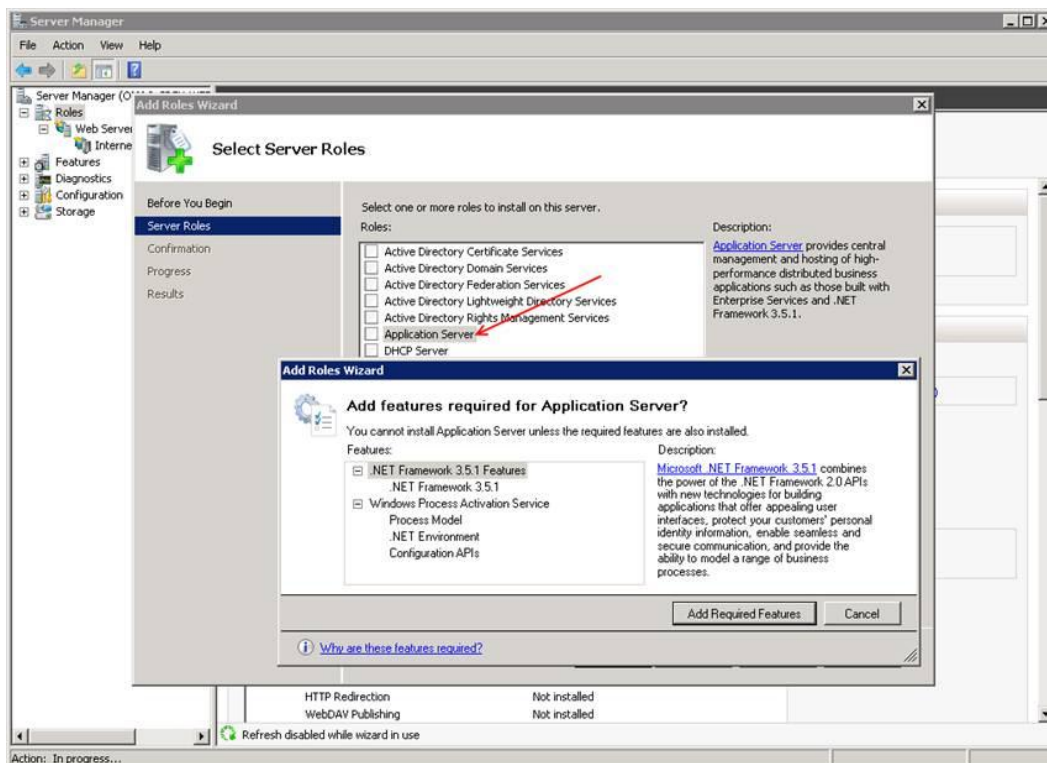


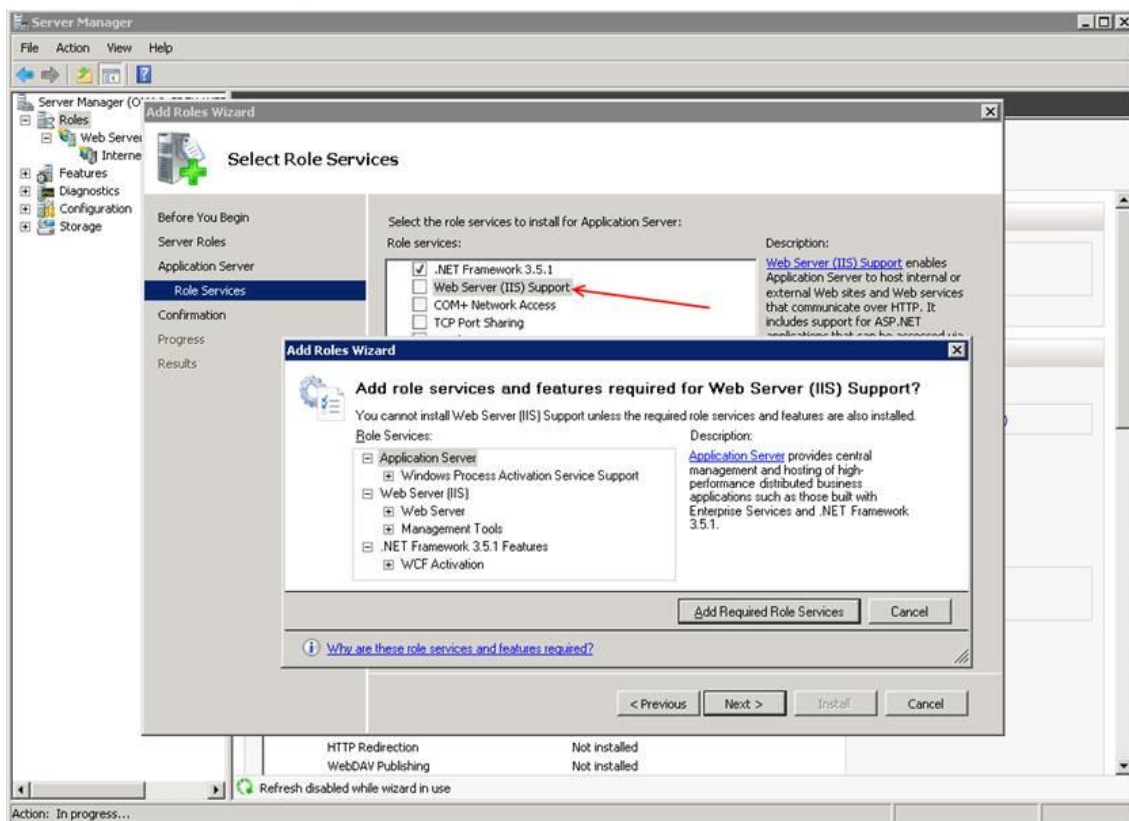
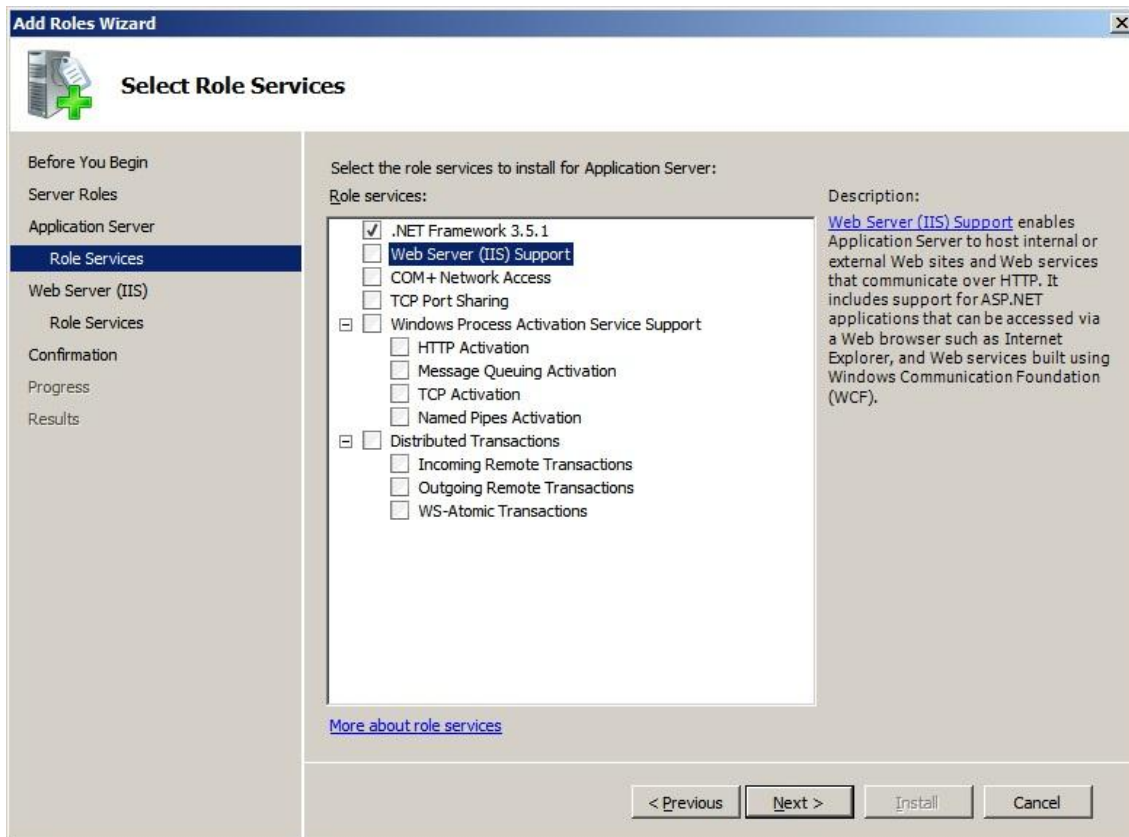
In the Server Manager window, scroll down to Roles Summary, and then click Add Roles. The Add Roles Wizard will start with a Before You Begin page. The wizard asks for verification of the following:

- a. The administrator account has a strong password.
- b. The network settings, such as IP addresses, are configured.
- c. The latest security updates from Windows® Update are installed.

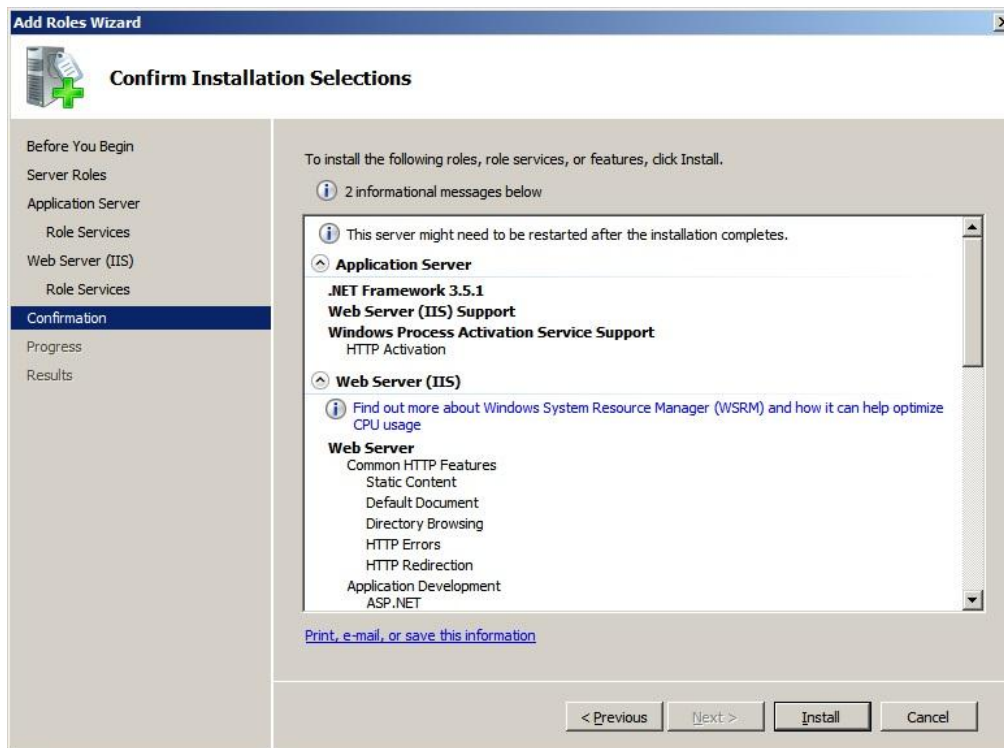


On the Select Role Services window, select Application Server and Web Server (IIS) Support and add any required role services to continue.

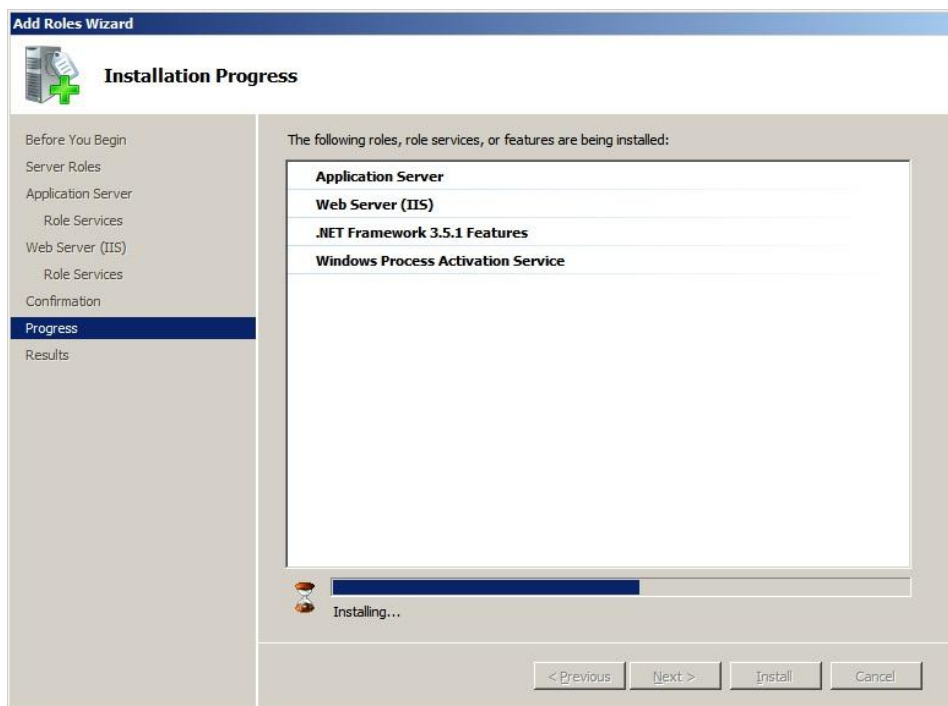




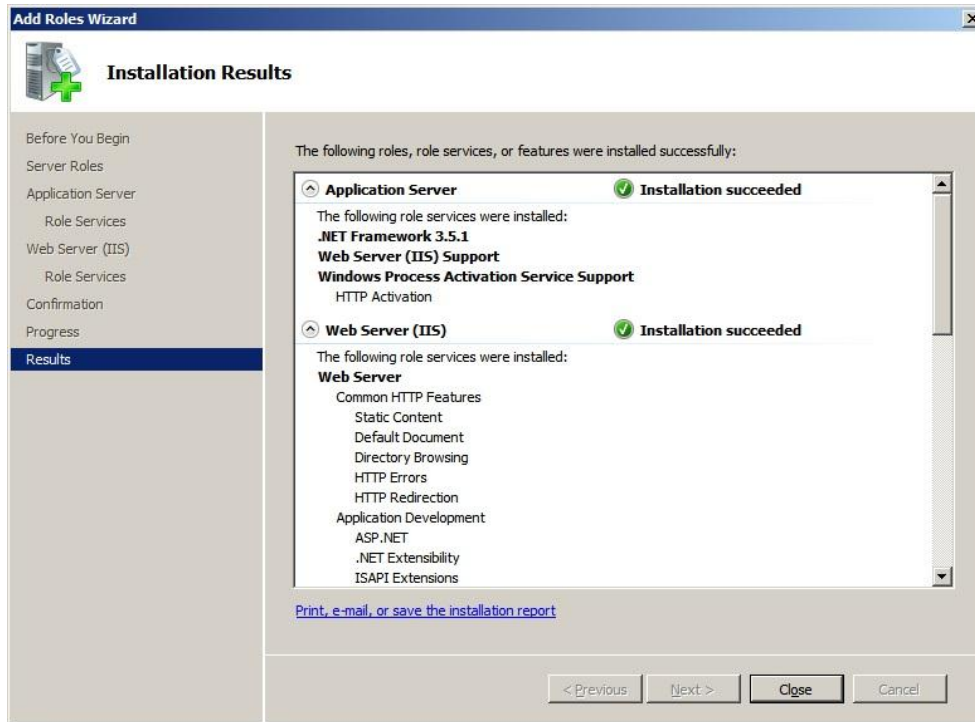
The Confirm Installation Selections window opens. Read through the list and click Install to complete the installation process.



The Installation Progress window opens. Wait until it finishes.



The Results window opens. Wait until it finishes.



Confirm that the Web server works by using <http://localhost> in Internet Explorer



11.2 Create Web Site on IIS 7.5

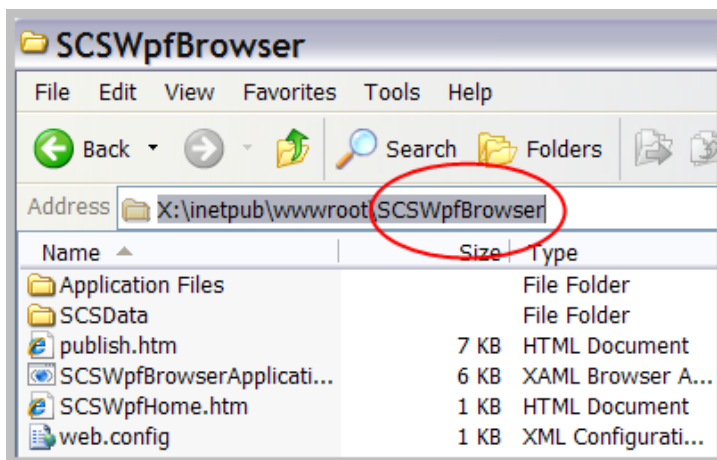
Introduction

During the installation of Internet Information Services 7 (IIS 7), a default Web site configuration is created in the \Inetpub\Wwwroot directory on your Web server. You can either use this default directory to publish your Web content, or create a directory at a file system location of your choice.

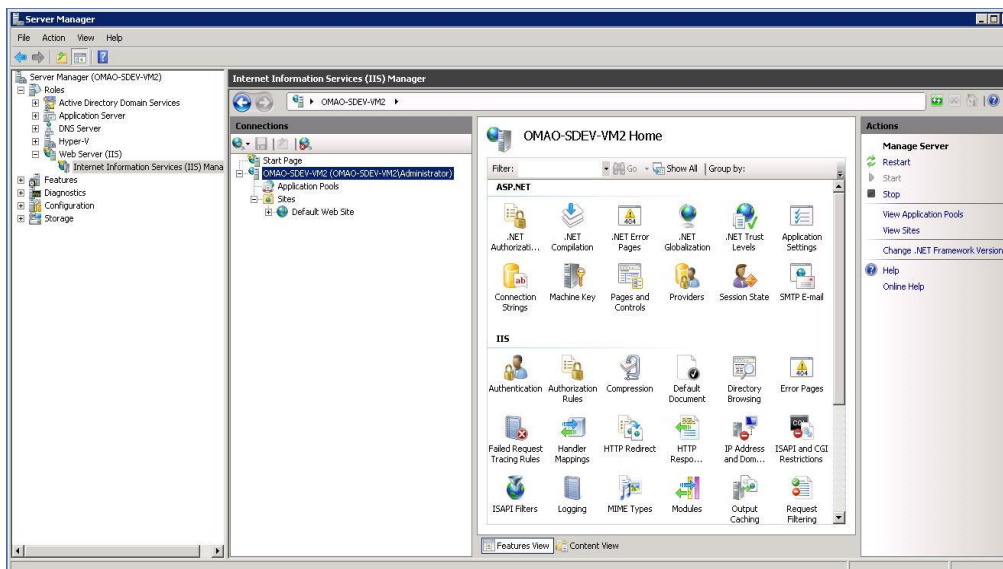
When you add a Web site in IIS 7, a site entry is created in the ApplicationHost.config file. The entry specifies the network binding for the site, maps the site to a location in the file system, and optionally specifies user credentials for content access.

Add a Web Site

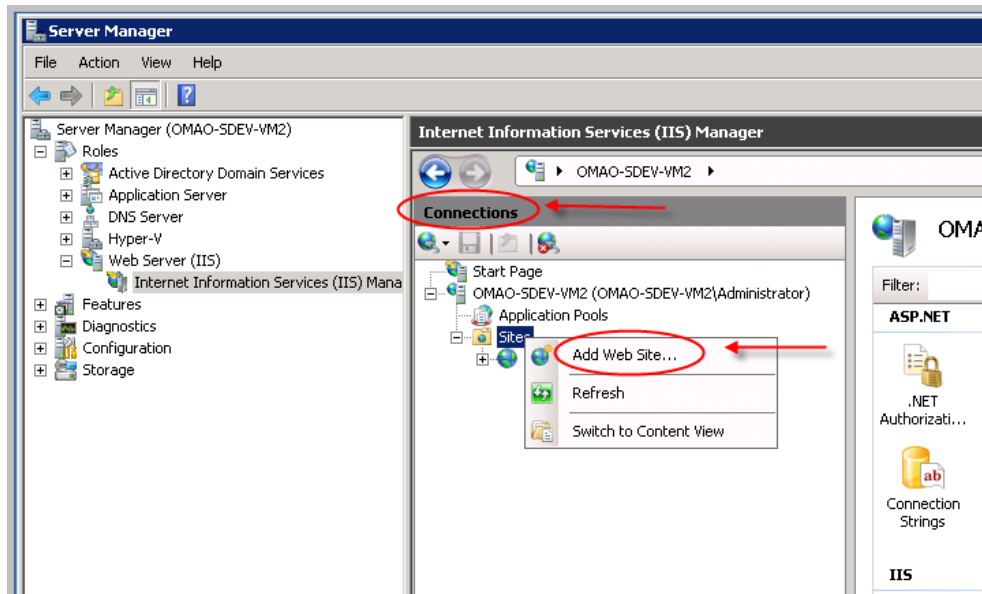
1. Insert DVD and copy folder and files to c:\inetpub\wwwroot



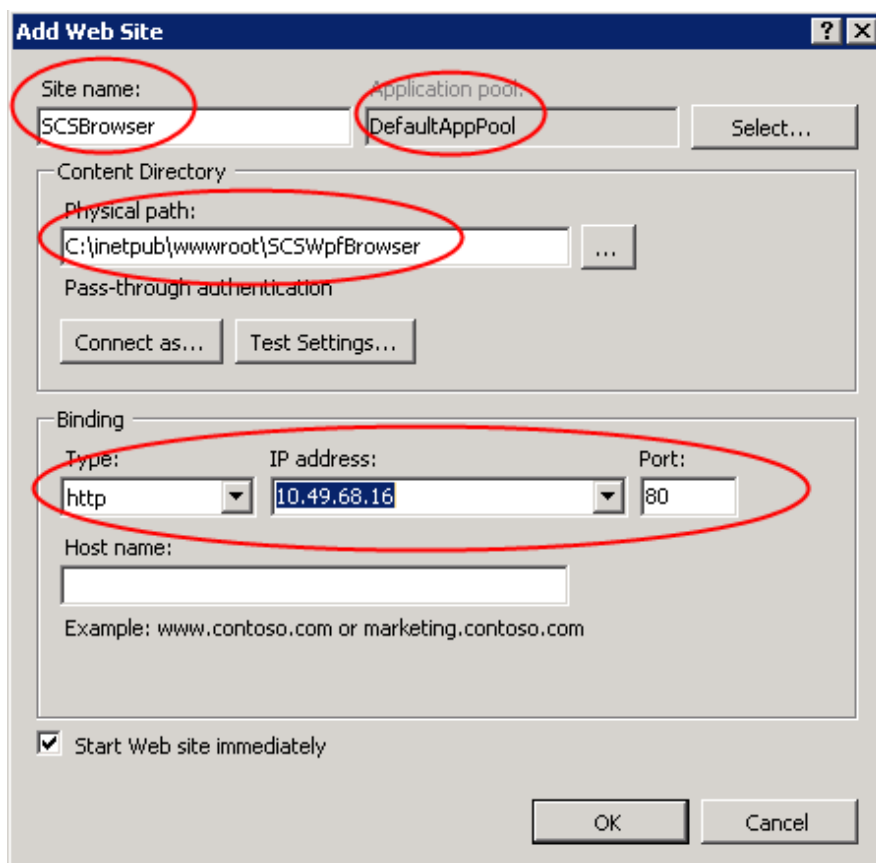
2. Start IIS Manager.



3. In the Connections pane, right-click the Sites node in the tree views, and then clicks Add Web Site.

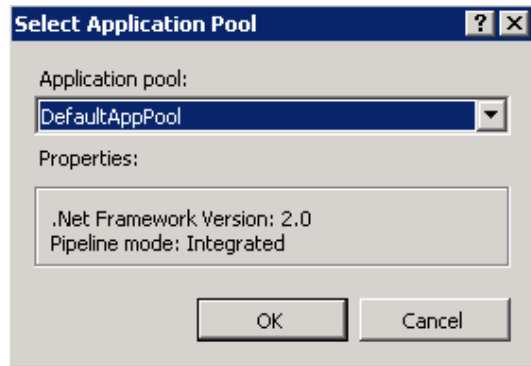


4. In the Add Web Site dialog box, type "SCSBrowser" for Web site in the Web site name box.



In the Physical path box, type the physical path of the Web site's folder, or click the browse button (...) to browse the file system to find the folder.

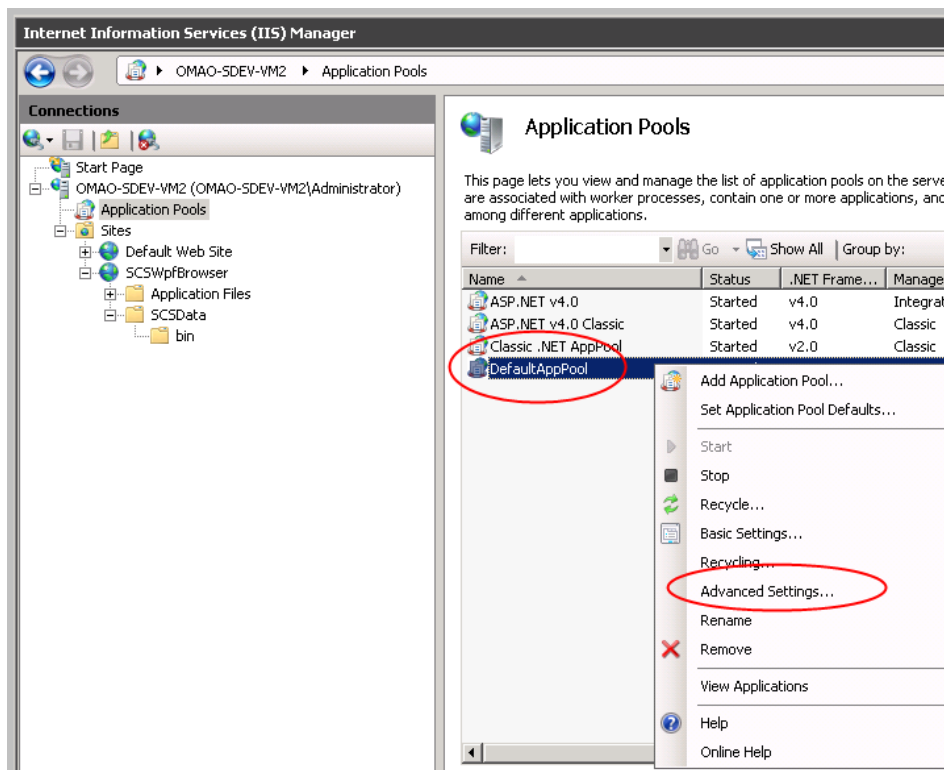
In the Select Application Pool dialog box, select DefaultAppPool from the Application Pool list, and then click OK.



The default value in the IP address box is All Unassigned. Please select the Web Server IP address by clicking the drop-down arrow. The default value in the Port text box is 80.

5. Click OK.

6. Click on Application Pools and right-click on DefaultAppPool to change the Advanced Settings. Select .NET Framework Version v4.0. Click on OK to save.



Advanced Settings

?

X

(General)

.NET Framework Version	v4.0
Enable 32-Bit Applications	False
Managed Pipeline Mode	Integrated
Name	DefaultAppPool
Queue Length	1000
Start Automatically	True

CPU

Limit	0
Limit Action	NoAction
Limit Interval (minutes)	5
Processor Affinity Enabled	False
Processor Affinity Mask	4294967295

Process Model

Identity	ApplicationPoolIdentity
Idle Time-out (minutes)	20
Load User Profile	False
Maximum Worker Processes	1
Ping Enabled	True
Ping Maximum Response Time (seconds)	90
Ping Period (seconds)	30
Shutdown Time Limit (seconds)	90
Startup Time Limit (seconds)	90

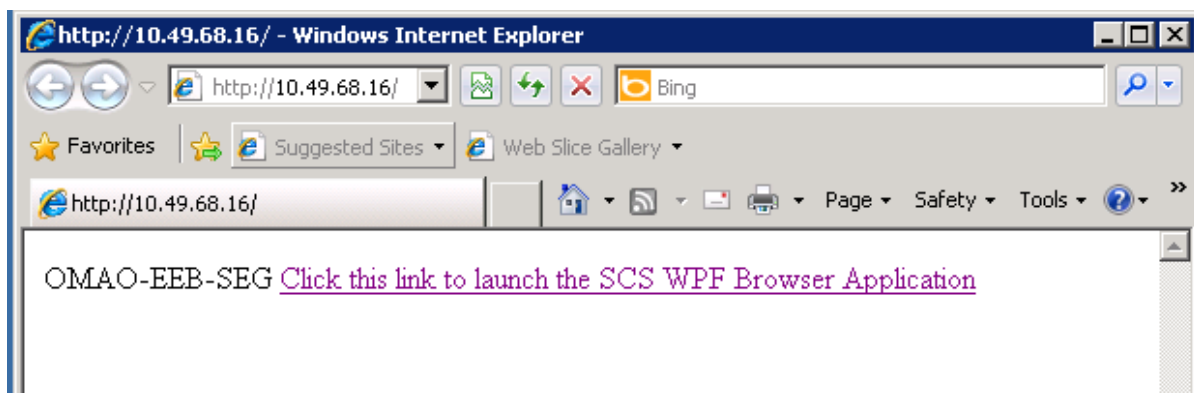
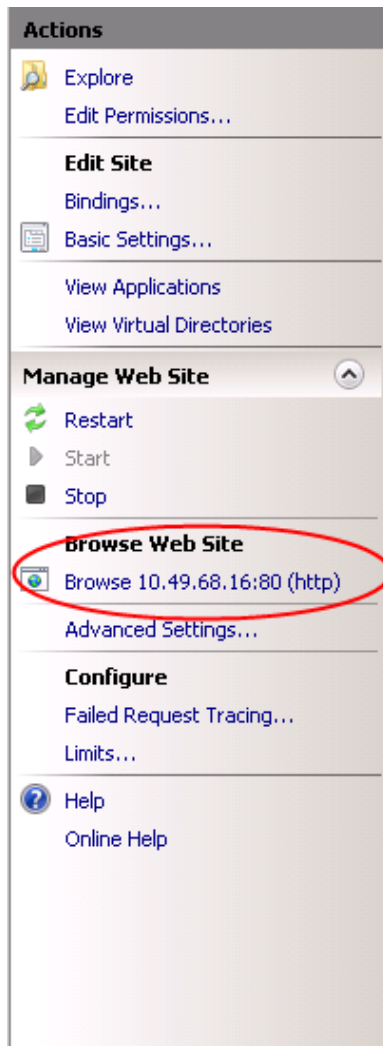
.NET Framework Version

[managedRuntimeVersion] Configures the application pool to load a specific version of the .NET Framework. Selecting "No Managed Code" will cause all ASP.NET requests to fail.

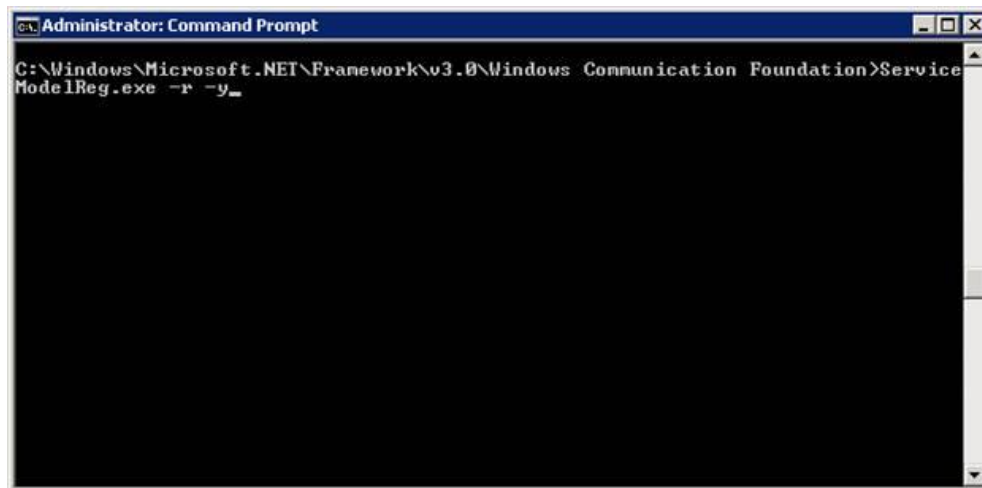
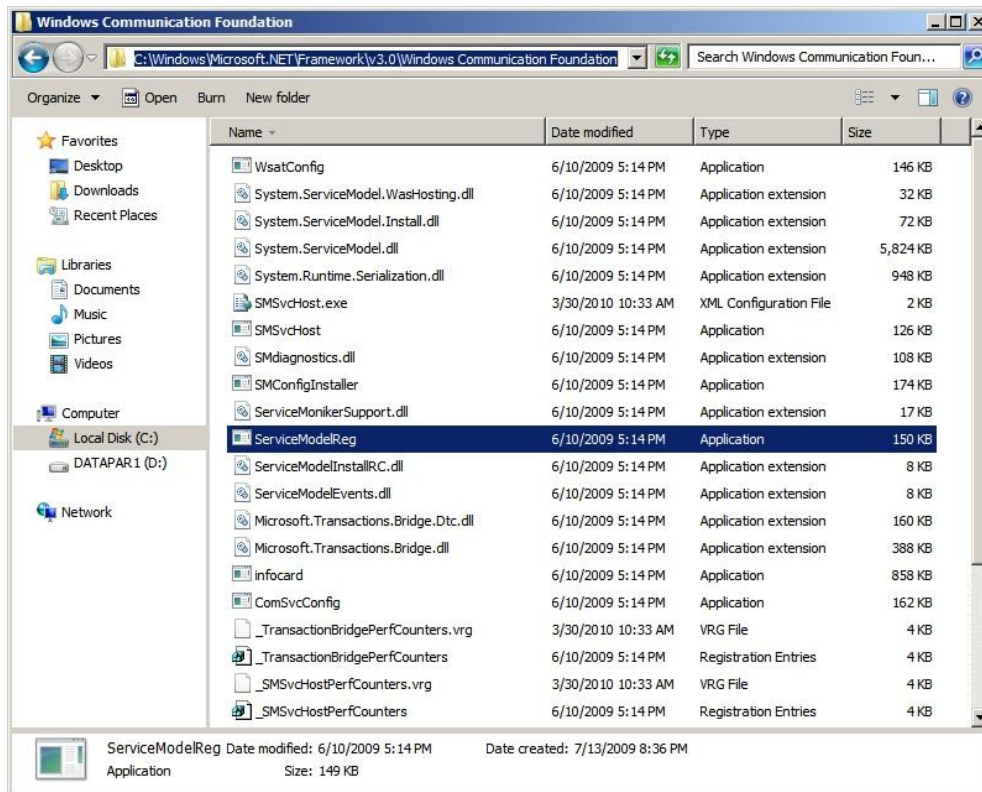
OK

Cancel

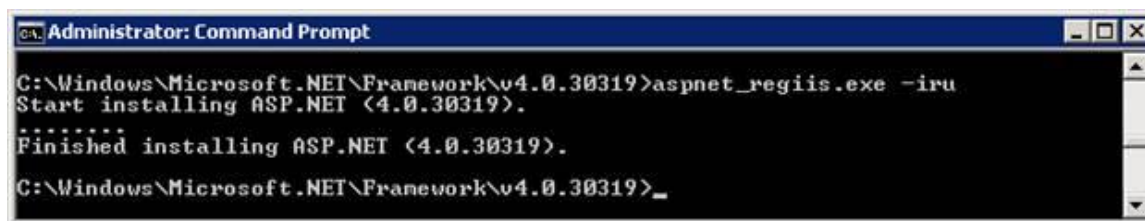
7. To test if the web site setup is successful, you can browse the site by clicking the site from Action pane.



8. Install Windows Communication Foundation (WCF)

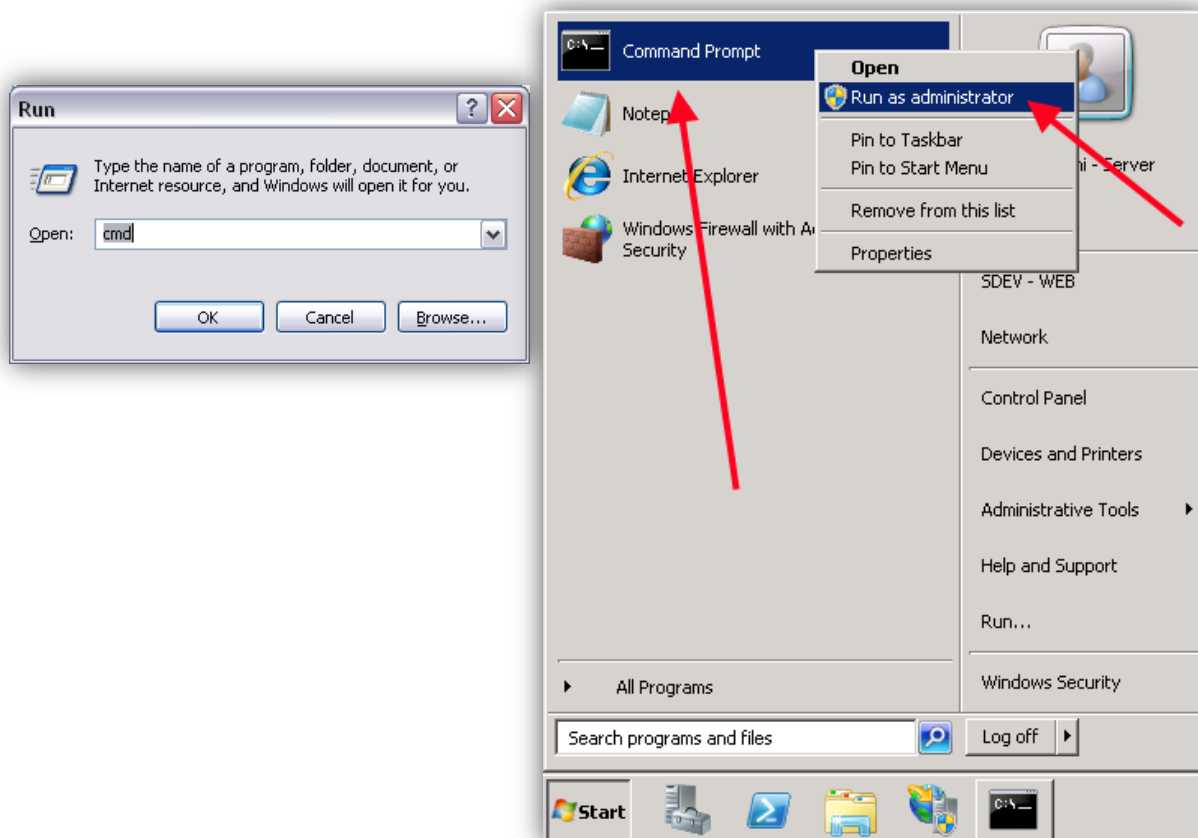


9. Run cmd.exe to Install ASP



Install SCS Data Web Service

1. Open a session (remote desktop or console) on the server running IIS that you'd like to install the web service on.
2. Open a DOS window
 - a. If you're running Windows 2003 Server. (Start → Run → cmd)
 - b. If you're running Windows Server 2008 or have UAC enabled
 - Open your start menu
 - Right click on '**Command Prompt**'
 - Choose '**Run as Administrator**'

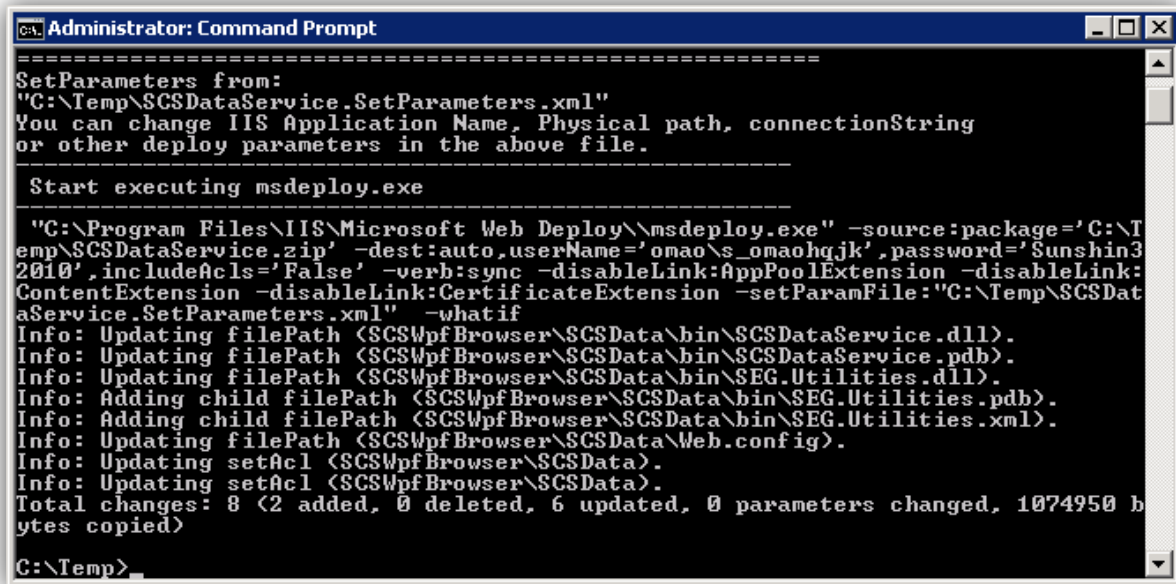


3. Browse to the SCS web service installation package on your DVD in the ??? folder. The main installation file is SCSDDataService.deploy.cmd
 - a. In your DOS window we will run the install. The optional parameters are
 - /t → Test mode
 - /u → Username (don't forget domain reference, i.e. SHIPNAME\Username)
 - /p → Password for user specified by /u
 - b. Ensure the result of your command passes without error
 - c. Once the test completes without error change the command parameter from /t to /y.

For example:

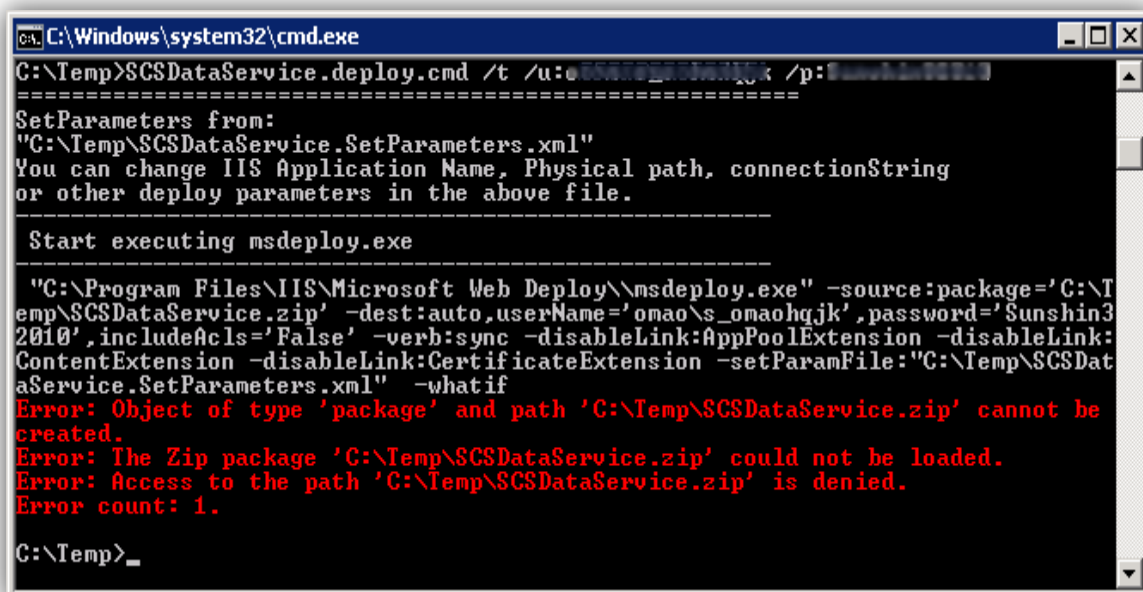
Test Run → *SCSDDataService.deploy.cmd /t /u:EXPLORER\Administrator /p:password01*

Real Run → *SCSDDataService.deploy.cmd /y /u:EXPLORER\Administrator /p:password01*



```
Administrator: Command Prompt
=====
SetParameters from:
"C:\Temp\SCSDDataService.SetParameters.xml"
You can change IIS Application Name, Physical path, connectionString
or other deploy parameters in the above file.
=====
Start executing msdeploy.exe
=====
"C:\Program Files\IIS\Microsoft Web Deploy\msdeploy.exe" -source:package='C:\T
emp\SCSDDataService.zip' -dest:auto,userName='omao\s_omaojqjk',password='Sunshin3
2010',includeAcls='False' -verb:sync -disableLink:AppPoolExtension -disableLink:
ContentExtension -disableLink:CertificateExtension -setParamFile:"C:\Temp\SCSDat
aService.SetParameters.xml" -whatif
Info: Updating filePath <SCSWpfBrowser\SCSDData\bin\SCSDDataService.dll>.
Info: Updating filePath <SCSWpfBrowser\SCSDData\bin\SCSDDataService.pdb>.
Info: Updating filePath <SCSWpfBrowser\SCSDData\bin\SEG.Utilities.dll>.
Info: Adding child filePath <SCSWpfBrowser\SCSDData\bin\SEG.Utilities.pdb>.
Info: Adding child filePath <SCSWpfBrowser\SCSDData\bin\SEG.Utilities.xml>.
Info: Updating filePath <SCSWpfBrowser\SCSDData\Web.config>.
Info: Updating setAcl <SCSWpfBrowser\SCSDData>.
Info: Updating setAcl <SCSWpfBrowser\SCSDData>.
Total changes: 8 (2 added, 0 deleted, 6 updated, 0 parameters changed, 1074950 b
ytes copied)
C:\Temp>_
```

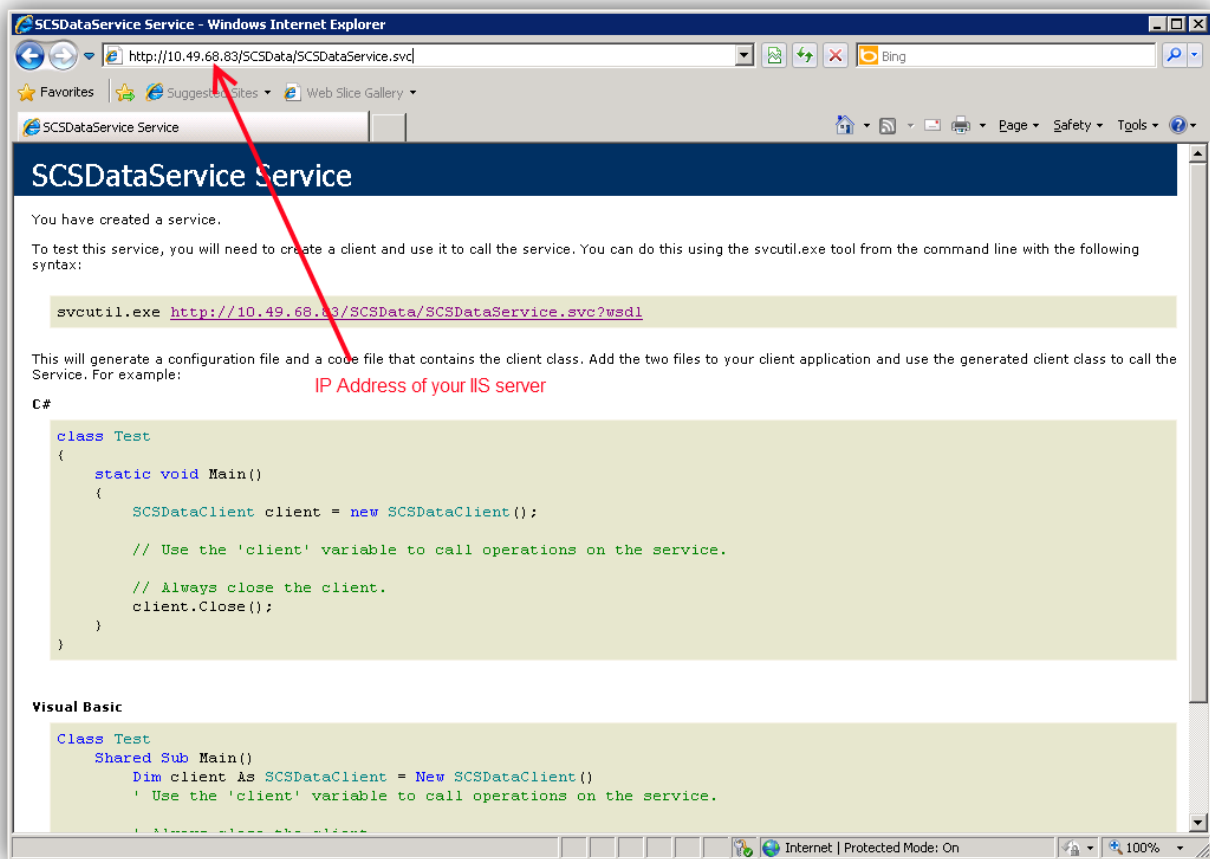
Note: If you get an error such as the one below you may need elevated privileges, ensure you're running as administrator (see step 2)



```
C:\Windows\system32\cmd.exe
C:\Temp>SCSDDataService.deploy.cmd /t /u:omao\s_omaojqjk /p:Sunshin32010
=====
SetParameters from:
"C:\Temp\SCSDDataService.SetParameters.xml"
You can change IIS Application Name, Physical path, connectionString
or other deploy parameters in the above file.
=====
Start executing msdeploy.exe
=====
"C:\Program Files\IIS\Microsoft Web Deploy\msdeploy.exe" -source:package='C:\T
emp\SCSDDataService.zip' -dest:auto,userName='omao\s_omaojqjk',password='Sunshin3
2010',includeAcls='False' -verb:sync -disableLink:AppPoolExtension -disableLink:
ContentExtension -disableLink:CertificateExtension -setParamFile:"C:\Temp\SCSDat
aService.SetParameters.xml" -whatif
Error: Object of type 'package' and path 'C:\Temp\SCSDDataService.zip' cannot be
created.
Error: The Zip package 'C:\Temp\SCSDDataService.zip' could not be loaded.
Error: Access to the path 'C:\Temp\SCSDDataService.zip' is denied.
Error count: 1.
C:\Temp>_
```

5. Test your installation by browsing to the service site. You should see something similar to the following:

<http://INSERT IP ADDRESS OF IIS SERVER/SCSDData/SCSDDataService.svc>



6. Setting up service to speak with SCS

- Open Windows explorer and browse to your IIS root (default is *c:\inetpub\wwwroot*)
- Find your website and the *SCSData* directory in it. Example:
C:\inetpub\wwwroot\SCSWpfBrowser\SCSData
- Located and open the *Web.Config* file inside this directory, it contains the information used by the service to properly talk to ACQ.
- This file is actually an XML file, but you can open it in Notepad. Look for the *ApplicationSettings* node, this has all the information you need to change.

```

<?xml version="1.0" ?>
- <configuration>
- <configSections>
- <sectionGroup name="applicationSettings" type="System.Configuration.ApplicationSettingsGroup, System, Version=4.0.0.0,
  Culture=neutral, PublicKeyToken=b77a5c561934e089">
  <section name="SCSDataService.Properties.Settings" type="System.Configuration.ClientSettingsSection, System,
    Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" requirePermission="false" />
  </sectionGroup>
</configSections>
- <system.web>
  <compilation debug="true" targetFramework="4.0" />
</system.web>
- <system.serviceModel>
- <behaviors>
- <serviceBehaviors>
- <behavior>
  <!-- To avoid disclosing metadata information, set the value below to false and remove the metadata
    endpoint above before deployment -->
  <serviceMetadata httpGetEnabled="true" />
  <!-- To receive exception details in faults for debugging purposes, set the value below to true. Set to
    false before deployment to avoid disclosing exception information -->
  <serviceDebug includeExceptionDetailInFaults="true" />
</behavior>
</serviceBehaviors>
</behaviors>
<serviceHostingEnvironment multipleSiteBindingsEnabled="true" />
</system.serviceModel>
- <system.webServer>
  <modules runAllManagedModulesForAllRequests="true" />
</system.webServer>
- <applicationSettings>
- <SCSDataService.Properties.Settings>
- <setting name="AcqPort" serializeAs="String">
  <value>505</value>
</setting>
- <setting name="AcqIP" serializeAs="String">
  <value>10.49.68.80</value>
</setting>
- <setting name="LoggerPath" serializeAs="String">
  <value>C:\inetpub\wwwroot\SCSDataService\SCSDataService.log</value>
</setting>
- <setting name="Ship40ShareName" serializeAs="String">
  <value>SHIP40</value>
</setting>
</SCSDataService.Properties.Settings>
</applicationSettings>
</configuration>

```

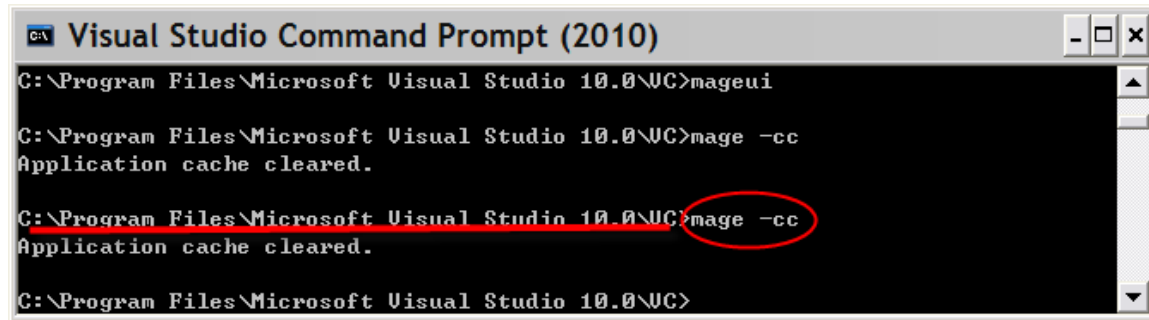
e. There are four parameters you may change:

- **AcqPort**
Default is 505, it's rare you would change this
- **AcqIP**
Default is 10.49.68.80, change this to point to your primary SCS server
- **LoggerPath**
Defaults to a path, change it to another path if you prefer a log file to be written elsewhere
- **Ship40ShareName**
Defaults to SHIP40, it's rare you would change this. Do not change unless you have manually modified the primary Ship directory shared to be displayed under a different name

7. Restart IIS & confirm the website gets ACQ templates & data.

8. Helpful hints

If you update or restore web server, change setting with IIS, on the server and client computers, please run “mage -cc” to clear application cache.



```
C:\Program Files\Microsoft Visual Studio 10.0\VC>mageui  
C:\Program Files\Microsoft Visual Studio 10.0\VC>mage -cc  
Application cache cleared.  
C:\Program Files\Microsoft Visual Studio 10.0\VC>mage -cc  
Application cache cleared.  
C:\Program Files\Microsoft Visual Studio 10.0\VC>
```


APPENDIX THREE

SAMOS DATA SUBMISSION PROCEDURES

12 APPENDIX THREE: SAMOS Data Submission Procedures

12.1 Introduction and Background

SAMOS (*Shipboard Automated Meteorological and Oceanographic System*) is a program operated by the Florida State University Center for Ocean-Atmosphere Prediction Studies (COAPS) to receive meteorological and oceanographic data from ships at sea in near real time, to verify the quality of that data, and provide feedback on the data quality to the ship.

NOAA OMAO has agreed to participate in this effort beginning with the NOAA Ship HENRY BIGELOW in the fall of 2006. Other ships will be added to the program as they are outfitted with SCS 4.0 and with VSAT internet connections.

Aboard NOAA ships the SCS event logger will execute the “SAMOS” event to assemble the necessary data into an observations file. Then, a standalone program, the “SAMOS Mailer” application will reformat the event log observations into a format appropriate for submission to the SAMOS project.

12.2 SAMOS Data Standard

The SAMOS program has defined a series of standards governing what data is to be submitted and the format of the submitted data.

In the SAMOS submission each sensor observation is reported as data value that is an average of the previous 60 seconds of sensor data. The SAMOS program requests at least 12 samples to be averaged over the 60 second period

Data values are to be computed once per minute continuously throughout the day (1440 samples per day), and an entire day’s data is to be mailed to the SAMOS project once per day at time 0001 GMT.

Each sensor observation consists of a designator and a sensor value separated by a colon. For example for true wind direction the designator is TWDIR and a sample observation would look like “TWDIR:214.” Each kind of data will have a NOAA-wide keyword assigned to it, and a standard unit of measurement. The following table summarizes the data to be submitted:

Designators	Data Type	Units
CS	Ship's Call Sign	
LAT	Latitude	Signed Decimal Degrees (-for South)
LON	Longitude	Signed Decimal Degrees (-for West)
GYRO	Ship Heading	Decimal Degrees
SOG	Speed Over Ground	Knots
COG	Course Over Ground	Decimal Degrees
ATEMP	Air Temperature	Degrees C
BARO	Barometric Pressure	hPa (NOT adjusted for Sea Level)
RELH	Relative Humidity	Percent
RWSPD	Relative Wind Speed	Knots
RWDIR	Relative Wind Direction	Decimal Degrees (from which wind is blowing)
TWSPD	True Wind Speed	Knots
TWDIR	True Wind Direction	Decimal Degrees (from which wind is blowing)
TSGWT	TSG Water Temperature	Degrees C
TSGS	TSG Salinity	Psu
TSGC	TSG Conductivity	S/M
TSGSV	TSG Sound Velocity	m/s

12.3 Setting up ACQ and SAMOS Derived Sensors

Using CFE-DB you must define a set of derived average sensors to compute the average for the previous 60 seconds. A new feature in SCS 4.x is the ability to compute averages over a specified time interval, independent of how many samples are sent by the sensor.

The Log Rate for the SAMOS average sensors should be set to 60 seconds, and the Timeout Sensor Parameter must be set to not less than 60 seconds (This is enforced by CFE-DB).

The Name of the SAMOS average sensors should start with the word SAMOS to distinguish them easily from the other sensors. For example SAMOS_LATITUDE. Since in SCS 4.0 sensor lists are alphabetized this serves to group all SAMOS sensors together in the list.

NOTE: If your sensor reports its data in units not in the above table, you will have to define another sensor, of type Derived Line Equation, to convert the native sensor units to the units in the table.

12.4 The SAMOS Event Log

Setting up the SAMOS event log template proceeds as follows. It is important that you follow these specs exactly. The correct operation of the SAMOS Mailer depends on it.

Template Name

The name of the event template should be SAMOS.

Sensor Metaitems

You must create a sensor metaitem for each kind of data to be submitted as follows:

- The name of the metaitem will be the designator shown in the table above.
- You will connect this metaitem to the sensor you chose to provide the data value, and this must be one of the SAMOS derived average sensors you set up previously in the sensor configuration file.
- Notes:
 - The order of the metaitems does not matter.
 - Metaitems may be put in separate tab groups at your convenience.
- In addition to the sensor metaitems there should be one manual metaitem whose name is “CS”, meaning “call sign”. The value of this metaitem will be the four-letter radio call sign of the ship, for example, “WTER”. This is important because the SAMOS database tracks submissions by call sign, not ship name.

SAMOS Data Log File

You must define a continuous output as follows:

- Name is SAMOS_OBS
- Type is continuous
- Log rate is 1 minute
- Select the “Lat/Lon format Decimal Degrees” radio button.
- The list of selected output elements should consist of all SAMOS derived average sensors you defined in the sensor configuration file above.

Note: The order of the output elements in the output definition does not matter.

12.5 Running the SAMOS Event

You should start the SAMOS Event as soon as you start ACQ; preferably as soon as possible after you leave port. Continue as long as you are at sea.

The SAMOS Event Logger can be stopped at any time, for example if ACQ needs to be stopped. However to avoid large holidays in the data it should be restarted as soon as possible. This will create data files with different event index numbers, but the SAMOS mailer software will handle that easily.

12.6 Important Files Created by the Event Logger

The Event Logger will create two files that are crucial to the operation of the SAMOS Mailer (described in chapter Eight - Utilities of this manual).

The files are created as usual in the EventTemplates\SAMOS subfolder of your Datalog directory as defined in the SCS Configuration file.

SAMOS_OBS_NNN.ELG, where NNN is the Event index number. This contains the observed data for all the SAMOS sensors. The first line in this file contains the names of the sensors that were logged.

METADATASENOSDESCRIPTION_NNN.CSV, where NNN is the Event index number. This file contains a list of all metaitems and the sensors that you connected to them when you created the event template.

Setting Up for SAMOS Mailer Operations

The SCS Configuration file, C:\SCS40.CFG, must be edited to include the following parameters:

- *SAMOS_EMAILSYSTEMNAME* is the name of the ship's mail server. Typically this parameter has the same value as *SHIP_EMAILSYSTEMNAME* in the SCS config file.
- *SAMOS_MAIL_BODY* defines a short text string you would like to include in the body of the emails.
- *SAMOS_MAIL_FROM* is the 'from' address which appears in the email.
- *SAMOS_MAIL_TO* is the email address to send the data files to. Note: ships of the NOAA fleet must send SAMOS email to SamosShip.SCS@noaa.gov, a forwarding account that sends the mail on to its final destination.
- *SAMOS_MAIL_SUBJECT* is a string which to appear in the subject line. It must contain the name of the ship and the ship's call sign.

Example settings appear below. They are specific to a SAMOS setup on the NOAA Ship Oscar Dyson.

SAMOS_EMAILSYSTEMNAME=mail.nems.noaa.gov;

SAMOS_MAIL_BODY=SAMOS data from the NOAA Ship OSCAR DYSON (WTER);

SAMOS_MAIL_FROM=chiefst.oscar.dyson@noaa.gov;

SAMOS_MAIL_TO=SamosShip.SCS@noaa.gov;

SAMOS_MAIL_SUBJECT=SAMOS data From the NOAA Ship Oscar Dyson (WTER);

12.7 Operation of the SAMOS Mailer

The SAMOS Mailer app (also known as the Event Log Mailer) is described in detail in *Chapter 8 - Utilities* of this manual. Note the following that are specific to SAMOS operations.

Reformatting the Data

The mailer app reads the SAMOS_OBS data file and the Metadata Sensor Description file and combines them into the SAMOS formatted data file. Again the order of the sensors and the metaitems in these files is not important.

The name of this file is constructed from the ship's call sign and the date of the observations.

A typical line in the reformatted SAMOS file looks like this (line wrapping in effect here).

```
$SAMOS:001, CS:WTDM, YMD:20070301, HMS:143450, LAT:55.47518, LON:-160.41834,  
GYRO:64.1, COG:, SOG:, RWDIR:225.25, RWSPD:5.95, TWDIR:, TWSPD:, RELHUM:59.3,  
TSGTEMP:, TSGSAL:30.48, ATEMP:12.05
```

It is a series of comma delimited observations. Each observation consists of the Designator for the data type and the data value, separated by a colon. In addition to the sensors you defined there are two fields, "\$SAMOS:001" and "YMD:20070301" generated by the mailer. The former contains the file format indicator, the latter contains the time of observation.

Each data file will contain data from one calendar day, 1440 lines in all.

Compressing the SAMOS File for Mailing

Before sending the data to the ship's mail server it compresses the data into a ZIP file to save mailing costs. For 24 hours of data the ZIP file size is about 40K to 50K bytes, depending on the number of sensors.

12.8 Examining the ZIP File

The SAMOS mailer will always delete the ZIP file after the file is successfully sent to the mail server. If you have any questions on what is being mailed to shore you can examine the file by temporarily changing the destination mail address to yourself, as follows.

- On the SAMOS Mailer main window, click on the Edit CFG button. This opens the SCS Configuration File Editor app. You can also access the editor from the SCS Menu.
- Change the SAMOS_MAIL_TO parameter to your own email address.
- Save the file and exit the SCS Configuration editor app.
- Restart the SAMOS Mailer app.
- Conduct a test mailing by pressing the Send Now button or Send All button. You should now have the zip file in your in box as an attachment.
- Reset the SAMOS_MAIL_TO parameter appropriately.
- Restart the SAMOS mail app.

APPENDIX FOUR

NODC DATA SUBMISSION PROCEDURES

13 APPENDIX FOUR: NODC Data Collection Procedures

13.1 Introduction

This document describes procedures to be implemented and executed aboard SCS ships in the NOAA in collecting and transmitting basic environmental data to NOAA's National Oceanographic Data Center (NODC).

This document supersedes the document of the same name dated November 2006, published in MOC Docs as ANC-10.

13.2 Background

The NOAA Offices of Marine and Aviation Operations (NMAO) and NODC (National Oceanic Data Center) have a cooperative program to record certain basic environmental data from any NOAA Ship having the NMAO Scientific Computer System (SCS). This program has been in place since 2003

NOAA ships use the SCS Event Logger to run the NODC event at all times while at sea, then send the accumulated data to NODC on CD or by FTP transfer once a month.

NODC reviews the data and provide feedback to NMAO headquarters and the ship's command about the completeness of the data and any Quality Assurance issues that may arise. This review will be supplemented by additional reviews and feedback from NMAO HQ.

Ultimately, *NODC archives all data for distribution to the public on its web site.*

13.3 Data to be collected

Who Does this Directive Apply to?

This directive applies to all ships that have a full set of meteorological and TSG sensors. Ships having "mini SCS" are exempt Mini-SCS systems acquire only GPS data in order to provide hourly position reports to the *Shiptracker* website.

The Basic Sensor Dataset

The basic NODC sensor suite consists of

- Primary GPS receiver
- Secondary GPS receiver
- Thermosalinograph (TSG)
- Wind sensor
- Air Temperature sensor
- Relative humidity sensor
- Barometer

13.3.1 The Observations File

The following sensor data shall be supplied in a single observations file as a basic dataset, subject to the availability of sensors.

1. GPS1 Time (Primary GPS)
2. GPS1 Latitude
3. GPS1 Longitude
4. GPS2 Time (Secondary GPS)
5. GPS2 Latitude
6. GPS2 Longitude
7. TSG Sea Surface Temperature
8. TSG Salinity
9. TSG Conductivity
10. True wind speed
11. True wind direction
12. Air temperature
13. Relative humidity
14. Barometric pressure

These data shall be recorded every 30 seconds.

The data shall be un-averaged. That is, *you should not use any derived average sensors*.

The SCS Administrator is free to add at his discretion additional environmental sensor data to the Observations file in the NODC template if sensors are available and providing good data. At the same time the administrator shall add metaitems to his NODC template for each additional sensor, indicating the manufacturer, model, calibration date and units of measurement.

13.3.2 Position Logs

In addition to the Observation file, NAV1 and NAV2 data files shall be provided, containing the RAW GPGBA messages of the primary and secondary, respectively, GPS receivers. Data shall be logged every thirty seconds.

Metadata

“Metadata” is descriptive information about the circumstances under which the sensor data was collected. Metadata enables users of the future to interpret the data correctly.

Two kinds of metadata will be provided

- Cruise information, such as Chief Scientist, Vessel Name etc. Refer to the sample event template file (discussed below) for a complete list.
- For each sensor providing data to the Observations file, the Event Header file shall contain metaitems for the manufacturer, model and calibration date of the sensor hardware, and the units of measurement.

13.4 Preparing for Data Collection

Load the Current Version of SCS on the SCS Servers

As of the date of this directive the current version of SCS is 4.2 or later.

Verify that, when you open up the SCS Menu, you see the version indicator in the title bar of the window. The same holds true when you start up the Event Builder or the Event Logger.

The Sample NODC Event Template File

On the SCS 4 installation CD in the Supplementary Files folder is a sample NODC event template called NODC-template.TPL. This template is not ready for use, but it does form the basis of each ship's actual template. The SCS administrator will modify this template by loading values of all metaitems and changing the names of all sensors in the output file definitions to match the ship's sensor names.

13.4.1 Cruise Information Group

- Cruise ID
“VV-YY-CC-leg NN”, VV=vessel code, YY= year, CC=Cruise number during year, NN= leg number of cruise if more than one.
- Captain
- Chief Scientist
- SCS Administrator (typically the chief ET or Survey Tech)
- Cruise Start Date (formatted as MM-DD-YYYY)
- Cruise Stop Date
- Data Start Date
- Data Stop Date

All items in the Cruise group except Data Start Date and Data Stop Date must be manually entered into the NODC template at the start of a cruise. The Data start/stop dates are supplied automatically by the Event Logger.

13.4.2 Navigation Sensors Group

This group of manual Metaitems provides the manufacturer, model and calibration date (if applicable) for “Nav1”, the primary navigation system and “Nav2”, the secondary navigation system.

13.4.3 Meteorology Sensors Group

Metaitems in this group provide the manufacturer, model and calibration date (if applicable) for instrumentation measuring air temperature, humidity, wind direction and speed, and atmospheric pressure.

13.4.4 Oceanography Sensors Group

The manual metaitems in this group describe the TSG sensor. As above, Manufacturer, model, calibration date, and units of measurement are required.

Modifying the NODC Template File.

Copy the sample NODC-Sample.TPL Template file from the SCS distribution disk to the “Templates\Event Templates” subdirectory of your SCSShip logical directory.

Run the Event Builder App (SSCMenu, Data Acquisition, Event Logging, Event Builder), and open the template file.

13.4.5 Manual Metaitems

Edit each metaitem, and fill in its proper value for this cruise. Most metaitems will not change often, so it is useful to check the “Use previous value” box. Metaitems that will change regularly should be made editable so that you can change the value when you start the Logger.

As stated above, your ship may not be carrying one or more of the basic sensors, or a sensor may be dead. In this case indicate the sensor’s status in the Annotations file after you start the event.

13.4.6 Sensor Metaitems

The sample template includes sensor metaitems for each of the basic sensor values. They will not be correct for your ship because your sensor names and Ids are no doubt different. To fix this, edit each sensor metaitem in turn, select the actual sensor from the list, and save the metaitem.

13.4.7 Save Edits as “NODC”

The NODC Event template must be named NODC.TPL for proper execution of the NODC logging procedure.

Starting the NODC Event Logger

Once the NODC Event Template has been built with the Event Builder application, it needs to be started. The NODC Event Log shall be started at the beginning of each and every cruise. This event logger shall be restarted whenever acquisition is restarted during a cruise.

To start the NODC Event Log, go through the following steps:

- Bring up the SSCMenu application (on the primary acquisition server PC)
- Select Acquisition->Events->Event Logger
- Select NODC.TPL from the list of events and then select the Start Event button.
- Change the values of any editable manual metaitems as necessary.

- Press the Start Event button. This will commence datalogging to the NODC_OBS_###.ELG file. ### represents the event's index number. The first time the event is run, ### will be 001. Each time the event is run thereafter, this number will increment by one.
- If any problems or specific events worth noting happen throughout the NODC event, the system manager shall press the Annotate button and enter in a brief description of the problem and/or event. This data will, in turn, be logged to the Annotations file (ANNOTATIONS_###.ELG).
- The system manager can minimize the NODC_### Event Log Running window to the bottom of the screen.

13.5 Ending the NODC Event Log

The NODC Event Log shall remain running throughout the entire cruise, up until the point where the system manager stops the primary data logging

To stop the NODC Event Log, go through the following steps:

1. Select the Annotate button and enter in any remaining pertinent information.
2. To close out of the NODC Event Log, press the Stop Event button in the Event Logger's main window.

13.6 NODC Event Output

All output files will be located in a directory named NODC. The Event Logger creates this directory under the directory EventData\NODC, which is under the folder specified by the SCSEventLog variable within the C:\SCS.CFG configuration file.

Header File

The Header file, NODC_###.HDR (### is the three-digit event index number) by definition contains the value of all metaitems at the time the Stop Event button is pressed in the Logger GUI. Please remember that if you change a metaitem value during a cruise the original value will *not* appear in the Header file. The Header file is by definition the value of all metaitems at the time the Stop Event button was pressed.

Navigation Files

Two data files, NAV1_###.ELG and NAV2_###.ELG shall be recorded, each recording the GPGGA message from their respective receivers. Each file is defined in the Event Builder as a continuous output with update interval of thirty seconds.

Sensor Data File (NODC Observation File)

There is one sensor data file, NODC_OBS_###.ELG (### represents the three-digit event index number). In the Event Builder, it is defined as a continuous output, updated every 30 seconds.

The output elements will consist of at least those in the basic sensor dataset specified above, assuming they are available.

The Event Logger writes the name of the sensor sbeing recorded in the first line of the Observations File.. You name the sensor in CFE-DB, the sensor configuration program. Please

name sensors so that it tells a scientist looking at this data 50 years from now what is being measured, for example salinity, humidity, etc.

Annotations Data File

The NODC Event Logger user interface will include an Annotation button to press to record a comment of any sort that you wish. This button will write the “Annotation” metaitem to the snapshot output ANNOTATIONS_###.ELG, whenever the button is pressed. The procedure for making an annotation is 1. Click on the Annotation textbox in the Logger main window, enter a new annotation, and press the annotation button.

Button Activity File

The file BUTTON_ACTIVITY_###.TXT is written to the disk along with the data file. It is a simple text file that has the record of all button presses made during the current Logger session. This includes the Start Event, Stop Event, and Annotation buttons.

13.7 Data Quality Assurance

The system manager running the NODC Event Log shall be responsible for at least a minimal amount of quality assurance on the data files logged as part of the NODC Event Log. There are essentially two primary methods of looking at the data collected to provide some assurance of the validity and quality:

Review Data as Text

With this method, the files created during the NODC Event Log will be viewed via a text editor (preferably the Programmer's File Editor (PFE)). This will basically ensure that the data is there within the files and there are no easily identifiable problems.

Go through the following steps to view the necessary data files:

1. Bring up the SCSMenu application (on the primary acquisition server PC)
2. Open up “PFE”, the SCS text editor application, through QA Processing->Read Data Files menu.
3. Once the PFE application is up, select File->Open
 - a. Parse through the directories until the EventData\NODC directory under the SCSLog logical directory is found Select the individual NODC event log data files (one at a time) and open them for viewing. Take a quick glance at the files and ensure the following:
4. there is data within the file
5. the starting and ending dates/times look correct
6. there are no strange characters or noticeable errors.
7. The Observation File and Navigation Data files record their data every 30 seconds.

Review Data as Graph

13.7.1 Real Time Plots

Real time series plots should be running for all basic sensors at all times while at sea. They provide a visual image of the data stream and are a much easier and reliable method of reviewing data quality over the “Review by Text” described above.

However, plots will not verify that data is being logged every 30 seconds to the Observation and Navigation files. You must still use “Review by Text” to verify that.

13.7.2 Post Time Series Plots (“Newspaper Plots”)

Sensors can be plotted one at a time (In the SCS Menu app, QA Processing->Plot Individual Sensors menu), or as a group (QA Processing->Plot All Sensors).

Plotting all sensors is also known as “Newspaper Plots”.

Standard Operating Procedure calls SCS managers to create Newspaper Plots once a day for review of sensor data quality.

With this method, the files created during the NODC Event Log will be viewed in a graphical format. By looking at the data in this way, as opposed to the text method, the system manager can get a clearer indication as to whether or not there were any breaks in the data.

13.8 Send Data to NODC

NODC Event data shall be transferred to CD and mailed to NODC, or it shall be placed in a ZIP file and uploaded to the NODC FTP site. This shall be done at the beginning of each month, or at the next import period following.

You may use whichever method is more efficient. In any case, send an e-mail to eed.nodc@noaa.gov, to notify NODC that you uploaded the data files. Note that eed.nodc is a mail list which includes both NODC and NMAO EED personnel.

In your e-mail notification, please include the following information:

- a. Your name, phone number, e-mail address, and ship’s name.
- b. Names of the zip files you uploaded.
- c. List all the cruises and/or legs included in the zip files as follows:

VV-YY-CC(-legN)

VV = vessel id,

YY = year,

CC = cruise number of that year, and

N = leg number in the cruise, if there is more than one leg.

The NODC Data Officer will assign a NODC Accession number to these data, notify you of this number by e-mail, and then archive these data.

In addition, this data should be backed up periodically as any other SCS data is.

Instructions for uploading SCS data to NODC's FTP server

“Zip” the NODC event directory. If the resultant file is too large to FTP, “zip” the SCS data by cruise instead and upload files separately

1. Use the following convention for naming the zip files:

shipname_yyyymmdd_YYYYMMDD.zip

shipname = your ship's name (e.g., freeman, kaimimoana, mcarthur, sette, jordan, brown, oregon, albatross, delaware, gunter, or foster),

yyymmdd = start date of data in file (4-digit year, 2-digit month, 2-digit day), and

YYYYMMDD = last date of data in file (year, month, and day).

For example for data from multiple cruises of the MILLER FREEMAN from Aug. 10 to Nov. 25, 2003, the filename would be freeman_20030810_20031125.zip.

2. Login to NODC's anonymous FTP server. Enter "ftp <ftp.nodc.noaa.gov>".
3. Go to the SCS directory on the FTP server. Enter "cd pub/incoming/scs".
4. Upload the zip file(s) (for example, enter "mput *.zip"), and then log off.

Copy the data to the CD Server

1. On the backup server (or the PC which contains the CD Writer), bring up a Windows Explorer window. As the backup server should be setup similar to the primary server, there should already exist a \DataLog directory on the data disk drive (if not, create the \DataLog directory now). Under the DataLog directory, there should be an EventData subdirectory (if not, create it now).
2. You should now have the following directory highlighted: D:\DataLog\EventData\.
3. On the backup server (or the PC which contains the CD Writer), bring up another Windows Explorer window. If you do not already have a Y:\ drive mapped to the SCSLog share on the primary SCS server, do so now. Select the Y:\ drive, and then select the EventData subdirectory (Y:\EventData\).
4. Select the NODC subdirectory from the Y:\EventData\ directory; then, drag it over to the first Windows NT Explorer Window (from step #1). Drop the NODC directory onto the D:\DataLog\EventData\ directory (if the directory already exists, overwrite it with the new one).

Burn the CD

When all data has been successfully moved from the primary SCS server to the backup SCS server (or the PC which contains the CD Writer), the NODC Event Log data should now be backed up to a CD.

NOTE: Several cruises of NODC event data maybe stored on a single CD therefore the above procedure would have been repeated. The frequency of data transfer to NODC is defined as once per month.

The standard COTS application being used for this backup is Adaptec's CD Writer Deluxe, but this may not be the case on all NOAA vessels. Either way, the steps involved should still remain the same:

1. Open the CD Writer application.
2. If there's a wizard available, select the Data CD choice, then select the D:\DataLog\EventData\NODC\ directory to be backup up. Then, follow along until the CD is written.
3. If no wizard is available, you'll simply need to drag the D:\DataLog\EventData\NODC\ directory from a Windows NT Explorer window into the CD Writer application window.
4. Most CD Writer applications will allow for a test to be performed before the CD is written. This test procedure is only necessary if it has never been run before (as it tests to see at what speed(s) the CD can be written). Otherwise, this test step can be skipped and you can go directly into creating the CD.
5. Once the CD is created, take it out of the CD Writer and place it into the standard CD reader on the server PC. Bring up a Windows NT Explorer window and make sure the CD is readable and the data files are all there.

Label the CD and Mail to NODC

An NODC data CD is to be created and mailed to NODC once per month.

Label the CD and its cover with the title “NODC Event Log Data”. Include the following information on both the CD and the its cover:

- Vessel Name
- All cruises and/or legs included on the CD, in the form (VV-YY-CC(-legN), where VV is the vessel id, YY is the year, CC is the cruise number of that year, and N is the leg number in the cruise, if there is more than one.
- Start and end dates for the data, in the form MM-DD-YYYY

Mail the CD to:

Officer in Charge of Data Collection

National Oceanographic Data Center

SSMC III, E/OC1, #4637

1315 East-West Highway

Silver Spring, MD 20910

Response from the NODC Office.

The NODC office will email you an acknowledgement of receipt of the data, together with an accession number. NODC will copy the NMAO mail list eed.nodc@noaa.gov.

If you do not receive an acknowledgment within seven days, email eed.nodc@noaa.gov to request one.

APPENDIX FIVE

TSG DATA SUBMISSION PROCEDURES

14 APPENDIX FIVE: TSG Data Submission Program

14.1 Introduction and Background

The NOAA Atlantic Oceanographic and Meteorological Laboratory (AMOL) have a long-standing program of collecting periodic XBT and TSG data from any and all volunteer ships at sea.

The NOAA Office of Marine and Aviation Operations (OMAO) has agreed to participate in this program by recording TSG data for daily transmission to the AOML data center.

Aboard NOAA ships the SCS event logger app will

- Edit the `TsgTransmitter` event log template to customize it for the ship's TSG configuration.
- Execute the `TsgTransmitter` event to assemble the necessary data into an observations file. Like the NODC and SAMOS events, this will run 24/7 while at sea.
- Run the SCS the "TSG Transmitter" application. This will reformat the event log observations into a format appropriate for submission to the AOML TSG project.

14.2 TSG Data Standard

AMOL has defined a series of standards governing what data is to be submitted and the format of the submitted data. These requirements translate into the setup of the `TsgTransmitter` event.

- Each sensor observation is an instantaneous value, as opposed to a value averaged over a time period.
- Data values are to be recorded every 30 seconds continuously throughout the day (2880 samples per day), and an entire day's data is to be mailed to the *AOML TSG* project once per day at time *0001 GMT*.
- Each sensor observation consists of a designator and a sensor value separated by a colon. For example for true wind direction the designator is `TWDIR` and a sample observation would look like "`TWDIR:214`." Each kind of data will have a NOAA-wide keyword assigned to it, and a standard unit of measurement. The following two tables summarize the data to be submitted:

TSG METADATA		
Designator	Parameter	Units
\$AOML001	Metadata record	N/A
\$AOML002	TSG DATA Record	N/A
WMOID	Ship Call Sign	
SHIPN	Ship Name	
IMONO	IMO Number	
TSGMOD	TSG Model	
TSGSN	TSG Serial Number	
IDP	TSG intake depth	
DCAL	TSG Calibration Date	
COTG	TSG Temp Coefficient G	
COTH	TSG Temp Coefficient H	
COTI	TSG Temp Coefficient I	
COTJ	TSG Temp Coefficient J	
COTF0	TSG Temp Coefficient F0	
COCG	TSG Conductivity Coefficient G	
COCH	TSG Conductivity Coefficient H	
COCI	TSG Conductivity Coefficient I	
COCJ	TSG Conductivity Coefficient J	
COCPR	TSG Conductivity Coefficient CPcor	
COCTR	TSG Conductivity Coefficient CTcor	
COA0	SBE 38 Temp Coefficient A0	
COA1	SBE 38 Temp Coefficient A1	
COA2	SBE 38 Temp Coefficient A2	
COA3	SBE 38 Temp Coefficient A3	
COSL	SBE 38 Temp Slope	
COTO	SBE 38 Temp Offset	
EXTMOD	SBE 38 External temp model	
EXTSN	SBE 38 External temp serial number	

EXTTD	SBE 38 External intake depth	
EXTDCAL	SBE 38 External calibration date	
SSTMOD	Sea surface model	
SSTSN	Sea surface serial number	
SSTD	Sea surface temperature depth	
SSTDICAL	Sea surface calibration date	
TEC	TSG technician name	
TECM	TSG technician email address	
TECP	TSG technician phone	
TECA	TSG Technician alternate name	
TECAM	TSG alternate email address	
TECAP	TSG alternate phone	
YMD	Year, month, day	
HMS	Hour, minute, seconds	UTC

TSG DATA VALUES All samples instantaneous. No averaging		
Designator	Data Value	Units
LAT	Latitude	Decimal Degrees, +N, -S
LON	Longitude	Decimal Degrees, +E, -W
INT	Internal TSG temperature*	Degrees Celsius
SAL	TSG salinity*	
COND	TSG conductivity*	
EXT	TSG secondary temperature*	Degrees Celsius
SST	Sea Surface Temperature*	Degrees Celsius

14.3 The TsgTransmitter Event Log

Setting up the TsgTransmitter event log template proceeds as follows. It is important that you follow these specs exactly. The correct operation of the TSG Transmitter app depends on it.

Template Name

The name of the event template should be TsgTransmitter.

Sensor Metaitems

You must create a sensor metaitem for each kind of data to be submitted as follows:

- The name of the sensor metaitem will be the designator shown in the TSG VALUES table above.
- You will connect this metaitem to the sensor you chose to provide the data value.
- Notes: The order of the metaitems does not matter. Metaitems may be put in separate tab groups at your convenience.

TsgTransmitter Data Log File

You must define a continuous output as follows:

- Name is TSG_OBS
- Type is continuous
- Log rate is 30 seconds.
- Select the “Lat/Lon format Decimal Degrees” radio button.
- The list of selected output elements should consist of all sensors you specified in setting up your sensor metaitems above

Notes: The order of the output elements in the output definition does not matter.

14.4 Running the TsgTransmitter Event

You should start the TsgTransmitter Event as soon as you start ACQ, preferably as soon as possible after you leave port. Continue as long as you are at sea.

The TsgTransmitter Event Logger can be stopped at any time, for example if ACQ needs to be stopped. However to avoid large holidays in the data it should be restarted as soon as possible. This will create data files with different event index numbers, but the TsgTransmitter mailer software will handle that easily.

14.5 Important Files Created by the Event Logger

The Event Logger will create two files that are crucial to the operation of the SAMOS Mailer (described in chapter Eight of this manual).

The files are created as usual in the EventTemplates\TsgTransmitter subfolder of your Datalogging directory as defined in the SCS Configuration file.

- TsgTransmitter_OBS_NNN.ELG, where NNN is the Event index number. This contains the observed data for all the SAMOS sensors. The first line in this file contains the names of the sensors that were logged.
- METADATASENOSDESCRIPTION_NNN.CSV, where NNN is the Event index number. This file contains a list of all metaitems and the sensors that you connected to them when you created the event template.

14.6 TSG Transmitter Operations

After the TsgTransmitter event log collects the data, the TSG Transmitter app mails it to AOML. All aspects of the setup and operation of the TSG Transmitter app are covered in Chapter Eight - Utilities.

APPENDIX SIX

RAW DATA SIMULATION

15 APPENDIX SIX - Raw Data Simulation

SCS has two ways of simulating raw data arriving at ACQ, external and internal simulation.

15.1 External Simulation

In external simulation, two separate apps working together, the RawSimulator and the SimulationManager1.0.

The two apps are not part of the standard suite of SCS software, so they are not available from the SCS Menu. However they are in the EXE40 folder after any SCS installation is completed.

RawSimulator and SimulationManager1.0 apps can be run from a separate machine that has several serial ports of its own, enough to send out the desired number of messages to ACQ. ACQ runs on a separate machine, using its own serial ports. The two sets of serial ports are tied together with serial cables.

Alternatively, they can run on the same machine as ACQ, sending data out a separate set of serial ports as above, and looping back onto ACQ's own set of serial ports.

A separate instance of the RawSimulator must be run for each serial port to be used. However, up to five messages may be sent per port. This allows for example a GPS to be simulated fully by sending out GPGGA, GPVTG and any other message out the same serial port. Young wind/met and POSMV equipment also send out multiple messages commonly.

The Raw Simulator is set up separately from each port. When the setup is complete, click "Save" to save the data. The simulation setup information is saved in a BAT file in the SimulatorRawData folder under your Ship directory. Place all raw files to be read in that same folder.

To run a raw simulation you must run the batch file appropriate for the port and RAW file name. Each batch file creates a new instance of the Raw Simulator.

The Simulation Manager is actually optional, to relieve you of the task of starting up multiple Raw Simulator batch files whenever simulation is to be started.

15.2 Internal Simulation

Internal simulation is more straightforward. You put RAW files into the AcqSimulatorData folder, also under your ship directory. Then, using CFE-DB, create devices of type Simulated and specify the name of the RAW file that is to be used. Using Legacy CFE, you would specify the data source of the Parent Sensor to be “simulation” and specify the RAW file name.

When ACQ starts it will read the RAW files themselves and treat them as incoming messages. All other functions of ACQ, and any client apps, are not aware of any data being simulated.

With Legacy CFE it is easy to change the source from “Simulated” to “Serial Port” without any other changes to the sensor configuration. Thus it is easy to simulate data on shore and switch to actual serial port inputs while at sea with minimal change to the sensor configuration.

CFE-DB is not as easy to manage. Simulated Devices, once created cannot be changed to serial port input. To switch data sources between simulated and serial port, two sets of devices, messages and fields must be created simulated and serial port, and one set must be disabled while the other set to be used.