



**R E F E R E N C E
M A N U A L**

Version 6.0

SMS 6.0

Copyright © 1999 Brigham Young University - Environmental Modeling Research Laboratory January 28, 1999

All Rights Reserved

Unauthorized duplication of the *SMS* software or user's manual is strictly prohibited.

THE BRIGHAM YOUNG UNIVERSITY ENVIRONMENTAL MODELING RESEARCH LABORATORY MAKES NO WARRANTIES EITHER EXPRESS OR IMPLIED REGARDING THE PROGRAM *SMS* AND ITS FITNESS FOR ANY PARTICULAR PURPOSE OR THE VALIDITY OF THE INFORMATION CONTAINED IN THIS USER'S MANUAL

The software *SMS* is a product of the Environmental Modeling Research Laboratory of Brigham Young University and is distributed by contract through Environmental Modeling Systems, Inc. (EMS-I). For more information about this software and related products, contact EMS-I:

EMS-I
719 North 1890 West Suite 38-B
Provo, UT 84601

Phone:801-373-5200 FAX:801-375-6410
info@ems-i.com
<http://www.ems-i.com>

TABLE OF CONTENTS

1	INTRODUCTION	1-1
1.1	OVERVIEW	1-1
1.2	COMMAND LINE ARGUMENTS	1-3
1.3	MODULES.....	1-4
1.3.1	<i>2D Mesh Module</i>	1-5
1.3.2	<i>2D Boundary Fitted Grid Module</i>	1-5
1.3.3	<i>2D Scatter Point Module</i>	1-5
1.3.4	<i>Map Module</i>	1-5
1.3.5	<i>River Module</i>	1-6
1.4	FUNCTIONAL DATA SETS	1-6
1.5	ABOUT THIS MANUAL	1-7
2	INTERFACE LAYOUT	2-1
2.1	SMS SCREEN	2-1
2.2	THE GRAPHICS WINDOW.....	2-3
2.3	TOOL PALETTE.....	2-4
2.3.1	<i>The Module Palette</i>	2-4
2.3.2	<i>The Static Tool Palette</i>	2-5
2.3.3	<i>The Dynamic Tool Palette</i>	2-6
2.3.4	<i>The Macros Tool Palette</i>	2-6
2.4	EDIT WINDOW.....	2-7
2.4.1	<i>The Edit Box</i>	2-7
2.4.2	<i>The Help Window</i>	2-7
2.5	MENU BAR.....	2-8
2.5.1	<i>File Menu</i>	2-8
2.5.2	<i>Edit Menu</i>	2-17
2.5.3	<i>Display Menu</i>	2-19
2.6	PLOT WINDOW	2-21
2.6.1	<i>Plot Options</i>	2-21
2.7	RIVER WINDOW.....	2-22
3	DATA VISUALIZATION	3-1
3.1	BROWSER.....	3-2
3.1.1	<i>File I/O</i>	3-3
3.1.2	<i>Active Data Set</i>	3-4
3.1.3	<i>Elevations & Automatic Data Sets</i>	3-4
3.1.4	<i>Deleting Data Sets</i>	3-5
3.1.5	<i>Data Set Info</i>	3-5
3.2	DATA CALCULATOR.....	3-5
3.3	VECTOR OPTIONS	3-6
3.4	CONTOUR OPTIONS.....	3-8
3.5	COLOR RAMP OPTIONS	3-11
3.6	CONTOUR LABELS	3-12
3.7	FILM LOOPS	3-13

3.7.1	<i>Film Loop Setup</i>	3-14
3.7.2	<i>Film Loop Playback</i>	3-17
3.7.3	<i>Saving Film Loops</i>	3-17
3.8	THE PLOT MANAGER.....	3-18
3.8.1	<i>Plot Objects</i>	3-18
4	MESH MODULE	4-1
4.1	MESH GENERATION.....	4-1
4.2	CURRENT NUMERICAL MODEL.....	4-2
4.3	TOOL PALETTE.....	4-2
4.3.1	<i>Create Mesh Nodes</i>	4-2
4.3.2	<i>Select Mesh Nodes</i>	4-2
4.3.3	<i>Create Nodestrings</i>	4-3
4.3.4	<i>Select Nodestrings</i>	4-4
4.3.5	<i>Create Elements</i>	4-4
4.3.6	<i>Select Elements</i>	4-6
4.3.7	<i>Swap Edges</i>	4-6
4.3.8	<i>Merge/Split Elements</i>	4-7
4.3.9	<i>Label Contours</i>	4-7
4.3.10	<i>Create Piers</i>	4-7
4.3.11	<i>Select Piers</i>	4-8
4.4	DATA MENU.....	4-8
4.4.1	<i>Mesh -> Scatterpoint</i>	4-8
4.4.2	<i>Mesh -> Grid</i>	4-9
4.4.3	<i>Material -> Feature</i>	4-9
4.4.4	<i>Mesh Contour -> Feature</i>	4-9
4.4.5	<i>Map Elevation</i>	4-10
4.5	MESH DISPLAY OPTIONS.....	4-11
4.5.1	<i>Mesh Quality</i>	4-13
4.6	NODE MENU.....	4-15
4.6.1	<i>Interpolation Options</i>	4-15
4.6.2	<i>Interpolate</i>	4-16
4.6.3	<i>Find Node</i>	4-16
4.6.4	<i>Duplicate Nodes</i>	4-17
4.6.5	<i>Select Disjoint Nodes</i>	4-17
4.6.6	<i>Locked</i>	4-18
4.6.7	<i>Transform</i>	4-18
4.6.8	<i>Options</i>	4-19
4.6.9	<i>Interpolate Nodal BC</i>	4-21
4.7	ELEMENTS MENU.....	4-21
4.7.1	<i>Options</i>	4-21
4.7.2	<i>Triangulate</i>	4-23
4.7.3	<i>Optimize Triangulation</i>	4-24
4.7.4	<i>Rectangular Patch</i>	4-25
4.7.5	<i>Triangular Patch</i>	4-26
4.7.6	<i>Select Thin Triangles</i>	4-27
4.7.7	<i>Find Element</i>	4-28
4.7.8	<i>Add Breaklines</i>	4-28
4.7.9	<i>Merge Triangles</i>	4-29
4.7.10	<i>Split Quadrilaterals</i>	4-29
4.7.11	<i>Quad8 <-> Quad9</i>	4-30
4.7.12	<i>Linear <-> Quadratic</i>	4-30

4.7.13	<i>Refine</i>	4-30
4.7.14	<i>Relax</i>	4-30
4.7.15	<i>Smooth Nodestring</i>	4-31
4.7.16	<i>Renumber</i>	4-32
4.7.17	<i>Assign Material Type</i>	4-33
4.7.18	<i>Assign Control Structure</i>	4-33
4.7.19	<i>Assign Junction Type</i>	4-34
4.8	MODEL MENUS	4-34
5	RMA2 INTERFACE.....	5-1
5.1	OPEN SIMULATION.....	5-1
5.2	SAVE SIMULATION.....	5-2
5.3	ASSIGN BC.....	5-3
5.3.1	<i>Assign Boundary Conditions to Nodes</i>	5-3
5.3.2	<i>Assigning Boundary Conditions to Nodestrings</i>	5-5
5.4	DELETE BC.....	5-6
5.5	ADD GC STRING	5-6
5.6	MATERIAL PROPERTIES.....	5-7
5.6.1	<i>Material Roughness</i>	5-7
5.6.2	<i>Roughness by Depth</i>	5-8
5.6.3	<i>Eddy Viscosity</i>	5-8
5.6.4	<i>Marsh Porosity</i>	5-10
5.7	MODEL CHECK.....	5-10
5.8	MODEL CONTROL.....	5-11
5.8.1	<i>Job Title</i>	5-12
5.8.2	<i>File Control</i>	5-12
5.8.3	<i>Iteration Control</i>	5-13
5.8.4	<i>Computation Time Control</i>	5-14
5.8.5	<i>Units Control</i>	5-14
5.8.6	<i>Other Options Control</i>	5-15
5.8.7	<i>Solution Type</i>	5-15
5.9	OPTIONAL BC CONTROLS.....	5-15
5.9.1	<i>Machine</i>	5-16
5.9.2	<i>Geometry</i>	5-17
5.9.3	<i>Dry Elements</i>	5-17
5.9.4	<i>Marsh Porosity</i>	5-18
5.9.5	<i>Roughness By Depth</i>	5-19
5.9.6	<i>Peclet Number Control</i>	5-19
5.9.7	<i>Echo Control</i>	5-19
5.10	SPECIAL ELEMENTS	5-19
5.10.1	<i>One-Dimensional Line Elements</i>	5-19
5.10.2	<i>Transition Elements</i>	5-20
5.10.3	<i>Junction Elements</i>	5-20
5.10.4	<i>Control Structures</i>	5-21
5.10.5	<i>One-Dimensional Geometry</i>	5-22
5.11	RUN GFGEN/RMA2.....	5-23
5.12	DISPLAY OPTIONS	5-24
5.12.1	<i>Nodal Display Options</i>	5-24
5.12.2	<i>Nodestring Display Options</i>	5-25
6	SED2D-WES INTERFACE.....	6-1
6.1	SED2D-WES FILE I/O.....	6-1

6.1.1	<i>New Simulation</i>	6-2
6.1.2	<i>Open Simulation</i>	6-2
6.1.3	<i>Save Simulation</i>	6-2
6.2	GLOBAL PARAMETERS.....	6-2
6.2.1	<i>Bed Type</i>	6-3
6.2.2	<i>Initial Concentrations</i>	6-7
6.2.3	<i>Diffusion Coefficients</i>	6-7
6.2.4	<i>Other Parameters</i>	6-7
6.3	LOCAL PARAMETERS.....	6-7
6.4	BC CONCENTRATIONS.....	6-9
6.5	MODEL CONTROL.....	6-9
6.6	PRINT CONTROL.....	6-11
6.7	CREATE DATA SETS.....	6-11
6.8	MODEL CHECKER.....	6-11
7	HIVEL INTERFACE.....	7-1
7.1	ACCESSING THE HIVEL2D INTERFACE.....	7-1
7.2	NEW SIMULATION.....	7-2
7.3	OPEN SIMULATION.....	7-2
7.4	SAVE SIMULATION.....	7-2
7.5	BUILD HOT START.....	7-3
7.6	ASSIGN BC.....	7-4
7.6.1	<i>Nodal Boundary Conditions</i>	7-5
7.6.2	<i>Nodestring Boundary Conditions</i>	7-5
7.7	DELETE BC.....	7-6
7.8	MODEL CONTROL.....	7-6
7.8.1	<i>Job Titles</i>	7-7
7.8.2	<i>Turbulence Coefficients</i>	7-7
7.8.3	<i>Units Control</i>	7-8
7.8.4	<i>Petrov-Galerkin Coefficients</i>	7-8
7.8.5	<i>Gravity and Manning's Conversion Constants</i>	7-8
7.8.6	<i>Iterations</i>	7-8
7.8.7	<i>Computation Time</i>	7-8
7.8.8	<i>Temporal Derivative</i>	7-9
7.8.9	<i>Reset Defaults</i>	7-9
7.9	MATERIAL PROPERTIES.....	7-9
7.10	MODEL CHECK.....	7-10
7.11	RUN HIVEL.....	7-11
7.12	DISPLAY OPTIONS.....	7-11
7.12.1	<i>Nodal Display Options</i>	7-12
7.12.2	<i>Nodestring Display Options</i>	7-12
8	ADCIRC INTERFACE.....	8-1
8.1	NEW SIMULATION.....	8-1
8.2	OPEN SIMULATION.....	8-2
8.3	SAVE SIMULATION.....	8-2
8.4	ASSIGN BC.....	8-3
8.4.1	<i>Normal Flow Parameters</i>	8-4
8.5	CREATE FUNCTIONS.....	8-5
8.6	MODEL CHECK.....	8-7
8.7	MODEL CONTROL.....	8-7
8.7.1	<i>Station Output</i>	8-9

8.7.2	<i>Global Output</i>	8-10
8.7.3	<i>Wind</i>	8-11
8.7.4	<i>Time Control</i>	8-12
8.7.5	<i>Bottom Friction</i>	8-13
8.7.6	<i>Tidal Forces</i>	8-14
8.7.7	<i>Harmonic Analysis</i>	8-14
8.7.8	<i>Solver</i>	8-15
8.8	RUN ADCIRC.....	8-16
9	CGWAVE INTERFACE	9-1
9.1	NEW SIMULATION.....	9-2
9.2	OPEN SIMULATION.....	9-2
9.3	SAVE SIMULATION.....	9-2
9.4	ASSIGN BC.....	9-2
9.5	CREATE FUNCTIONS.....	9-3
9.6	MODEL CHECK.....	9-5
9.7	MODEL CONTROL.....	9-5
9.7.1	<i>Title</i>	9-6
9.7.2	<i>Incident Wave Conditions</i>	9-6
9.7.3	<i>Open Boundary</i>	9-7
9.7.4	<i>1-D</i>	9-7
9.7.5	<i>Solver</i>	9-8
9.7.6	<i>Iteration Control</i>	9-8
9.7.7	<i>Model Parameters</i>	9-8
9.8	RUN CGWAVE.....	9-8
10	FESWMS INTERFACE	10-1
10.1	OPEN SIMULATION.....	10-2
10.2	SAVE SIMULATION.....	10-3
10.3	NODAL BOUNDARY CONDITIONS.....	10-3
10.3.1	<i>Specify x/Tangent Condition</i>	10-4
10.3.2	<i>Specify y/Normal Condition</i>	10-5
10.3.3	<i>Specify Water Surface or Source/Sink</i>	10-5
10.4	BOUNDARY SECTION.....	10-5
10.4.1	<i>Flow</i>	10-5
10.4.2	<i>Water Surface Elevation</i>	10-6
10.4.3	<i>Rating Curve/Friction Slope</i>	10-6
10.5	INITIAL CONDITIONS.....	10-7
10.6	WIND CONDITIONS.....	10-7
10.7	WEIRS.....	10-8
10.8	CULVERTS.....	10-9
10.9	DROP INLETS.....	10-9
10.10	PIERS.....	10-9
10.11	NODE CEILINGS.....	10-10
10.12	FLUX STRING.....	10-10
10.13	MATERIAL PROPERTIES.....	10-10
10.14	MODEL CHECK.....	10-12
10.15	FESWMS CONTROL.....	10-12
10.15.1	<i>Project Title</i>	10-13
10.15.2	<i>FESWMS Version</i>	10-13
10.15.3	<i>FLO2DH Input</i>	10-13
10.15.4	<i>FLO2DH Output</i>	10-14

10.15.5	Units.....	10-14
10.15.6	Solution Type.....	10-14
10.15.7	Slip Conditions.....	10-14
10.15.8	Higher Order Integration.....	10-15
10.15.9	Bottom Stresses.....	10-15
10.15.10	Control Buttons.....	10-15
10.16	FESWMS DISPLAY OPTIONS.....	10-19
10.16.1	Nodal Display Options.....	10-20
10.16.2	Nodestring Display Options.....	10-20
11	RIVER MODULE	11-1
11.1	RIVER MODULE TOOLS.....	11-2
11.1.1	Creating Sections.....	11-3
11.1.2	Editing Sections.....	11-3
11.2	RIVER PLOTS.....	11-6
11.2.1	River Profile Plots.....	11-6
11.2.2	River Cross Section Plots.....	11-8
	SELECT THE CURRENT PLOT.....	11-8
12	WSPRO INTERFACE	12-1
12.1	NEW SIMULATION.....	12-2
12.2	OPEN SIMULATION.....	12-2
12.3	SAVE SIMULATION.....	12-2
12.4	SECTION CREATION AND ATTRIBUTES.....	12-2
	Create Road Section.....	12-3
	Create Culvert Section.....	12-4
12.4.3	Cross Section Attributes.....	12-5
12.4.4	Bridge Section Attributes.....	12-7
12.4.5	Guide Bank Section Attributes.....	12-10
12.5	ROUGHNESS PARAMETERS.....	12-11
12.6	WSPRO RUN CONTROL.....	12-12
12.7	JOB PARAMETERS.....	12-13
12.8	MODEL CHECK.....	12-14
12.9	RUN WSPRO.....	12-15
13	SCATTER POINT MODULE.....	13-1
13.1	SCATTER POINT SETS.....	13-2
13.2	INPUTTING SETS.....	13-2
13.3	SAVING SETS.....	13-2
13.4	TOOL PALETTE.....	13-3
13.4.1	Select Scatter Point.....	13-3
13.4.2	Select Scatter Point Set.....	13-3
13.5	DISPLAY OPTIONS.....	13-3
13.6	SCATTER POINT CONVERSION.....	13-4
13.6.1	Scatter Points -> Mesh Nodes.....	13-4
13.7	INTERPOLATION.....	13-4
13.7.1	Interpolation From Scatter Point Sets.....	13-4
13.7.2	Interpolation Options.....	13-4
13.7.3	Linear Interpolation.....	13-5
13.7.4	Inverse Distance Weighted Interpolation.....	13-6
13.7.5	Natural Neighbor Interpolation.....	13-12
14	MAP MODULE.....	14-1

14.1	FEATURE OBJECTS.....	14-2
14.1.1	Feature Object Types.....	14-2
14.1.2	Feature Object Tools.....	14-4
14.1.3	Feature Object Commands.....	14-5
14.1.4	Coverages.....	14-9
14.1.5	Feature Object Attributes.....	14-12
14.1.6	Constructing Numeric Models.....	14-17
14.1.7	Display Options.....	14-20
14.2	DRAWING OBJECTS.....	14-22
14.2.1	Drawing Object Tools.....	14-22
14.2.2	Display Attributes.....	14-23
14.2.3	Display Options.....	14-25
14.2.4	Drawing Order.....	14-25
14.3	IMAGES.....	14-26
14.3.1	Importing an Image.....	14-26
14.3.2	Registering an Image.....	14-27
14.3.3	Resampling an Image.....	14-29
14.3.4	Fit Entire Image.....	14-30
14.3.5	Deleting Images.....	14-30
14.3.6	Exporting the Resampled Region.....	14-30
14.3.7	Export TIFF vs. Save Image.....	14-32
14.4	DXF FILES.....	14-32
14.4.1	Importing DXF Files.....	14-32
14.4.2	Display Options.....	14-32
14.4.3	DXF -> Feature Objects.....	14-33
14.4.4	DXF -> Scatter Points.....	14-34
14.4.5	Deleting DXF Files.....	14-34
14.5	READING AND SAVING MAP FILES.....	14-35
15	OBSERVATION TOOLS.....	15-1
15.1	OVERVIEW OF VERIFICATION PROCESS.....	15-2
15.2	OBSERVATION COVERAGE.....	15-2
15.2.1	Interpolation Method.....	15-3
15.2.2	Measurement Type List.....	15-4
15.2.3	New/Delete/Name.....	15-4
15.2.4	Steady State vs. Transient.....	15-4
15.3	OBSERVATION POINTS.....	15-5
15.3.1	Creating Observation Points.....	15-5
15.3.2	Importing Observation Points.....	15-7
15.3.3	Calibration Targets.....	15-9
15.3.4	Point Error Statistics.....	15-10
15.4	ARCS.....	15-10
15.5	DISPLAY OPTIONS.....	15-11
15.5.1	Points.....	15-12
15.5.2	Arcs.....	15-13
15.6	PLOTTING OPTIONS.....	15-13
15.6.1	Open/Close Plot Window.....	15-14
15.6.2	Plot Options Dialog.....	15-14
16	XY SERIES EDITOR.....	16-1
16.1	XY SERIES LIST.....	16-1
16.2	THE XY EDIT FIELDS.....	16-2

16.2.1	<i>Delete</i>	16-3
16.2.2	<i>Interpolate</i>	16-3
16.2.3	<i>Update</i>	16-3
16.2.4	<i>Insert</i>	16-3
16.2.5	<i>Compress</i>	16-3
16.2.6	<i>XY Options</i>	16-3
16.3	THE XY SERIES PLOT.....	16-4
16.3.1	<i>The Plot Tools</i>	16-5
16.3.2	<i>The Plot Macros</i>	16-5
17	FILE FORMATS	17-1
17.1	2D-MESH FILES.....	17-2
17.2	2D SCATTER POINT FILES.....	17-4
17.3	ASCII DATA SET FILES.....	17-9
17.4	ASCII SCALAR DATA SET FILES (VERSION 4).....	17-13
17.5	ASCII VECTOR DATA SET FILES (VERSION 4).....	17-15
17.6	BINARY DATA SET FILES.....	17-17
17.7	BINARY SCALAR DATA SET FILES (VERSION 4).....	17-21
17.8	BINARY VECTOR DATA SET FILES (VERSION 4).....	17-23
17.9	BOUNDARY ID FILES.....	17-25
17.10	BOUNDARY XY FILES.....	17-26
17.11	COASTLINE FILES.....	17-26
17.12	DROGUE FILES.....	17-26
17.13	MAP FILES.....	17-27
17.13.1	<i>General</i>	17-29
17.13.2	<i>Feature Objects</i>	17-29
17.13.3	<i>Feature Object Attributes</i>	17-32
17.13.4	<i>Drawing Objects</i>	17-35
17.14	IMAGE FILES.....	17-39
17.15	MATERIAL FILES.....	17-41
17.16	MESH FROM POLYGON FILES.....	17-43
17.17	SMS SUPER FILES.....	17-45
17.18	TIN FILES.....	17-46
17.19	XY SERIES FILES.....	17-48
17.20	XYZ FILES.....	17-51
18	REFERENCES	18-1
19	INDEX	19-1

Introduction

SMS is a comprehensive environment for one-, two-, and three-dimensional hydrodynamic modeling. New enhancements and developments continue at the Environmental Modeling Research Laboratory (EMRL) at Brigham Young University (formerly known as the Engineering Computer Graphics Laboratory) in cooperation with the U.S. Army Corps of Engineers Waterways Experiment Station (USACE-WES), and the U.S. Federal Highway Administration (FHWA).

1.1 Overview

SMS is a pre- and post-processor for surface water modeling and analysis. It includes two-dimensional finite element, two-dimensional finite difference, three-dimensional finite element and one-dimensional backwater modeling tools. Interfaces specifically designed to facilitate the utilization of several numerical models comprise the modules of *SMS*. Supported models include the USACE-WES supported *TABS-MD* (*GFGEN*, *RMA2*, *RMA4*, *RMA10*, *SED2D-WES*), *ADCIRC*, *CGWAVE*, *STWAVE*, and *HIVEL2D*. Comprehensive interfaces have also been developed for facilitating the use of the FHWA commissioned analysis packages *FESWMS* and *WSPRO*.

These hydrodynamic modeling programs will calculate water surface elevations and flow velocities for shallow water flow problems and support both a steady-state and dynamic model. Additional tools are provided in *SMS* to support the modeling of contaminant migration and sediment transport.

The finite element mesh, finite difference grid, or cross section entities along with associated boundary conditions necessary for analysis may be created within *SMS* and saved to model specific files. These files are used to perform the hydrodynamic,

contaminant migration, and sediment transport analyses. The numerical models create solution files that contain the water surface elevations, flow velocities, contaminant concentrations, sediment concentrations or other functional data at each node, cell, or section. *SMS* reads this data and generates profiles and cross sectional plots, two-dimensional vector plots, drogue plots, color-shaded contour plots, time variant curve plots, and dynamic animation sequences.

SMS can also be used as a pre- and post-processor for other finite element or finite difference programs provided the programs can be made to read and write files in a supported format. *SMS* is well suited for the construction of large, complex meshes (several thousand elements) of arbitrary shape. Sample meshes are shown in Figure 1.1 and Figure 1.2.

This document is meant to be a reference manual for *SMS* only. While a chapter describing model interface components is included, details on the analysis software are contained in separate documents. It is assumed that the user is familiar with the overall modeling process and terms defined in the appropriate supporting reference manual.

SMS was designed as a comprehensive hydrodynamic modeling system. As part of an ongoing effort, other analysis models will be supported in future versions.

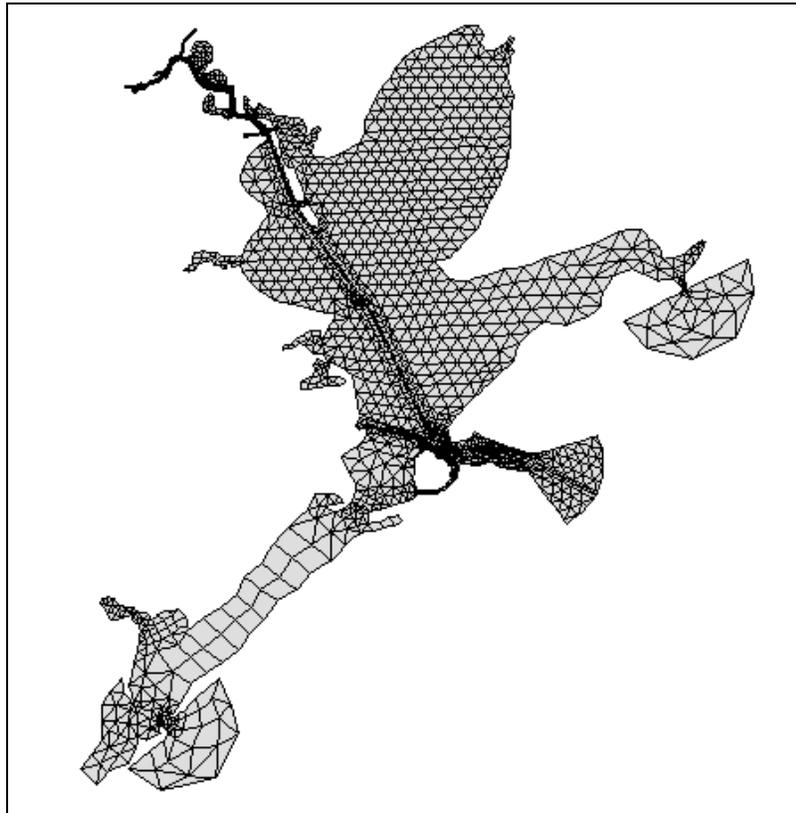


Figure 1.1 *Finite Element Mesh Representing Galveston Bay.*

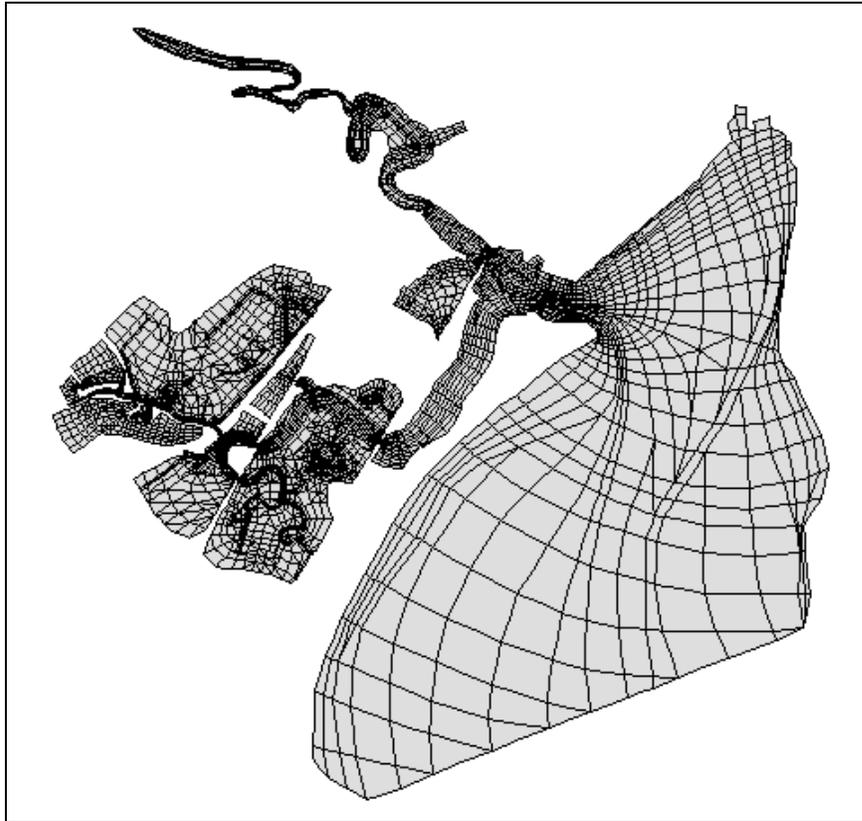


Figure 1.2 Finite Element Mesh Representing Saugus River Estuary.

1.2 Command Line Arguments

SMS provides for a few command line arguments, which facilitate the use of SMS. These arguments are simply entered after the command for SMS at the command line. Some of the arguments are not available on the PC and are so noted.

The following commands are available:

help (Unix only)

The help command displays usage information SMS commands.

Usage: sms -help

about (Unix only)

The about command displays SMS copyright and vendor information.

Usage: sms -about

dm <module>

The default module command is used to specify the default module.

Possible values include tabs, hivel, rma10, adcirc, cgwave, feswms, scat, map, and wspro.

Usage: sms -dm tabs

r <file spec> (Unix only)

The resource directory command is used to specify the path name for the resource directory.

Usage: sms -r /sms/bin

ini <file spec> (Unix only)

The initialization path command is used to specify the path where the initialization file should be read and saved.

Usage: sms -ini /sms/User

tmp <file spec> (Unix only)

The temp path command is used to specify the path where temporary files should be saved.

Usage: sms -tmp /sms/User

Command line arguments may also be used to open the files associated with a current project. After *SMS* processes the above arguments, any other information included on the command line will be read and interpreted as file names that the user wishes to open.

1.3 Modules

The interface for *SMS* is divided into five separate modules. A module is provided for each of the basic data types supported by *SMS*. As the user switches from one module to another, the *Tool Palette* and the menus change. This allows you to focus only on the tools and commands related to the data you are currently processing. Switching from one module to another can be done instantaneously to facilitate the simultaneous use of several data types when necessary. In many cases, the data may be converted from one format to another. The commands and tools available are also specialized based on the numerical model to be employed. For example, the mesh module provides the basic tools for both *TABS* and *FESWMS*. The user selects which model will be used and the tools and menus specific to that model become available. Switching from

one model to another results in the conversion of applicable data. However, data specific to one model may be lost in this conversion. Commands specific to an analysis model reside in a menu corresponding to that model. The following modules are supported in *SMS*.

1.3.1 2D Mesh Module

The *2D Mesh Module* is used to manipulate 2D finite element meshes. A mesh consists of a finite element network and the boundary conditions applied to that network. A variety of tools are provided for mesh generation and mesh editing. In *SMS*, 2D meshes are used as the basis for analysis for the *TABS* suite of software, the supercritical flow package *HIVEL-2D*, the *FESWMS* analysis package and the coastal models *ADCIRC* and *CGWAVE*. After an analysis, output data at each node of the mesh can be used to generate contour, fringe and vector plots to represent the solution. Multiple time steps from time variant solutions can be strung together to form an animation of the dynamic solution.

1.3.2 2D Boundary Fitted Grid Module

The *2D Boundary Fitted Grid Module* will be used to construct 2D boundary fitted grids. These grids can be exported for use in analysis packages or used as the basis of three dimensional finite difference grids. An interface for the three dimensional model *CH3D* is currently under development.

1.3.3 2D Scatter Point Module

The *2D Scatter Point Module*; is used to interpolate from groups of 2D scattered data points to any of the other data types. For example, the user may gather field data points representing the bathymetry of the region to be modeled. The elevation data from these points can be interpolated to a well-structured set of elements to create the bathymetry of the entire mesh. The *2D Scatter Point Module* can be used to interpolate from a set of scattered xy points representing the empirical data to a finite element mesh. A variety of interpolation schemes are supported.

1.3.4 Map Module

The *Map Module* is used to manipulate four types of objects: feature objects, image objects, drawing objects and DXF objects.

Feature objects are patterned after the data model used by geographic information systems (GIS) such as ARC/INFO®. A user constructs a conceptual model of the site to be modeled. *SMS* uses the conceptual model to automatically generate a numerical model (mesh, grid, one-dimensional model, etc). Geometric features, bathymetric

information, and attributes for regions and lines are all incorporated into the numerical model.

The other three object types are primarily used as graphical tools to enhance the development and presentation of a model. Image objects are digital images representing aerial photos or scanned maps in the form of TIFF files. These can be displayed in the background for guides in model construction or visualization aids. Drawing objects include a simple set of tools that are used to draw text, lines, polylines, arrows, rectangles, etc., to add annotation to the graphical representation of a model. DXF objects consist of drawings imported from standard CAD packages such as *AutoCAD*® or *Microstation*®. This data may be used as background information or converted to features or scattered data for model generation.

1.3.5 River Module

The *River Module* is used to construct one dimensional river profiles for cross sectional based models such as the step backwater model *WSPRO*. Tools are provided in this module for creating a “tree” of data to describe the river being modeled. The one-dimensional model can be edited directly or generated from features in the *Map Module*. Tools are provided for the creation of river sections. Properties can be assigned to sections, and structures such as bridges and culverts can be added. After step backwater analysis is performed the results can be viewed either in the cross sections or as profile curves.

1.4 Functional Data Sets

An important feature of *SMS* is that the interface to each of the separate modules is designed in a consistent fashion. Once the user becomes familiar with the interface to one of the modules, the other modules can be used with little additional training. In order to help provide a consistent interface, the concept of generic functional data sets is used in *SMS*. A functional data set, which may be referred to simply as a data set consists of a scalar or vector value for each node in a mesh, each vertex in a scattered data set, or each cell in a grid. A data set can be either steady-state (constant) or dynamic (transient). Dynamic data sets are represented using data sets for different points in time. Meshes, grids, and scatter point sets have independent lists of data sets.

A functional data set of scalar values is used to represent quantities such as the water surface elevation or depth computed by a hydrodynamic model or empirical values used as initial conditions for input to a dynamic model. A functional data set of vector values is used to represent quantities such as flow velocities or stresses. Data sets can be imported from a file, created by interpolating from a scatter point set, or computed using other data sets, constants and mathematical operators. An example of the use of data sets is to compare the difference in the solutions from two separate simulations on the same finite element mesh. The two solutions can be read as data sets and the *Data*

Set Calculator can be used to compute the absolute value of the difference between the two data sets. The resulting data set can be contoured just like any other data set.

1.5 About This Manual

This reference manual has been designed to parallel the modular concept used in *SMS*. Chapter 2 describes the layout of the user interface and some general concepts that apply to numerical modeling. Chapter 3 discusses visualization tools that can be used with functional data sets on all data types. These chapters should be read regardless of which module the user intends to use. Chapter 4 describes the *Mesh Module* and chapters 5-10 describe the model specific interfaces such as *RMA2*, *FESWMS*, and *ADCIRC* for two dimensional finite element models. Chapter 11 contains details of the *River Module* and should be referred to along with chapter 12, which discusses the interface to the one dimensional river model *WSPRO*. Chapters 13 and 14 describe the *Scatter point* and *Map Modules*, which contain utilities useful for all types of modeling supported by *SMS*. Chapter 15 details the tools for model verification. Chapter 16 discusses the *XY Series Editor*, and Chapter 17 describes the *File Formats* for non-model specific files used in *SMS*.

This manual applies to both the Unix version and the MS Windows version of *SMS*. Most of the features are identical between the two versions. Any differences in the two versions are noted explicitly in the documentation.

Interface Layout

The interface to *SMS* has been designed in a modular fashion. Five separate modules representing different data types are supported. As the user switches from one module to another, the available menus and tools change.

This chapter contains information on the portion of the interface that remains the same and provides access to general tools that are used by all of the modules. It also discusses some interface concepts that apply to specific features of numerical models that have common applicability.

2.1 SMS Screen

The *SMS* screen is divided into several sections or windows depending on the data types being processed and the numerical model being used. Three of these sections are common to all data types and models. These are the *Tool Palette*, the *Edit Window*, and the *Menu Bar*. Three additional windows may be part of the screen layout. The *Graphics Window* is the most common and is almost always part of user interaction. The *Plot Window* and *River Window* can be opened when the user wishes to operate with data displayed in these windows. Three sample layouts are displayed in Figure 2.1, Figure 2.2, and Figure 2.3.

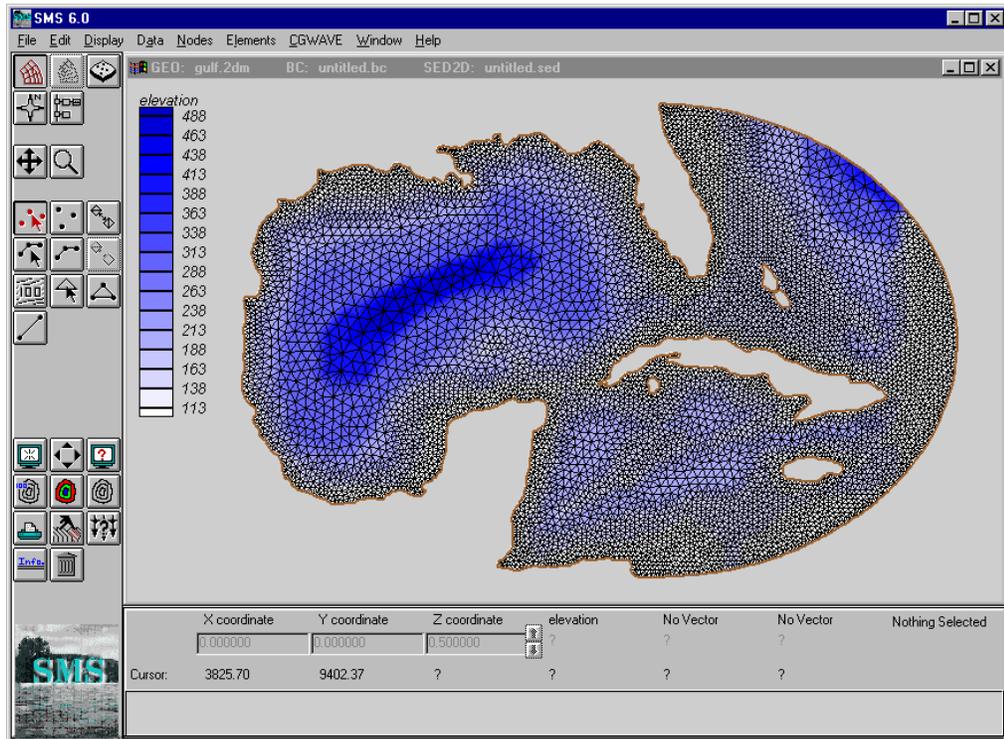


Figure 2.1 The SMS Screen.

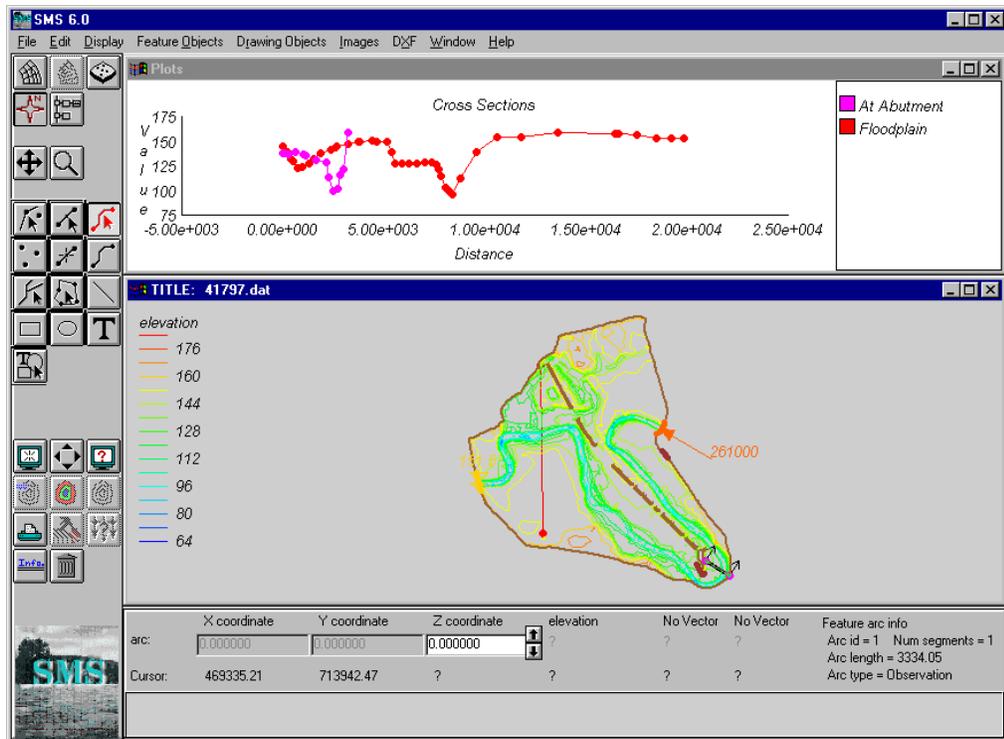


Figure 2.2 SMS Screen w/Plot Window.

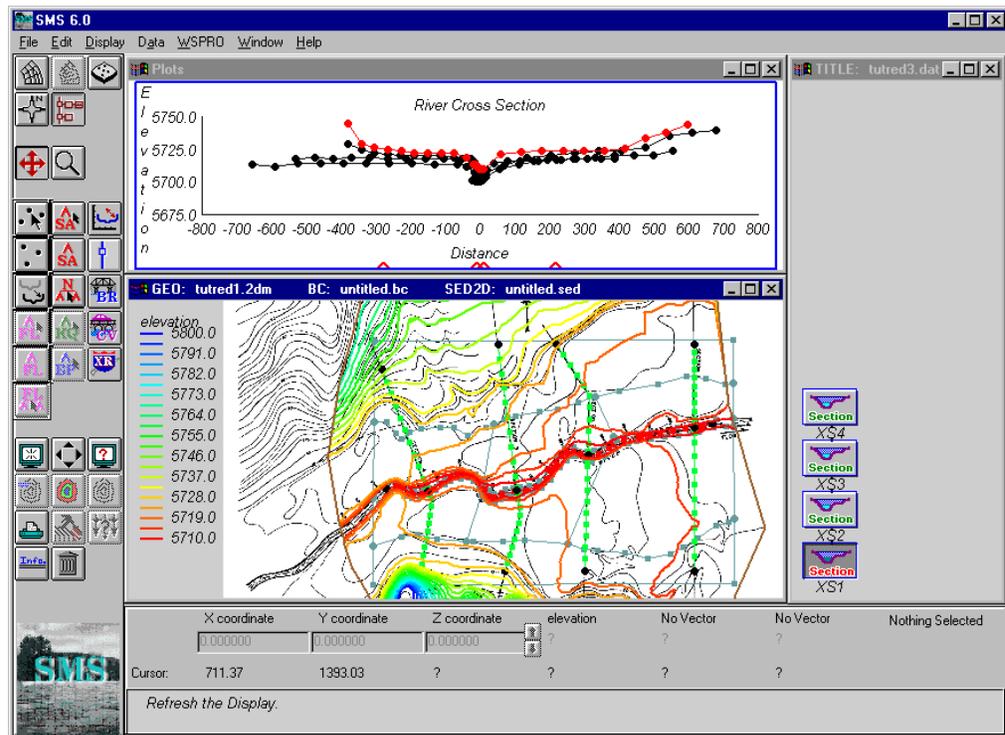


Figure 2.3 The SMS Screen w/Plot and River Windows.

2.2 The Graphics Window

The *Graphics Window* is where *SMS* displays two and three-dimensional data. It is also where the user interacts with that data in *SMS*. The selected tool in the *Tool Palette* determines the type of interaction that can be performed in the *Graphics Window*. For example, if the *Create Node* tool is currently selected any click in the *Graphics Window* will result in the creation of a node at the location of the click.

The user has control of how all data appears in the *Graphics Window*. Each type of entity has an associated set of display attributes. These attributes include visibility, color, line thickness, and font type. Each data type is associated with a specific module and the attributes for that type are controlled via that module's *Display Attributes* dialog (see section 2.5.3).

The *Graphics Window* is integral in the creation, editing and visualization of two-dimensional finite element meshes (see Chapter 4) and two-dimensional finite difference grids. It is also the main means of interacting with a conceptual model and site maps (see Chapter 14). The only time the *Graphics Window* is not utilized is for one-dimensional modeling after the one-dimensional model is generated (see Chapter 12).

2.3 Tool Palette

The *Tool Palette* is divided into four parts as shown in Figure 2.4.

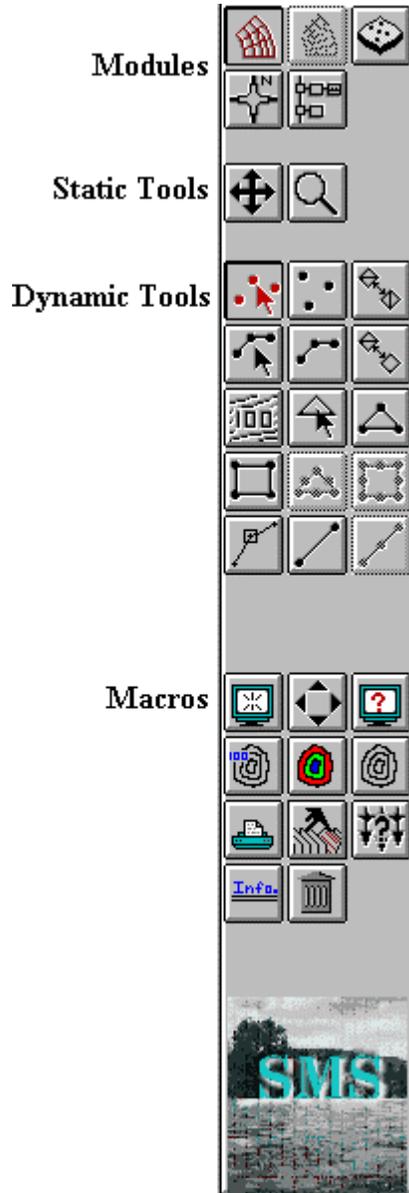


Figure 2.4 SMS Tool Palette.

2.3.1 The Module Palette

The *Module Palette* is used to switch between modules. Only one module is active at any given time. However, the data associated with a module (e.g. a 2D finite element mesh) is preserved when the user switches to a different module. Activating a module only changes the set of available tools and menu commands.

This should not be confused with changing the current model inside of a module. When a new model is selected, the tools and menus may change, and the data will be converted as much as is possible. However, some data may be lost (see Section 4.2).

2.3.2 The Static Tool Palette

The *Static Tool Palette* contains tools that are available in every module and model interface. These tools are used for display manipulation. *SMS 6.0* includes two static tools, pan and zoom.



The Pan Tool

The *Pan* tool is used to pan the viewing area of the *Graphics Window*. When the *Pan* tool is active, clicking the mouse in the *Graphics Window* has the following results:

- If the mouse is clicked (depressed and released) at a single point, the selected location is moved to the center of the window.
- If the mouse is dragged while holding the button down an arrow appears which shows what movement will take place. When the button is released, the point where the button was depressed is moved to the release location. The image isn't updated until the mouse button is released.



The Zoom Tool

The viewing area can be magnified or shrunk using the *Zoom* tool. When this tool is active, the following actions can be used to redefine the viewing area of the *Graphics Window*:

- A rectangle can be dragged around a portion of the display to zoom in on a particular region. The display is refreshed and the area inside the rectangle is expanded to fill the entire screen. (Note: the aspect ratio of the screen is not changed.)
- If the mouse is clicked at a single point, that point is moved to the center of the window, and the size of the data displayed in the window is enlarged by a factor of two.
- A rectangle can be dragged around a portion of the display while the *SHIFT* key is held down to zoom out to a particular region. The display is refreshed and the entire screen area shrinks to the inside of the rectangle.
- If the mouse is clicked at a single point while the *SHIFT* key is held down, that point is moved to the center of the window, and the size of the data displayed in the window is decreased by half.

If a user starts dragging either an arrow or a zoom box, and decides that no change in view is desired, the operation may be aborted by holding the escape key and releasing the button, or dragging outside of the graphics window and releasing the button.

2.3.3 The Dynamic Tool Palette

The *Dynamic Tool Palette* contains tools that apply to the selected module and model. These tools are called dynamic because the available tools change whenever the module or model is changed.

Selection Tools

Many of the dynamic tools are used for selecting a specific entity such as a mesh node, or a grid cell. It is necessary to first select some objects before issuing many of the commands in *SMS*. For example, to delete a specific node requires that the node be selected and then the *Delete* command issued. Specific tools are described individually in later chapters.

Most of the selection tools follow a standard selection protocol. One item can be selected by clicking on the item. When a new item is selected, any other selected items are unselected.

In many cases, multiple items need to be selected. If the *SHIFT* key is held down while clicking on individual items, the items are added to the set of selected items. A previously selected item can be unselected by holding down the *SHIFT* key and clicking on it again. This removes the item from the set of selected items without affecting other selected items. Multiple objects can also be selected by dragging. The user may drag a box around the items to be selected, or hold down the *CONTROL* key and drag an arrow through the items to be selected. If the *SHIFT* key is held down while dragging a selection, previously selected entities are deselected, and previously unselected entities are selected.

Other commands for selecting multiple objects such as *Select With Poly* can be found in the *Edit* menu and are described later in this chapter.

2.3.4 The Macros Tool Palette

The *Macro Tool Palette* contains buttons to perform frequently used menu commands. These buttons serve as shortcuts to menu commands. Not all macros are available in all modules. In this document, when a macro exists for a menu item being described, a picture of the button is shown next to the title.

2.4 Edit Window

There are two sections in the *Edit Window* (Figure 2.5): the *Edit Box* and the *Help Window*.

	X coordinate	Y coordinate	Z coordinate	elevation	No Vector	No Vector	Node info
node:	1567.849310	1356.647900	5720.000000	5720.00	0.00	0.00	Node id = 27 Attached to 9 elements
Cursor:	481.09	1330.23	?	?	?	?	
<i>Split Quads/Merge Triangles tool.</i>							

Figure 2.5 SMS Edit Window.

2.4.1 The Edit Box

The *Edit Box* is on the top half of the *Edit Window*. The top row of the edit box is used to edit the coordinates of the selected mesh node, grid node, or scatter point. The coordinates are changed by typing in new values and hitting the *ENTER* or *TAB* key. If more than one node is selected, only the z coordinate is available for editing. To the right of the z coordinate edit box are two arrows for interactively adjusting the z coordinate of the selected object. Entering a new value here will modify the bathymetry of each of the selected nodes. This allows the user to quickly model a feature such as a dredged channel.

The second row of numbers in the edit box is used to display the coordinates of the cursor. The z coordinate corresponds to an interpolated elevation value from either the mesh or grid, depending on which module is active.

The scalar and vector data values associated with the selected object and the cursor are displayed to the right of the coordinate values.

Other information about the selected object(s) is displayed in the right side of the edit window. The displayed information includes features such as the ID of a selected node, element type of selected element, nodestring type, area of selected elements, etc. This information is only visible for 1024 x768 or higher resolution on PC systems.

2.4.2 The Help Window

SMS includes three methods of online help, including automatic help messages, the Windows help utility (accessed through the *Help* menu), and status prompts. The *Help Window* on the bottom of the *Edit Window* is where *SMS* provides the user with context sensitive help messages or information regarding the current operation. As the cursor is moved over tools, macros, or menu items, a description of the item appears in the *Help Window*. Help messages also appear as the cursor is passed over items in dialog boxes. Status messages appear in bold red text in the same location.

2.5 Menu Bar

The commands in *SMS* are accessed through pull down menus located in the menu bar. Each menu can be accessed either with the mouse or by pressing the highlighted letter in the menu title. Once a menu is visible the individual commands can be selected with the mouse or by pressing the highlighted letter in the menu command.

The menus available at any time are dependent on the active module and current numerical model. The first three menus (*File, Edit, Display*) are the same for every module. The remaining menus change with the module and the model. This system is designed to partition the available commands into usable groups to avoid unnecessary complexity.

2.5.1 File Menu

The *File* menu is one of the standard menus available in all modules. The commands in the file menu are described here.

File Types

The file types supported in the *File* menu correspond to the generic file types only. The commands for opening and saving files associated with specific numerical models such as *RMA2* and *FESWMS* are found in the menus associated with that numerical model. Project or simulation files that reference model specific files are generic. This allows access to model specific files together with related generic files in an easy manner.

The generic file types:

- **Simulation File:** File containing data identifiers along with file names where the specified data is stored. Opening a simulation file can open multiple files in a single command.
- **2D Scatter Point File:** File containing one or more sets of 2D scatter points. There are actually two versions of scatter point file supported by *SMS*. Both include the capability to append multiple functional data sets to the list of (x,y) locations.
- **Material File:** File containing definition of general material parameters. Since materials are different in the different models, this generic file consists of display parameters such as color, name and ID.
- **Image file:** The file name and registration data related to a TIFF image used for background display. The actual TIFF image is saved in a separate file. Importing and registering a TIFF image and then saving the registration file creates this file.

- **Map file:** Feature objects, drawing objects and boundary conditions placed on the feature objects are saved in this file. This file also contains all of the attributes associated with the features that are used for automatic mesh, grid, or tree generation.
- **2D Mesh file:** This is a generic geometry file. It may be used as a translation file to allow *SMS* to create data for a non-supported numerical model. It does not include boundary conditions.
- **Data Set file:** This is a generic file to store functional data sets. It may be ASCII or binary, and may contain a single data set or multiple sets. Each data set may be flagged as steady state or transient. A data set file is flagged to determine whether the data applies to a set of scattered data points, a mesh, or a grid.
- **Settings file:** This file saves the current settings and default values of the program (display options, defaults, etc.).

All of the generic files can be stored as ASCII text files for flexible portability across platforms. Since data set files may contain large amounts of data they can be saved in binary format for economy. The first item in each of these files is a keyword signifying the file type. The formats for these files are described in Chapter 17.

New

The *New* command deletes all the data associated with all modules. It resets the status of the program so that all display option and default values match the values in the “settings” file. This command should be selected when a new modeling problem is started.

Open

The *Open* command is used to read one or more of the generic file types (described above). This command opens a file browser from which a single file can be selected. *SMS* reads the keyword at the beginning of the file to determine its type, reads the file, and displays the data.

Save

The *Save* command is used to save a generic file type. This command opens the *Save Files* dialog (see Figure 2.6). This dialog contains a check box and a filename for each of the basic file types, and a file browser button for the directory to save the files.

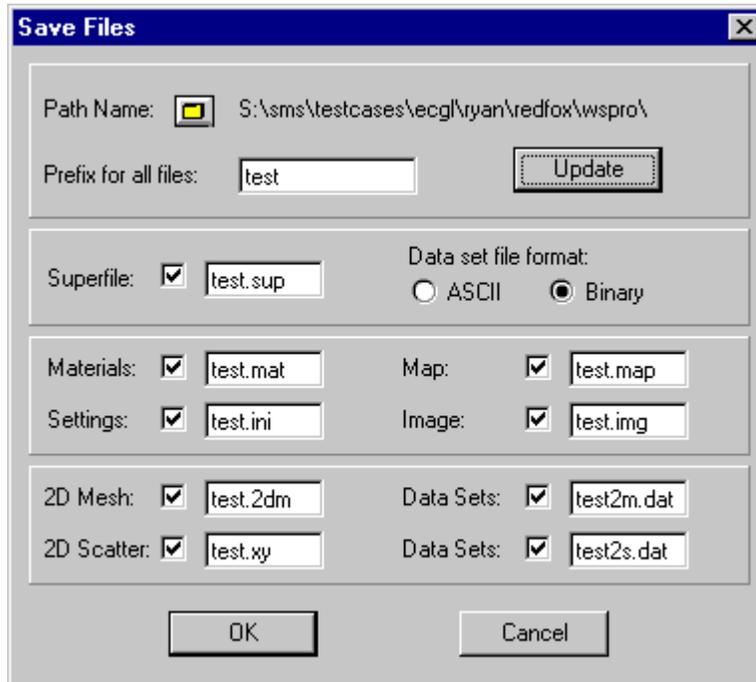


Figure 2.6 The Save Dialog.

If a data type does not currently exist in *SMS* memory, the check box, button, and filename for that data type is dim. The check box can be toggled off or on to indicate whether or not the data type is to be saved. The button is used to bring up the browser to select the target directory. A superfile that contains a list of all the saved files can be generated. The superfile can then be used to open all of the files at once using the *Open* command.

Import File

The *Import* command opens the *Select Import Format* dialog that allows the user to import data from one of several non-standard file types. The formats of these non-standard files are described in Chapter 17 or in model specific documentation. Currently supported data file formats include:

TIN File

A TIN file stores the data for a triangulated irregular network. When *SMS* imports a TIN file, each triangle is converted to a linear triangular element. If you wish to append a TIN to an existing mesh, you must convert the mesh to linear before importing the TIN file. Any quadratic elements are deleted when the TIN is imported. This can be a useful way of generating a background mesh. The use of a background mesh is discussed in Chapter 14.

XYZ File

An *XYZ* file contains a list of coordinate points. When *SMS* imports an *XYZ* file the data points are converted to nodes. This provides a convenient way to import a set of points for the mesh construction operations described in Chapter 4

TIFF image File

An *TIFF* file is an image that can be imported into *SMS*. *TIFF* files of aerial photographs, and scanned or digitized maps can be imported as a background display for aid in digitization, visualization and model construction. (see section 14.3)

DXF File

DXF is a file format defined by AutoDesk to store Computer Aided Drafting (CAD) data. These files can be imported through the file menu. They include points, lines, and polylines. Data from *DXF* files can serve as a background information to aid in model construction. They can also be converted to feature or scatter points.

GFGEN Geometry File

A *GFGEN* file contains the geometric description of a mesh along with some other mesh renumbering information. When importing this file type, all information other than the geometry is ignored. Therefore, this is recommended only for merging sections of a mesh contained in separate *GFGEN* files.

FESWMS File

A *FESWMS* file contains the geometric description of a mesh in addition to other parameters to control the analysis model. When importing this type of file, all information other than the geometry is ignored. Therefore, import is recommended only for merging sections of mesh contained in separated *FESWMS* files.

2D Mesh File

A 2D Mesh file is a generic file format that can store the nodes and elements of a mesh or mesh section. It is generally better to store a mesh in the model specific file format such as an *GFGEN* “.geo” file. However, if the model to be used is unknown, or a section of mesh has been created by another utility and is to be imported by *SMS*, a 2D Mesh file may be used.

Polygon File

A polygon file is a list of polylines. Each polyline consists of a list of (x,y) or (x,y,z) points. The polylines may close to form a polygon, or may be left open. *SMS* reads these list of points and converts them to feature arcs and polygons in the current coverage (see Section 14.1). Previous versions of *SMS* utilized polygon files directly to assist in the meshing process. This is now done through the *Map Module*.

Observation Point File

An observation point file contains a list of values associated with one or more observation points. These values are used for statistical comparisons to numerical model results (see Section 15.3.2).

Shape File

Shapefiles are the name used to refer to data exported by ARC/INFO®. This file contains GIS objects with a set of associated attributes. *SMS* converts the objects to feature objects in the *Map Module*. It also attempts to map features for the objects to supported features in *SMS*.

Coastline File

Coastline files include lists of two-dimensional polylines. These polylines may be closed or open. *SMS* will only import a coastline file if the current model is a coastal model (ADCIRC, CGWAVE, and STWAVE). The open polylines are converted to feature arcs and are interpreted as open sections of coastline. Closed polylines are converted to arcs and polygons and are assigned the attributes of islands.

Drogue File

Drogue files contain particle-tracing data. An analysis program may create this type of file to track constituent migration or to visualize velocity fields.

Arcview ASCII Grid File

An Arcview ASCII grid file contains elevation data distributed over a regular grid. This is useful for interpolation to a mesh or boundary fitted grid. *SMS* imports this information as scattered data..

Export File

SMS supports an *Export* command to communicate or interface with programs that utilize similar data. *SMS* supports geometric data in four formats for this purpose and image data in one format. The geometric data formats supported via the export command include:

TIN (Triangulated Irregular Network)

This format includes a set of nodes connected to form Delaunay criterion triangles. Such programs as the Watershed Modeling System (WMS), which uses TINs to model terrain, use this format. In order for *SMS* to export a TIN, all elements must be linear triangles and the mesh must be renumbered.

QUAD4

This format includes four sided elements for use with the QUAD4 analysis package.

2DMESH

This is a generic file format defined by the Environmental Modeling Research Laboratory (see Chapter 17). It provides a simple format for proprietary analysis models that wish to interface with SMS without a complete model specific interface. The *Groundwater Modeling System (GMS)* also supports this format. Therefore data may be transferred from SMS to GMS or vice-versa using 2DMESH files.

DXF

DXF is a standard file format specified by AutoDesk's AutoCAD®. Once the data from SMS is saved into DXF format, it may be read and edited by any program that supports this format.

BOUNDARY IDS/XYS

These commands create a file consisting of the boundary of the finite element mesh. One command stores the node identifier for each boundary node in order. The other command outputs the (x,y) value of every point in the boundary.

TIFF

Exporting a TIFF image allows the user to create an image file of the current SMS display screen. This is an alternative to performing a screen capture with a third party image manipulation package.

Get Info

The *Get Info* command opens a dialog that reports basic information concerning the data type associated with the active module. For example, for meshes, the *Get Info* dialog reports the number of nodes, the number of elements, the number of linear elements, etc.

Save Environment

The *Save Environment* command is used to save the current settings of the program (display options, defaults, etc.) to a default settings file. SMS reads the “default settings” file each time it is launched or the new command is invoked.

Print

Printed copies of SMS screen displays can be generated by using the *Print* command. The UNIX version of SMS will create a PostScript file that can be sent to a PostScript printer. The UNIX version will also create encapsulated PostScript files that can be imported into many other programs. The MS-Windows version will print to any printer supported by Windows.

If the *Plot Window* is up, the user is prompted to select a window to be printed (see Figure 2.7).

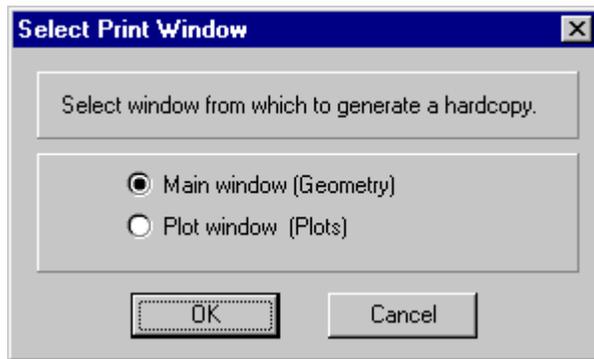


Figure 2.7 Window Select Dialog

When the *Print* command is selected, the *Print* dialog will appear (see Figure 2.8). This dialog allows the user to change a number of printing parameters. The *Printer Setup (PC)/Postscript Setup (UNIX)* button accesses the *Windows Printer Setup* dialog in the MS-Windows version and the *Page Size* dialog in the UNIX version. The *Page Layout* button accesses the *Page Layout* dialog. The data that will be printed to the output device is identical to that which appears in the selected window when a *Refresh* command is issued.

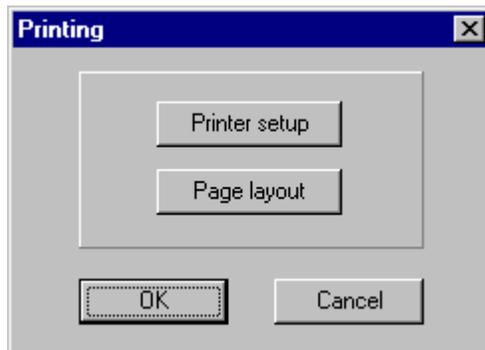


Figure 2.8 The Print Dialog.

Printer/Postscript Setup

The *Printer/Postscript Setup* command allows the user to control the orientation of the printed image and the paper size in UNIX (see Figure 2.9). The MS-Windows version opens the printer setup dialog, which allows the user to change relevant parameters of the currently selected printer. The current printer can also be changed using Windows Print Manager.

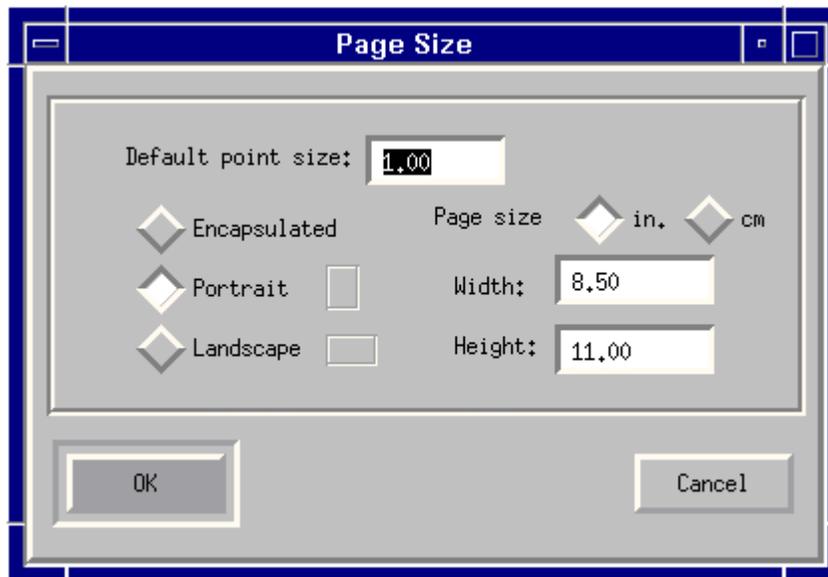


Figure 2.9 The Page Size Dialog.

Page Layout

The *Page Layout* dialog (see Figure 2.10) allows the user to change the size and position of the printed image on the paper. The two scroll bars just under the page display control the image size. When the *Maintain Aspect Ratio* box is checked, moving one of the scroll bars will also move the other scroll bar. The image size is displayed to the right of each scroll bar. The *Center* button allows the user to center the image on the page. The *Max Aspect* button sets the image to a size that will just fill the paper, maintaining the aspect ratio, with a 0.25 inch margin on either the left/right or top/bottom borders (depending on the paper direction and paper orientation)

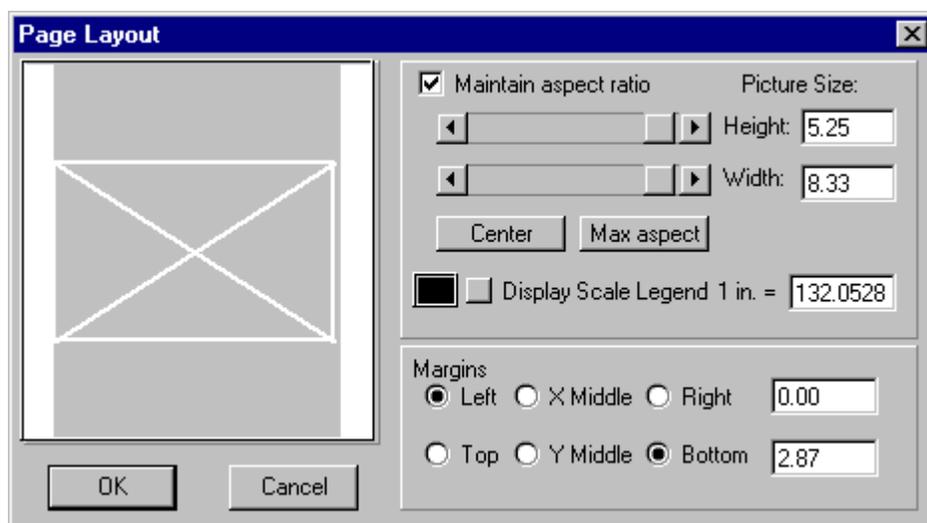


Figure 2.10 The Page Layout Dialog.

The top portion of this dialog displays how the image in the graphics window will be positioned on the printed page. The grayed region represents the paper, and its size and orientation will reflect the size and orientation specified in the *Printer/Postscript Setup* dialog. The white box with an “X” through it represents the size of the printed output.

The image may be interactively positioned on the paper by dragging the box within the paper display. Alternately, the position can be specified using the margin control items at the bottom of the dialog. The size of the image can be edited using the scroll bars and edit fields in the center of the dialog or by using the image scale legend. The size of the image is displayed in the same units as that used for the paper size and is displayed in the text edit fields on the right side.

Demo Mode

Since some users may not require all of the modules or model interfaces provided in *SMS*, modules and model interfaces can be licensed individually. The icons for the unlicensed modules or the menus for unlicensed model interfaces are dim and cannot be accessed. The *Demo Mode* command provides a way of evaluating additional modules you may wish to consider licensing in the future. This is particularly useful when using the tutorials provided with *SMS*.

When the *Demo Mode* command is selected, all modules of the program will be enabled. The only exceptions are that the *Print*, *Save*, and *Export* options will be disabled. It is important to note that when the mode is changed all current data will be deleted. When the program is in demo mode, a check mark appears next to the menu item. To return to normal operating mode, select the *Demo Mode* command again. If an evaluation copy of the software is being used, or if all modules are enabled this menu item is unavailable.

Register

The *Register* command runs the registration wizard. This reports what modules are currently registered and help you register any modules you have purchased.

The *Details* button allows the user to clearly read the registration string needed to obtain a password. Clicking on this button will open a dialog that gives a name to each letter and symbol. Using these names facilitates the issuing of a password.

Recent Files

SMS remembers the last five files opened during operation. These files are added to the File Menu and stored in the Settings file. A file can be reopened, either at the beginning of a new session or to restart a session by choosing it from the list.

Quit

The *Quit* command is used to exit the program. If your data has not been saved, *SMS* warns you before it exits.

2.5.2 Edit Menu

The *Edit* menu is one of the standard menus and is available in all of the modules. The commands in the *Edit* menu are used to select objects, delete objects, and set basic object and material attributes.



Delete

The *Delete* command is used to delete the selected objects. This command is equivalent to hitting the *DELETE* or *BACKSPACE* key on the keyboard.

Select Commands

Select All

The *Select All* command selects all items associated with the current selection tool.

Select With Poly

The *Select With Poly* command selects items associated with the current selection tool which are inside a user defined polygon. Create the polygon after selecting the command by clicking in the *Graphics Window*. The polygon is closed with a double click.

Select By Material Type

The *Select By Material Type* command selects all items of a specified material. This command opens a dialog with a list of the defined materials and waits for the user to select a material type. This enables all nodes or elements that reference a specific material to be selected together. The application of materials is described below.

Confirm Deletions

By default, whenever a set of selected objects is about to be deleted, the user is prompted to confirm the deletion. This helps ensure that objects are not deleted accidentally. Selecting the *Confirm Deletions* command in the *Edit* menu toggles this request for confirmation. When the option is off, the check mark next to the *Confirm Deletions* line in the menu disappears.

UTM <-> Lat/Lon

This menu item is reserved for future coordinate conversion capabilities.

Materials Data

Many of the data types supported by *SMS* (i.e., elements, cells) have a material ID associated with each object. This material ID is an index into a list of material types. These material types often represent different types of bed material or areas of fluid

properties. A global list of material attributes is maintained that can be edited using the *Materials Data* command in the *Edit* menu. The command brings up the *Materials Editor* (see Figure 2.11) dialog where each material is assigned an ID number. This dialog can be used to delete unused materials, create new materials, and assign a descriptive name or color to a material. This general information is saved in the material file which is described in Sections 2.5.1 and 17.15.

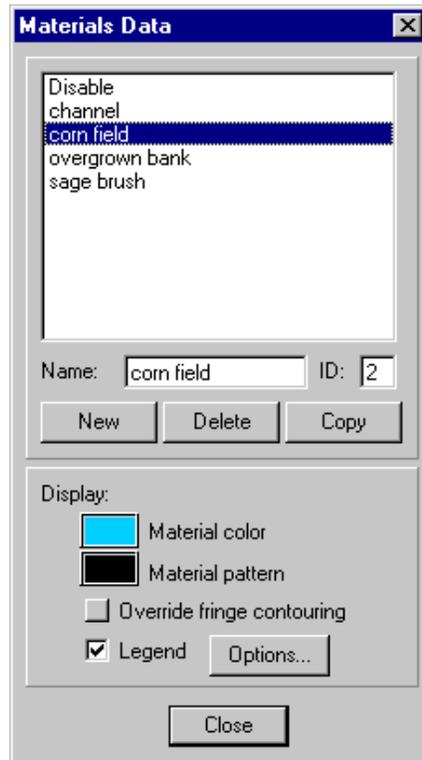


Figure 2.11 The Material Editor Dialog.

The user can specify if a material pattern should be displayed instead of contouring the element or cell by choosing the *Override fringe contouring* option. The *Legend* toggle option controls the display of a legend of the materials in the *Graphics Window*. The options for the legend are edited in the *Material Legend Options* dialog (see Figure 2.12). These options include:

- The name to be displayed on the legend.
- The font to be used in the legend.
- The specification of which corner of the screen the legend will appear in.
- The size of each entry in the legend.
- Whether all materials will be included in the legend, or only active materials.

When a new mesh element or grid cell is created, a user specified default material is assigned to the new object.

Model specific material properties such as Manning's n and Eddy viscosity are edited using commands local to the model menu.

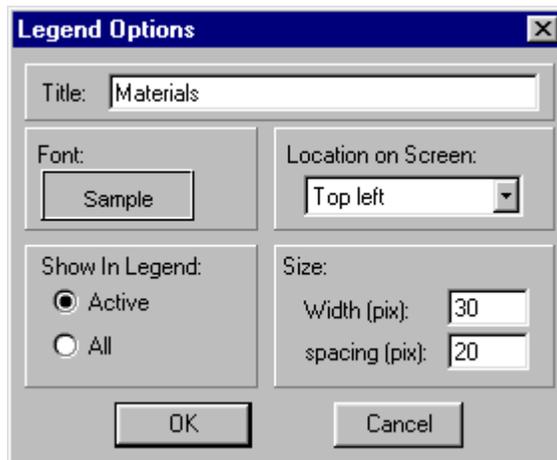


Figure 2.12 The Material Legend Options Dialog.

2.5.3 Display Menu

The *Display* menu is the third standard menu available in all modules. The commands in the *Display* menu are used to control what entities are displayed, and the attributes of those entities. It also allows the user to control the display of a guideline grid in the background, the background color, and the range of values mapped to the display.



Refresh

When editing the image in the *Graphics Window* it occasionally becomes necessary to refresh the screen by redrawing the image. By default, *SMS* automatically updates the display when it is required (see Automatic Refresh below). To force the display to update, select the *Refresh* command from the *Display* menu or click the *Refresh* macro. The process of redrawing can be aborted by pressing the *ESC* key.

Automatic Refresh

Depending on the capabilities of the computer and the amount of data being displayed, significant time can be taken for the *Graphics Window* to refresh. The *Automatic Refresh* toggle provides the option to only refresh the *Graphics Window* when the user forces the *Refresh* command.

When a check mark is shown next to the *Automatic Refresh* command, *SMS* refreshes the screen when it is needed. Selecting the command will toggle the refresh mode to manual. No further automatic updates to the *Graphics Window* are made until the *Refresh* command in the display menu (or the *Refresh* macro) is selected. If the image displayed in the *Graphics Window* is not up to date, the *Refresh* macro is highlighted in red. After the user issues the *Refresh* command, the macro is unhighlighted. To turn the *Automatic Refresh* command back on, select the command again.



Display Options

Each data type in *SMS* has a set of display options that can be modified using the *Display Options* command in the *Display* menu. The *Display Options* command opens a different dialog for each module. This dialog controls which entities of the data type are to be displayed. Each display entity associated with a data type is listed in the *Display Options* dialog. The check box next to the entity named can be toggled on or off to control its visibility. The attribute button to the left of the check box can be used to set the color of the feature. The specific display options for each module are described in the appropriate chapter.



Frame Image

Selecting the *Frame Image* command in the *Display* menu centers displayed data. This command adjusts the window boundaries so that all visible objects fit in the *Graphics Window*.

Set Window Boundaries

The numerical model resides in a virtual world. The extents of that world displayed in the *Graphics Window* are the window boundaries. These boundaries can be altered using the *Pan* and *Zoom* tools. Alternatively, it is possible to precisely control the visible region by using the *Set Window Boundaries* command from the *Display* menu. The *Set Window Boundaries* dialog box appears, and the x and y limits of the viewing area can be set.

Grid Options

When entering new nodes or entering a polygon, it may be useful to have the coordinates snap to a uniform grid. This allows accurate placement of the objects when the desired coordinates are even multiples of some number.

A drawing grid can be activated using the *Grid Opts* command in the *Display* menu. If the *Snap to grid* option is selected, all new vertices will snap to the closest grid point. The grid spacing and options for displaying the grid can also be set using the *Drawing Grid Options* dialog.

Background Color

The user may select any color supported by their machine as the background color. By selecting the background color item, a standard dialog for selecting a color from a palette appears, from which the user specifies the color to be used for the *Graphics Window* background.

View Data File

Since the process of numerical modeling often utilizes many input files and generates many output files, it is not uncommon to review an ASCII data file. When the *View Data File* command is selected, *SMS* asks the user to select a file and then asks which editor is to be used. A separate process is created for editing/viewing the selected file using the selected editor. It should be remembered that this is now a separate process and the data in the file is not part of the *SMS* database. The data may be saved and incorporated into *SMS* using file read and import capabilities.

Plot Window

SMS uses the *Plot Window* to display two-dimensional plots such as time variant data at a single point, cross sections, profiles, and statistical comparisons (see section 3.8). Since this type of plot may not be desired, the display of the *Plot Window* is user controlled. Selecting the *Plot Window* command opens the plot window. A check mark is placed next to this menu item when the *Plot Window* is open. Selecting the command again closes the window.

River Window

The *River Window* displays the structure of a one-dimensional hydrodynamics model. For two-dimensional modeling, this window will not be used. Therefore, the user specifies when the *River Window* will be displayed. Selecting the *River Window* command opens the *River Window*. A check mark is placed next to the menu item when the window is open. Selecting this command again closes the *River Window*.

2.6 Plot Window

The *Plot Window* is used to plot two-dimensional curves such as time variant data at a single point, cross sections, profiles, and statistical comparisons. These curves may be associated with the one-dimensional river model, or two-dimensional grids and networks. *SMS* includes tools to plot curves to represent variation of data along a curve or over time. Plots are created and edited using the *Plot Manager* (see Section 3.8).

The *Plot Window* is displayed at startup only if the *WSPRO* model in the river module is selected as default (see sec. 1.2). The display of the *Plot Window* is toggled using the *Plot Window* command in the *Display* menu.

2.6.1 Plot Options

The user controls the number of plots to be generated, the type of data to be viewed in each plot, and the other attributes of the plot. Each plot may have multiple curves. The attributes of the plot include the color, legend display options, grid sizing and grid

display. Details of how plots are generated and controlled are included with the visualization tools in Section 3.8.

2.7 River Window

The *River Window* is associated with one-dimensional models. *SMS 6.0* supports only one one-dimensional model, *WSPRO*. However, final testing is being done for an interface to *DAMBRK* and several other one-dimensional model interfaces are under development.

One-dimensional models have been the main workhorses for practical hydrodynamic engineering. For that reason, many models and a lot of data are available in this form. *SMS 6.0* includes a first generation approach at integrating the one-dimensional modeling work with the two-dimensional approach. Tools exist to create one-dimensional models from topographic and bathymetric data sets. Tools are under development to convert existing one-dimensional model data and the solutions generated for this data to two-dimensional domains.

Data Visualization

SMS was designed as a general purpose modeling system. One of the main purposes of *SMS* is to provide a consistent interface for a variety of models and grid types. In order to accomplish this goal, input data and solution data are handled in a simple, consistent fashion using data sets.

A data set is a set of values associated with each node, cell, vertex, or scatter point in an object. A data set can be steady-state (one value per item) or dynamic (one value per item per time step). The values in the data set can be scalar values or vector values. In *SMS*, a two-dimensional finite element mesh has an associated list of scalar data sets and a list of vector data sets. Sets of scattered data points have an associated list of scalar data sets. The commands for manipulating data sets are located in the *Data* menu. The *Data* menu is one of the standard menus and is available in each of the modules (see Section 4.4).

Data sets are used for both pre- and post-processing of models. For example, a scalar data set associated with a two dimensional mesh can represent starting or initial values of water surface elevation. Another data set associated with the same mesh may represent initial velocity values. *SMS* can export these values as an initial condition file for *FESWMS* and *HIVEL2D*. All data sets can be used to generate contours, color fringes, vector plots, animation sequences and other data sets. A detailed discussion of the visualization tools is presented in this chapter.

One advantage of the data set list approach for managing information is that it facilitates transfer of information between different types of models or models with differing resolution. This is accomplished through sets of scatter data points and interpolation. Meshes and grids can be converted to 2D scatter point sets. When an object is converted to a scatter point set, all data sets associated with the object are

copied to the new scatter point set. The data sets can then be interpolated from the scatter point set to other objects of any type using one of the supported interpolation schemes.

3.1 Browser



Most of the interaction with data sets is accomplished with the *Data Set Browser* (see Figure 3.1). Selecting the Data Browser command in the Data menu activates the Data Set Browser. The two parallel list boxes in the browser contain the lists of scalar and vector data sets for the current object. The active module determines the current object. If the *Scatter Point Module* is current, the data sets shown in the browser correspond to the active scatter point set. In this case, the vector section of the *Data Set Browser* is disabled or dimmed. If the *Mesh Module* is current, the data sets correspond to the loaded mesh. (Note: only one mesh may be in memory at any given time.)

The *Data Set Browser* displays the available data sets in the windows on the left. The selected data set(s) are highlighted. Timesteps for the selected data sets are displayed on the right.

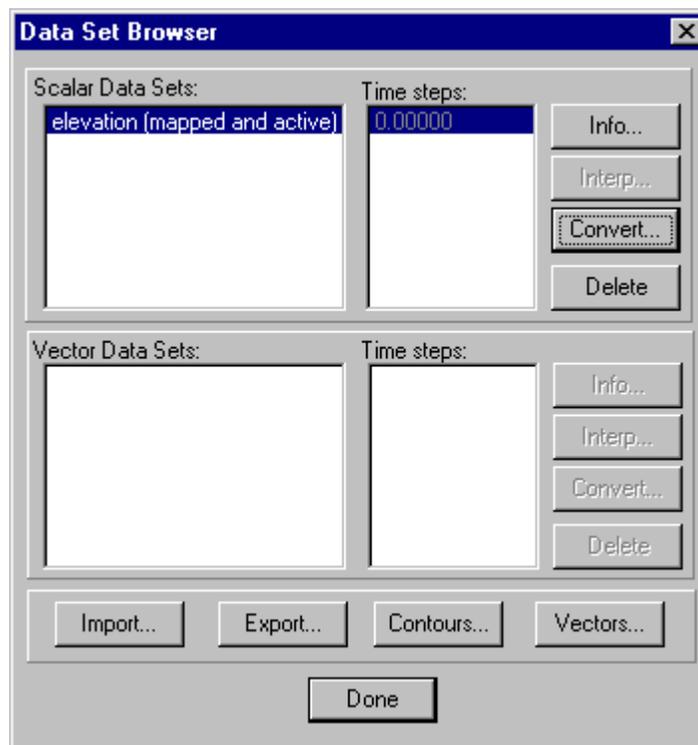


Figure 3.1 Data Browser Dialog.

3.1.1 File I/O

Previously defined or computed data sets can be read into *SMS* by selecting the *Import* button in the *Data Browser*. This will bring up the import type dialog box (see Figure 3.2). The user then selects the file type desired, from the supported types, which include:

- Generic *SMS* (ASCII or Binary) Data Set Files
- *TABS* (*RMA2/RMA4/SED2D-WES*) Solution Files
- *FESWMS* Solution Files
- *ADCIRC* Solution Files

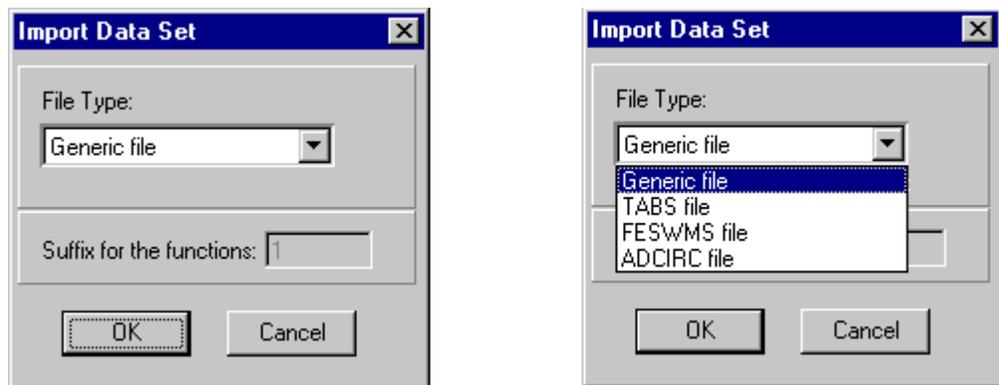


Figure 3.2 *Import Data Set Dialog.*

The most common source of these data files is the numerical models. Most models support their own file formats that must be imported by *SMS* to visualize the results of analysis. These include *RMA2* and *ADCIRC*. Some models, such as *HIVEL* have been modified to output generic *SMS* data set files instead of a model specific format. Other models, such as *FESWMS*, can create output in either its own format, or the *SMS* generic format, or both. Model specific files are accessible only in the module that contains that model interface. Generic *SMS* files contain a record indicating what data they apply too. Therefore, they can be imported in any module. The format for the *SMS* data set files is described in Chapter 17.

Generic files also include a specification of names for each functional data set in the file. For model specific file types, a predetermined set of functions are created from the data in the file. For example, when *SMS* reads a solution file for *RMA2* it creates a function for water surface elevation, depth and velocity magnitude. If two solution files were read, two of each function would be created. To distinguish one solution from another, the user specifies a suffix that will be added to the default function names.

Once one of the file type options has been chosen, a file browser dialog appears and the user must select a file corresponding to the type selected.

Data sets can be exported from *SMS* to files by selecting the *Export* button in the *Data Browser*. Data sets can be saved as either binary or ASCII data set files in the *Data Browser* or from the save command (see section 2.5.1).

When a data set is imported to *SMS*, a copy of the data set may be written to a temporary file on disk in binary form to make access faster. Do not delete these files from your hard drive while inside of *SMS*. When a data set is needed it is loaded from the hard disk into internal memory and becomes the *active* data set. Only one time step of one scalar data set and one vector data set is active at any given time. The active data is marked in the list of functions in the *Data Browser*. This method of file manipulation reduces the amount of RAM required but it requires extra hard disk space. It also requires that a temporary location be defined for the system on which *SMS* is running.

When a new data set is created through interpolation or using the data calculator, a temporary binary file is created for the data set. To save the data set to disk permanently, the user must select the *Export* button from the *Data Browser*, or save the data from the *File* menu.

3.1.2 Active Data Set

One data set is always highlighted in both the scalar and vector data set lists. In addition, if a dynamic data set is highlighted, the time steps for the data set are listed in the text box directly to the right of the list of data sets. One of the time steps is highlighted. The highlighted sets are the active data sets for the object. The values corresponding to the active data sets and time steps are used whenever contour, color fringe, and vector plots are generated. In addition, the entire range of time steps of the active data set is used whenever animation film loops are generated.

Whenever a new data set is created, whether by importing from a file, interpolating, or using the data calculator, the data set becomes the active data set for the object.

3.1.3 Elevations & Automatic Data Sets

Whenever a new mesh or grid object is created or read from a file, a scalar data set is created containing the elevations of the nodes of the object. Thus, there is always at least one data set associated with a mesh or grid. It is labeled as *mapped* in the *Data Browser*. Any function may be assigned to be this elevation function using the *Map* command in the *Data* menu. *SMS* treats the currently mapped function differently than other functions in three ways. First, the mapped function value is stored as the “z” component of nodes and is displayed as the “z” value in the edit window. Therefore, the user may edit these values. Second, the mapped function cannot be deleted. Third, for models that support disabled elements, such as *FESWMS* or *RMA2*, *SMS* ignores

all functions except the mapped function when fringes or contours are computed and displayed.

There are several other situations in which SMS will automatically create data sets. These include *FESWMS* ceiling values (see section 10.11), *SED2D-WES* bed property sets (see section 6.2.1) and *RMA10* layer data sets.

3.1.4 Deleting Data Sets

Data sets can be deleted by selecting the data set in the list box and selecting the Delete button in the *Data Browser*. This removes the data set from the list and deletes the binary copy of the data set on disk. If the original data set file was already in binary form, the file is not deleted.

All data sets associated with an object are automatically deleted, except the mapped function (see section 3.1.3), whenever the object is deleted or whenever the number of nodes or scatter points in the object is changed due to an editing command.

3.1.5 Data Set Info

The *Info* buttons in the *Data Browser* will bring up a dialog listing some of the main characteristics of the active vector or scalar data set. These characteristics include statistics such as maximum, minimum, and range as well as mean and standard deviation. The name of the active data set can be edited from the *Info* dialog.

3.2 Data Calculator

SMS performs mathematical operations with data sets to create new data sets using the *Data Calculator* (see Figure 3.3). The user accesses the *Data Calculator* by selecting the *Data Calculator* command from the *Data* menu.

The *Data Calculator* can be used to perform any set of mathematical operations shown as icons in the center of the dialog. Some of the operators are binary (i.e., "+", "-") and some are unary (i.e., "1/x", ln(x)). The user simply composes a mathematical equation using operators and data sets. The operator then specifies a name for the new data set to be created as a result of the operation. Once the mathematical operation is defined, the user clicks on the COMPUTE button to execute the operation. The new function then appears in the *Data Set* list.

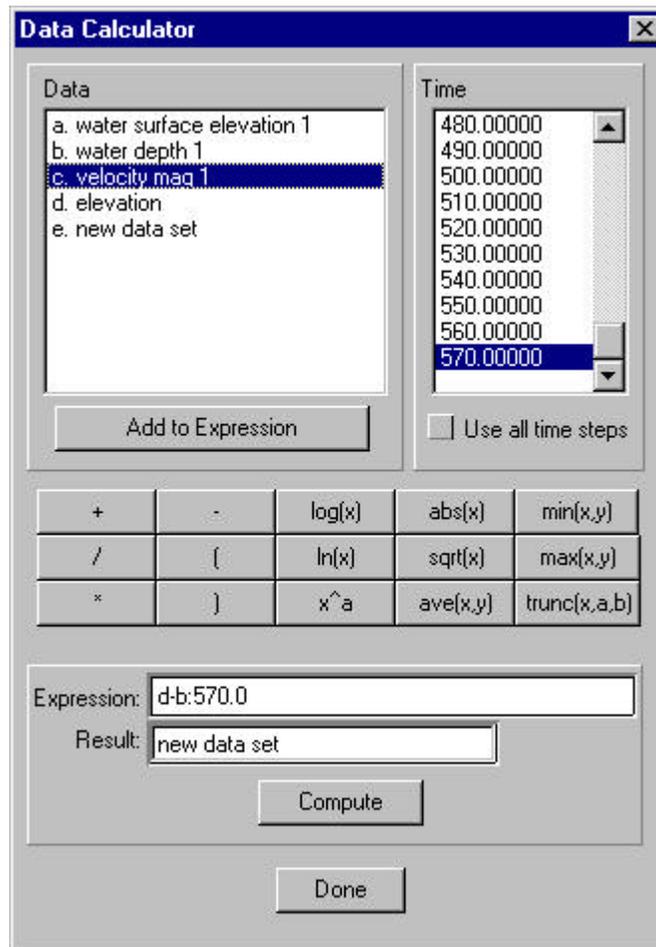


Figure 3.3 The Data Calculator.

The *Data Calculator* is useful for a variety of tasks. For example, to generate a data set representing the absolute difference between two other data sets, the user enters the equation $|a-b|$ where 'a' and 'b' correspond to the two data sets. Such a data set would be useful for comparing the results of two separate solutions computed by a numerical model.

3.3 Vector Options

If the *Vectors* item in the *Display Options* dialog is toggled for display, vector plots can be generated using the active vector data set for the object. Vectors are placed at specified nodes or cells, or displayed across the mesh or grid with uniform spacing based on user defined options. The display of vectors can be controlled using the *Vector Options* dialog (see Figure 3.4) accessed through the *Vector Opts* command in the *Data* menu or from the *Data Browser* or *Display Options* dialogs.

The upper left portion of the *Vector Options* dialog allows the user to specify the shape of the vector arrows by specifying proportional head length, head width, shaft length and shaft width.

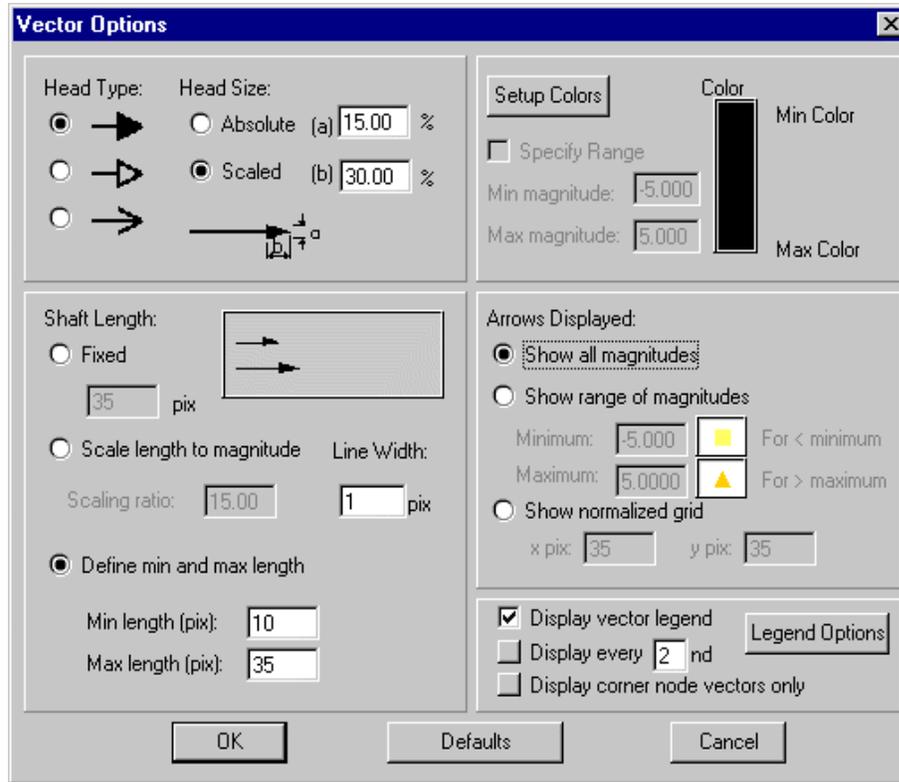


Figure 3.4 *Vector Options Dialog.*

In the lower left, the user specifies how big the arrows will appear in the graphics window. This can be a constant length, a scaled length, or a range of lengths. The min and max arrow sizes are displayed in the canvas just above this section of the dialog.

The upper right portion of the dialog is used to specify the arrow color. Arrows may be a constant color, or shaded according to magnitude. If a ramp of colors is desired, the color of the vector is extracted from a ramp. By default, the arrow with the smallest magnitude is displayed in the color at the bottom end of the ramp, and the arrow with the largest magnitude is displayed in the color at the top of the ramp. Intermediate magnitudes are interpolated to select an appropriate intermediate color. Alternately, the user can define the magnitudes that map to the top and bottom of the ramp. If this option is used, any arrow with a magnitude lower than the minimum is displayed in the color at the bottom of the ramp, and any arrow with a magnitude greater than the maximum is displayed with the color at the top of the ramp.

The lower right portion of the dialog allows the user to control the density of arrow to be displayed. In a very dense mesh, a large number of data points may be displayed very close together on the screen. Therefore, if a vector is displayed at every point, the picture can become a jumble of vectors on top of each other. One way to treat this is to

zoom in on a specific portion of the mesh, so the nodes are not displayed so close together. However, if the desired region of the mesh is still too dense, or zooming in is not acceptable, the user can filter the displayed vectors. SMS provides four methods of filtering vectors. These include:

1. Vectors above and below extremes may be turned off.
2. Vectors at midside nodes may be turned off.
3. A random sampling of vectors may be used.
4. Vector arrows may be displayed on a uniform grid overlaying the mesh or grid. The user specifies the density of this overlaying uniform grid in screen space.

The default values button at the bottom fills in a set of values based on the values in the initial conditions file.

The lower right portion of the vector options also includes a toggle for a vector legend. The vector legend displays the significance of the size of the vectors displayed on the grid. Options are shown in Figure 3.5.

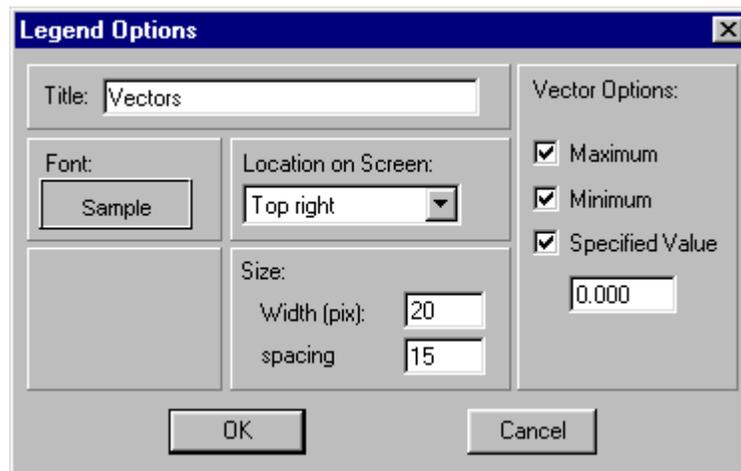


Figure 3.5 Vector Legend Options Dialog.

3.4 Contour Options

The scalar values associated with the active data set represent a sampling of values from a continuous function. Contour lines of constant value created by interpolating the sample values are very useful when visualizing this function. SMS uses linear interpolation across the elements or grid cells to generate these contours. Contour display is enabled using in the *Display Options* dialog of each applicable module.

The options used to generate contours can be edited by selecting the *Contour Options* command in the *Data* menu or by selecting the *Contour Options* button in either the *Data Browser* or *Display Options* dialogs. The *Contour Options* dialog is shown in Figure 3.6. The values shown in the upper left corner of the dialog correspond to the maximum and minimum values in the active data sets active timestep. These values are sometimes useful when choosing an appropriate contour interval.

The number of contours is user controlled. Options include the specification of a number of contour intervals, a specified contour interval, or a set of explicit contour values. For either a specified interval or specified number of contours options, a maximum and a minimum contour value can be specified and the contouring can be restricted to this specified range. If no explicit range is specified, *SMS* will choose a range to optimize the color distribution inside the extreme values for the data set. If the *Values* button is selected, the *Contour Values* dialog is displayed. Up to ten specific contour values can be typed into the dialog.

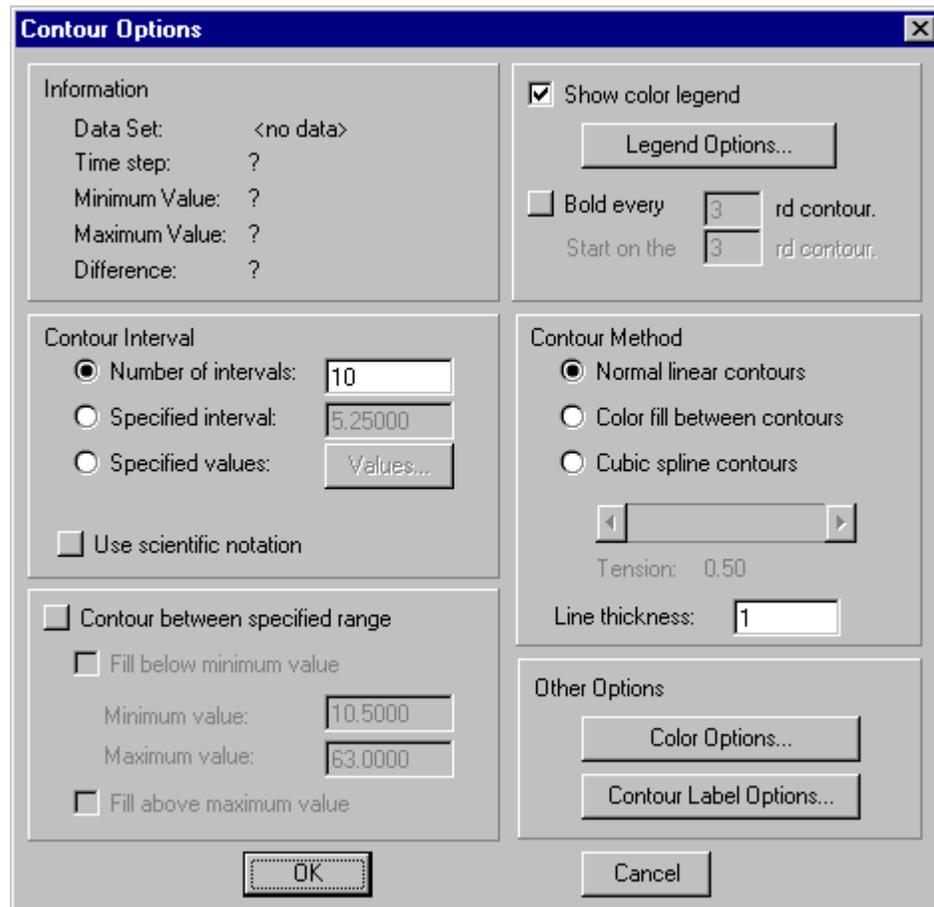


Figure 3.6 Contour Options Dialog.

The items in the upper right section of the *Contour Options* dialog control the display of a contour legend and the option to accentuate some of the contours. If the *Show color legend* option is selected, and the contours are not being displayed as a single

color, a legend of colors and corresponding data set values is displayed in a corner of *Graphics Window*. For color filled contours, this legend is a vertical strip of colors with text labels for the contour levels. If the contours are being displayed as linear segments or cubic splines, the legend is displayed as a series of contour level values and a line drawn in the color corresponding to that level. The size, location, label and font for the legend are set using the *Legend Options* dialog (Figure 3.7). If the user enters the title “DS” for the legend title, the name of the current data set is used. If the user enters “DS:TS” the current dataset and timestep are used as the title.

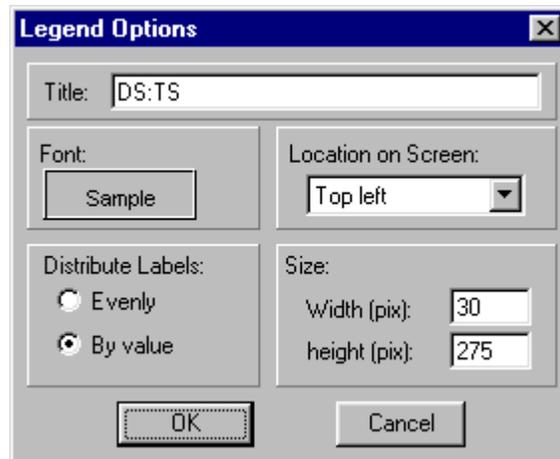


Figure 3.7 Contour Legend Options Dialog.

The options in the middle of the right side of the dialog control how the contours are computed and displayed. Three contouring methods are available:

- The default method is *Normal Linear Contours* and causes the contours to be displayed as piece-wise linear strings.
- If using the *Color fill between contours* method, the same linear contour strings are computed, but the regions between adjacent contour lines is filled with a solid color.
- If using the *Cubic Spline Contours* method, the contours are computed in strings and drawn as cubic splines. Drawing the contours as splines can cause the contours to appear smoother. Occasionally, loops appear in the splines or the splines cross neighboring contour splines. These problems can sometimes be fixed by adding tension to the splines. A tension factor greater than zero causes the cubic spline to be blended with or converge to a linear spline based on the same set of points. A tension factor of unity causes the cubic spline to coincide with the linear spline.

In the lower right corner of the *Contour Options* dialog, two buttons allow the user to specify the contour colors and the contour labeling options.

3.5 Color Ramp Options

The *Color Ramp Options* dialog (see Figure 3.8) lets the user determine how the contours and vectors will be colored. The user may invoke the *Color Options* dialog either from the *Data* menu, or from a button in the *Data Browser*, *Vector Options* or *Contour Options* dialogs.

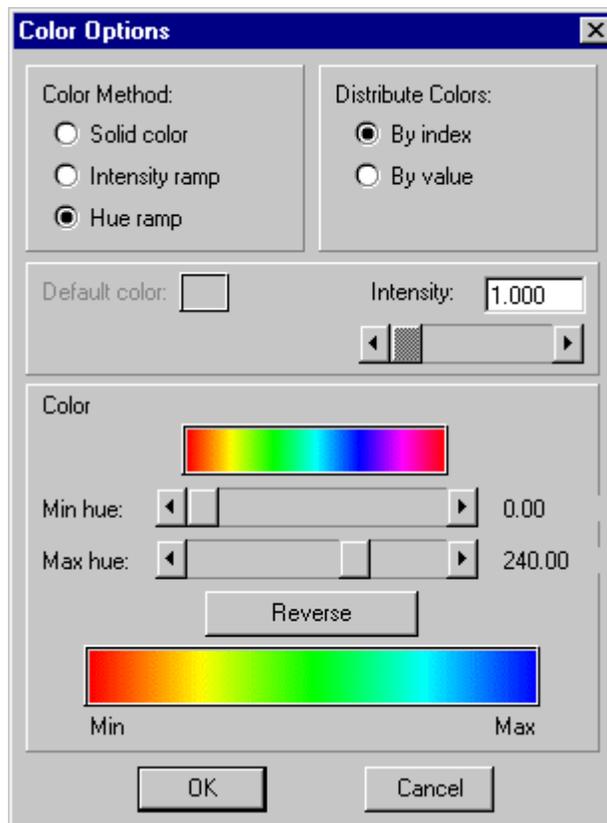


Figure 3.8 Color Ramp Options Dialog.

The default color method is the *Solid color* option. This method allows the user to select a color from the standard *Windows Color Chooser* and all contours are displayed with that color. As an alternative, the user can define a ramp of colors. These colors are distributed across the range of contour values in a continuous fashion, giving each contour its own color. If specific contour values are specified, the user controls whether the colors are distributed by index or by value in the upper right portion of the dialog. There are two types of color ramps supported by *SMS*.

- If using the *Intensity ramp* option, the color ramp will be defined as a continuous variation of the intensity of the default solid color. This is the same color used for the *Solid color* option.
- If the *Hue ramp* option is chosen, the ramp will be defined as a continuous variation of hues using the hue-saturation-value color model.

In either case, the ramp can be edited to include only a portion of the entire ramp. The user does this by editing the *Min hue* and *Max hue* scroll bars in the *Color Options* dialog. These controls specify where the minimum value will be mapped into the ramp and where the maximum value will be mapped. The reverse button changes the direction of the color gradation in the color ramp

3.6 Contour Labels



The *Contour Label Opts* command in the *Data* menu is used to access the *Contour Labels Options* dialog which can be used to set the label color, font, spacing, size, etc. The dialog may also be invoked through the *Contour Options* dialog.

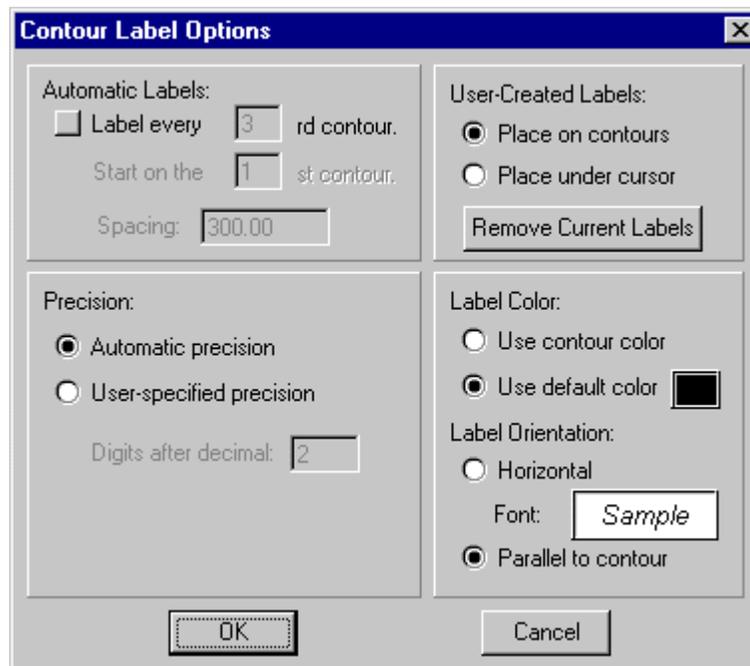


Figure 3.9 Contour Label Options Dialog.

Labels can be added to contours one of two ways:

- The upper left portion of the *Contour Label Options* dialog controls the generation of automatically spaced contour labels. The user can toggle the generation of automatic contour labels on or off. If the toggle is on, the user also specifies which contours should be labeled and the distance along the contour between labels.



- In some modules, contour or function labels can be added manually to an image by selecting the *Contour Labels* tool in the *Tool Palette* and clicking on the mesh or grid where a label is desired. If the *Place on contours* option in

the upper right portion of the *Contour Label Options* dialog is selected, the label is moved to the closest contour and the contour is labeled there. If the *Place under cursor* option is selected, the label shows the value of the point at the click location and is placed there. This option is useful to post data set value labels in regions where there are no contours. If the mouse button is held down, a box showing the outline of the label is drawn. The box can then be positioned precisely with the mouse. A line is drawn from the box to the point that was clicked to help the user keep track of the contour that was selected. Contour labels can be deleted by holding down the *SHIFT* key while clicking on a label.

The bottom portion of the *Contour Label Options* dialog control how the labels appear. On the left side, the user can control how many digits of accuracy are desired. The default will match the contour legend. On the right side, the user can select a color and font for the label. For labels on contours, the user may also specify that the contour be oriented to lie along the contour.

3.7 Film Loops

One of the most powerful visualization tools in *SMS* is animation. An animation sequence can be generated for an object with a dynamic data set to illustrate how vectors, contours and/or fringes vary as a function of time. Each frame of the animation may be stored as a pixel map. The entire set of frames in an animation sequence is referred to as a film loop. Animation sequences can also be generated to visualize a steady state vector field. This process is referred to as *flow tracing* and is discussed below.

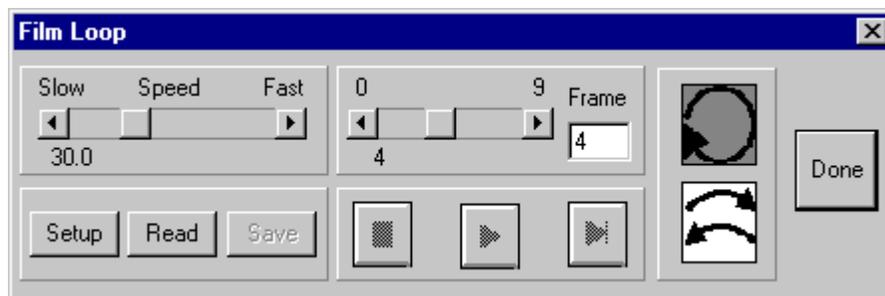


Figure 3.10 The Film Loop Dialog.

An animation film loop is initiated by selecting the *Film Loop* command in the *Data* menu. This command brings up the *Film Loop* dialog (shown in Figure 3.10) which controls the generation and play back of a film loop. Clicking the *Setup* button brings up the *Film Loop Options* dialog to generate a new film loop (section 3.7.1). Once a film loop has been generated, it can be saved to a file using the *Save* button. Previously saved film loops can be read from disk using the *Read* button. On the Windows platforms, film loops are saved as *MS Video for Windows* files (also known as “AVI” files) which can be played with public domain players or embedded in

presentations. On UNIX systems, the film loops are saved in a proprietary format referred to as an animation loop or “alp”. The *EMRL* provides utilities for converting “ALP” files to “AVT” format.

3.7.1 Film Loop Setup

SMS generates a new film loop via the *Film Loop Options* dialog (see Figure 3.11).

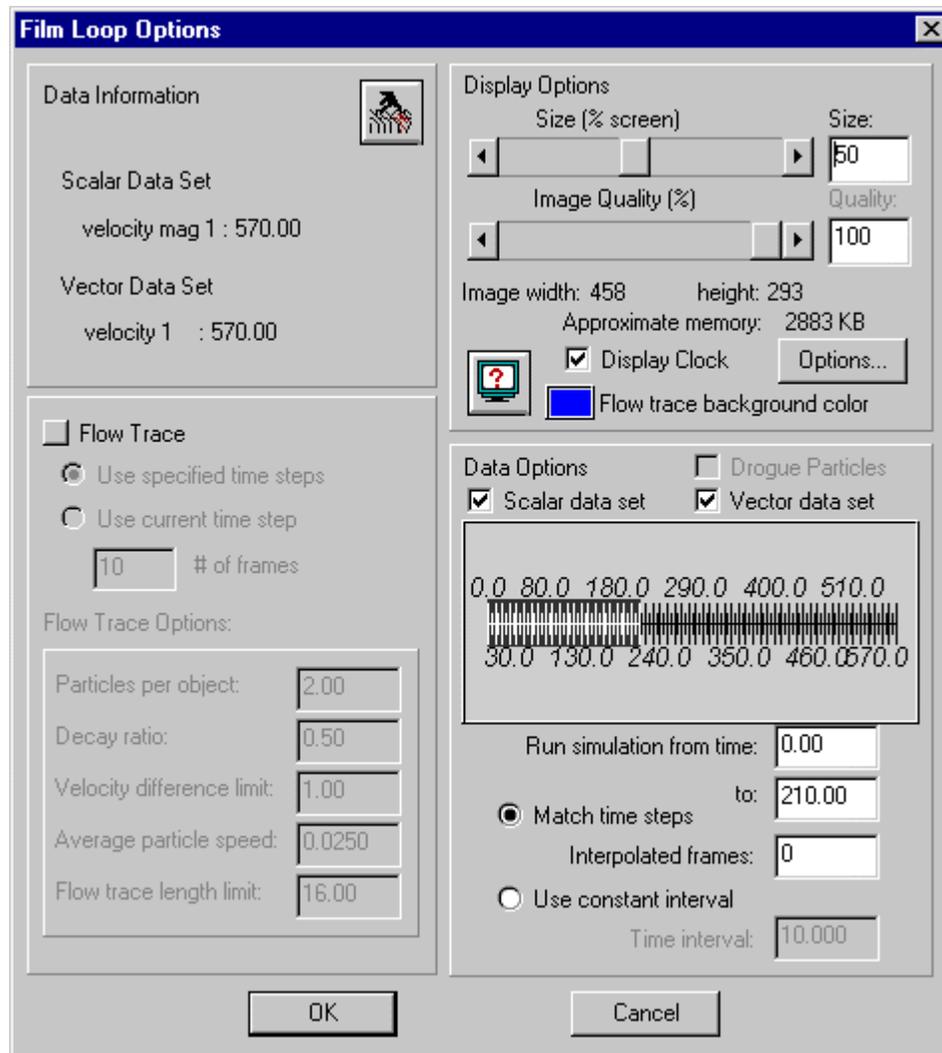


Figure 3.11 The *Film Loop Options* Dialog.

Data Set

Film loops are always generated using the active data set. The *Data Browser* button in the *Data Information* portion of the dialog can be used to change the active scalar and vector data sets. The current active data sets are displayed to the left of the *Data Browser* button.

Image Size

Animations or film loops can be generated at various resolutions. *SMS* allows the specification of resolution as a percentage of the *Graphics Window*. This value is controlled in the *Display Options* portion of the dialog. Naturally, the most pleasing animations are comprised of full screen images. However, this may result in film loops which require a significant amount of memory and which are difficult to playback at a high speed. By reducing the resolution or size, the film loop becomes both smaller and faster. Since the size affects the resolution in both the *X* and *Y* directions, the memory required for a film loop is quadratically proportional to the screen size. For example, an image generated at 50% of the *Graphics Window* size requires 25% as much memory as an image generated at full *Graphics Window* size. The memory required to store the frames of an animation is also affected by the image compression. The user controls the level of compression using the *Image Quality* scroll bar. The standard compression with 100% quality is generally quite good. Therefore, unless memory space is very limited, it is recommended that the quality be left at 100%.

Animation Clock

Since animations are simulating the passage of time, it is natural to display a clock, which indicates the time reference for each frame of the animation. The *Display Clock* toggle controls whether a clock will be displayed. The *Options* button brings up the *Legend Options* dialog (see Figure 3.12) with a control to specify a digital clock face or analog.

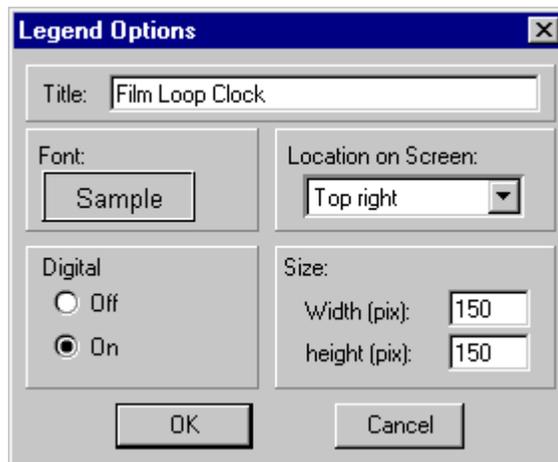


Figure 3.12 Filmloop Clock Options dialog.

Animation Time Control

Animation can be applied to any object with a dynamic data set. The user defines the beginning and ending time for the animation sequence and the time step between subsequent frames. As each frame is generated, data values corresponding to the current time are loaded into memory and the image is redrawn using the current

display options. The display options may be modified while setting up an animation using the display options button in the *Display Options* portion of the dialog.

The strip in the center of the *Data Options* portion of the *Film Loop Setup* dialog displays the allowable time values for the current data function(s) and the selected range to be animated. The user can select a time range to animate graphically on this scale, or explicitly in the edit fields below the time step strip. The legal time range displayed in the strip is based on the current scalar and vector data set(s). *SMS* allows animation of only scalar or vector data while the other remains constant. This normally is only used when a static field such as elevation is displayed with a varying velocity field or a static velocity field is displayed over a changing scalar field such as constituent dispersion or sediment deposition.

The total number of frames generated in the film loop can be defined by either matching the time steps (one frame per time step) or by using a constant interval (e.g., one frame for every two-hour interval). If the *Match Time Steps* option is chosen, extra frames can be created between each time step using linear interpolation of the data values at the specified time steps.

Flow Trace Animation

Flow trace animation is a technique used to visualize vector fields in *SMS*. It can be thought of as dropping tiny drops of dye into a fluid field in a random distribution and watching the flow pattern created. The process can also be thought of as creating particles of zero mass, and letting the vectors in the vector field be forces pushing the particles around. The *Flow Trace* portion of the *Film Loop Setup* dialog allows the user to control the flow trace. This entire portion of the dialog is disabled if no vector data exists for the current data set. The top radio group allows the user to specify whether the flow trace should be created for a steady state or dynamic system. Below this the user can specify the density of particles or dye droplets by specifying the average number of particles for each cell or element. The number of frames required for a droplet to become dispersed is represented as a portion of the animation in the *Decay ratio* field.

The path of each particle is defined by tracing the particle. A starting position is defined randomly in the mesh or grid. Successive particle locations are computed by applying the forces of the vector field to the current location. At the new point, the velocity and direction are sampled. If the particle has traveled farther than the *Flow trace length limit*, or the velocity has changed more than the *Velocity difference limit*, the step is broken into two steps of half the step size. This process is repeated, until a sequence of valid points within the limits are defined for each frame. Therefore, the smaller the values of the *Flow trace length limit* and *Velocity difference limit*, the more precisely the particles will imitate the vector field. Generally, the default values are sufficient.

The *Average particle speed* is used to scale the vector field, thus changing the distance each particle or droplet travels. This is useful for vector fields with extreme magnitudes. For a low magnitude data set, the particles may not move very far. While

this sluggish motion is accurate for the data, scaling the vector field up, and exaggerating the motion causes the flow patterns to be more visible. Similarly, in high magnitude fields the particles may become long streaks and scaling the values down may result in a clearer picture of the flow patterns.

3.7.2 Film Loop Playback

Once a new film loop has been generated or a film loop has been read from disk, several options are available for playing back the film loop. The buttons at the lower middle of the *Film Loop* dialog are designed to mimic the buttons on a VCR or CD player. The *Play* button  causes the film loop to cycle continuously. The *Stop* button  halts the playback. The *Step* button  can be used to advance the film loop forward one frame at a time. In addition, the frame scroll bar can be used to interactively move the frames forward or backward.

The speed of playback can be adjusted using the *Speed* scroll bar. The maximum speed depends on the speed of the computer and the size of the image being animated. For smaller images, the maximum playback speed is faster.

Two options are available for cycling the film loop playback. The *continuous playback* option  starts a new cycle at the first frame in the loop after the last frame is encountered. The *oscillation* option  plays the loop in the forward direction to the end of the loop and then in the reverse direction back to the beginning of the loop.

3.7.3 Saving Film Loops

Saving and reading film loops is useful since some film loops may take a significant amount of time to generate depending on the complexity of the image. The film loops are saved to disk in a compressed binary format. When a film loop is read from disk, it is first uncompressed on the hard disk. Then, if sufficient internal memory is available, the entire film loop is read into RAM for playback. If sufficient internal memory is not available, the playback is performed reading one frame at a time from the hard disk.

With the PC version of *SMS*, film loops are saved in the *MS Video for Windows* (*.AVI) format. AVI files can be embedded into Presentations and PowerPoint presentations and other multi-media documents. They may also be viewed using any stand-alone AVI video for windows player. This offers the ability to present and review film loops outside of the *SMS* environment.

3.8 The Plot Manager

The amount of data generated by multi-dimensional numerical models can be enormous. Therefore, it can be very useful to extracting part of this data to be viewed as two-dimensional plots. *SMS* includes a set of tools to create, view, and save a variety of such plots. These tools are accessed through the *Plot Options* command in the *Display* menu that is available in all modules.

Two-dimensional plots are displayed in the *Plot Window*, which was described in section 2.6. The *Plot Window* can be opened and closed using the *Plot Window* command in the *Display* menu. A check mark will appear next to the command when the window is visible.

3.8.1 Plot Objects

Two-dimensional plots are useful for many purposes. These include extracting data from two or three dimensional objects, model verification (see Chapter 0) and one-dimensional river modeling (see Chapters 11- 12). *SMS* organizes the data to be displayed into *Plot Objects*. For simplicity this document will refer to these objects simply as plots. A plot consists of a group of related curves.

The plots are controlled using the *Plot Options* dialog (Figure 3.13). The options are as follows:

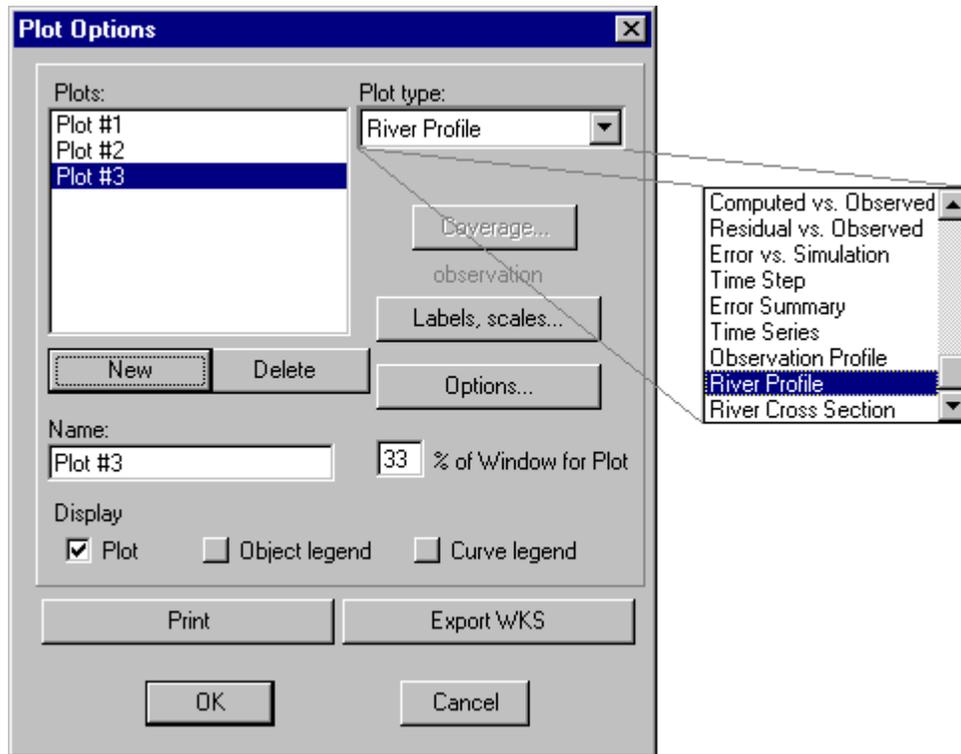


Figure 3.13 The Plot Manager Dialog.

Plot List

The list of currently defined plots is shown in the *Plots* section. A new plot is created by selecting the *New* button. An existing plot is deleted by selecting the plot and selecting the *Delete* button. The name of the plot is only for user reference. It can be edited by selecting the plot and using the *Name* edit field.

By default, all of the plots in the plot list are displayed in the *Plot Window*. However, selected plots can be hidden temporarily without deleting the plot using the *Display Plot* toggle.

Type of Plot

When a new plot is created, the user selects a plot type from the combo box to the right of the list of plots. The available plot types include comparison plots, error summaries, time series at a point, observation profiles, and one-dimensional river profiles and cross. The one-dimensional river plot types are described in detail in Chapters 11 through 12. All other types are described in Chapter 0. The options specific to each plot type can be edited by selecting the *Options* button.

Labels, Scales...

The *Labels, Scales* button in the *Plot Options* dialog shown in Figure 3.13 brings up a dialog which can be used to customize the titles, labels, tick marks, etc. for any of the plot types.

Coverage

Plot types other than the one-dimensional river plots are associated with a single observation coverage (see Chapter 14). More than one such coverage can be defined at once time. The *Coverages* button in the *Plot Options* dialog brings up a dialog that can be used to designate which coverage is associated with a specific plot.

Legends

Two types of legends can be turned on or off in the *Plot Options* dialog: the *Object legend* and the *Curve legend*. These options are used with a *Profile* plot, a *Time Series* plot or one-dimensional river plots. The object legend lists the color of each of the selected points or arcs. The curve legend lists the name, line type, and symbol type of each of the curves in the plot. Both legends aid in identifying specific data curves.

Print

The *Print* button in the *Plot Options* dialog prints a copy of either the selected plot or all current plots.

Export WKS

The *Export WKS* button in the *Plot Options* dialog can be used to save the data used to generate either the selected plot or all current plots in a simple text file that can be opened in a spreadsheet.

Mesh Module

The *Mesh Module* is used for pre- and post- processing of finite element meshes. Various tools and menu commands are provided for manipulating data. A limited number of finite element analysis models are directly supported by *SMS*. Other finite element analysis models can use *SMS* for pre- and post- processing as long as generic *SMS* file formats are used. This chapter describes pre-processor finite element mesh generation. Later chapters describe pre-processor boundary condition definition for specific models. The previous chapter discussed the interpretation of results and use of *SMS* as a post-processor.

4.1 Mesh Generation

A finite element mesh is defined as a network of triangular and quadrilateral elements constructed from nodes. The creation of a finite element mesh requires the user to provide bathymetric information and to define the study area extremities.

Digitized or survey points can be imported to provide the bathymetry. This type of data is generally not appropriate for use as mesh nodes due to random location and distribution. In this case the data should be converted to scatter points (see section 4.4.1). If the bathymetric points are to be directly used as nodes, the triangulate command (see section 4.7.2) can generate elements from the points.

The *Map Module* (see chapter 14) provides tools for defining the study area boundaries and features from which a finite element mesh can be created. *SMS* then interpolates the bathymetry data onto the mesh (see chapter 13). This process is also described in Lesson 2 of the tutorials. The *Mesh Module* provides various tools for manually editing the finite element mesh.

4.2 Current Numerical Model

The mesh module is set up to be used with a single numerical model analysis engine at any given time. The current numerical model is changed using the *Data / Switch Current Model* menu command. *SMS* shows only those tools in the *Tool Palette* and those menus in the *Menu Bar* which are relevant to the current numerical model. After a finite element mesh has been created, boundary conditions and material properties can be assigned using the commands in the menus associated with the current numerical model. *SMS* currently supports the following numerical models: *TABS (RMA2/SED2D)*, *FESWMS*, *HIVEL2D*, *RMA10*, *ADCIRC*, *CGWAVE*. By default, *SMS* starts inside the *Mesh Module* with the *TABS* model current. The current model on startup can be changed using command line arguments as explained in section 1.2. The mesh generation tools and menu commands are described in this chapter. Commands in the numerical model menus are described in later chapters.

4.3 Tool Palette

The following tools are contained in the *Dynamic Tools* portion of the *Tool Palette* when the *Mesh Module* is active. Unless otherwise noted, these tools are available in all numerical model interfaces.

4.3.1 Create Mesh Nodes



The *Create Mesh Nodes* tool is used to manually create a node using the mouse. A node will be created at the location where the mouse button is clicked inside the *Graphics Window*. If the node is created inside the triangulated area of a current mesh, and the *Insert nodes into triangulated mesh* option is turned on (section 4.6.8), then the new node will be added as part of the mesh. If the new node is not added as part of the current mesh, then the z-value assigned depends on the Nodal z-value option (section 4.6.8).

4.3.2 Select Mesh Nodes



The *Select Mesh Nodes* tool is used to select nodes. A single node is selected by clicking on it. A second node can be added to the selection list by holding the *SHIFT* key while selecting it. Multiple nodes can be selected at once by dragging a box around them. A selected node can be deselected by holding the *SHIFT* key as it is clicked.

If the nodes are not locked (see section 4.6.8), then a single node can be clicked and dragged to a new location. As the node is being dragged, its new location is shown in the *Edit Window*. If a single node selected, the *X*, *Y*, and *Z Coordinate* fields in the *Edit Window* become available to set the node location exactly. If multiple nodes are

selected, the *Z Coordinate* field in the *Edit Window* becomes available. The value shown is the average elevation value of all selected nodes. If this value is changed, the new value will be assigned to all selected nodes.

With one node selected, the *Edit Window* shows the node id number and the number of elements to which it is attached. With two nodes selected, the *Edit Window* shows both node id numbers and the distance between the nodes. With multiple nodes selected, the *Edit Window* shows the number of selected nodes.

4.3.3 Create Nodestrings

The *Create Nodestrings* tool is used to create node string. Nodestrings are used for operations such as assigning boundary conditions, forcing breaklines into the mesh, and renumbering the mesh. To create a nodestring:

1. Click on a node. The node will be highlighted in red and a prompt will be shown in the *Help Window*.
2. Click on any node to add it to the nodestring. The selected node is also highlighted in red and a solid red line is drawn between the two nodes. Continue adding nodes to the nodestring in this manner.

Note: For most operations, nodes in the nodestring should be adjacent, but this is not required. A breakline, for example, will usually be made of nodes which are not adjacent.

3. Press the *BACKSPACE* key to backup one node. Press the *ESC* key to abort the nodestring creation.
4. Double-click a node or press the *ENTER* key to end the nodestring creation.

The *SHIFT* and *CTRL* keys assist in creating large nodestrings which are made up of adjacent nodes. These can be used after at least one node has been selected and function as follows:

- ? *SHIFT*. Holding down the *SHIFT* key and selecting another node will add to the nodestring all nodes between the two. The path chosen is the smallest distance between the two nodes. This is useful for creating continuity strings which run along a cross section of the mesh.
- ? *CTRL*. Holding down the *CTRL* key and selecting another node will add to the nodestring all nodes on the mesh boundary between the two, going counter clockwise from the first node to the second node. Both nodes must be on the boundary of the mesh or SMS will beep.

- ? *CTRL + SHIFT*. Holding down both the *CTRL* and *SHIFT* keys and selecting another node will add to the nodestring all nodes on the mesh boundary between the two, going clockwise from the first node to the second node. Both nodes must be on the boundary of the mesh or SMS will beep.

4.3.4 Select Nodestrings



The *Select Nodestrings* tool is used to select nodestrings. When this tool is chosen, a small icon appears near the center of each nodestring. A nodestring is selected by clicking inside this icon. A second nodestring can be added to the selection list by holding the *SHIFT* key while selecting it. Multiple nodestrings can be selected by dragging a box around their icons. A selected nodestring can be deselected by holding the *SHIFT* key as its icon is clicked.

When nodestrings are selected, the *Z Coordinate* field in the *Edit Window* becomes available. The value shown is the average elevation value of all nodes in the selected nodestrings. If this value is changed, the new value will be assigned to all nodes in the selected nodestrings.

With one nodestring selected, the *Edit Window* shows the number of nodes in the nodestring, its type, and its length. With multiple nodestrings selected, the *Edit Window* shows the number of selected nodestrings and their total length.

4.3.5 Create Elements

Most elements in *SMS* will be created using automatic mesh generation techniques. At times, however, it is necessary to manually create a single element or a small group of elements.

Although *SMS* supports various types of elements, only those element types supported by the current numerical model (see section 4.1) will be available in the *Tool Palette*. Some of these element types are linear while others are quadratic. A linear element has only corner nodes, while a quadratic element has midside nodes between the corner nodes.

The following linear elements are supported:

-  Two-node lines.
-  Three-node triangles.
-  Four-node quadrilaterals.

The following quadratic elements are supported:

-  Three-node lines.
-  Six-node triangles.
-  Eight-node quadrilaterals.
-  Nine-node quadrilaterals.

Linear and quadratic elements cannot coexist in a single mesh. If linear elements exist in a mesh, then the quadratic element creation tools are dimmed out. Similarly, if quadratic elements exist in a mesh, then the linear element creation tools are dimmed out. To create a single linear or quadratic element:

1. Select the tool which corresponds with the type of element to be created.
2. Click on the corner nodes which will make the element, one-by-one. *Do not click midside nodes.* As each node is clicked, it becomes highlighted in red.
3. Alternatively, a box can be dragged around the corner nodes which will make the element. A beep will sound if the box does not surround the exact number of corner nodes required by the selected tool.

The midside nodes in quadratic elements are created automatically, as is the center node of a nine-node quadrilateral. Before a new element is created, *SMS* performs the following quality checks. If any of these fails, the new element is not created.

- ? An new element cannot overlap other elements.
- ? A quadrilateral element cannot twist or overlap itself.
- ? A quadrilateral element cannot be concave.

The following special element is supported:

-  Control Structure.

A 1D control structure can be created at the junction of any two line elements. This element is used to define a special situation such as equal head or momentum conservation. These control structures are used only by the *TABS* and *RMA10* models. To create a control structure:

1. Select the *Create Control Structure* tool.

2. Click on a node which connects exactly two line elements.

When a control structure is created, an arrow shows the expected flow direction. For information on changing the flow direction, as well as other parameters associated with it, see section 4.7.18.

4.3.6 Select Elements



The *Select Elements* tool is used to select elements. A single element is selected by clicking inside it. A second element can be added to the selection list by holding the *SHIFT* key while selecting it. Multiple elements can be selected at once by dragging a box around them. Holding the *CTRL* key and dragging the mouse selects any elements through which the line is drawn. A selected element can be deselected by holding the *SHIFT* key as it is clicked.

When elements are selected, the *Z Coordinate* field in the *Edit Window* becomes available. The value shown is the average elevation value of all nodes in the selected elements. If this value is changed, the new value will be assigned to all nodes attached to the selected elements. Caution must be used when changing node elevations in this manner. Do not create large flat areas where surrounding elements may become dry because this can cause ponds to form when the finite element analysis is performed.

With one element selected, the *Edit Window* shows the element id number, its type, and its area. With multiple elements selected, the *Edit Window* shows the number of selected elements and their combined area.

4.3.7 Swap Edges



The *Swap Edges* tool is used to manually swap the edges of two adjacent triangles. This is useful in such cases as preserving a geometrical feature in the mesh or avoiding an artificial dam in a channel.

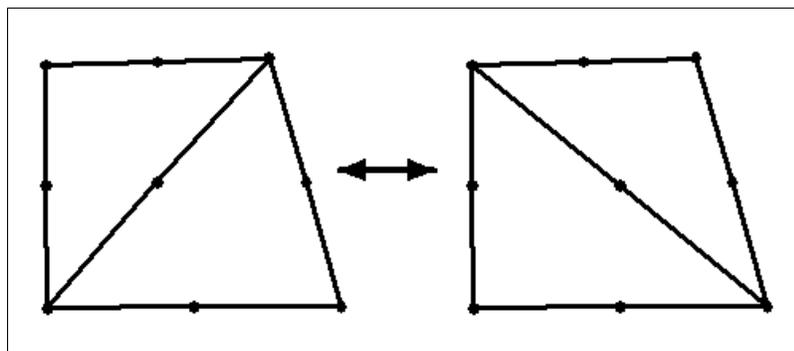


Figure 4.1 Swap tool example.

Two adjacent triangles form a quadrilateral with an element edge down one diagonal. When the diagonal is clicked, it gets swapped to the other diagonal, as long as the quadrilateral is not concave. An example of swapping is shown in Figure 4.1.

4.3.8 Merge/Split Elements



The *Merge/Split Elements* tool is used to either merge two triangles into a quadrilateral or split a quadrilateral into two triangles. This is a useful tool to use when trying to avoid certain mesh drying problems.

To split a quadrilateral element, click inside it. An element edge appears on the diagonal which will make the triangles uphold the Delaney criteria. To merge two adjacent triangles into a quadrilateral, click the common edge. A quadrilateral will form as long as it is not concave. An example using this tool is shown in Figure 4.2.

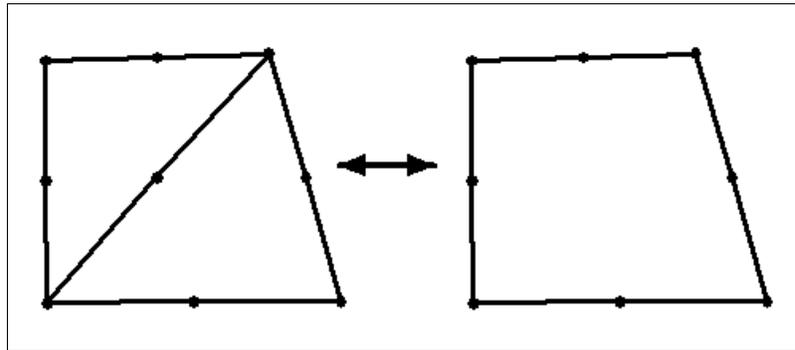


Figure 4.2 Merge/Split tool example.

4.3.9 Label Contours



The *Label Contours* tool is used to manually create a contour label using the mouse. To add a label, click on the point where the label should be created. The label will remain on the screen until either it is manually removed or the automatic contour label options are changed. To manually remove a contour label, hold the *SHIFT* key and clicking on it. See section 3.6 for more information on automatic contour label options.

4.3.10 Create Piers



The *Create Piers* tool is used to manually create a pier using the mouse. To create a pier, click the mouse in the *Graphics Window* at the desired location. When the pier is created, its exact coordinates are shown in the *Edit Window*. The pier can be moved by changing these coordinates.

4.3.11 Select Piers



The Select Piers tool is used to select piers. A single pier is selected by clicking on it. A second pier can be added to the selection list by holding the SHIFT key while selecting it. Multiple piers can be selected at once by dragging a box around them. A selected pier can be deselected by holding the SHIFT key as it is clicked.

A single pier can be clicked and dragged to a new location. As the pier is dragged, its new location is shown in the Edit Window. If a single pier is selected, the X and Y Coordinate fields in the Edit Window become available to set the pier location exactly.

With one pier selected, the Edit Window shows the pier width, length, and scour shape. With multiple piers selected, the Edit Window shows the number of selected piers.

4.4 Data Menu

Most of the SMS modules have a *Data* menu, but the items in this menu are different for each module. The first half of the *Mesh Data* menu has already been described in chapter 2.5.2. These items include manipulation of functional data sets through the *Data Browser* and *Data Calculator*, as well as visualization tools including *Contour Options*, *Vector Options*, *Drogue Options* and *Film Loops*. The rest of the *Mesh Data* menu is described in this section.

4.4.1 Mesh -> Scatterpoint

The *Mesh -> Scatterpoint* command in the *Mesh Data* menu is used to create scatter point data from existing mesh nodes. When this command is performed, a prompt appears to request a name (see Figure 4.3) for the new scatter point set.

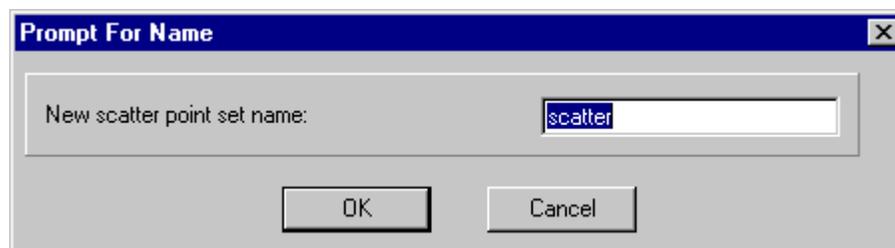


Figure 4.3 Mesh -> Scatterpoint prompt.

When the scatter point set is created, it contains one scatter point for each mesh node, including midside nodes and center nodes. Any mesh data sets that have been read into SMS are copied into the new scatter point set. The scatter point data can then be used for interpolation. The use of the *Scatter Point Module* is described in chapter 13.

4.4.2 Mesh -> Grid

This command is reserved for future use.

4.4.3 Material -> Feature

The *Material -> Feature* command in the *Mesh Data* menu is used to create feature objects from an existing finite element mesh. When this command is performed, a prompt appears (see Figure 4.4) to choose whether to add to the current coverage or to create a new coverage for the new feature objects.

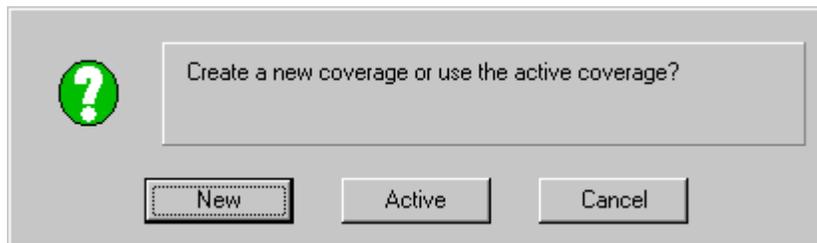


Figure 4.4 *Material -> Feature* prompt.

Unless the *Cancel* button is pressed in this dialog, feature arcs are created from the mesh boundary and material boundaries. Feature polygons are then built out of the feature arcs to represent existing material regions. After the feature arcs and polygons are created, the arc vertices can be redistributed and the finite element mesh can be regenerated.

Note: Before regenerating the mesh from feature object data, the original node elevations should be converted to scatter points (see section 4.4.1) so that they can be interpolated onto the new mesh.

4.4.4 Mesh Contour -> Feature

The *Mesh Contour -> Feature* command in the *Mesh Data* menu is used to create a feature arc from a contour line. When this command is performed, a prompt appears (see Figure 4.5) to specify the contour value to use. If the value specified is not one of the displayed contour lines, then the feature arc will be created from an imaginary contour line of the value.

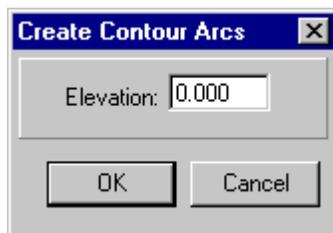


Figure 4.5 Mesh Contour -> Feature prompt.

All contour lines of the specified value will be converted to feature arcs and added to the active coverage. If the value does not exist in the finite element mesh, then no feature arcs will be created.

4.4.5 Map Elevation

The *Map Elevation* command in the *Mesh Data* menu is used to define the data set used for nodal elevation values. When this command is performed the *Select Data Set* dialog opens (see Figure 4.6) to allow any scalar data set to be chosen.

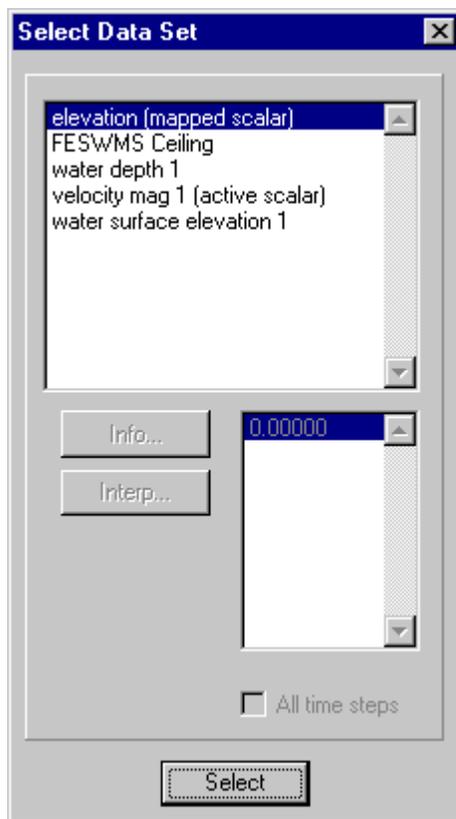


Figure 4.6 Select Data Set dialog.

By default, *SMS* creates a data set named *elevation* to store the nodal elevation values. The data set being used to store nodal elevations is referred to as the *mapped* data set. Any time step of any scalar data set can be used as the mapped data set and override the previous nodal elevation values. This is used mainly when interpolating new elevation data from scatter points (see chapter 13).

4.5 Mesh Display Options

The properties of all mesh data that *SMS* displays on the screen can be controlled through the mesh *Display Options* dialog (see Figure 4.7). This dialog is opened by selecting the *Display Options* command from the *Display* menu.

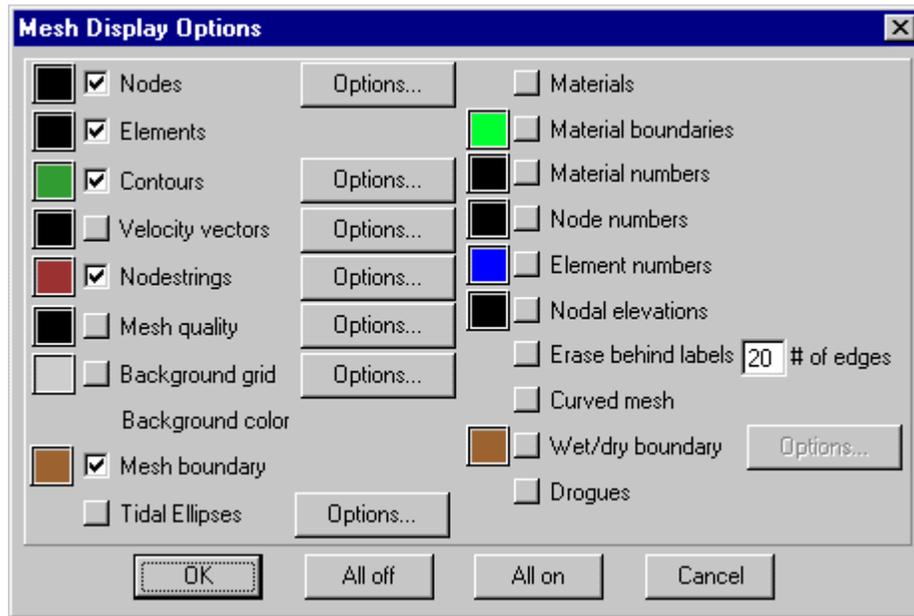


Figure 4.7 Mesh Display Options dialog.

This dialog has a toggle box for each entity to tell whether or not it should be displayed. Most of these entities also show a small rectangle to the left which is filled with the color in which they should be displayed. Click in this rectangle to change the display attributes, including the color, font, fill pattern, line thickness, and point radius. Some of these entities also show an *Options* button to the right. For these entities, additional display controls are available. The available mesh display options include the following:

- *Nodes*. A circle is filled around each node. The *Options* button is used to set the display of nodal boundary condition data. The dialog that opens when this button is clicked depends on the current numerical model (see section 4.1). When exporting data to an AutoCAD DXF file, this is saved if it is on.
- *Elements*. Element edges are drawn using the specified line attributes. When exporting data to an AutoCAD DXF file, this is saved if it is on.
- *Contours*. The mesh contours are drawn for the active scalar data set. The specific contour display options are described in section 3.4. When exporting data to an AutoCAD DXF file, this is saved if it is on.

- *Velocity Vectors.* The mesh vectors are drawn for the active vector data set. The specific vector display options are described in section 3.3.
- *Nodestrings.* The color in which a nodestring is drawn depends upon its type. Unassigned nodestrings are drawn in the color shown at the left of the toggle box. For the display of boundary condition nodestrings, click the *Options* button. The dialog that opens when this button is clicked depends on the current numerical model (see section 4.1).
- *Mesh Quality.* The mesh quality shows potential problems with the finite element mesh. An element is highlighted in a color corresponding to the criterion which it violates. The *Options* button opens the *Mesh Quality Options* dialog in which various criteria can be set (see section 4.5.1).
- *Background Grid.* A Cartesian grid can be displayed in the background to serve as a guide when creating nodes. Interactively created nodes can be snapped to the grid corners. This option is generally turned off.
- *Background Color.* The *Graphics Window* background can be set to any desired color.
- *Mesh Boundary.* A line around the perimeter of the mesh can be drawn. This is useful when the elements are turned off. When exporting data to an AutoCAD DXF file, this is saved if it is on.
- *Tidal Ellipses.*
- *Materials.* Elements can be filled with the color and pattern which define their materials. See section 2.5.2 for changing material display attributes.
- *Material Boundary.* The boundary between zones of elements with a common material type is drawn using specified line attributes. When exporting data to an AutoCAD DXF file, this is saved if it is on.
- *Material Numbers.* The *material id number* can be displayed in the center of each element.
- *Node Numbers.* The *node id number* can be displayed next to each node.
- *Element Numbers.* The *element id number* can be displayed in the center of each element.
- *Nodal Elevations.* The *Z Coordinate* can be displayed next to each node.
- *Erase Behind Labels.* This causes the region behind labels to be painted in the background color before it is written. This makes labels more visible.

- *Curved Mesh.* When midside nodes of quadratic elements are dragged from the midpoint of the two corner nodes, they can be displayed with curved edges.
- *Wet/Dry Boundary.* After a simulation has been opened, the interface between wet and dry nodes can be displayed. When exporting data to an AutoCAD DXF file, this is saved if it is on.
- *Drogues.* The display of drogues can be turned on or off. See section 2.5.1 for more information on drogue options.

4.5.1 Mesh Quality

The quality of the finite element mesh has a large effect on the validity of the finite element analysis. *SMS* provides certain quality checks to help find potential problems with the mesh. Inside the *Mesh Display Options* dialog, there is an option named *Mesh Quality*, as discussed in the previous section. Click the *Mesh Quality Options* button to open the *Element Quality Checks* dialog (see Figure 4.8).

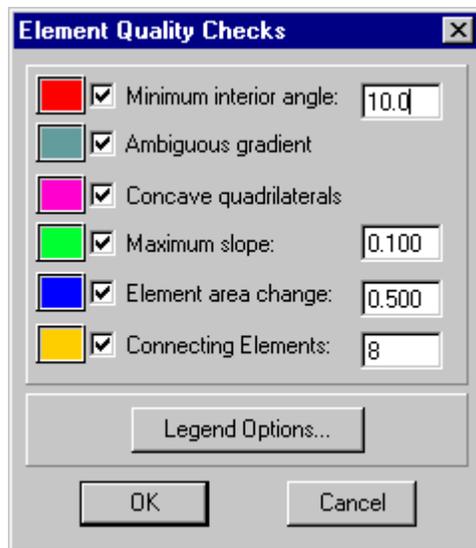


Figure 4.8 *Element Quality Checks* dialog.

Each of the element quality checks can be individually turned on or off. This way, only the desired quality checks will be made. The available quality checks include:

- ? *Minimum interior angle.* A triangular element that has an angle less than this value will be drawn in the specified color. A quadrilateral element that has an angle less than twice this value will be drawn in the specified color. This option warns of skinny triangles and skinny diamond-shaped quadrilaterals.
- ? *Ambiguous gradient.* Only a quadrilateral can have an ambiguous gradient. This occurs when two diagonal corner nodes are higher than the other two nodes, forming a saddle point in the middle of the element. These elements are

undesirable because a drop of water at the middle of the element could flow downhill in two different directions.

- ? *Concave quadrilaterals.* SMS attempts to prevent the creation of concave quadrilateral elements. However, they happen to be created, this check will highlight the element. There should never be concave quadrilateral elements in a finite element mesh.
- ? *Maximum slope.* In the flow direction, a steep element slope indicates the possibility of supercritical flow. Since not all models support supercritical flow, this can be a problem. In the direction perpendicular to flow, a steep element slope indicates possible wet/dry shocking. If a large element with a steep slope perpendicular to the flow direction becomes dry, then a large volume of water is forced into the rest of the mesh and the subsequent shock can cause divergence.
- ? *Element area change.* It is suggested that the area difference between two adjacent elements not exceed 50%. In other words, an element whose area is 20 ft² should not have an element next to it whose area is greater than 40 ft² or less than 10 ft². This check draws a line between any two elements whose area change is greater than the specified percent. The percent should be entered as a decimal, such as the default of 0.50 for 50%.
- ? *Connecting elements.* Although SMS can support any number of elements being connected to a specific node, most of the finite element analysis engines can only support up to eight elements connected to a single node. This check highlights any node that is connected to more than the specified number of elements, the default being eight.

When these mesh quality checks are turned on, a legend appears to show the meaning of the different colors. This legend uses the general legend options. Only the checks that are turned on will be shown in the legend.

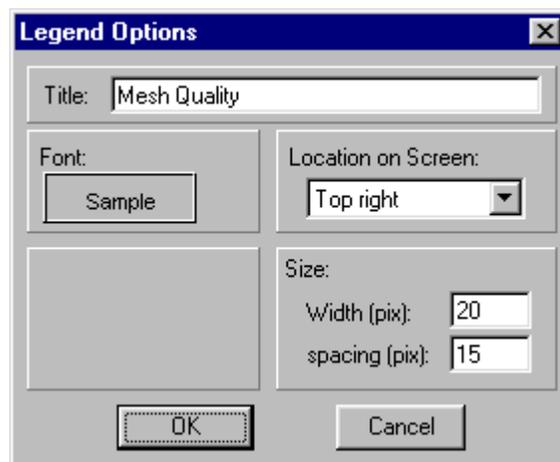


Figure 4.9 Mesh Quality Legend Options Dialog.

4.6 Node Menu

Nodes are the basic building blocks of elements in finite element meshes. Nodes are also used for building nodestrings and assigning certain boundary conditions. The method for creating individual nodes was described in section 4.3.1. The *Node Menu* contains special commands for operating on existing nodes.

4.6.1 Interpolation Options

The *Interpolation Options* item from the *Nodes* menu opens the *Node Interpolation Options* dialog (see Figure 4.10). Using the options that are set in this dialog, a set of new nodes can be interpolated between any two existing nodes (see section 4.6.2).

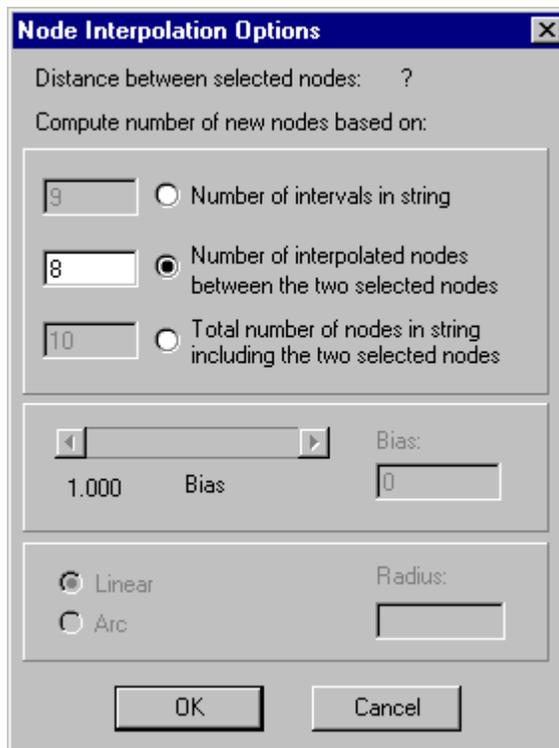


Figure 4.10 Node Interpolation Options.

If two nodes are selected when this dialog is invoked, the distance between the two nodes is displayed at the top of the dialog. The number of new nodes can be specified in three ways:

- ? *Number of intervals in string.* If this option is chosen, the number of new nodes is one less than the number of intervals specified.
- ? *Number of interpolated nodes.* If this option is chosen, the number of new nodes is exactly specified.

- ? *Total number of nodes in string.* If this option is chosen, the number of new nodes is two less than the number of nodes specified.

The *Bias* factor controls the distribution spacing of the new nodes. This factor can be any number between 0.1 and 10.0. A smaller factor will make new nodes be closer to the first selected node while a larger factor will make new nodes be closer to the second selected node. For example, a bias of 2.0 makes the first new node spaced twice as far as the last new node (see Figure 4.11).

The *Linear/Arc* option controls the distribution shape of the new nodes. The *Linear* option causes all new nodes to be in a straight line while the *Arc* option causes all new nodes to form an arc. If the arc option is used, a *Radius* must also be specified. The arc will be created counter-clockwise from the first selected node to the second (see Figure 4.11).

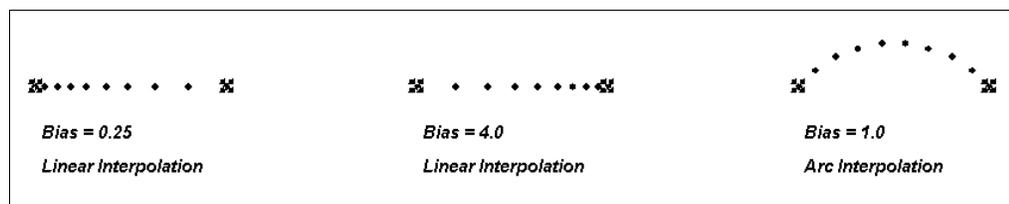


Figure 4.11 Interpolation option examples.

4.6.2 Interpolate

After the interpolation options are set up, nodes can be interpolated between any two selected nodes by choosing the *Interpolate* item from the *Nodes* menu. This operation may be performed multiple times with a single set of interpolation options by selecting any two nodes and invoking the command again. See section 4.3.2 for help with selecting mesh nodes.

The elevation of each new node depends on the *Insert nodes into triangulated mesh* option in the *Node Options* dialog (see section 4.6.8). If this option is turned on and the new node is inside the finite element mesh, then the elevation is interpolated from the mesh. If this option is turned off or the new node is not inside the finite element mesh, the elevation is interpolated from the two selected nodes.

4.6.3 Find Node

The *Find Node* , command from the *Nodes* menu is used to locate a node either with a specific ID, or near a specific location. When this command is executed the *Find Node* dialog opens (see Figure 4.12).

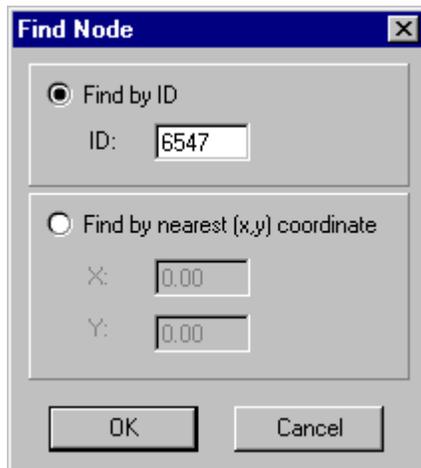


Figure 4.12 Find Node dialog.

When the *Find by ID* option is selected, then the node with the specified *ID* is highlighted with a red circle. If there is no node with the specified id, then an error message is given. Conversely, when the *Find by nearest (x,y) coordinate* option is selected, the node closest to the specified (x, y) location is highlighted with a red circle.

With either of these methods, if the current tool is the *Select Nodes*  tool, then the found node becomes selected in addition to being highlighted.

4.6.4 Duplicate Nodes

Duplicate nodes are either selected or deleted, according to the option defined in the *Node Options* dialog (see section 4.6.8). The behavior that will be carried out is shown in this menu item text, as it will read either *Select Duplicate Nodes* or *Delete Duplicate Nodes*. Two nodes are considered to be duplicates if they are closer together than the *Tolerance* in the *Node Options* dialog. When deleting duplicate nodes, elements attached to deleted nodes will also be removed, unless the *Merge adjacent elements when deleting* option is turned on in the *Node Options* dialog.

4.6.5 Select Disjoint Nodes

Disjoint nodes can be found automatically and selected by choosing the *Select Disjoint Nodes* option from the *Nodes* menu. Disjoint nodes are nodes that are not connected to any elements. Before saving a simulation, it is important to make sure there are no disjoint nodes in the mesh.

4.6.6 Locked

The nodes in a mesh can be dragged with the mouse cursor if they are unlocked and the *Select Nodes*  tool is selected. The *Locked* item in the *Nodes* menu toggles on and off the node locked status. If nodes are locked, a check mark is shown next to the menu text. The default status is locked so that nodes are not accidentally moved.

4.6.7 Transform

The *Transform* command from the *Nodes* menu is used to move a group of selected nodes. If there are no selected nodes, the transformation will be applied to all nodes of in mesh. When this command is executed, the *Nodes Transform* dialog opens (see Figure 4.13).

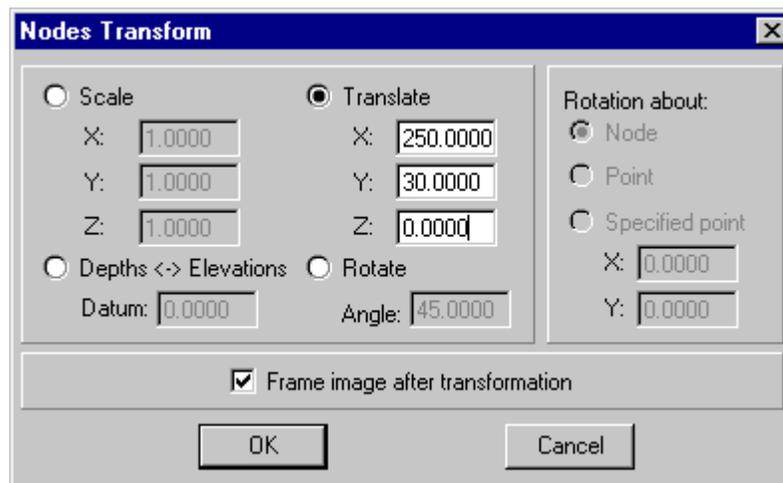


Figure 4.13 *Nodes Transform dialog.*

In this dialog, the transformation type can be chosen and then appropriate parameters can be entered. The following transformation types are available:

- *Scale*. Scaling factors for the *X*, *Y*, and/or *Z* directions are entered. To avoid scaling a specific direction, the default value of 1.0 should be used.
- *Translate*. The change in *X*, *Y*, and *Z* are entered. To avoid translation in a specific direction, the default value of 0.0 should be used.
- *Depths <-> Elevations*. Nodal *Z*-values can be converted from elevations to depths, and vice versa. A constant water surface elevation (*WSE*) value is defined for the conversion process. Note that this is different than simply translating or scaling the *Z*-coordinate. This transformation is governed by the following simple equation:

$$\text{new } Z = \text{WSE} - \text{old } Z$$

- *Rotate*. When rotation is selected, the set of options on the right side of the dialog become available to define the center of rotation. If the *Specified Point* option is used, then the center of rotation is explicitly defined. Otherwise, after clicking the *OK* button from the *Nodes Transform* dialog, the user must click in the graphics window at the *Point* or on the *Node* about which the rotation should occur. The rotation will occur counter-clockwise by the specified angle around the specified center of rotation.

By default, the image will be framed after the transformation takes place. However, this can be turned off by using the *Frame image after transformation* option.

4.6.8 Options

Certain parameters governing the creation and manipulation of nodes are set using the *Node Options* dialog (see Figure 4.14), which is opened by selecting the *Options* command from the *Nodes* menu.

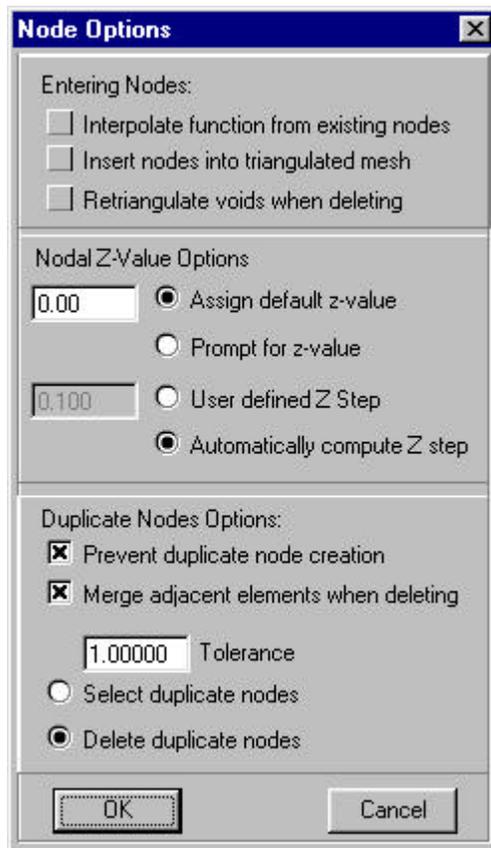


Figure 4.14 *Node Options* dialog.

The top section of the *Node Options* dialog has the following three toggle boxes:

- ? *Interpolate function from existing nodes.* When new nodes are added inside the mesh boundary using the *Create Nodes*  tool (see section 4.3.1), the Z coordinate can be determined by interpolation from the existing mesh. If this option is turned off, then the default Z coordinate depends on the *Nodal Z-Value Options* section of this dialog.
- ? *Insert nodes into triangulated mesh.* When a node is added inside the mesh boundary, it can become part of the mesh. If this option is turned off, then new nodes remain disjoint. This option also applies to nodes created using the *Interpolate* command from the *Nodes* menu (see section 4.6.2).
- ? *Retriangulate voids when deleting.* When deleting a node that is attached to elements, the elements are also removed. The void in the mesh left by the deleted elements can be automatically filled by triangulating the surrounding nodes. If this option is turned off, then the void will remain.

The center section of the *Node Options* dialog has values to control nodal Z values.

- ? *Default Z Coordinate.* The Z coordinate assigned to a node created with the *Create Nodes*  tool depends on this option. If the *Prompt for z-value* option is set, then the user is prompted for the Z coordinate of the new node. If the *Assign default z-value* option is set, then the specified value is assigned. However, this specified value can be overridden if the new node is inside the mesh and the *Interpolate function from existing nodes* option is on. For this latter case, the Z coordinate is interpolated from the existing mesh.
- ? *Z Step.* The Z coordinate of selected nodes can be changed using the *Z Coordinate* field in the *Edit Window* (see section 2.4). To the right of the *Z Coordinate* field are arrows to increase or decrease the Z coordinate by a small amount. This small amount can be explicitly defined using the *User defined Z step* option, or *SMS can Automatically compute the Z step*.

The bottom section of the *Node Options* dialog has the following options for dealing with duplicate nodes:

- ? *Prevent duplicate node creation.* When turned on, *SMS* does not allow duplicate nodes to be created.
- ? *Merge adjacent elements when deleting.* When deleting duplicate nodes (see section 4.6.4), any elements attached to the deleted nodes will also be deleted, unless this option is turned on.
- ? *Tolerance.* This field defines how close two nodes need to be to be considered as duplicates.

? *Select/Delete duplicate nodes.* This determines the action of the *Duplicate Nodes* menu command as described in section 4.6.4.

4.6.9 Interpolate Nodal BC

If two non-adjacent boundary nodes have been assigned boundary conditions, and the two nodes are selected, this command interpolates the boundary conditions to each of the boundary nodes between the two.

4.7 Elements Menu

Elements fill in the area between the nodes to define the flow area. The creation of individual elements was discussed in section 4.3.5. This process is labor intensive, so it is usually better to use automatic mesh generation techniques. The *Map Module* (see chapter 14) provides powerful, fully automatic mesh generation. However, some semi-automatic mesh generation methods are available using commands in the *Elements* menu of the *Mesh Module*.

4.7.1 Options

Certain parameters governing the creation and manipulation of nodes are set using the *Element Options* dialog (see Figure 4.15), which is opened by selecting the *Options* command from the *Elements* menu. This dialog is divided into four sections.

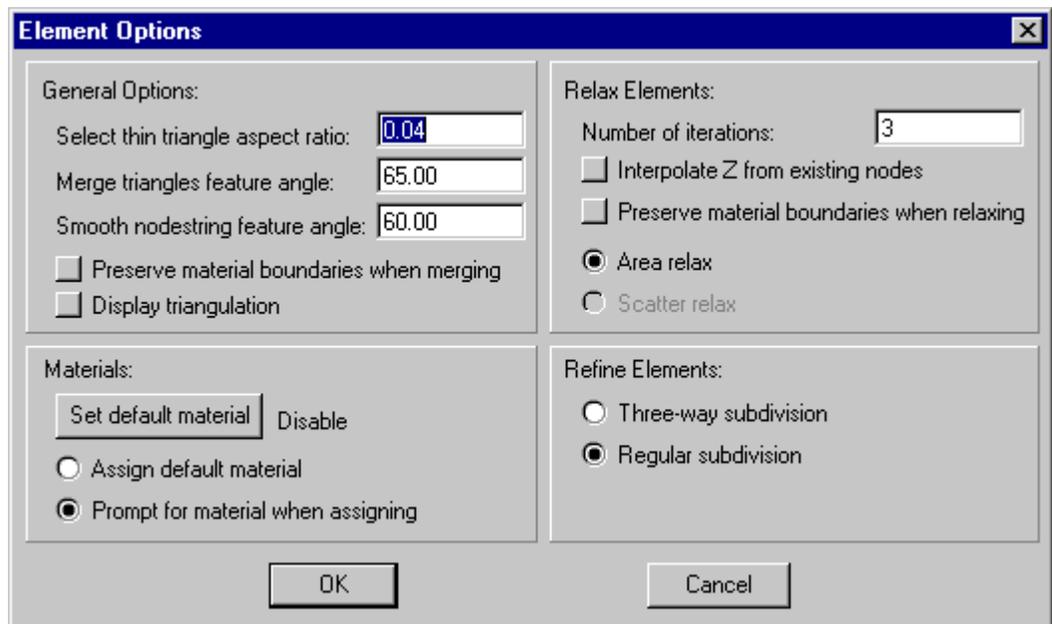


Figure 4.15 The *Element Options* dialog.

General Options

The *General Options* section of the *Element Options* dialog specifies the following parameters for general element operations:

- *Select thin triangle aspect ratio.* When SMS finds thin triangles (see section 4.7.6), only elements with an aspect ratio (element width divided by element length) less than this value are selected.
- *Merge triangle feature angle.* When triangles are automatically merged into quadrilaterals (see section 4.7.9), all angles inside the resulting quadrilateral must be larger than this value.
- *Smooth nodestring feature angle.* When a nodestring gets smoothed (see section 4.7.15), the smoothing will not be applied around a corner whose angle is greater than this value.
- *Preserve material boundaries.* When this is turned on, triangles will not be merged into quadrilaterals (see section 4.7.9) if they are assigned different materials types, even if they satisfy the *Merge triangle feature angle* criteria.
- *Display triangulation.* When this is turned on, the triangulation process (see section 4.7.2) will be displayed as it occurs. Although this is interesting to watch, it slows down performance.

Materials

The *Materials* section of the *Element Options* dialog controls how materials are assigned to elements using the following options:

- ? *Set default material.* This defines the material assigned to elements as they are created.
- ? *Assign default material.* If this option is selected, the material selected as the “default” material is assigned to selected elements when the assign material command is issued.
- ? *Prompt for material when assigning.* If this option is selected, the user must select from a list of existing materials to assign to the selected elements when the assign material command is issued.

Relax Elements

The *Relax Elements* section of the *Element Options* dialog controls the following relaxation parameters. Relaxation is described in section 4.7.14.

- *Number of iterations.* This is the number of iterations to perform during the relaxation process.

- *Interpolate Z from existing mesh.* When this is turned on, the nodal Z coordinate is interpolated from the old mesh so that the contours do not change. When this is turned off, the nodal Z coordinates are not changed when they are moved.
- *Preserve material boundaries when relaxing.* When this is turned on, a node will not be moved if it lies on a material boundary.
- *Area relax / Scatter relax.* Relaxation will be based on either element areas or underlying scattered data.

Refine Elements

When refining an element, it is traditionally cut into fourths. There is a new algorithm used only for triangles which refines it into three elements, called the Three-way subdivision. These two refinement types are shown in Figure 4.16.

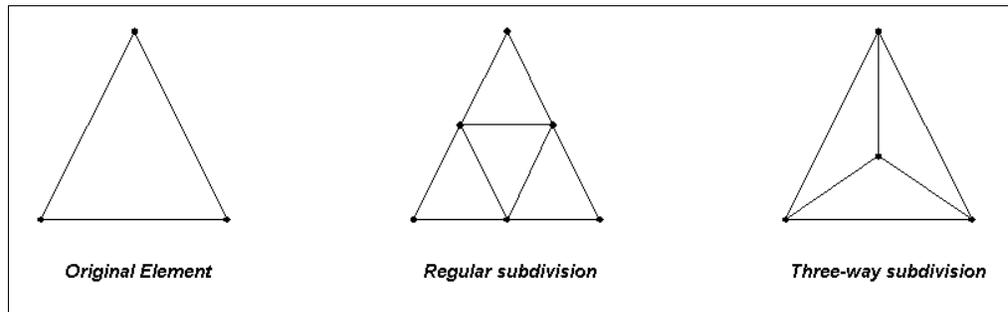


Figure 4.16 Types of refinement for triangular elements.

4.7.2 Triangulate

New elements are constructed in mass by triangulating a set of nodes when the Triangulate command from the Elements menu is executed. The selected nodes are connected with a series of triangles as shown in Figure 4.17. If nodes are not selected, then all nodes will be triangulated. If linear elements exist, or a linear element creation tool has been selected, then this command creates linear triangles. Otherwise, quadratic triangles are created.

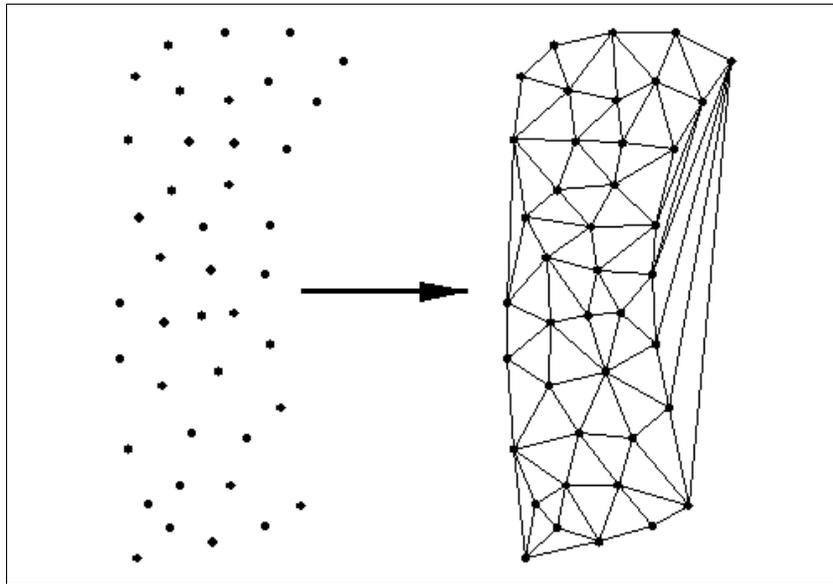


Figure 4.17 Triangulation of nodes.

The triangulation algorithm ensures that the Delaunay criterion is satisfied. The Delaunay criterion is such that the circumcircle of a triangle does not enclose a node on any other element. The circumcircle of a triangle is the circle that passes through its vertices as shown in Figure 4.18.

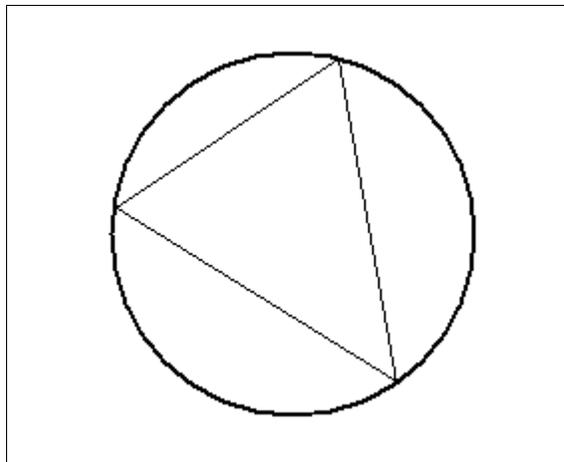


Figure 4.18 Circumcircle of a Triangle.

4.7.3 Optimize Triangulation

At times, the user will perform manual mesh editing using the *Swap Edge*  tool (see section 4.3.7). This makes the Delaunay criterion no longer hold. Selected elements can be returned to the Delaunay state by choosing the *Optimize Triangulation* command from the *Elements* menu.

4.7.4 Rectangular Patch

Elements can be made to fill a rectangular area by choosing the *Rectangular Patch* option from the *Elements* menu. To define a rectangular patch, four nodestrings must be selected. The nodestrings must connect at the ends. The *Rectangular Patch Options* dialog is shown in Figure 4.19.

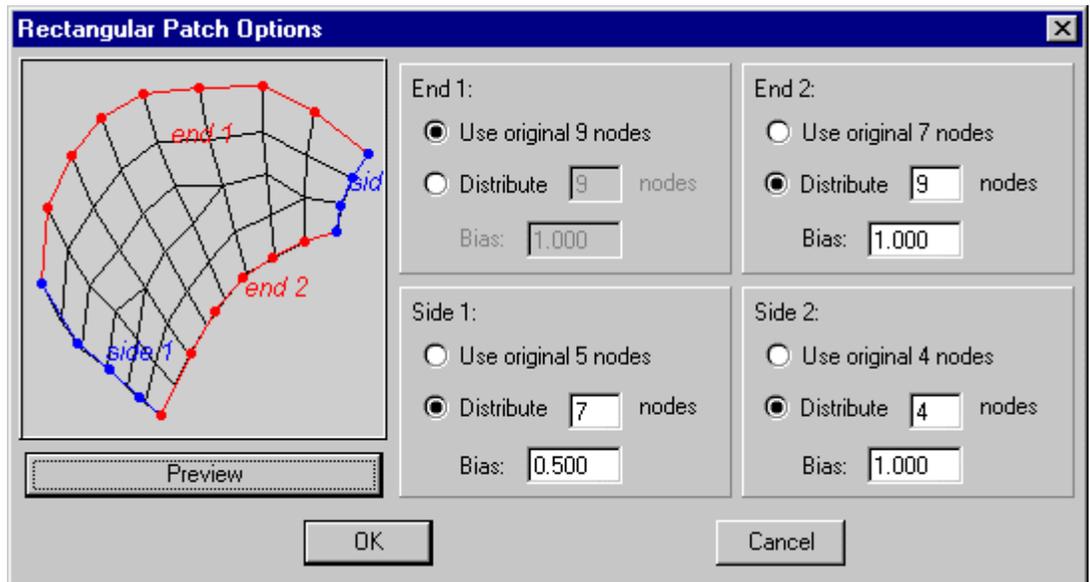


Figure 4.19 Rectangular Patch Options Dialog.

The coordinates of the new nodes on the interior of the patch are computed by constructing a partial bicubic Coons patch using the nodestrings as patch edges. This ensures that interior nodes are smoothly interpolated from the nodes making up the perimeter of the patch. Patches are applicable when the data points are gathered along parallel lines, such as cross sections in a river. The following options are available for each edge of the rectangular patch:

- ? *Use original nodes.* This option causes the original nodes from the nodestring to be used as corner nodes of elements along the boundary.
- ? *Distribute nodes.* This option distributes the specified number of nodes as corner nodes of elements along the boundary. If elements already exist on the boundary, then this option is unavailable.
- ? *Bias.* This is used with the *Distribute nodes* option. It causes the spacing of nodes along the nodestring to be weighted more to one of the corners.

After the spacing on each side is defined, click the *Preview* button to see how the patch will look. If changes are desired, they can be made. When the patch looks good, click the *OK* button to accept it. The patch can be canceled by clicking the *Cancel* button. Be careful to use the preview button because THERE IS NO UNDO FOR THIS OPERATION.

The elements in a new patch are checked to make sure they do not overlap each other. If any problems are detected, an error message is given and the patch is not created. Errors may occur especially when the region is highly irregular in shape. In such cases, the region can either be divided into smaller patches, or it can be filled using a different mesh generation technique.

The following are some hints when using rectangular patches:

- ? The curvature of the patch can change somewhat, but it should not switch directions. If it does, then the patch should be split at the inflection point of the curve.
- ? Although opposite sides in the rectangular patch are not required to have the same number of nodes, the best patches occur when this is close. In the example shown in Figure 4.19, the two ends have the same number of nodes and the two sides only differ by three nodes.

4.7.5 Triangular Patch

Elements can be made to fill a triangular area by choosing the *Triangular Patch* option from the *Elements* menu. To define a triangular patch, three nodestrings must be selected. The nodestrings must connect at the ends. The *Triangular Patch Options* dialog is shown in Figure 4.20.

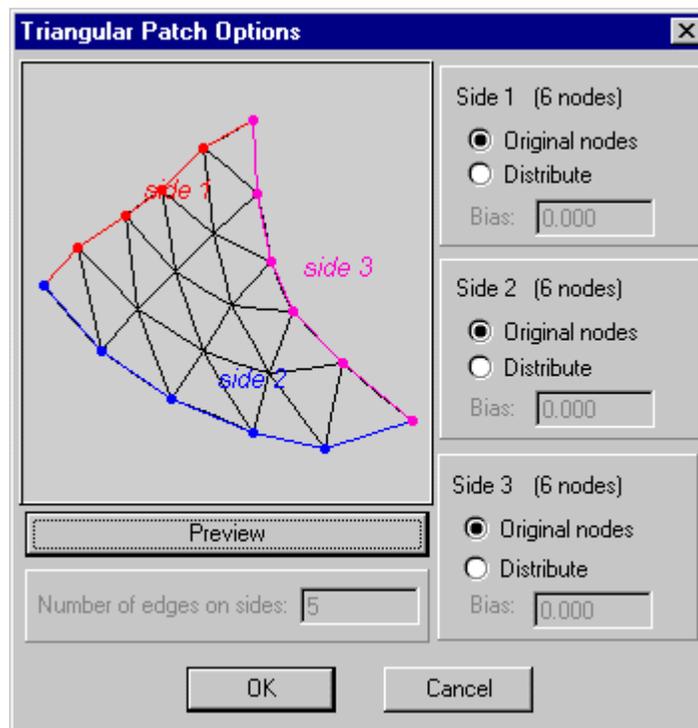


Figure 4.20 Triangular Patch Options dialog.

The coordinates of the new nodes on the interior of the patch are computed by constructing a partial bicubic Coons patch using the nodestrings as patch edges. This ensures that interior nodes are smoothly interpolated from the nodes making up the perimeter of the patch. The following options are available for each edge of the triangular patch:

- ? *Use original nodes.* This option causes the original nodes from the nodestring to be used as corner nodes of elements along the boundary.
- ? *Distribute nodes.* This option distributes the specified number of nodes as corner nodes of elements along the boundary. If elements already exist on the boundary, then this option is unavailable.
- ? *Bias.* This is used with the *Distribute nodes* option. It causes the spacing of nodes along the nodestring to be weighted more to one of the corners.

All three sides of a triangular patch must have the same number of nodes. After the spacing on each side is defined, click the *Preview* button to see how the patch will look. If changes are desired, they can be made. When the patch looks good, click the *OK* button to accept it. The patch can be canceled by clicking the *Cancel* button. Be careful to use the preview button because **THERE IS NO UNDO FOR THIS OPERATION.**

The elements in a new patch are checked to make sure they do not overlap each other. If any problems are detected, an error message is given and the patch is not created. Errors may occur especially when the region is highly irregular in shape. In such cases, the region can either be divided into smaller patches, or it can be filled using a different mesh generation technique.

4.7.6 Select Thin Triangles

During the process of triangulation (see section 4.7.2), a mesh of triangular elements is created around existing nodes. This usually creates triangular elements outside the desired mesh boundary. Many of these exterior triangles are very skinny, and some are virtually invisible. The *Select Thin Triangles* command from the *Elements* menu finds and selects skinny triangular elements which are on the mesh boundary.

Thin triangles interior to the mesh will not be selected when this command is performed, since deletion of interior triangles would result in gaps in the mesh. The criterion to define a skinny element is defined in the *Element Options* dialog (see section 4.7.1). After the thin triangles have been selected, they can be removed by

selecting the *Delete*  macro button.

4.7.7 Find Element

The *Find Element* command from the *Elements* menu is used to locate an element either with a specific ID, or surrounding a specific location. When this command is executed the *Find Element* dialog opens (see Figure 4.12).

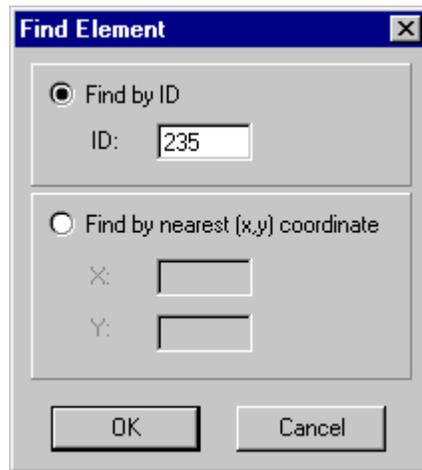


Figure 4.21 Find Element dialog.

When the *Find by ID* option is selected, then the element with the specified *ID* is highlighted in red. If there is no element with the specified id, then an error message is given. Conversely, when the *Find by nearest (x,y) coordinate* option is selected, the element which surrounds to the specified (x, y) location is highlighted in red. With either of these methods, if the current tool is the *Select Elements*  tool, then the found element becomes selected in addition to being highlighted.

4.7.8 Add Breaklines

The *Add Breaklines* command from the *Elements* menu can be executed when at least one nodestring has been selected (see section 4.3.4). This forces element edges along the nodestring line. When this command is performed, elements are sliced along the nodestring to ensure that the edges will conform to the breakline. The elevations of any new nodes are interpolated from the original mesh. All new triangles satisfy the Delauney criterion (see section 4.7.2).

A breakline example is shown in Figure 4.22. This example has some long, skinny quadrilaterals which will be split across the width. The dotted line in the left part of the figure represents the location of the breakline. When the elements are split, triangles are formed. These can be merged together using the *Split/Merge*  tool (see section 4.3.8), as shown in the right part of the figure.

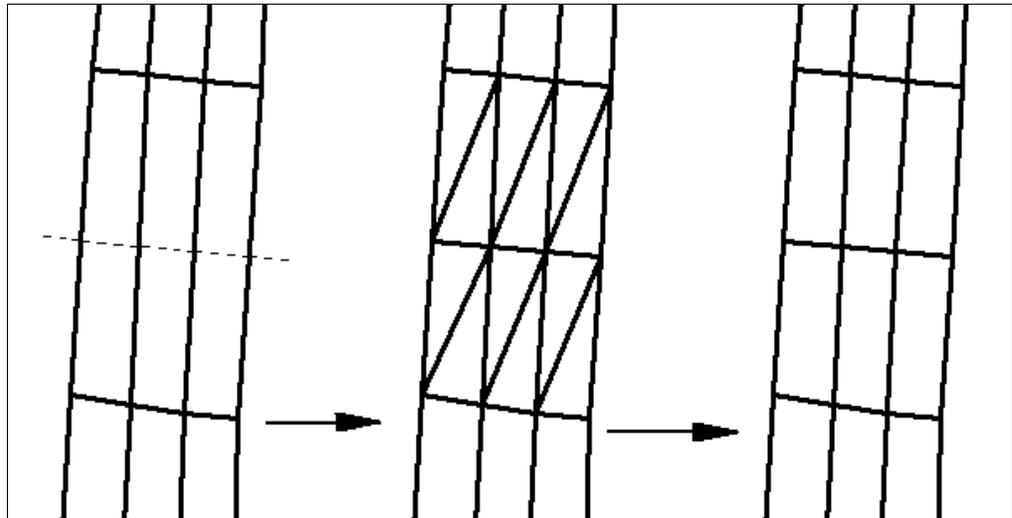


Figure 4.22 Example of a breakline.

4.7.9 Merge Triangles

The merging of individual pairs of triangles using the *Split/Merge*  tool was previously discussed in section 4.3.8. Doing this manually for large numbers of elements takes a lot of time. The *Merge Triangles* command from the *Elements* menu automatically merges a selected set of triangles simultaneously. If no elements are selected when this command is executed, then all triangles in the finite element mesh will be processed.

This command uses the *Merge triangles feature angle* specified in *Element Option* dialog (see section 4.7.1). This angle should be between zero and ninety degrees. Any two adjacent triangles are merged into a quadrilateral if all angles in the resulting quadrilateral are greater than the *Merge triangles feature angle*.

In order to form quadrilateral elements with the best aspect ratios, *SMS* starts with a feature angle of ninety degrees and checks for any elements that can be merged. Then, a series of steps are performed, each time lowering the feature angle and checking for elements that can be merged. This ensures that the quadrilaterals which are formed are as close to rectangular as possible. In general, after the automatic merging process is complete, a limited number of triangles will still exist.

4.7.10 Split Quadrilaterals

The *Split Quadrilaterals* command in the *Elements* menu is used to split a set of quadrilaterals into triangles. If no elements are selected, all quadrilateral elements in the mesh will be split. The quadrilaterals are split along the shortest diagonal.

4.7.11 Quad8 <-> Quad9

The *Quad8<->Quad9* item from the *Elements* menu is used to convert between eight- and nine- noded quadrilaterals. Currently, only *FESWMS* supports nine-noded quadrilaterals, while both *FESWMS* and *TABS* support eight-noded quadrilaterals. If no elements are selected when this command is performed, all elements will be converted.

4.7.12 Linear <-> Quadratic

The *Linear<->Quadratic* item from the *Elements* menu converts the finite element mesh back and forth between linear and quadratic. A finite element mesh must be made of either all linear elements or all quadratic elements. Linear elements do not have midside nodes while quadratic elements do.

4.7.13 Refine

At times, there is not enough definition in a finite element mesh. The *Refine* command from the *Elements* menu splits each of the selected elements into four elements as shown in Figure 4.23. After a section has been refined, SMS automatically creates transitions, from new areas of higher density to old areas of lower density, using triangular elements.

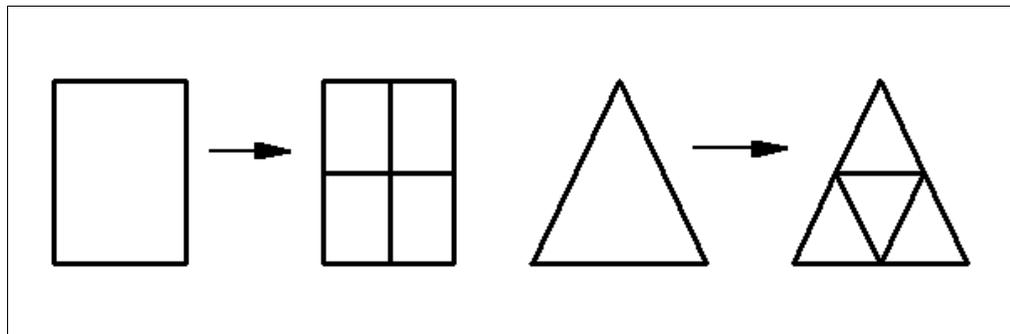


Figure 4.23 Traditional element refinements.

When refining triangular elements, there is a new option that can be used to split a triangle into three elements. This option is described in section 4.7.1.

4.7.14 Relax

During the process of creating and editing a finite element mesh, elements may violate the 50% size difference guideline for adjacent elements (see section 4.5.1). The *Relax* command from the *Elements* menu can improve adjacent element areas by slightly moving nodes. This command moves a node to the centroid of all elements to which it is attached. Since moving nodes changes the element boundaries, relaxation is an

iterative process. The number of iterations performed for a relaxation is specified in the *Element Options* dialog (see section 4.7.1). If no elements are selected, then the relaxation is performed on all elements in the mesh.

Since mesh nodes are being moved during the relaxation process, SMS provides a few options to assure that the desired results occur. These options are accessed using the *Element Options* dialog and described in section 4.7.1.

4.7.15 Smooth Nodestring

As described in section 4.3.5, quadratic elements have a node located at the midpoint of each edge. These nodes are generally referred to as midside nodes. The angular corners resulting from such elements are discontinuous. Such a discontinuity may result in inaccuracy in the numerical model sometimes referred to as a mass loss (see Figure 4.24). Mass loss occurs because water artificially flows out of the mesh.

To minimize the abrupt change in flow direction, element edges can be curved by slightly moving the midside node. This can be done by hand using the *Select Nodes*  tool with the nodes unlocked (see section 4.6.6). Moving large numbers of nodes becomes tedious. However, element edges along a selected nodestring can be smoothed by SMS with the *Smooth Nodestring* command from the *Elements* menu. An example of element edge smoothing is shown in Figure 4.24.

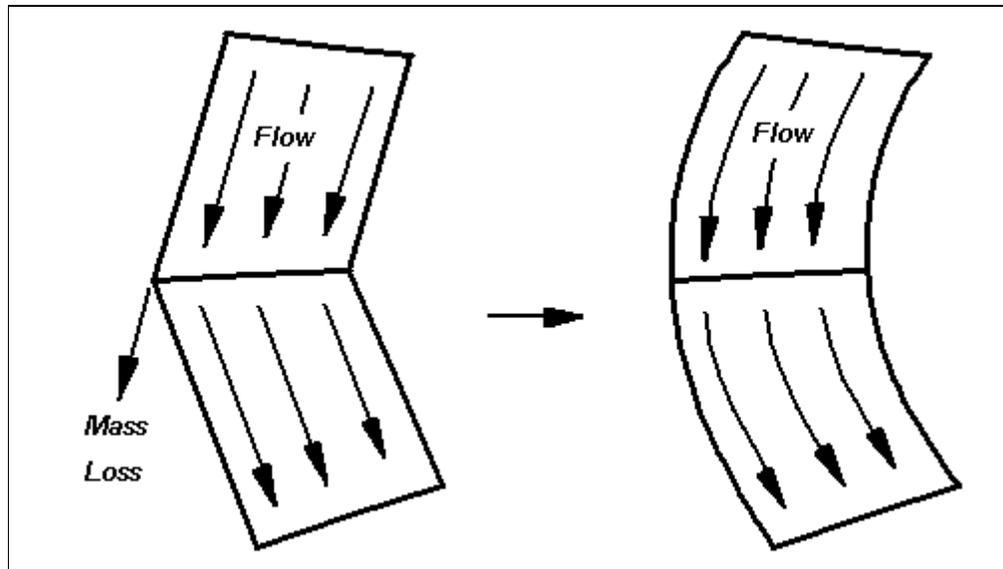


Figure 4.24 Example of the *Smooth Nodestring* command.

Normally, element edge smoothness is only a concern along the mesh boundary. However, if the analysis includes regions that become dry, interior boundaries should also be smoothed. To avoid smoothing corners that should be sharp, SMS provides a *Smooth nodestring feature angle* in the *Element Options* dialog (see section 4.7.1). A corner will only be smoothed if it is less than the specified angle.

4.7.16 Renumber

The nodes and the elements need to be renumbered before an analysis is performed to minimize the *Front Width* and *Half Band Width* in the model. The mesh is renumbered by selecting the *Renumber* command from the *Elements* menu after having selected a nodestring (see section 4.3.4). The nodestring is used as a seed to start the renumbering process.

After invoking the *Renumber* command, the *Renumber* dialog opens for choosing the renumbering method to used (see Figure 4.25). Currently, only two options are provided. The *Band Width* option attempts to minimize the half band width, while the *Front Width* option attempts to minimize the front width.

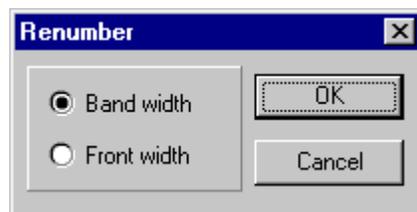


Figure 4.25 *Renumber* dialog.

Upon execution of this command, the nodes and elements are renumbered by a wave that passes through the mesh. This wave proceeds from elements connected to the selected nodestring and continues through the entire mesh. If a section of elements or nodes are disjoint, the wave will not continue and a warning message is given. Disjoint mesh sections are assigned arbitrary numbers, but before running the analysis, there should be no disjoint sections. Nodes which are not connected to any elements can be found and selected automatically (see section 4.6.5).

It is important to realize that after renumbering the finite element mesh, any previous boundary condition file or solution file is no longer valid!

Front Width and Band Width Described

Due to the number of questions that are asked regarding this subject, this section will attempt to describe, in a broad sense, why renumbering is important.

The finite element solvers use an iterative, banded numerical solver to solve the governing differential equations. If the computer had to simultaneously solve the thousands of equations, it would run out of memory. This is why it uses a banded solver. The front width and the half-band width determine the size of the matrix which is used by the finite element solvers. A smaller front width and band width lead to a smaller required matrix. The front width and band width depend on the node and element numbering of the finite element mesh. To minimize the front width and band width, the mesh should be renumbered.

SMS provides an estimate as to how large the front width and half band width may become when running the finite element solver. These estimates are shown in the *Mesh Information* dialog, which can be opened by performing the *File / Get Info* command while in the *Mesh Module*. There is not a single renumbering scheme which always produces the lowest values for these, so it is suggested that you renumber from various locations in the mesh and check the mesh information before deciding which renumbering scheme to use. As a guideline, the mesh should be renumbered using a nodestring which extends across a small section of the mesh, such as a river inflow boundary.

4.7.17 Assign Material Type

Each element in the mesh is assigned a material type. The default material ID can be set in the *Element Options* dialog (see section 4.7.1). A selected element is assigned a new material type by choosing the *Assign Material Type* command from the *Elements* menu. If the *Assign default material* option is selected in the *Element Options* dialog, then the default material is automatically assigned to the selected element. If the *Prompt for material when assigning* option is selected in the *Element Options* dialog, then the *Materials Data* dialog opens from which a material type can be chosen.

4.7.18 Assign Control Structure

When using a *TABS* model, special elements called *Control Structures* can be created (see section 5.10.4). When a control structure is selected, the *Assign Control Structure* command from the *Elements* menu becomes available. This command opens the *RMA2 Control Element* dialog (see Figure 4.26).

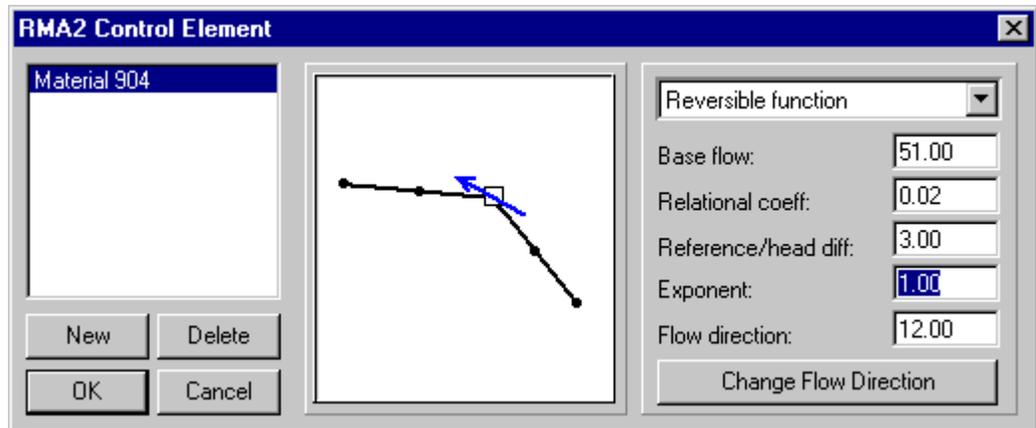


Figure 4.26 RMA2 Control Element dialog for defining the control structure.

This dialog shows a preview of the flow direction as well as the parameters for the control structure. The flow direction can be changed by clicking the *Change Flow Direction* button.

4.7.19 Assign Junction Type

When using a *TABS* model, special elements called *Junction Elements* can be created (see section 5.10.3). When a junction element is selected, the *Assign Junction Type* command from the *Elements* menu becomes available. This command opens the *RMA2 Junction Element* dialog (see Figure 4.27).

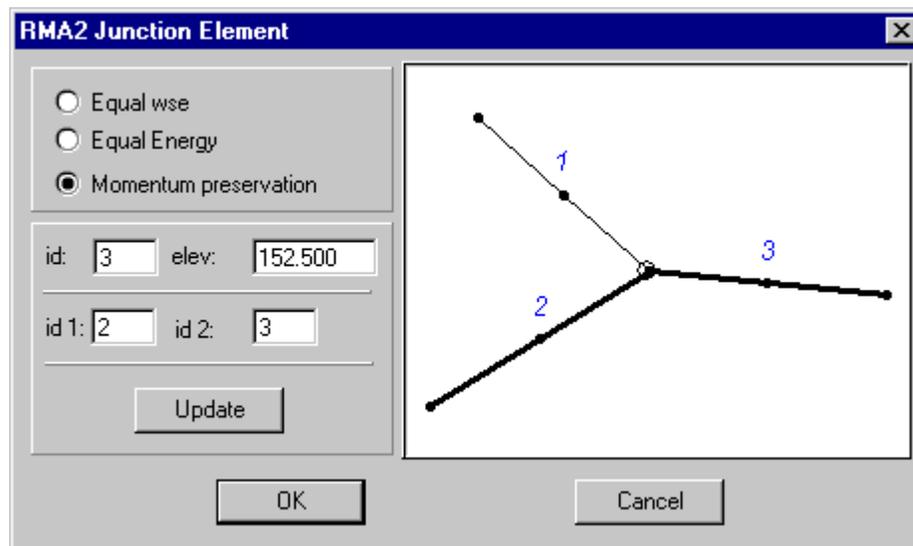


Figure 4.27 RMA2 Junction Element dialog for defining the junction element.

This dialog shows the three types of junction elements that can be created, *Equal wse* (water surface elevation), *Equal energy*, and *Momentum preservation*. When the *Momentum preservation* option is selected, two of the elements must be defined as the primary channel.

4.8 Model Menus

In the current version of *SMS*, the supported 2D finite element analysis engines are *TABS*, *FESWMS*, *HIVEL2D*, *RMA10*, *CGWAVE*, and *ADCIRC*. Anything specific to a numerical model is accessed through the appropriate model menu. These model specific attributes include the material attributes, boundary conditions, and model run control. For details on a specific model, refer to the chapter in this manual which discusses the particular model interface.

RMA2 Interface

RMA2 is a hydrodynamic modeling code that supports subcritical flow analysis, including wetting and drying and marsh porosity models. It is part of the TABS analysis package written by the U.S. Army Corps of Engineers Waterways Experiment Station (USACE-WES). The methods of analysis used by the TABS codes along with their file formats and input parameters are described in their own documents. *SMS* supports both pre- and post-processing for *RMA2*.

A mesh for use with *RMA2* is created and edited in *SMS* using the *Mesh Module*. The modeling parameters required by *RMA2* are generated and applied to the mesh using commands grouped in the *RMA2* menu. These commands are described in this chapter. Post-processing is generic, once the solution file for *RMA2* has been imported into the *Data Browser*. See section 3.1 for importing the solution file into the *Data Browser* and Chapter 3 for post-processing techniques.

5.1 Open Simulation

The *Open Simulation* command in the *RMA2* menu reads in a simulation that has been previously created and saved. The simulation file written previously has the “.sim” extension. The simulation file contains a list of file names which will be used to run the finite element model. When an *RMA2* simulation file is used, both the geometry and boundary condition files will be opened together. Opening a new simulation causes any existing simulation to be removed from memory but it does not remove old data from the disk.

Geometry files typically have the file extension “.geo”. The name of an open geometry file is displayed at the top of the *Graphics Window*. The geometry file stores the location of nodes that define element corners and the element connectivity of the nodes. Midside nodes are only required in the geometry file at edges that are curved. See the *GFGEN* documentation for the geometry file format.

Boundary condition files typically have the file extension “.bc”. The name an open boundary condition file is displayed at the top of the *Graphics Window* next to the name of the geometry file. The boundary condition file stores inflow and outflow boundaries, run control parameters, material properties, and various other model-specific information. See the *RMA2* documentation for the boundary condition file format.

A single geometry or boundary condition file may be opened using the *Open Simulation* command in the *RMA2* menu. If a simulation file is not chosen from the file browser, the dialog shown in Figure 5.1 will open. The correct file type should be chosen and *SMS* will attempt to read the file.

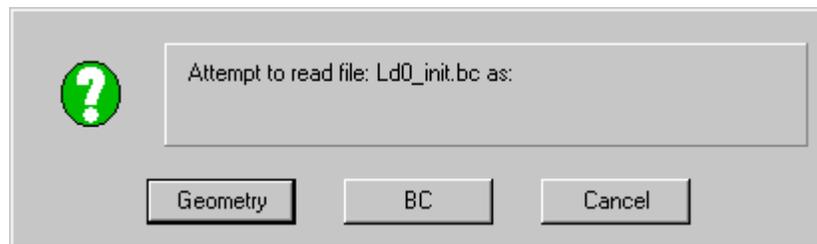


Figure 5.1 Choose to read a single RMA2 geometry or boundary condition file.

5.2 Save Simulation

The *Save Simulation* command in the *RMA2* menu invokes the dialog shown in Figure 5.2. In this dialog, you can choose to save the geometry and/or the boundary condition data. Generally, the entire simulation would be saved and a single prefix would be used for all file names. To do this:

1. Choose *RMA2 / Save Simulation*.
2. Enter the desired simulation name in the *Simulation file* edit field.
3. Click the *Update* button.
4. Click the *OK* button.

After saving a simulation, *GFGEN* and *RMA2* can be run directly from inside *SMS* as described in section 5.11.

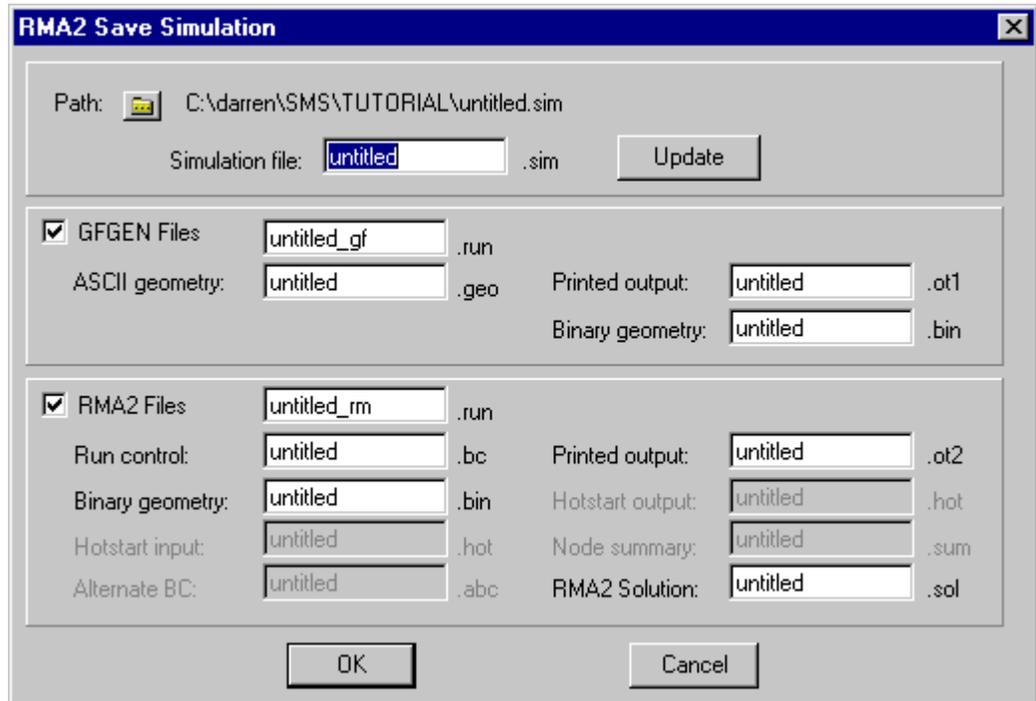


Figure 5.2 RMA2 Save Simulation dialog.

5.3 Assign BC

The *Assign BC* command is used to assign boundary conditions to nodestrings or nodes, depending on which is selected. It is generally advisable to assign boundary conditions only to nodestrings for stability considerations.

5.3.1 Assign Boundary Conditions to Nodes

The *Nodal Boundary Conditions* dialog (see Figure 5.3) may be used to assign boundary conditions to individual nodes. Before assigning boundary conditions to nodes, at least one boundary node must be selected using the *Select Nodes* tool. All selected nodes must be on the mesh boundary. Attempting to assign nodal boundary conditions to an interior node will result in an error message. All selected nodes are assigned identical boundary conditions. If different boundary conditions are desired for different nodes each node should be selected and assigned separately.

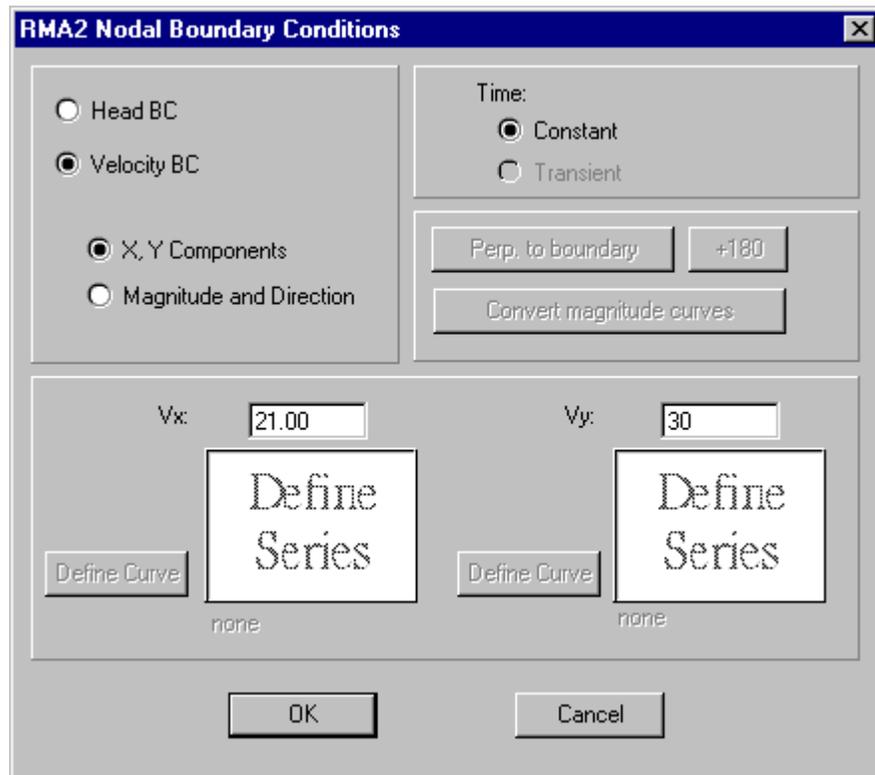


Figure 5.3 RMA2 Nodal Boundary Conditions Dialog.

If a dynamic simulation is requested (see section 5.8.4) either constant or transient boundary conditions can be created. If a steady state simulation is requested, only constant boundary conditions can be created. For either steady state or transient boundary conditions, the velocity is entered as either a combination of magnitude and direction or as x- and y- components.

To assign constant boundary conditions to the selected node(s):

1. Select the *Constant* option in the *Time* section.
2. Choose either a *Head* or *Velocity* boundary condition type.
3. Enter the required values in the fields available.

To assign transient boundary conditions to the selected node(s):

1. Select the *Transient* option in the *Time* section.
2. Choose either a *Head* or *Velocity* boundary condition type.
3. Click the desired *Define Curve* button and select an existing curve or create a new one. For help on creating time series curves, see Chapter 16.

5.3.2 Assigning Boundary Conditions to Nodestrings

The *Nodestring Boundary Conditions* dialog (see Figure 5.4) may be used to assign boundary conditions to individual nodestrings. Before assigning boundary conditions to nodestrings, at least one nodestring must be selected using the *Select Nodestrings* tool. All selected strings must be on the mesh boundary. Attempting to assign a boundary conditions to nodestring not on the mesh boundary will result in an error message.

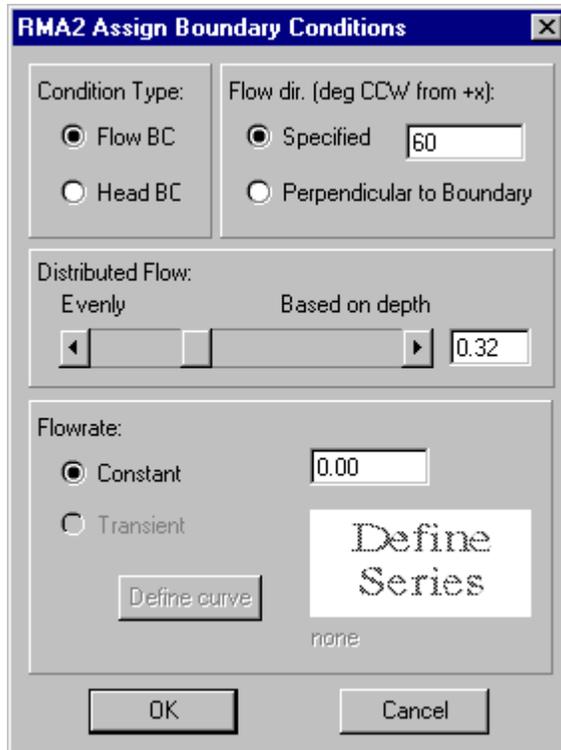


Figure 5.4 RMA2 Nodestring Boundary Conditions Dialog.

If a dynamic simulation is requested (see section 5.8.4) either constant or transient boundary conditions can be created. If a steady state simulation is requested, only constant boundary conditions can be created. For either steady state or transient boundary conditions, the velocity is entered as either a combination of magnitude and direction or as x- and y- components.

To assign constant boundary conditions to the selected nodestring(s):

1. Select the *Constant* option in the *Time* section.
2. Choose either a *Head* or *Flow* boundary condition type.
3. Enter the required values in the fields available.

To assign transient boundary conditions to the selected nodestring(s):

1. Select the *Transient* option in the *Time* section.
2. Choose either a *Head* or *Flow* boundary condition type.
3. Click the desired *Define Curve* button and select an existing curve or create a new one. For help on creating time series curves, see Chapter 16.

When assigning flow boundary conditions, two values are specified in addition to the flow rate. First, the flow direction is specified as the angle in degrees counter-clockwise from the positive x-axis to which the water is flowing (i.e. an angle of zero degrees means the water is flowing in the positive x-direction). *SMS* can automatically compute the angle perpendicular to the boundary. The second value specified when using flow boundary conditions is the flow distribution. A value of zero will distribute an even amount of water to each node on the boundary, while a value of one will distribute more flow to nodes that have deeper water.

5.4 Delete BC

The *Delete BC* command will delete any boundary conditions previously assigned to selected nodes or nodestrings. If neither nodes nor nodestrings are selected, boundary conditions assigned to all nodes and/or nodestrings can be deleted. If a boundary condition is deleted from a nodestring, the nodestring will not be deleted.

5.5 Add GC String

Any selected nodestring(s) can be defined as a geometry continuity (GC) string by selecting the nodestring and invoking the *Add GC String* command from the *RMA2* menu. When *RMA2* runs, a table will be printed in the output file (.ot2) which gives the flux through each GC string. Nodestrings which have defined head or flow boundary conditions are automatically defined as GC strings, so this command is only used for interior nodestrings to test continuity. It is very important that GC strings cross the mesh in the same direction or the flow rate results will not be reported correctly. **Create all GC strings (including boundary condition nodestrings) from right to left when facing downstream.**

The flux for each GC string is reported as a percentage of the total flow. The total flow for a specific simulation is defined as the flow through the very first GC string written to the boundary condition file. **Inside SMS, the very last nodestring that was created is considered to be the total flow nodestring.** If only one inflow exists, create the inflow nodestring last. If more than one inflow exists, but only one outflow exists, create the outflow last. It is very important that the total flow nodestring include all the flow that is in the mesh or the flow percentages reported by *RMA2* will not be correct.

5.6 Material Properties

The *RMA2 materials* menu item allows you to assign material properties to any defined material. The dialog that opens when this command is invoked is shown in Figure 5.5.

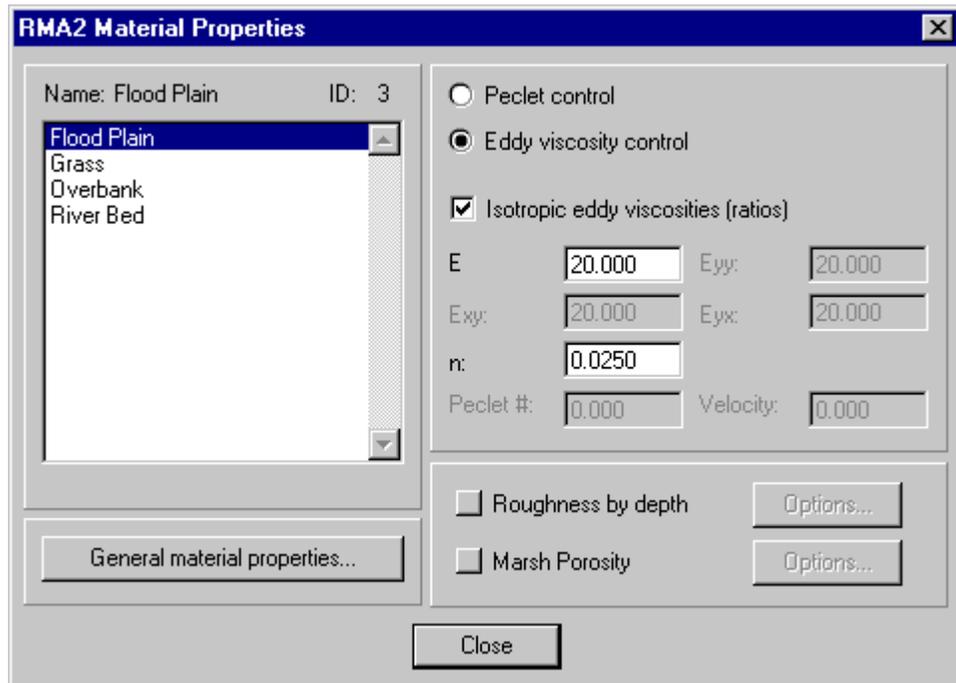


Figure 5.5 RMA2 Material Properties Dialog.

This dialog shows all available materials and their properties. To add or delete materials from the list, or to change the name or id of a specific material, click the *General Material Properties* button. See section 2.5.2 for information on the *General Materials* dialog.

5.6.1 Material Roughness

One of the material properties specified for *RMA2* is the material roughness. Generally, the roughness through each element will be computed using *Manning's* equation. To be able to do this, *Manning's n* must be specified by the user. Manning's *n*-value is always less than unity. If a value greater than one is entered, *RMA2* will use the *Chezy* equation to calculate roughness instead of Manning's equation. No guidelines for roughness values will be given here because they vary widely and can be found in several hydraulics reference books.

5.6.2 Roughness by Depth

At times, it is desirable to define lower roughness values for deep water than for shallow water. *RMA2* allows a depth vs. roughness curve to be created for each material if desired. When the roughness by depth Options button is pressed, the dialog shown in Figure 5.6 appears.

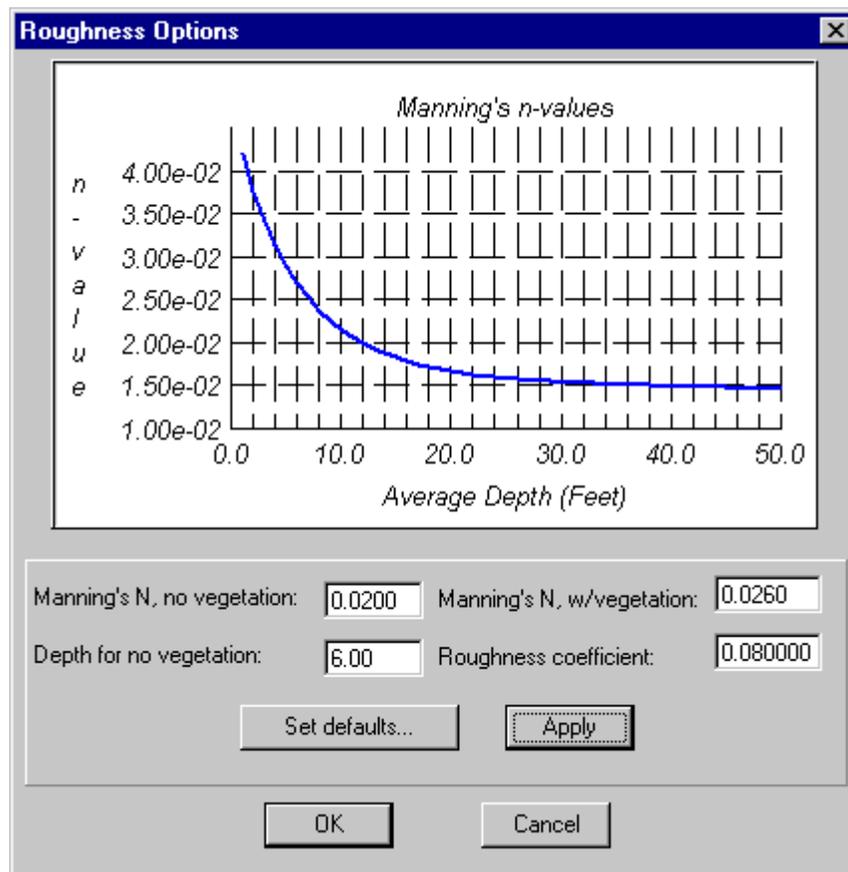


Figure 5.6 The *RMA2* Roughness Options dialog.

Four values are required to define the roughness curve: Manning's roughness where no vegetation exists, the depth for no vegetation, the Manning's roughness *increase* for vegetated water, and a roughness coefficient. The Depth for no vegetation value is how deep the water must be before vegetation no longer affects Manning's roughness. The exact equation used to generate the curve from these values is available in the *RMA2* documentation. To update the curve drawn in *SMS* after entering values in the available fields, click the *Apply* button.

5.6.3 Eddy Viscosity

Another material property specified for *RMA2* is the eddy viscosity. This is used for helping with convergence as well as the calibration effort. The eddy viscosity is

defined in $\text{lb-sec}/\text{ft}^2$ for English units and Pascal-sec for SI units. The following four eddy viscosity values are required:

- x-momentum turbulent exchange in the x direction (Exx).
- x-momentum turbulent exchange in the y direction (Exy).
- y-momentum turbulent exchange in the x direction (Eyx).
- y-momentum turbulent exchange in the y direction (Eyy).

Unless good data is available in the area being modeled, all these eddy viscosity directions should be defined to have the same value. This can be done easily with SMS by selecting the *Isotropic Eddy Viscosity* option. When this option is selected, only one eddy viscosity value needs to be entered and the value will be applied to all directions.

True viscosity of water is on the order of 10^{-5} for English and 10^{-3} for SI units. However, the eddy viscosity defined in the numerical model is on a much larger scale. Table 5-1 was taken from the documentation provided by USACE-WES giving suggested eddy viscosity values. These are suggested values only, and they may vary widely from project to project. It is generally wise to start with the default values and change them as needed for convergence and/or calibration.

Table 5-1 Sample eddy viscosities.

Type of Problem	E, lb-sec/ft ²	E, Pascal-sec
Homogenous horizontal flow around an island	10-100	500-5000
Homogenous horizontal flow at a confluence	25-100	1100-5000
Steady-state flow for thermal discharge to a slow moving river	20-1000	1000-50000
Tidal flow in a marshy estuary	50-200	2500-10000
Slow flow through a shallow pond	0.2-1.0	10-50

As an alternate method of defining the eddy viscosity, the *Peclet Control* option can be used. The peclet method applies an eddy viscosity to each element based on its size, the velocity of water through it and the water density. Using this option, the eddy viscosity is constantly being adjusted. To use the peclet method, a peclet number must be specified as well as a scale factor for the xx, xy, yx, and yy directions. The scale factors should all be the same unless specific data proves otherwise. A lower peclet number will cause a higher computed eddy viscosity, but it should always be between 15 and 40. The exact formula for computing eddy viscosity from the peclet number can be found in the RMA2 documentation.

5.6.4 Marsh Porosity

Marsh porosity values are defined globally as described in section 5.9.4. The global values can be overridden for specific material types. To do this, highlight the material, turn on the *Marsh porosity* check box, and click the *Options* button.

When writing out the marsh porosity values for materials, the order written is often important. SMS will always write the values for materials in sequential order, starting with material one up to the number of materials that are defined.

5.7 Model Check

A *Model Check* should be performed on all *RMA2* models before attempting an analysis. The model check will perform a basic check to insure that all of the needed information to run the analysis is present. The *Model Check* command in the *RMA2* menu causes the *Model Check* dialog (see Figure 5.7) to appear.

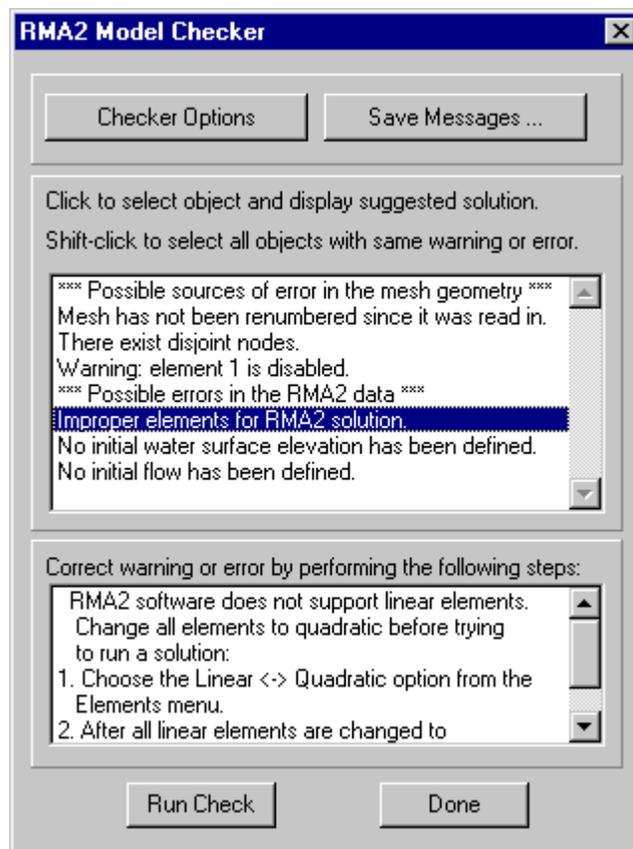


Figure 5.7 RMA2 Model Checker Dialog.

Select the *Checker Options* button to open the *RMA2 Model Checking Options* dialog. This dialog lists the checks that may be performed during the model checking procedure. By default all checks are enabled. The checks include:

- *Check Water Surface Elevation.* The initial water surface elevation warning is given if the defined head value is lower than the highest nodal elevation.
- *Check Boundary Conditions.* Boundary conditions should consist of at least a head boundary condition (assigned to a node or a string). Flow and velocity boundary conditions can also be defined (see section 5.3).
- *Mesh Check Options.* Generic model checking options include: renumbered mesh, material definition, duplicate nodes, and small interior voids. In this dialog, the number of reported messages of a single type can be limited. See section 4.5.1 for more information on the general mesh check options.
- Other miscellaneous checking options are performed such as a check for linear elements and 1D geometry data.

Click the *Run Check* button to perform the model check. After *SMS* has completed the model check, error messages will appear in the top text window. When the user clicks on a specific error message, the lower text window shows a set of instructions on how to fix the error. To save the error messages to a text log file, click the *Save Messages* button and choose a filename. To close the *Model Checker*, click the *Done* button.

5.8 Model Control

The *RMA2* numerical model requires several parameters to control the analysis such as water temperature, time definition, and model units. This data is stored in the boundary condition file. The *RMA2* analysis model supports about 50 different data records. Some of these are very specialized and rarely used, so *SMS* does not support them. A complete list of all records available for geometry and boundary condition files is contained in the *RMA2* documentation.

The *Model Control* command from the *RMA2* menu opens the *RMA2 Model Control* dialog (see Figure 5.8). This dialog contains parameters which control the execution of *RMA2*. The parameter description for a each field is displayed in *SMS* using the interactive help messages. The rest of this section explains the parameters available in this dialog.

Figure 5.8 RMA2 Model Control dialog.

Note: An unsupported record must be added to the boundary condition file after SMS has written it. As a boundary condition file is read by SMS, unsupported records are saved internally and connected to their time step. These records are then written to the boundary condition file in the appropriate time step when SMS outputs a boundary condition file.

5.8.1 Job Title

The *Job Title* section allows the user to give up to three titles to and make comments about the problem being modeled. These fields are stored in the *T1*, *T2*, and *T3* records.

5.8.2 File Control

The *Files* section allows the user to control what file I/O will occur during the analysis. The following files are available:

- ? *Hotstart input file.* Turn this on to tell *RMA2* to read initial conditions from a hotstart file. The hotstart file will usually have been written by a previous *RMA2* simulation. The hotstart file contains the water surface elevation and velocity components for each node, as well as the derivative of those. This data is associated with a specific time step. The default for this option is off.
- ? *Hotstart output file.* Turn this on to tell *RMA2* to write out a hotstart file. This should be done whenever a run may need to be continued, such as when spinning down a steady state model using *REV* cards. The hotstart file will be written after each time step. The default for this option is off.
- ? *GFGEN geometric data.* This option should always be turned on. It tells *RMA2* that the nodes and element connectivities are stored in a separate geometry file. Unexpected results may occur if this is turned off.
- ? *Save RMA2 results.* This option tells *RMA2* to save the computed results to a file after each time step has been computed. The only time that it is advisable to not save a simulation is if you are running an intermediary solution to get a good hotstart file. However, even in this situation, it is advisable to create a solution file so that the simulation progress can be monitored by reading the solution file into *SMS* for post-processing. The default for this option is on.
- ? *Alternate dynamic BC file.* This option tells *RMA2* to read a separate file which contains the dynamic boundary conditions. It is not advisable to use this method, as the alternate file must be created manually and it cannot be read by *SMS*. The default for this option is off.
- ? *Full print output.* This option tells *RMA2* to save various data to an ASCII file as it runs. Such data includes model input, convergence data, continuity (see section 5.5), and debug information. These options are set in the *Optional BC Controls* dialog (see section 5.9.7). The default for this option is on.
- ? *Summary by node.* This option tells *RMA2* to save a special ascii file for writing detailed summary information at specific nodes. If this option is chosen, the user must manually specify at which nodes the detailed summary information is desired. This can be done using a text editor such as *notepad*.

Depending on which file options are turned on, some fields in the *Save Simulation* dialog (see section 5.2) will be dim. These file options are stored in the *\$L* record.

5.8.3 Iteration Control

The *Iterations* section controls the number of iterations performed at each time step and convergence criteria *RMA2* uses at each iteration of the analysis. At each time step, *RMA2* performs up to the specified number of iterations with the finite element solver to find a solution. It also keeps track of the change in water depth for each node

between two iterations. *RMA2* will continue to iterate on a solution until one of the following conditions is satisfied:

- ? The specified number of iterations are performed.
- ? The maximum change in water depth at any node between two iterations is less than the convergence tolerance specified.
- ? The maximum change in water depth at any node between two iterations exceeds 25.0.

In the first two cases, the time step is considered to be converged and *RMA2* moves on to solving the next time step. In the third case, the model is on a divergent path, so an error message is given and execution is terminated.

A steady state simulation and time step zero of a dynamic simulation are controlled by the number of *Initial solution* iterations and the *Steady state depth convergence*. All other time steps are controlled by the number of *Each time step* iterations and the *Dynamic depth convergence*. The iteration data is stored in the *TI* record.

5.8.4 Computation Time Control

The *Computation time* section allows the user to specify the simulation run time. For a steady state simulation, these values are unused. For a dynamic simulation, three of four values are specified. These values available include the following:

- ? *Total time*. This is the total simulation time, in decimal hours.
- ? *Time step*. This is the length of time between time steps, in decimal hours.
- ? *Number of time steps*. This is the number of time steps that will be performed during the analysis. It does not include the initial solution. For example, to model 12 hours at half-hour time steps, set this to 24.
- ? *First time step*. This is the number of time steps to skip when continuing an analysis using a hotstart file. If no hotstart file is being used, or the hotstart file came from a steady state simulation, this value should be set to zero.

Only two of the first three values are required because the other will be computed. For example, using 24 half-hour time steps, the computation time is computed to be 12 hours. The time control data is stored in the *TZ* record.

5.8.5 Units Control

The *Units* section allows the user to control the type of units used during the analysis as well as feedback shown on the screen. When this value is changed, no automatic

conversion is performed on the nodal locations. It is simply a flag to tell *RMA2* in which units the data is specified. This is stored in the SI record.

5.8.6 Other Options Control

The *Other Options* section allows the user to control the fluid temperature (*FT* record) in degrees Celsius. It also includes the *Optional BC Controls* button which causes the *Options BC Controls* dialog to appear.

5.8.7 Solution Type

The *Solution Type* section specifies whether a steady state or dynamic simulation will be performed. If a steady state simulation is chosen, transient boundary conditions (section 5.3) and time data (section 5.8.4) cannot be defined.

5.9 Optional BC Controls

The *Optional BC Controls* dialog (Figure 5.9), allows the user to set the most commonly used optional global parameters. Since these values are optional, a check box is provided in the upper left corner of each section. If the box is not checked, data for that record is not required.

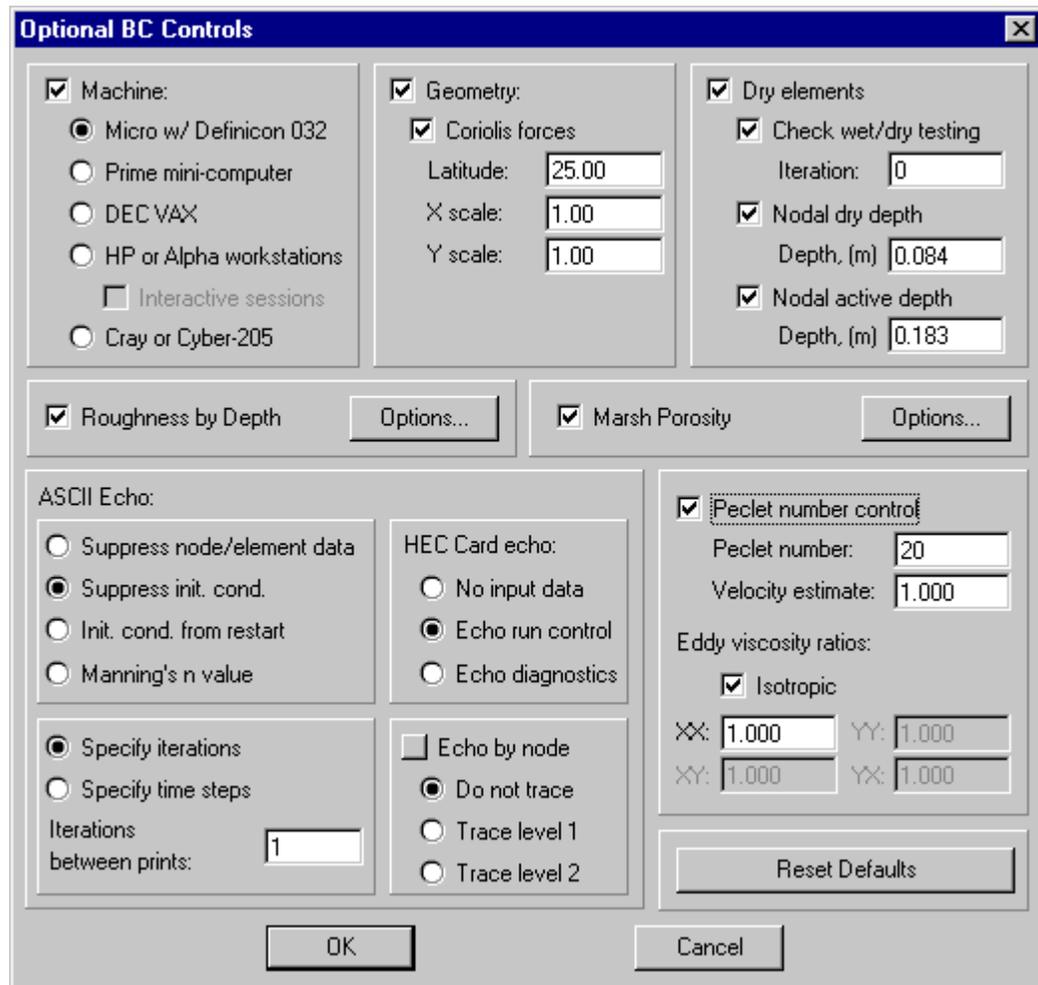


Figure 5.9 Optional Boundary Condition Control Dialog.

5.9.1 Machine

The *Machine* section allows the user to specify what machine type will be used to perform the numerical analysis. The machine type is used to define the *FORTRAN* word size used in writing temporary matrix files. If not correct, problems can result. If this option is not turned on, the *Alpha/HP workstation* is selected by *RMA2* and when the model runs, the following warning message is given:

```
IVRSID=          0 IS NOT PERMITTED ... BEWARE!
--> Auto setting  IVRSID =4 as a guess ...
```

For running on a *PC*, set this to the first value, *Micro w/ Definicon 032*. This value is stored in the *\$M* record.

5.9.2 Geometry

The *Geometry* section allows the user to specify global geometric modifications to the geometry. The following modifiers are available:

- ? *Coriolis forces*. This option tells *RMA2* to take into account the earth's rotation when running the analysis. This option is needed only when there is a wide water surface in the area of interest which can cause large circulation zones, such as a shallow lake. This value is stored in the *LA* record.
- ? *Scale factors*. If specified to be something other than unity, *RMA2* will multiply these scale factors by each x- and y- nodal value before performing any computations. Note that *SMS* is capable of performing these and other operations, so these scale factors are generally not used (see section 4.6.7). These values are stored in the *GS* record.

5.9.3 Dry Elements

The *Dry Elements* section must be turned on if it is anticipated that elements might dry out during the analysis. If this is not turned on and elements do become dry, *RMA2* may crash. When a node becomes dry, all elements attached to that node also become dry. The following values are specified:

- ? *Nodal dry depth*. This value is used to determine when a node goes dry. If the water depth at any wet node gets to be less than this value, then the node becomes dry. A positive value will cause the node to become dry before the water surface reaches the node. A negative value will cause the node to become dry after the water surface is below the node. The default value is 0.275 feet or 0.084 meters.
- ? *Nodal active depth*. This value is used to determine when a node gets wet. If the water depth at any dry node gets to be above this value, then the node becomes wet. The default value is 0.60 feet or 0.183 meters.
- ? *Check wet/dry testing*. This value specifies how many iterations to perform between testing nodes for a change in their wet/dry status. It is important to allow *RMA2* to perform some iterations between wet/dry checks otherwise rapid divergence can occur. **The suggested minimum value is 4.** It is a good idea to set the *Number of iterations* (section 5.8.3) to be two to three times this value to allow for convergence of the wet/dry process. A final wet/dry test is performed at the end of each time step.

It is very important that the *Nodal active depth* value be greater than the *Nodal dry depth* value. This gives a transition zone so that *RMA2* does not rewet a node immediately after it dries out. If no transition zone were provided, an infinite wet/dry loop could occur. These values are stored in the *DE* record.

5.9.4 Marsh Porosity

An alternate method of wet/dry testing is the marsh porosity method. In marsh-type applications, the water surface elevation can be at or just below the bathymetric elevation, while still conducting a significant amount of flow. The marsh porosity option allows elements that would otherwise be considered dry to still conduct a percentage of flow through them.

Three values are specified for the marsh porosity option. These values are shown in Figure 5.10 and will be explained in this section of the document. The *Marsh Porosity* section of the *Optional BC Controls* dialog allows for specification of global marsh porosity values. After these are specified, marsh porosity data can be defined at the material level (see section 5.6.4).

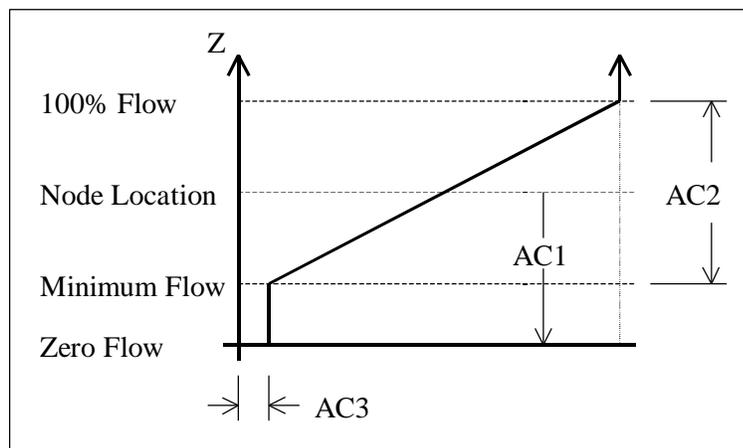


Figure 5.10 Definition of marsh porosity values.

The *Node Location* in the figure corresponds with the nodal z value at a given corner node in the finite element mesh. The following values define marsh porosity:

- ? *AC1*. This is the distance below the node for which zero flow occurs. In other words, the node is not considered dry until the water surface gets *AC1* below the nodal z value. This is the *Zero Flow* elevation.
- ? *AC2*. This defines a transition range during which flow through the node is reduced. If the water surface is $\frac{1}{2}AC2$ above the node elevation value, then the node is considered fully wet, otherwise, it is only partially wet. The transition range is between the *100% Flow* and *Minimum Flow* elevations.
- ? *AC3*. This is the minimum percentage of flow transmitted through the node until it becomes fully dry. This percentage of flow is used when the water surface is between the *Minimum Flow* and *Zero Flow* elevations.

These global marsh porosity values are stored in the *DM* record.

5.9.5 Roughness By Depth

RMA2 allows for global roughness by depth assignment. In general, the roughness will only be assign at the material level. However, if all materials have the same roughness values, then it can be assigned globally. (The global values defined can be overwritten for a specific material by defining these parameters for it.) See section 5.6.2 for information on the roughness by depth definition and assigning these parameters to materials. The global roughness by depth values are stored in the *RD* record.

5.9.6 Peclet Number Control

RMA2 also allows the user to specify global *peclet number control*. This acts just like the global *Roughness By Depth* control defined in section 5.9.5. The global peclet control numbers are stored in the *PE* record.

5.9.7 Echo Control

The ASCII echo region controls the printed output generated during an *RMA2* analysis. The user can specify the desired message suppression type, the card echoing, how frequently a print should be made, and whether nodes should be traced. These options are stored in the *TR* record.

5.10 Special Elements

RMA2 supports certain types of special elements integrated in the 2D model. A detailed definition of these elements is found in the *GFGEN* documentation. A brief discussion of the types of 1D elements and their parameters is defined here.

5.10.1 One-Dimensional Line Elements

Line elements are made of three nodes, two corner nodes and one midside node. These elements can either be curved or straight. There are two uses for these types of elements. First, they are used to define narrow channels which enter the 2D model. To define the flow or head at such a channel, create a *nodestring* that only has one node in it. Define the boundary conditions at this *nodestring*. An example of line elements used in this way is shown in Figure 5.11 on the left side. The second use for line elements is to model a flow control structure in the middle of the mesh. An example of this is also shown in Figure 5.11 on the right side.

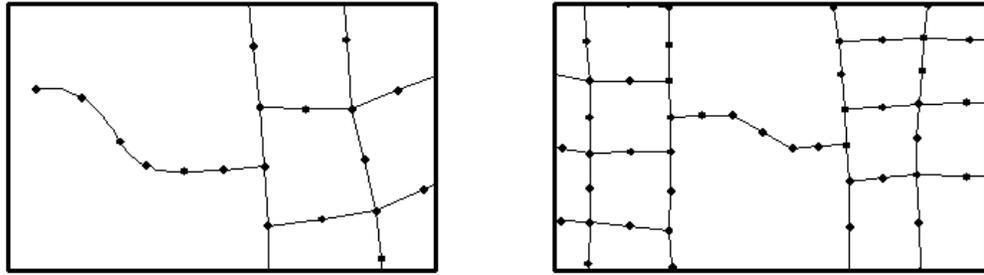


Figure 5.11 One-dimensional line elements inside SMS.

The 1D channel must be defined for these 1D elements. For information on doing this, see section 5.10.5.

5.10.2 Transition Elements

A transition element is a special type of line element. It looks like a “T” and is used to combine the 1D and 2D portions of the finite element network. **The 1D part of the transition element must be perpendicular to the 2D element that it connects.** To create a transition element inside *SMS*, simply create a line element between the midside node of an existing 2D element and the corner node on a 1D element. Figure 5.12 shows the correct use of transition elements in two situations. Boundary conditions should never be assigned to transition elements (as shown in the left part of the figure) and two transition elements should never touch each other (as shown in the right part of the figure).

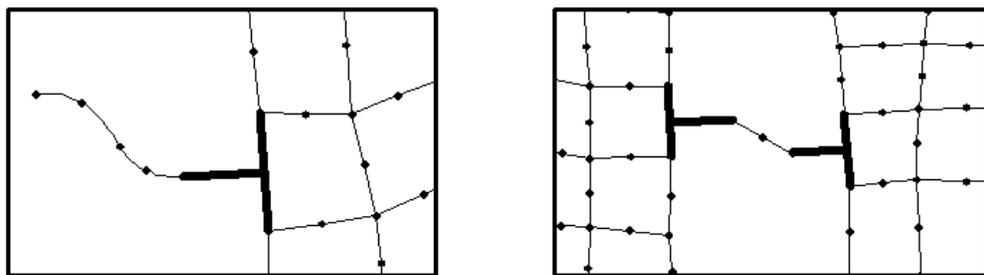


Figure 5.12 RMA2 Transition elements inside SMS.

5.10.3 Junction Elements

A junction element must be created when three or more 1D line elements join at a single location. *SMS* automatically creates and deletes junction elements as required. These junction elements are shown in *SMS* as a circle. An example of a junction element is shown in the left side of Figure 5.13. At a junction element, there is actually one node for each of the elements that comes into the junction. This can be seen in the right part of the figure, which shows the nodes separated.

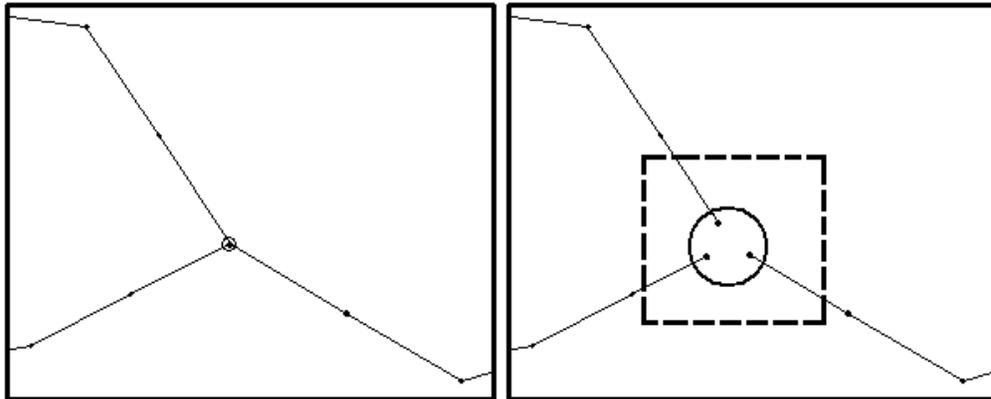


Figure 5.13 RMA2 Junction elements inside SMS.

A junction element must be defined as one of the following:

- ? Water surface elevation is the same at each of the nodes that converge to define the junction.
- ? Total energy head is the same at each of the nodes that converge to define the junction.
- ? Momentum is conserved in the primary channel of the junction. The primary channel is defined by two of the elements which converge at the junction. The water surface of nodes at the junction which are not part of the primary channel are all set to the average water surface elevation of the two nodes that do make up the primary channel.

To define the junction type, select the junction and choose *Elements / Assign Junction Type*. See section 4.7.19 for more information.

5.10.4 Control Structures

A control structure can optionally be defined where two 1D Line elements join. To create a control structure:

1. Select the Create Control Structure  tool from the toolbox.
2. Click on a node that joins two 1D Line elements.

There is no limit to the number of control structures that can be defined. Two examples of control structures are shown in Figure 5.14. The left part of the figure shows a control structure as part of a 1D element network while the right part of the figure shows a control structure between two sections of a 2D finite element mesh. In this second case, notice that there is first a transition element (labeled 1), then a 1D line element (labeled 2) before reaching the control structure.

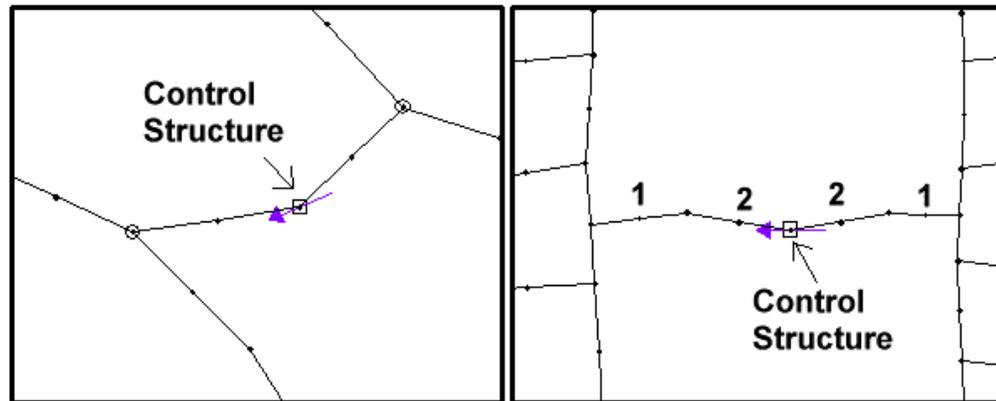


Figure 5.14 RMA2 Control Structures inside SMS.

When *SMS* draws a control structure, it shows up as a small square. The arrow shows the flow direction that *RMA2* expects. To change the properties of a control structure, including the flow direction, select the control structure and choose *Elements / Assign Control Structure*. See section 4.7.18 for more information. As with junction elements, one node is created for each element connected to the structure.

5.10.5 One-Dimensional Geometry

The channel for nodes in 1D elements must be defined so that *RMA2* can calculate the velocities and water surface elevations at those nodes. This data must be defined at all corner nodes attached to 1D line elements. To define the channel:

1. Choose *RMA2 / 1D Geometry*.
2. Enter the desired values
3. Click the *OK* button or press the *ENTER* key.

When the *1D Geometry* item is selected from the *RMA2* menu, the dialog shown in Figure 5.15 appears. In this dialog, four values are specified to define a trapezoidal channel plus some extra storage.

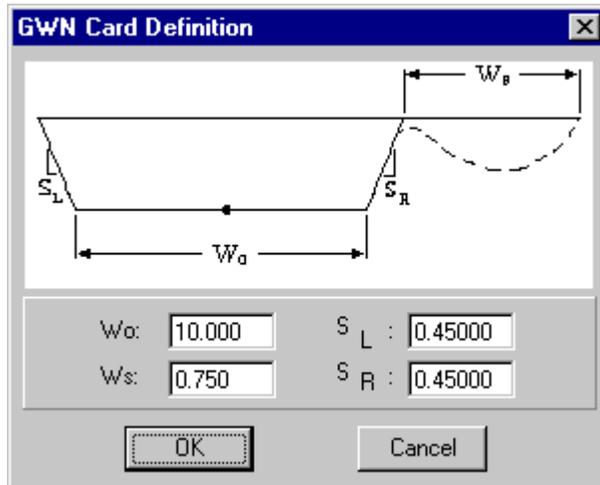


Figure 5.15 One-Dimensional geometry data.

Remember that there are multiple nodes on top of each other in a single junction element or control structure. These generally require their own channel definition, so they must be selected separately. To select a specific node in the junction element or control structure:

1. Choose the select nodes tool.
2. Click *and hold* the mouse button on the junction element or control structure.
3. With the mouse still held down, drag it to the desired node, then let the mouse button down.

When this is done, the display will return to normal view, and the desired node will be selected. With this done, the 1D geometry data can be assigned as usual.

5.11 Run GFGEN/RMA2

After a simulation has been read or saved, *SMS* can launch the finite element analysis engines. The model *GFGEN* converts the ascii geometry file written by *SMS* into a binary geometry file which is used by *RMA2*, *SED2D*, *RMA4*, and *RMA10*. To run *GFGEN*:

1. Select *RMA2 / Run GFGEN*. A dialog appears showing the engine that will execute as shown in Figure 5.16.
2. If the executable program is the wrong version, or if you are given the message that it was not found, then click the file browser icon and choose the correct version of the program that should run.
3. Click the *OK* button or press the *ENTER* key.

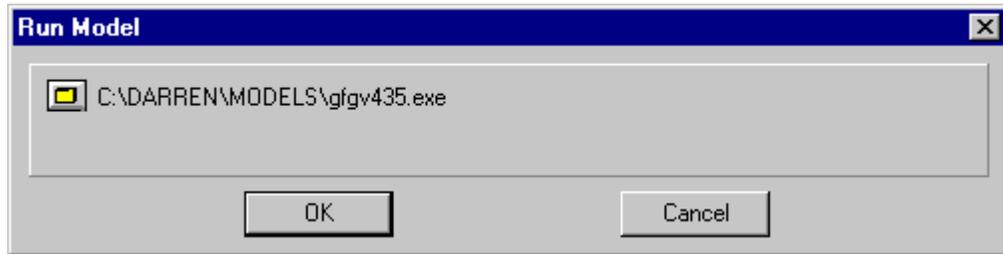


Figure 5.16 Run GFGEN dialog.

A window will appear which displays various information as *GFGEN* is runs. You should not need to type anything in this window. If the window prompts you for filenames, you need an updated version of the model. When *GFGEN* has finished, you will be prompted to press the *ENTER* key. If the window goes away before doing this, it encountered a problem and crashed. If this happens, look at the output file for information on why the model crashed (.ot1).

After *GFGEN* successfully runs, the finite element analysis can be run using *RMA2*. To run *RMA2*, follow the same process as when running *GFGEN*, except choose the Run *RMA2* menu item from the *RMA2* menu instead of the Run *GFGEN* menu item. Once again, the model should not prompt you for any filenames and you should be prompted to press the *ENTER* key before the window goes away. It is important to look at the message given in the *RMA2* window before making it go away. Many times, *RMA2* will give an error message and the solution will not be valid. If you see the message "RMA2 has finished initial solution", then the solution was successfully completed. If this message is not given, then there is a problem in your model.

5.12 Display Options

There are various boundary condition display options that can be set when dealing with the *RMA2* model. As discussed previously, boundary conditions are assigned to either nodes or nodestrings (see section 5.3). The display of these is accessed through the *Mesh Display Options* dialog (section 4.5).

5.12.1 Nodal Display Options

Next to the *Nodes* toggle box in the *Mesh Display Options* dialog, there is a button named *Options*. This button opens the *RMA2 Node Display Options* dialog (see Figure 5.17).

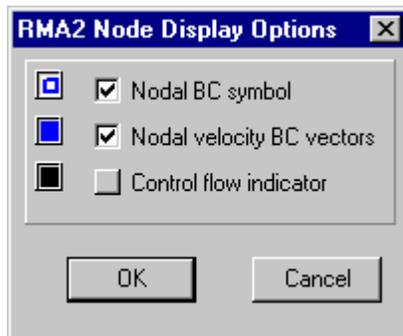


Figure 5.17 RMA2 Node Display Options dialog.

The items that can be specified in this dialog are:

- *Nodal BC Symbol*. This controls the symbol drawn at all nodes where nodal boundary conditions have been defined, whether those boundary conditions be velocity or head. Click the box to change the symbol attributes.
- *Nodal velocity BC vectors*. This controls the vector drawn at all nodes where velocity boundary conditions have been defined. Click the box to change the line attributes.
- *Control flow indicator*. This controls the vector drawn over a control element indicating the flow direction. Click the box to change the line attributes.

5.12.2 Nodestring Display Options

Next to the *Nodestring* toggle box in the *Mesh Display Options* dialog, there is a button named *Options*. This button opens the *RMA2 Nodestring Display Options* dialog (see Figure 5.18).

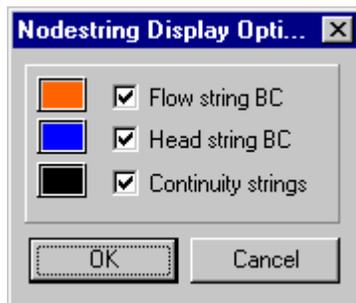


Figure 5.18 RMA2 Nodestring Display Options dialog.

This dialog contains the display settings for all nodestrings used with RMA2.

SED2D-WES Interface

SED2D-WES is part of the *TABS* modeling system developed by the U.S. Army Corp of Engineers Waterways Experiment Station. It had been distributed previously under the name of *STUDH*. It has the ability to compute sediment loading and bed elevation changes when supplied with a hydrodynamic solution computed by *RMA2*. The model supports both clay and sand beds individually, but the two bed types cannot be contained within the same model.

The *SED2D-WES* menu is available when in the mesh module and the current model is switched to the *TABS* model. Everything in the *SED2D-WES* menu is disabled until an *RMA2* mesh and boundary conditions file have either been read or saved. This is due to the fact that *SED2D-WES* relies on the existence of an *RMA2* solution, and because some of the control variables defined for *RMA2* are applied to the *SED2D-WES* model. The user can define *SED2D-WES* parameters as soon as the mesh exists. However, it is recommended that *SED2D-WES* pre-processing be done after the *RMA2* analysis has been completed. All *SED2D-WES* solutions are computed in metric units.

6.1 SED2D-WES File I/O

Sediment transport data describing such parameters as the deposition rate and the sediment load, along with the bed conditions and layer definitions used by *SED2D-WES* are stored in an ASCII file. This file is referred to as a sediment transport file and has a *.sed* file extension. *SED2D-WES* is run in an iterative fashion. The original file for a problem will be created by *SMS*. As *SED2D-WES* computes the scour and deposition of sediment across the mesh, a new bed definition is created for the problem, and *SED2D-WES* outputs a new sediment transport data file for successive runs.

6.1.1 New Simulation

The *New* command deletes all *SED2D-WES* data currently in *SMS*. This allows the user to start over in the definition of sediment transport parameters and bed definition and characteristics. This command does not delete the geometry or *RMA2* boundary conditions.

6.1.2 Open Simulation

The *Open* command allows the user to read in an existing sediment transport file to apply to the geometry. It is important to remember that before a *SED2D-WES* simulation file can be opened, there must first be read an *RMA2* simulation.

6.1.3 Save Simulation

The *Save* command allows the user to save a sediment transport file for *SED2D-WES*.

6.2 Global Parameters

To initialize the definition of a sediment transport problem, global parameter values need to be specified. Any region of the model that is not redefined with local parameters inherits these global parameters. These global parameters include: bed type, diffusion coefficients, initial concentration and settling velocity (see Figure 6.1).

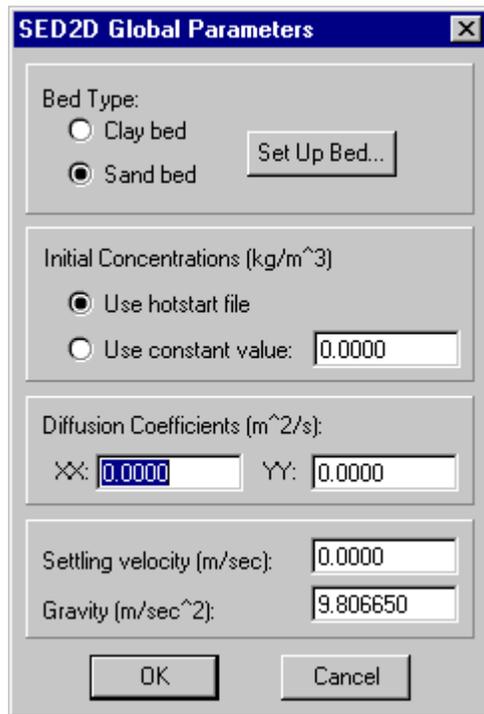


Figure 6.1 SED2D-WES Global Parameters Dialog.

6.2.1 Bed Type

SED2D-WES supports problems involving either clay beds or sand beds. Natural conditions may involve both, but a dominant bed type must be defined and used. *SMS* allows the user to select which bed type is desired. Based on the selected type, the *Set Up Bed* button invokes the appropriate bed parameter dialog and allows the user to define the default bed conditions for the model.

Clay Bed

When the bed type is set to clay and the *Set Up Bed* button is clicked, the *Global Clay Layers* dialog (see Figure 6.2) opens. In this dialog, the global clay layer parameters are defined.

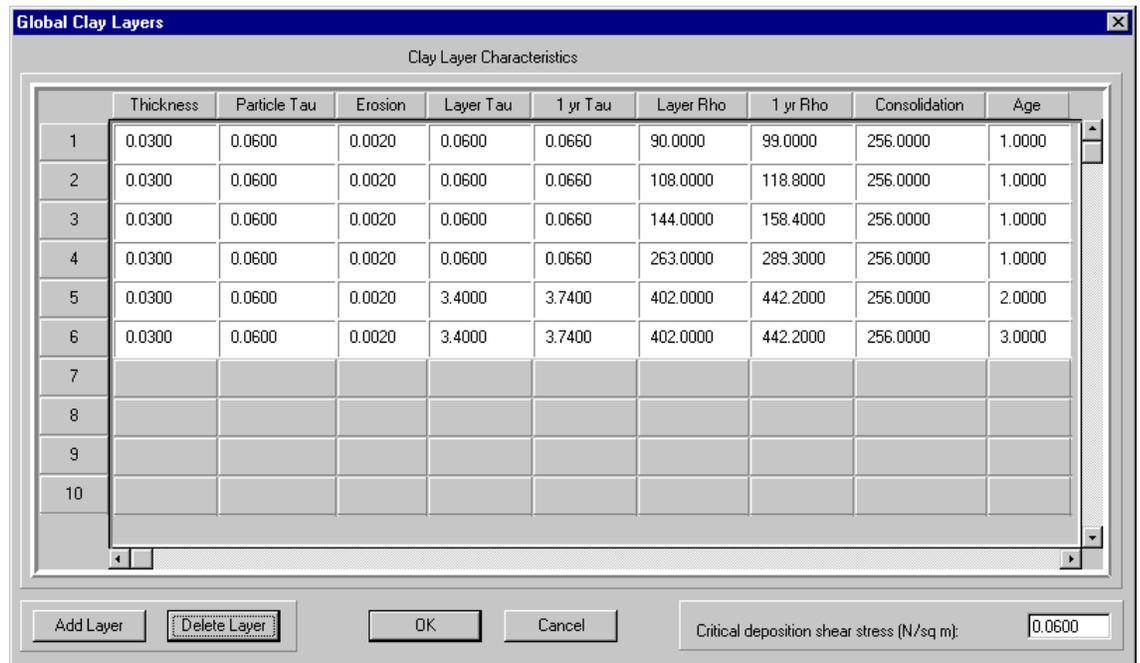


Figure 6.2 The Global Clay Layers dialog for SED2D-WES.

The default number of clay layers is four. At least one must exist, and up to ten can be defined. The first is the top and youngest clay layer, while the last is the bottom and oldest clay layer. For each clay layer, the following parameters are defined:

- *Thickness*. This is the typical thickness for the layer, specified in meters. The default value is 0.03 meters.
- *Particle Tau* (τ). This specifies the bed shear stress at which cohesive particles begin to erode away, specified in newtons per square meter. The default value is 0.06 n/m^2 .
- *Erosion*. This is an erosion rate constant for particle erosion, specified in kilograms per square meter per second. The default value is 0.002 $\text{kg/m}^2/\text{sec}$.
- *Layer Tau* (τ). This specifies the bed shear stress at which the entire layer begins to erode in mass, in newtons per square meter. The default is 0.06, 0.12, and 0.41 for layers 1, 2, and 3, respectively, and then 3.4 for any other layer.
- *1 yr Tau* (τ). This is the bed shear stress at which the layer begins to erode in mass after it is a year old, specified in newtons per square meter. If this value is greater than the *Layer Tau* value, the clay layer becomes more resistant to erosion with time (and consolidation). The default is 1.1 times the *Layer Tau* value.

- *Layer Rho (r)*. This is the initial dry density of clay material in the layer, specified in kilograms per cubic meter. The default is 90, 108, 144, and 263 kg/m³ for layers 1-4, respectively, and 402 kg/m³ for any other layer.
- *1 yr Rho (r)*. This is the dry density after the layer is a year old, specified in kilograms per cubic meter. If this value is greater than the *Layer Rho* value, then the clay layer becomes denser with time and the layer density depends also on the consolidation coefficient. The default is 1.1 times the *Layer Rho* value.
- *Consolidation*. This is the consolidation coefficient used in determining the layer density, specified in kilograms per square meter. It applies only to clay layers that are at least one year old. The default is 256 kg/m².
- *Age*. This is the initial age in years of the layer, specified in years.

In addition to the layer-specific parameters just explained, the following parameter is also defined:

- *Critical deposition shear stress*. This is the same for all layers and it is the shear stress at which deposition of sediment occurs.

Sand Bed

When the bed type is set to sand and the *Set Up Bed* button is clicked, the *Global Sand Layers* dialog (see Figure 6.3) opens. In this dialog, the global sand bed parameters are defined.

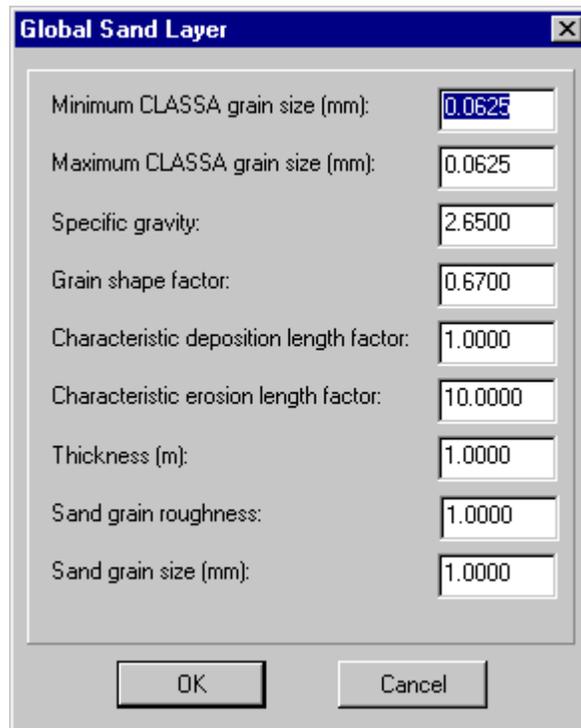


Figure 6.3 The Global Sand Layer dialog for SED2D-WES.

For a sand bed type, there is only one layer, although the thickness is variable. The following parameters are defined for the sand layer:

- Minimum and Maximum CLASS A grain size. These define the range of grain sizes that exist in the bed and in suspension, specified in millimeters. Currently, *SED2D-WES* requires that these two values be the same. In future versions, this will change. The default value is 0.0625 mm.
- Specific gravity. This is the specific gravity of sand grains. The default value is 2.65.
- Grain shape factor. This defines how smooth the sand grains are. The default value is 0.67.
- Characteristic deposition and erosion length factors. These are used in the equations that calculate the deposition and erosion that occurs. It is strongly recommended that the default values of 1.0 and 10.0, respectively, be used.
- Thickness. This defines the initial constant thickness of the sand layer. It is assumed that non-erodable bedrock exists under the sand.
- Sand grain roughness and size. These define the size of particles that get transported.

6.2.2 Initial Concentrations

The *Initial concentration* section of the *SED2D-WES Global Parameters* dialog (Figure 6.1) allows the user to specify the initial suspended sediment concentration. For a *SED2D-WES* coldstart simulation, the *Use constant value* option must be chosen. If a previous *SED2D-WES* solution is available, then the *Use hotstart file* option may be chosen.

In general, the specific concentration of suspended sediment varies across the flow field. For this reason, the following suggested procedure should be followed when running a *SED2D-WES* simulation.

1. Run *SED2D-WES* with a constant initial suspended sediment concentration.
2. Run *SED2D-WES* a second time with the same flow field and boundary conditions, using the first solution as hotstart input. This file gives better initial suspended sediment concentrations.
3. Use the second solution as the *SED2D-WES* solution file.

6.2.3 Diffusion Coefficients

The *Diffusion Coefficients* section of the *SED2D-WES Global Parameters* dialog allows the user to specify effective diffusion coefficients to be used by *SED2D-WES*, in the x- and y- directions. These values define the turbulent exchange coefficients in each of the principle directions. It is recommended that the two values be identical.

6.2.4 Other Parameters

The following other parameters are specified on the global level for the *SED2D-WES* analysis:

- ? *Settling velocity*. This value defines how fast sediment drops out of suspension. It is defined in meters per second.
- ? *Gravity*. This value defines the acceleration due to gravity. It is defined in meters per second squared.

6.3 Local Parameters

The global parameters discussed in the previous section are applied by *SED2D-WES* to each node in the mesh during the analysis. Local parameters can be defined for nodes, elements, nodestrings, or material types to override the global parameters. These local parameters are set using the *Local Parameters* menu command.

Note: While these parameters can be specified by any object or group, all the data is converted to nodal information inside SMS before being written to the SED2D-WES input file.

To override the global default value for a node, element, or nodestring:

1. Select the desired item.
2. Choose the *SED2D / Local Parameters* menu item.

The local parameters dialog which opens depends on the global bed type that was defined. Either the *Local Parameters for Clay* (Figure 6.4) or the *Local Parameters for Sand* (Figure 6.5) dialog will appear. To override a global parameter, turn off the *Use Default* toggle box and enter a new value.

The following values are available for a clay bed type:

- ? Diffusion coefficients, initial concentrations, settling velocity, layer thickness, and layer age.

The following values are available for a sand bed type:

- ? Diffusion coefficients, initial concentrations, settling velocity, layer thickness, grain roughness, and grain size.

Use	Layer	Thickness (m)	Age (yrs)
<input checked="" type="checkbox"/>	1	0.0300	1.0000
<input checked="" type="checkbox"/>	2	0.0300	1.0000
<input type="checkbox"/>	3	0.0300	1.3500
<input type="checkbox"/>	4	0.0300	1.5000
<input type="checkbox"/>	5		
<input type="checkbox"/>	6		
<input type="checkbox"/>	7		
<input type="checkbox"/>	8		
<input type="checkbox"/>	9		
<input type="checkbox"/>	10		

Use Default

- XX Diffusion: 0.0000
- YY Diffusion: 0.0000
- Initial concentration: 0.0000
- Settling velocity (m/sec): 0.0000

OK Cancel

Figure 6.4 The Local Clay Parameters dialog for SED2D-WES.

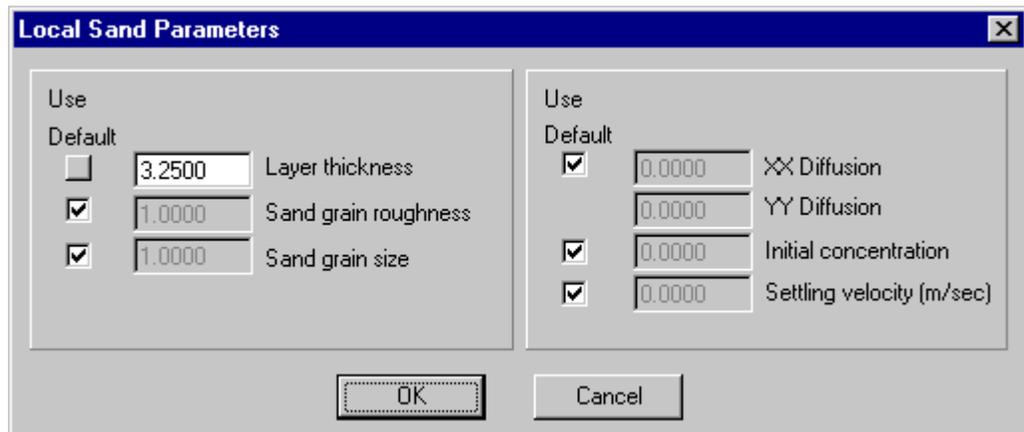


Figure 6.5 The Local Sand Parameters for SED2D-WES.

6.4 BC Concentrations

SED2D-WES requires sediment concentration to be specified at all inflow boundaries. To do this:

1. Select the nodestring where *RMA2* boundary conditions are defined.
2. Choose the *SED2D / BC Concentrations* menu item.

The suspended sediment concentration can be specified as either constant or transient. A transient value is defined using the *XY Series Editor* (see chapter 16).

6.5 Model Control

SED2D-WES requires several additional parameters to control how the numerical model functions. These parameters are defined using the *SED2D Model Control* dialog (see Figure 6.6).

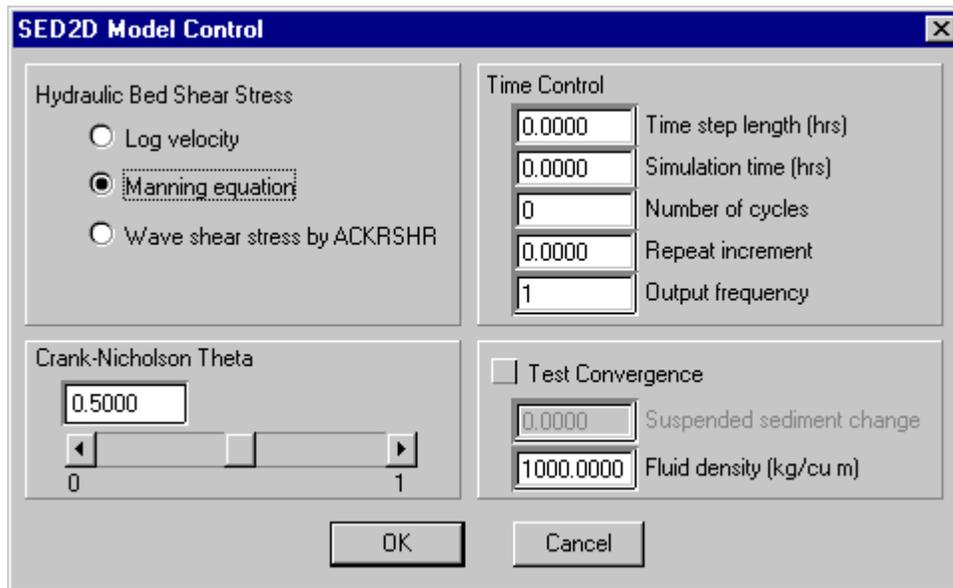


Figure 6.6 The SED2D-WES Model Control Dialog.

The *SED2D Model Control* dialog provides control over the following parameters:

- *Hydraulic Bed Shear Stress*. This option defines which equation should be used when computing the bed shear stress. This value is stored in the *HS* record.
- *Crank-Nicholson Theta*. This value is an implicitness factor used when performing calculations. This value should be limited between 0.5 (equal weighting of the current time step and the previous time step) and 1.0 (no influence from the previous time step). The default value is 0.66. This is stored in the *TT* record.
- *Time Control*. The *Time step length* and *Simulation time* define the total time modeled in the *SED2D* analysis. When a dynamic *RMA2* flow field is used, it is very important that the time step length for *SED2D* be exactly equal to the time step length from the *RMA2* simulation. Otherwise, accuracy errors can occur. The *Number of cycles* to perform defines the number of times to repeat the entire *SED2D* simulation. The *Repeat increment* defines at what time step the *RMA2* solution file should be rewound. This should be set equal to the *Simulation time* unless the *RMA2* flow field is dynamic and shorter than the *SED2D* simulation. The *Output frequency* defines how often to save the simulation results to the binary file. To save all time steps, set this value to one. These values are stored in the *TZ* and *TO* records.
- *Test Convergence*. Turn this option on to specify a convergence criteria for the model. This value is a percent between zero and one. If the maximum deposition or erosion at any node is the greater than this percent of the original water depth, then *SED2D* will stop running. This is used because when the

bed changes too much, the flow field is no longer valid. The default value is 0.25. This value is stored in the *EF* record.

- ? *Fluid Density*. Enter the desired fluid density in kilograms per cubic meter. This value is stored in the *FD* record.

6.6 Print Control

The *Print Echo Control* dialog (Figure 6.7) allows the user to specify what information is echoed to an output file and how often that information is written. This is stored in the *TO* record. The user also may specify a subroutine trace. This is used more for debugging *SED2D* than analyzing a model. This is stored in the *TR* record.

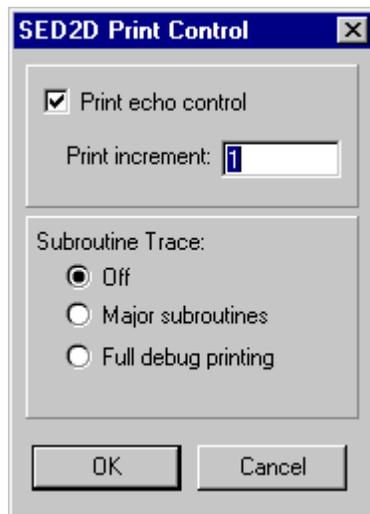


Figure 6.7 The *SED2D-WES* Print Control Dialog

6.7 Create Data Sets

When the *Create Data Sets* option is turned on, a check mark is shown next to it in the menu and data sets are created by *SMS* for parameters used in the *SED2D* analysis. These parameters include the x- and y- diffusion coefficients, initial concentrations, and settling velocities. The data sets appear in the *Data Browser* and they may be contoured to show their variation across the mesh.

6.8 Model Checker

The *SMS* interface to *SED2D-WES* is equipped with a *Model Checker*. This feature evaluates the current model and performs standard checks to assure the quality of the

model for numerical analysis. The *Model Checker* is an ongoing development. Additional checks will be added as SMS users provide suggestions. The current version of *SMS* includes an option to assure that concentration values are specified at all inflow boundaries. The *SED2D* model checker also runs the *RMA2* model checker. The user may disable any model checking option, but this is not recommended. Possible errors are shown in the window at the top of the dialog. Additional information about an error is shown in the bottom window when the error is selected.

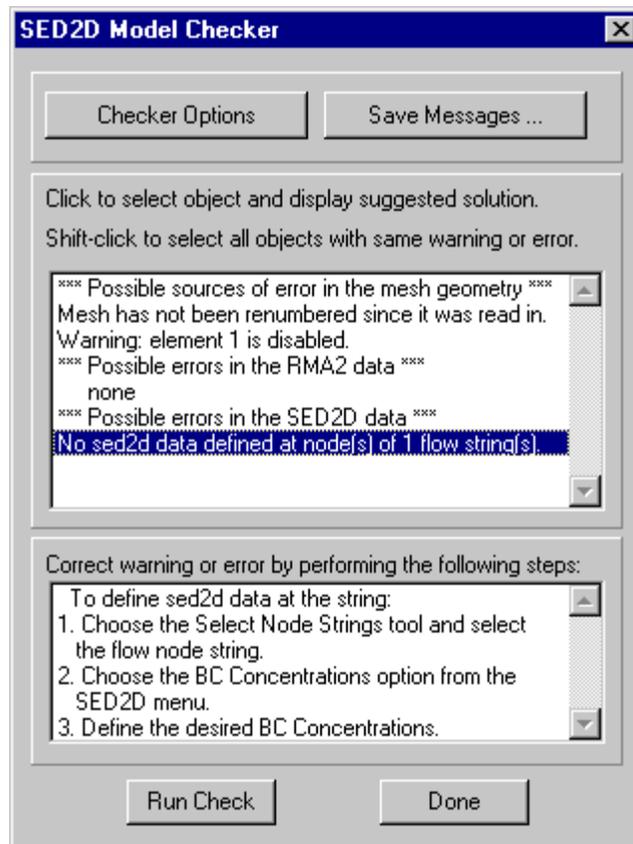


Figure 6.8 The SED2D Model Check dialog.

HIVEL Interface

HIVEL2D is a two-dimensional model used to analyze high velocity flow. It was written to model concrete-lined channels with hydraulically steep slopes, and has been adapted for use with both supercritical and/or subcritical flow fields. *HIVEL2D* was developed at the U.S. Army Corps of Engineers Waterways Experiment Station (USACE-WES) and continues to be maintained by WES. *HIVEL2D* is a linear solver so only linear elements are created (no midside nodes).

This chapter describes the commands used in *SMS* to create and edit *HIVEL* model parameters and boundary conditions. These commands are all found in the *HIVEL* menu. For information on using *SMS* to create the finite element mesh, see chapter 4. Once the analysis is complete, post-processing of the analysis results is performed by importing the solution file using the data browser. (See Lesson 9 of *SMS Tutorials* and the *HIVEL2D User Manual* for more about running *HIVEL*).

7.1 Accessing the HIVEL2D Interface

By default, the *HIVEL* interface inside *SMS* is not available on startup. To change to the *HIVEL* model with a registered version of *SMS*:

1. Be sure that you are in the *Mesh Module* .
2. Select *Data / Switch Current Model*.
3. In the *Current Model* dialog, choose the *HIVEL2D* option

4. Click the *OK* button or press the *ENTER* key.

If you have not yet registered *SMS*, you can change to the *HIVEL* model using the above steps if you are running in *Demo Mode* (see section 2.5.1). You can have *SMS* always start with the *HIVEL* interface by using command line arguments (see section 1.2). When the *HIVEL* interface is available, the tools and menus change so that only those useful for *HIVEL* are available.

7.2 New Simulation

The *New Simulation* command in the *HIVEL* menu deletes all of the *HIVEL*-specific data associated with the current model. This data includes the finite element mesh and boundary conditions, model control data, hot start data, and any solutions that have been imported into *SMS*. This data is only removed from memory, not from the disk. Boundary conditions include inflow/outflow specifications and material properties. Model control data includes simulation titles, computation time, and default constants such as gravity and turbulence coefficients. Data such as scatter sets and feature objects are not removed from memory.

7.3 Open Simulation

The *Open Simulation* command in the *HIVEL* menu allows you to select a simulation that has been previously saved. When a *HIVEL* simulation is opened, any existing finite element mesh and boundary conditions are erased from memory. The file to open is the *superfile*, typically with the extension “.sup”. The superfile contains a list of names of all the files associated with the simulation. For more about this file, see the *HIVEL* documentation.

After reading the superfile, *SMS* opens the geometry, boundary condition, and hotstart files, and displays the data in the *Graphics Window*. (See the *HIVEL User Manual* for more information of data file formats.) The name of the simulation is written at the top of the *Graphics Window*.

7.4 Save Simulation

The *Save Simulation* command in the *HIVEL* menu opens the *HIVEL Save Simulation* dialog (see Figure 7.1). Specified in this dialog is the superfile name along with all other files that can be created. The path where the files will be saved is shown at the top of the dialog and may be changed by clicking on the file button  to the left of it. If the *Hot start* option is turned on, the hot start file will be written according to the values set up in the *Build Hot Start* dialog (see section 7.5). After clicking the *OK* button, the saved simulation can be opened at a later time or used in an analysis.

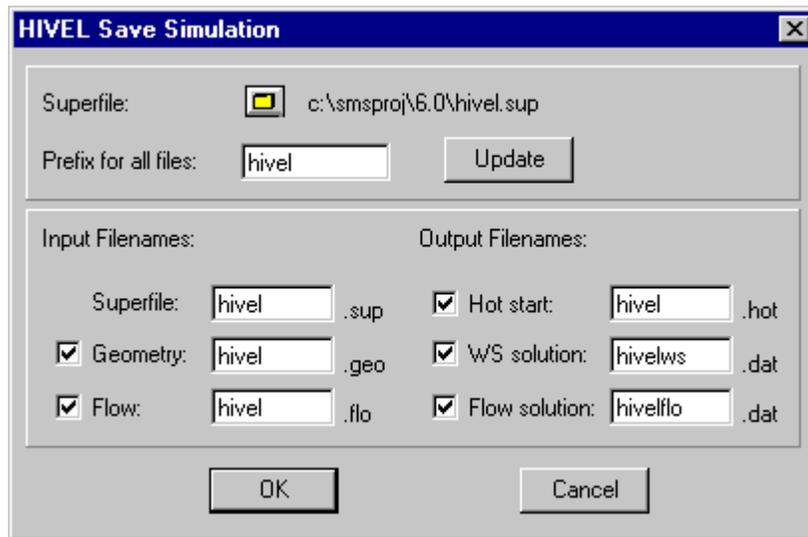


Figure 7.1 HIVEL Save Simulation dialog

7.5 Build Hot Start

In general, when using HIVEL, a hotstart file should be used. The *Build Hot Start* command in the *HIVEL* menu invokes the *Setup HIVEL Build Hot Start* dialog (see Figure 7.2). At the top of the dialog is the name of the file that will be used to save the hotstart data. Under the file name is a field for *Time associated with step m*. This corresponds with the last time step from a previous *HIVEL2D* run. This should be set to 0.0 for an initial solution. *HIVEL2D* calculates derivatives of water depth and velocity using the depth and velocity values from the last two time steps computed. The rest of the *HIVEL Build Hot Start* dialog is used to specify the water depth and velocity of the last two time steps from the previous run.

The hotstart velocity values can be defined either by a vector data set, which has one (x,y) pair per node, or by a constant velocity assigned to all nodes. Similarly, the water depth values can be defined either by a scalar data set, which has one depth value per node, or the depth can be computed at each node using a constant water surface elevation. For an initial run, it is suggested that the velocity be set to zero and that the water depths be computed from a constant water surface elevation equal to the specified outflow condition. However, it is important that the initial water surface elevation be greater than all the mesh nodal elevations because *HIVEL2D* does not handle dry nodes.

The Write Hotstart Now button causes the hotstart file to be written according to the specified parameters. This generally does not need to be done because the hotstart file will be written when the simulation is saved (see section 7.4).

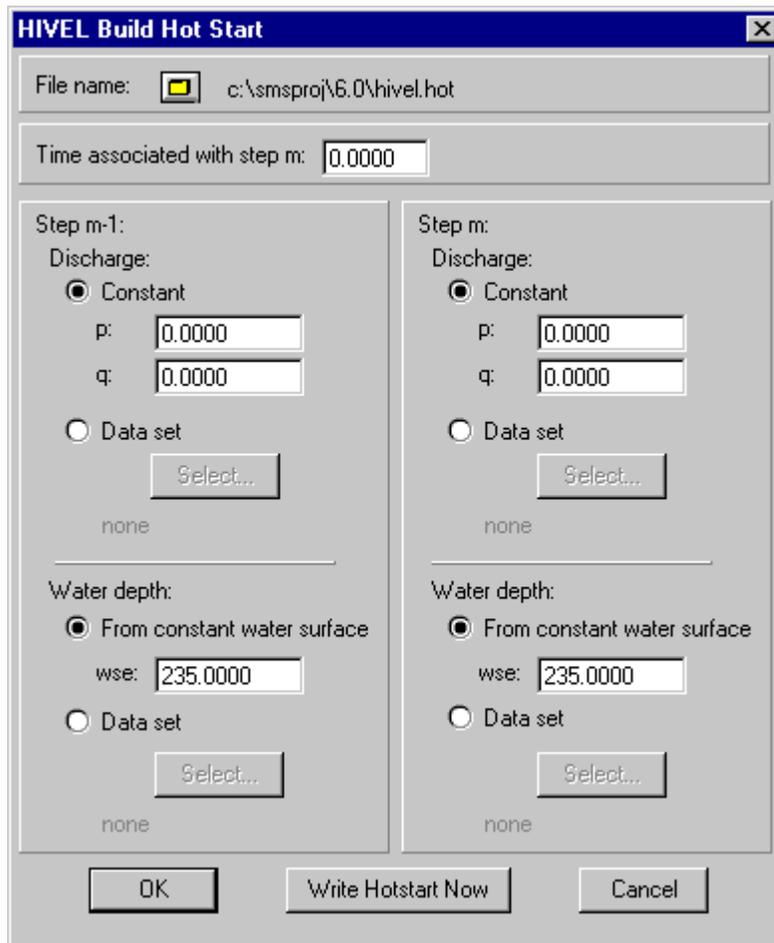


Figure 7.2 HIVEL Build Hot Start dialog

When SMS opens a HIVEL simulation, data sets are created from the hotstart file and they are automatically assigned to time steps m and $m-1$. This allows the boundary conditions to be changed without changing the hotstart data.

It is important to understand that after running a simulation, HIVEL overwrites the hotstart file with the values from the last two time steps of the simulation. Therefore, if it is desired that a specific hotstart file be saved, a backup copy should be created.

7.6 Assign BC

The *Assign BC* command in the HIVEL menu is used to assign boundary conditions to nodes or nodestrings. If nodes are selected when this command is issued, then boundary conditions are assigned to the nodes. If nodestrings are selected when this command is issued, then boundary conditions are assigned to the nodestrings.

7.6.1 Nodal Boundary Conditions

Nodal boundary conditions can be assigned to any node that lies on the mesh boundary. If a selected node is not on the boundary, *SMS* will give an error message. The *HIVEL Nodal Boundary Condition* dialog is shown in Figure 7.3.

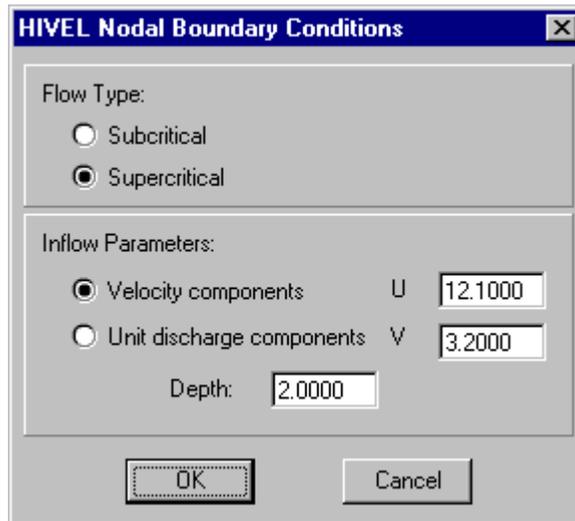


Figure 7.3 *HIVEL Nodal Boundary Conditions Dialog*

In *HIVEL2D*, a node can only be assigned an inflow boundary condition, not an outflow boundary condition. This inflow boundary condition can be either velocity components or unit discharge components. The inflow boundary must be specified as either subcritical or supercritical. If the supercritical option is selected, then the water depth at the node must also be specified.

7.6.2 Nodestring Boundary Conditions

Nodestring boundary conditions can be assigned to any nodestring that lies fully on the mesh boundary. If a selected nodestring contains any nodes not on the boundary, *SMS* will give an error message. A selected nodestring must also be continuous along the boundary. If the nodestring skips any nodes along the boundary, then an error message is given. The *HIVEL Nodestring Boundary Condition* dialog is shown in Figure 7.4.

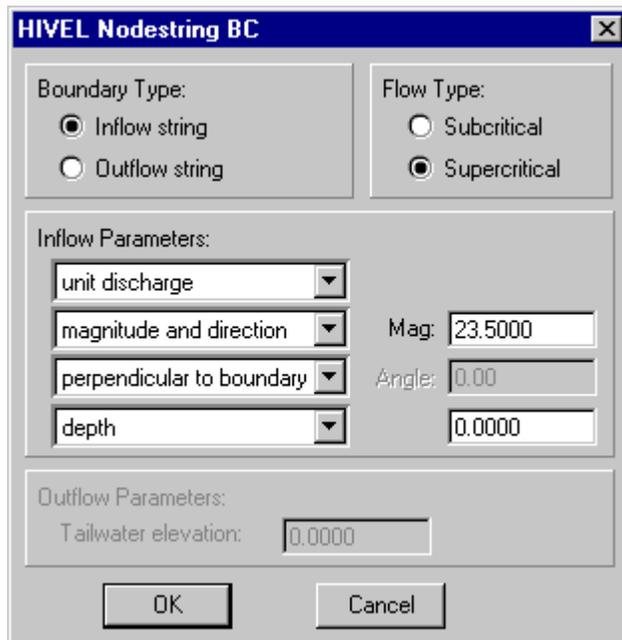


Figure 7.4 HVEL Nodestring Boundary Condition Dialog

In *HVEL2D*, a nodestring is defined as either an inflow or an outflow boundary condition. Flow through the boundary is defined as either subcritical or supercritical.

An inflow boundary condition is given either an average velocity or a unit discharge. The velocity or discharge value can be specified using x and y components or magnitude and direction components. If the magnitude and direction option is chosen, then the angle can be automatically computed as perpendicular to the boundary or it can be entered manually. For a supercritical flow condition, the water depth or water surface elevation must be specified.

An outflow boundary condition is much simpler. If an outflow boundary condition is subcritical, then the tailwater elevation must be specified. If it is supercritical, the water surface is not specified.

7.7 Delete BC

The *Delete BC* command will delete boundary conditions previously assigned to all selected items. If neither nodes nor nodestrings are selected, you will be given an error message that nothing is selected.

7.8 Model Control

The *Model Control* command in the *HVEL* menu opens the *HVEL Model Control* dialog (see Figure 7.5). This dialog is used to specify model control parameters.

Figure 7.5 HIVEL Model Control Dialog

7.8.1 Job Titles

The *Job Titles* section allows the user to give up to three titles and/or comments about the problem being modeled. These fields correspond to the *T1*, *T2*, and *T3* records. Although *HIVEL2D* allows any number of *T1* and *T2* records in the boundary condition file, *SMS* only stores one of each.

7.8.2 Turbulence Coefficients

The *Turbulence Coefficients* section allows the user to specify the turbulence coefficient that will be used for computations. One value is used for smooth flow and the other is used for rough flow conditions, such as near a hydraulic jump. These values are used by *HIVEL2D* to determine turbulent eddy viscosity based on depth, velocity magnitude, and roughness. Default values are 0.10 for smooth flow and 0.50 for rough flow conditions. These values are stored in the *TURB* record.

7.8.3 Units Control

The *Units* section allows the user to control the type of units are used during the analysis. When this is changed, the *Gravitational acceleration* value and *Conversion coefficient for Manning's n* value (see section 7.8.5) are automatically converted to the new units. The *Units* value is stored in the *SI* card.

7.8.4 Petrov-Galerkin Coefficients

The *Petrov-Galerkin Coefficients* section allows the user to specify the Petrov-Galerkin weight coefficients for the model. If no values are specified, the defaults of 0.25 and 0.50 for smooth and rough regions, respectively, will be used. These values are stored in the *PGWC* record.

7.8.5 Gravity and Manning's Conversion Constants

In the lower left of the dialog, two constants for the model are specified. The first is the acceleration of gravity (*GRAV* record). The default value is 32.189 for English units or 9.8066 for Metric units. The second constant is an empirical conversion coefficient for use in Manning's equation (*MCON* record). The default value is 2.208 for English units or 1.00 for Metric units.

Note: When entering these values, be careful that the units are set (see section 7.8.3) because these values are converted when the units are changed.

7.8.6 Iterations

The *Iterations* section allows the user to specify the maximum number of iterations that will be performed for each time step and a convergence tolerance. At each time step, *HIVEL2D* will perform the specified number of iterations unless convergence is reached. Convergence is reached when the maximum change in *Froude Number* at all nodes in the mesh from the previous iteration is less than the specified tolerance. If the convergence criteria is not reached, the solution may not be valid. These values are stored in the *ITER* record.

7.8.7 Computation Time

The *Computation Time* section allows the user to specify the time step in hours (*TIME* record) and the total number of time steps *HIVEL2D* should perform (*STEP* record). The initial time value is the value that was specified in the hotstart file (see section 7.5). Also specified in the *Computation Time* section is the frequency for saving solution data. The default value is one, meaning the solution will be saved after each time step in the simulation.

7.8.8 Temporal Derivative

The *Temporal Derivative* section allows the user to specify if *First order backward* differencing or *Second order backward* differencing will be used to calculate initial guesses at each time step. This value is stored in the *TIME* record.

7.8.9 Reset Defaults

The *Reset Defaults* button will reset all values in this dialog to their default values. The default values may be customized from what the factory defaults were. This is accomplished by first entering the desired values and then executing the *File / Save Environment*. All values that were entered will become the new defaults. To return to the original factory defaults, it is necessary to delete the *SMS* initialization file (see section 1.2).

7.9 Material Properties

The *Material Properties* command in the *HIVEL* menu opens the *HIVEL Material Properties* dialog (see Figure 7.6).

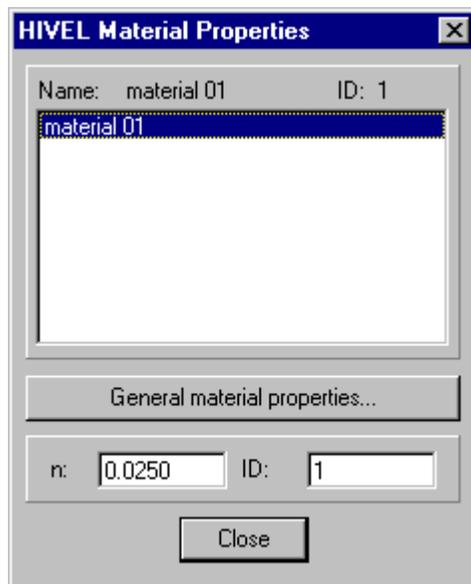


Figure 7.6 HIVEL Material Properties dialog

Materials are assigned to elements using the *Elements / Assign Material Type* command. In *HIVEL2D*, these materials have an identification number and a Manning's roughness value. The Manning's roughness value gets applied to elements of the specific material type.

Clicking the *General Material Properties* button will open the *Materials Data* dialog (see Section 2.5.2), where the name, identification number, color, and pattern of each material is defined. Hivel2D does not store the material names, but this can be stored using a separate materials file (see section 2.5.1).

7.10 Model Check

The *Model Check* command in the *HIVEL* menu opens the *Model Check* dialog (see Figure 7.7). A model check should be performed on all models before running an analysis. The model check will perform a basic check to insure that all of the required information is present as well as check the finite element network for errors.

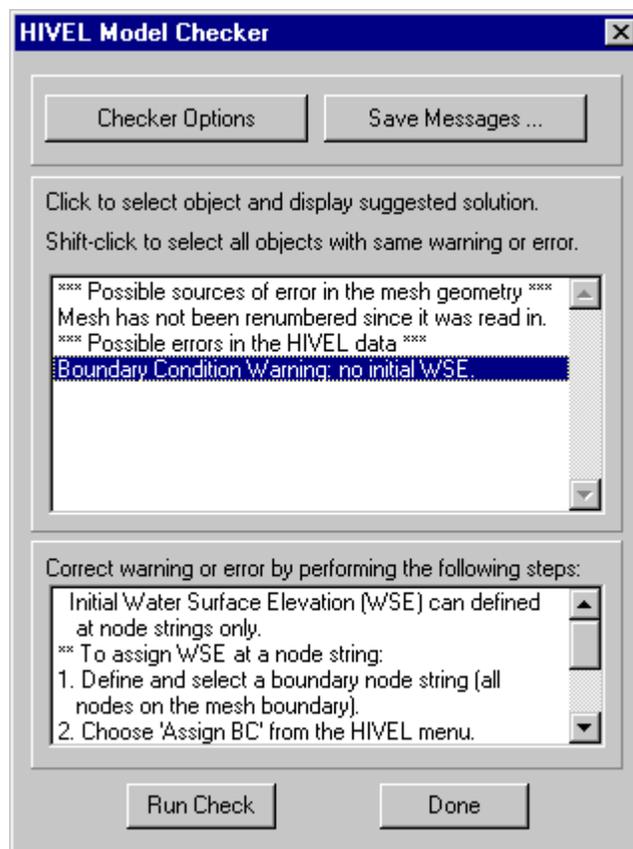


Figure 7.7 HIVEL2D Model Checker Dialog

Selecting the *Checker Options* button opens the *HIVEL Model Checking Options* dialog. This dialog lists the items that may be included in the model checking procedure. By default all supported checks are enabled. The checks include:

- *Check boundary conditions.* When this option is turned on, *SMS* will check to make sure both inflow and outflow have been defined. It will also check if the outflow water surface elevation is higher than the highest node elevation.

- *Check model controls.* Parameters that are checked when this option is turned on are the of turbulence coefficients, Petrov-Galerkin weight coefficients, and backwards differencing used. See section 7.8 for more information on these model parameters.
- *Mesh Check Options.* Generic model checking options include: renumbered mesh, material definition, duplicate nodes, and small voids. You can also limit the number of similar messages to report.

Click the *Run Check* button to perform a model check with the selected options. After *SMS* has completed its model check, the error and warning messages will appear in the top text window. When one of these messages is highlighted, the lower text window shows directions on how to fix the error or warning. To save the results of a model check to a text log file, click the *Save Messages* button and enter a file name. To close the *Model Checker*, click the *Done* button. The model checker may remain open while changes are made to the mesh.

7.11 Run HIVEL

The *Run HIVEL* command from the *HIVEL* menu allows you to run a finite element analysis on the model open in *SMS*. If the current data in *SMS* has been edited since it was last saved, you will be prompted to save it. When this command is issued, a dialog appears showing the default location of the *HIVEL2D* executable (see Figure 7.8). If the executable shown is the wrong version, or if you are given a message that it was not found, then click the file browser icon and choose the correct executable to run. Upon clicking the *OK* button, a new window appears to run *HIVEL2D*. If the window goes away before prompting you that the simulation was finished, this means that the program crashed and you should look in the printed output file for more information.

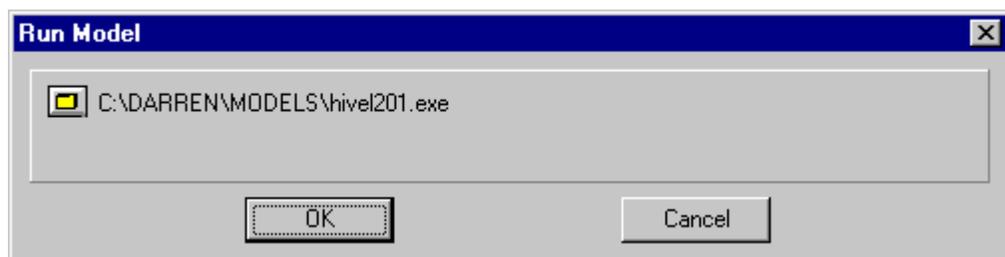


Figure 7.8 The Run HIVEL simulation dialog.

7.12 Display Options

There are various boundary condition display options that can be set when dealing with the *HIVEL2D* model. As discussed previously, boundary conditions are assigned

to either nodes or nodestrings (see section 7.6). The display of these is accessed through the *Mesh Display Options* dialog (section 4.5).

7.12.1 Nodal Display Options

Next to the *Nodes* toggle box in the *Mesh Display Options* dialog, there is a button named *Options*. This button opens the *HIVEL Node Display Options* dialog (see Figure 7.9).

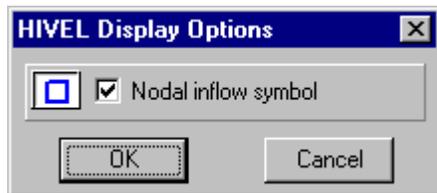


Figure 7.9 HIVEL Nodal Display Options dialog.

Only inflow boundary conditions can be assigned to nodes in HIVEL. Click on the box to change the nodal boundary condition display attributes.

7.12.2 Nodestring Display Options

Next to the *Nodestring* toggle box in the *Mesh Display Options* dialog, there is a button named *Options*. This button opens the *HIVEL Nodestring Display Options* dialog (see Figure 5.18).

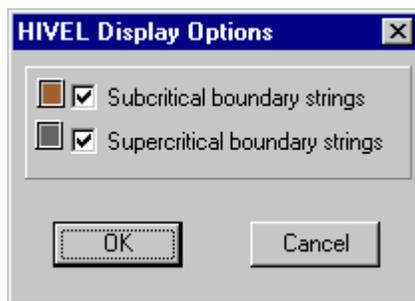


Figure 7.10 HIVEL Nodestring Display Options dialog.

This dialog contains the display settings for all nodestrings used with *HIVEL2D*.

ADCIRC Interface

The ADvanced CIRCulation model (*ADCIRC*), is a two-dimensional, depth-integrated, barotropic time-dependent long wave, hydrodynamic circulation model. *ADCIRC* can be applied to computational domains encompassing the deep ocean, continental shelves, coastal seas, and small-scale estuarine systems for simulations that require months to years time. In a single simulation, *ADCIRC* can provide tide and storm surge elevations and velocities corresponding to each node over a very large domain encompassing regional domains such as the western North Atlantic Ocean, the Caribbean Sea, and the Gulf of Mexico.

A mesh for use with *ADCIRC* can be created in *SMS* using either the *Map Module* or the *Mesh Module*. The *Mesh Module* editing tools can be applied to these meshes, and the *Map Module* mapping can be used to update portions or all of the mesh. The modeling parameters required by *ADCIRC* are generated and applied to the mesh using commands grouped in the *ADCIRC* menu of the *Mesh Module*. Once the analysis is complete, the user imports solution files for post-processing through the data browser. See section 3.1 for importing the solution file into the *Data Browser* and Chapter 3 for post-processing techniques.

8.1 New Simulation

The *New Simulation* command in the *ADCIRC* menu deletes all of the *ADCIRC*-related data associated with the current model. This data includes the finite element mesh and boundary conditions, model control data, and any solutions that have been imported into *SMS*. This data is only removed from memory, not from the disk.

Boundary conditions include the nodestring type. Model control data includes simulation titles, initial values, and model type. Data such as scatter sets and feature objects are not removed from memory.

8.2 Open Simulation

The *Open Simulation* command in the *ADCIRC* menu reads in geometry and control files that have been previously created and saved. The geometry file written previously by *SMS* has the “.grd” extension and the control file has the “.ctl” extension. (Note: These files are accessed as units 14 and 15 from within *ADCIRC*. Therefore, they may also be accessed as fort.14 and fort.15) Opening a new geometry causes any existing geometry to be removed from memory along with control data associated with that geometry. Opening a control file replaces any data that exists in the Model Control.

The name of an open geometry file is displayed at the top of the *Graphics Window*. The geometry file stores the location of nodes that define element corners and the element connectivity of the nodes. The file also contains the geometric definition of nodestrings that compose the boundary condition locations. See the *ADCIRC* documentation for the format of the geometry file.

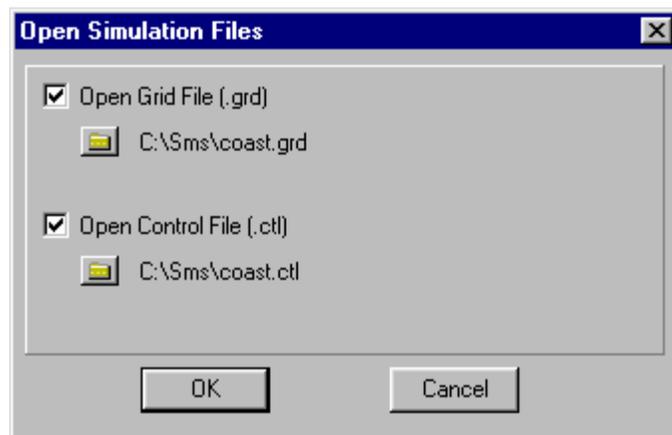


Figure 8.1 Open Simulation Files dialog.

8.3 Save Simulation

The *Save Simulation* command in the *ADCIRC* menu opens the dialog shown in Figure 8.2. In this dialog, you can choose to save the geometry and/or the control file. Generally, the entire simulation would be saved. A single prefix may be defined in the upper portion of the dialog. The names of the files to be saved are updated when the *Update* button is selected. The *Path Name* icon at the top of the dialog allows the user to select the directory for the files to be saved in.

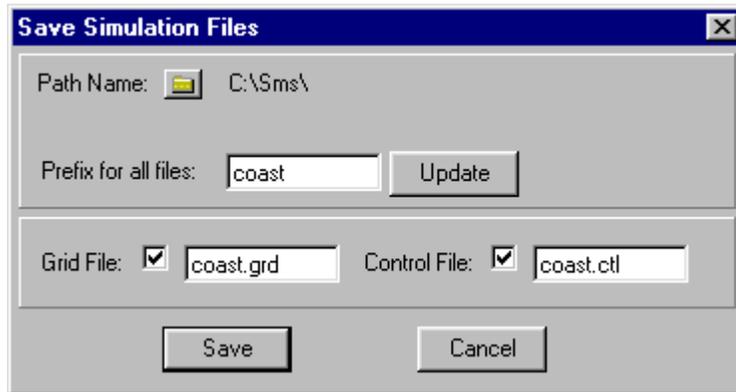


Figure 8.2 ADCIRC Save Simulation dialog.

After saving a simulation, ADCIRC can be run directly from inside SMS as described in section 8.8.

8.4 Assign BC

The ADCIRC *Nodestring Atts* dialog (Figure 8.3) is used to assign a type of boundary conditions to individual nodestrings. Before assigning boundary conditions to nodestrings, at least one nodestring must be selected using the *Select Nodestrings* tool.

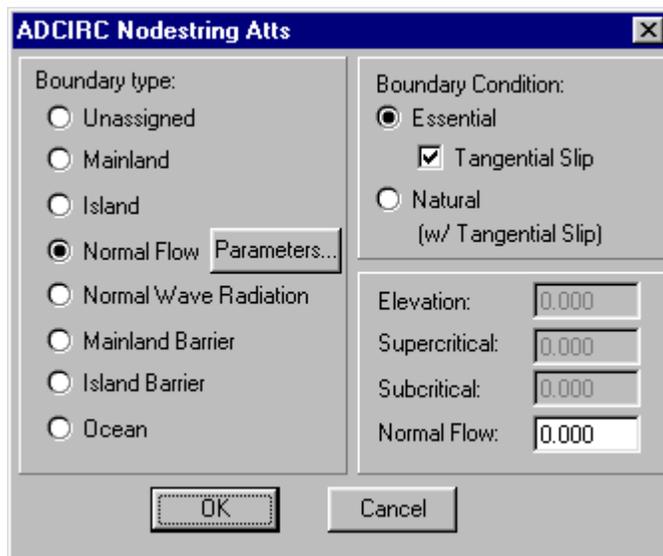


Figure 8.3 ADCIRC Nodestring Atts Dialog.

To assign boundary conditions to the selected nodestring(s), select one of the *Boundary Type* options. Options on the right side of the dialog will become available, as different *Boundary Type* options are selected.

8.4.1 Normal Flow Parameters

When the *Normal Flow* option is selected from the *ADCIRC Nodestring Atts* dialog, the *Parameters* button becomes available. Pushing the *Parameters* button brings up the dialog shown in Figure 8.4.

The screenshot shows the "Normal Flow Parameters" dialog box. At the top, there is a checkbox for "Non-Periodic Normal Flow" which is unchecked. Below it is a "Time Increment" field set to "0". To the right, there is a "Minimum Angle For Tangential Flow:" field set to "110" with "deg." next to it. The "Frequencies" section contains a list box with "Newfreq" selected. To the right of the list box are fields for "Name:" (Newfreq), "Frequency:" (0.000), "Nodal Factor:" (0.000), and "Equilibrium Arg (deg):" (0.000). Below the list box are "New" and "Delete" buttons. The "Flow/Unit Width" section contains a list box with "Node 13" selected. To the right are fields for "Name:" (Newfreq), "Amplitude:" (0.000), and "Phase:" (0.000). At the bottom are "OK" and "Cancel" buttons.

Figure 8.4 Normal Flow Parameters dialog.

The *Frequencies* and *Flow/Unit Width* for the nodes on the *Normal Flow* nodestrings can be set when the *Non-Periodic Normal Flow* is disabled. Pushing the *New* button creates a new frequency set that includes all nodes on *Normal Flow* nodestrings. The name, frequency, nodal factor, and equilibrium argument can be set for the entire frequency set. The amplitude and phase can be set for each individual node in the set. Pushing the *Delete* button deletes the selected frequency set.

The time increment and minimum angle for tangential flow can also be set inside of the dialog.

8.5 Create Functions

The *Create Functions* dialog is used to create functions for the entire geometry.

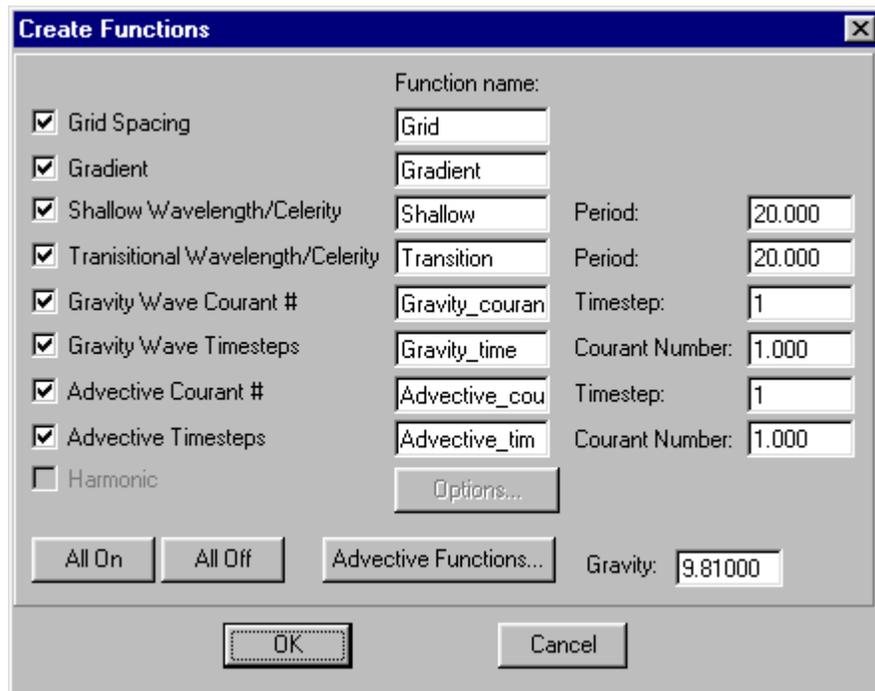


Figure 8.5 ADCIRC Create Functions Dialog.

Each function that is toggled on will be created. All of the available functions can be turned on by pushing *All On*. All of the functions can be turned off by pushing *All Off*. The *Gravity* can be set and is used in several of the function calculations. The *Advective Functions* is described below in the advective function sections. The functions that can be created include:

- *Grid Spacing*. Creates a function that gives the average distance between a node and its neighbors.
- *Gradient*. Creates a function that gives the gradient at each node. The gradient is calculated as the run divided by the rise.
- *Shallow Wavelength/Celerity*. Creates two functions that calculate the celerity and wavelength at each node in shallow water. The celerity is calculated as: $Celerity = (Gravity * Nodal\ Elevation)^{0.5}$. The *Wavelength* is: $Wavelength = Period * Celerity$.
- *Transitional Wavelength/Celerity*. Creates two functions that calculate the celerity and wavelength at each node for any depths.

- *Gravity Wave Courant Number*. Creates a function that gives the courant number for each node given the *Timestep*. The equation is: $\text{Courant Number} = \text{Timestep} * (\text{Gravity} * \text{Nodal Elevation})^{0.5} / \text{Nodal Spacing}$.
- *Gravity Wave Timesteps*. Creates a function that calculates the gravity wave timestep given the *Courant Number*. The equation is the same as for the *Gravity Wave Courant Number*, solved for the Courant Number.
- *Advective Courant Number*. Creates a function that calculates the courant number given the *Timestep* and a velocity function. The velocity function can be selected by clicking on the *Advective Functions...* button. This brings up a *Select Data Set* dialog that lists the vector functions currently in memory. The courant number is calculated as: $\text{Courant Number} = \text{Nodal Velocity Magnitude} * \text{Timestep} / \text{Nodal Spacing}$. This option is disabled if no vector functions exist.
- *Advective Timesteps*. Creates a function that calculates the timestep given the *Courant Number* and a velocity function. The velocity function can be selected as described above in the description of the *Advective Courant Number*. The equation is the same as for the *Advective Courant Number*, solved for the timestep. This option is disabled if no vector functions exist.
- *Harmonic*. Creates a scalar harmonic function and/or a vector harmonic function. Pushing the Options button brings up the *Harmonic Options* dialog shown in Figure 8.6. The name of the function(s) to be created can be set in the *Name* fields. The frequencies to be used in creating the function can be chosen by double-clicking on a frequency name shown in the *Scalar Frequencies* window or by clicking on the name and by pushing *Select*. A frequency can be unselected by double-clicking on the name again or by selecting it and pushing *Unselect*. The time values that will be used in calculating the timesteps can be set in the fields at the bottom of the dialog.

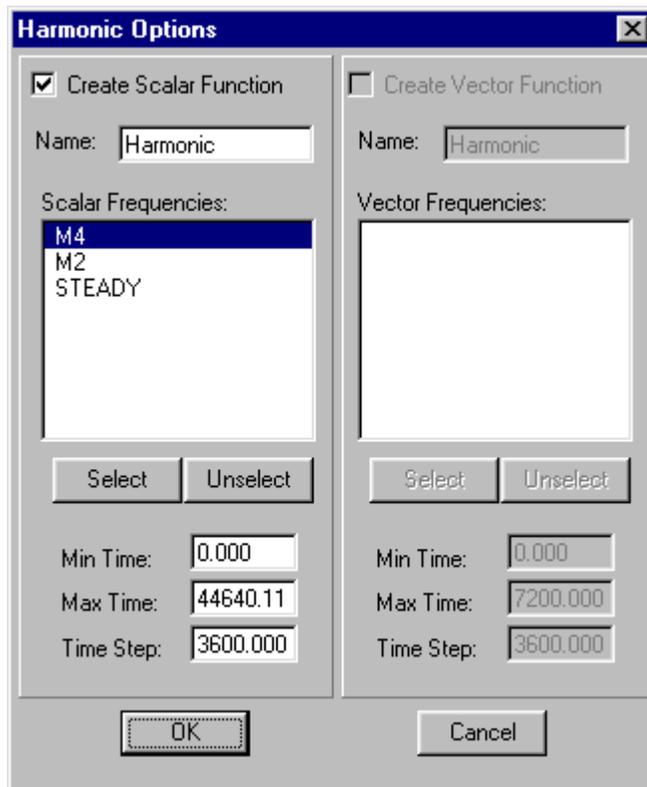


Figure 8.6 ADCIRC Harmonic Options dialog.

8.6 Model Check

The model check capability for ADCIRC is under development. Suggestions for items to check automatically can be sent to the developers.

8.7 Model Control

The *ADCIRC* numerical model requires several user-specified parameters to control the analysis. The *Model Control* command from the *ADCIRC* menu opens the *Model Control* dialog (see

Model Control

Project Title:

Run ID:

Non Fatal Error Override
 Abbreviated Output
 Echo Screen

Model Type
 2D DI Transport
 3D VS
 3D DSS

Initial Values
 Cold Start
 Hot Start 1 (unit 67)
 Hot Start 2 (unit 68)

Finite Amplitude Terms On
 Wetting/Drying
 Advective Terms On
 Time Derivative Terms On

Coordinate System
 Cartesian
 Spherical
 Gravity:

Generalized Wave Continuity:
 Lateral Viscosity: L²/T
 Lateral Diffusivity:

Hot Start Output
 Generate Output File
 # Time Steps:

Coriolis Option
 Constant
 Variable

Model Center
 Latitude: deg.
 Longitude: deg.

Figure 8.7). This dialog contains parameters that control the execution of ADCIRC. The parameter description for each field is displayed in SMS using the interactive help messages.

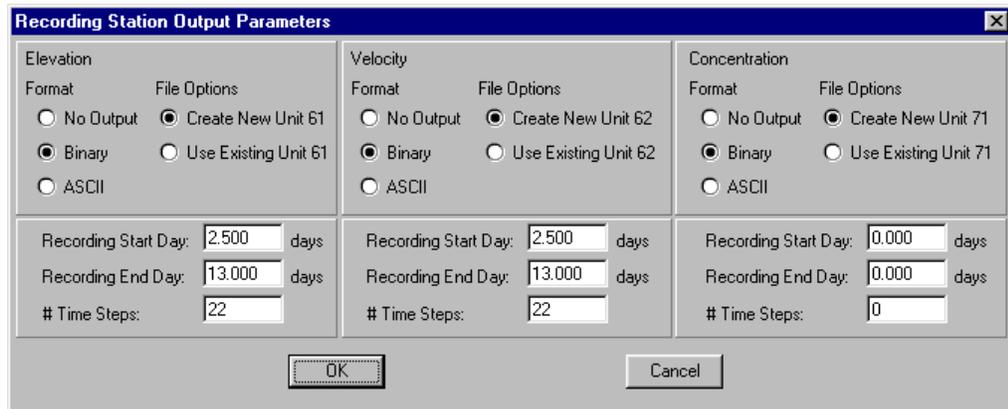
Figure 8.7 ADCIRC Model Control dialog.

The rest of this section explains the parameters available in this dialog.

The parameters that may be set in the Model Control are described in the ADCIRC documentation. A brief description of the dialogs brought up by clicking on the buttons in the model control follows.

8.7.1 Station Output

The *Station Output* button brings up the *Recording Station Output Parameters* dialog. The dialog controls the output at the recording stations for elevation, velocity, and concentration. The output will be formatted according to the *Format* option chosen. The *File Options* controls if the output is put into an existing file or a new file. The recording time section controls when the output will begin to be recorded and how often it will be recorded.



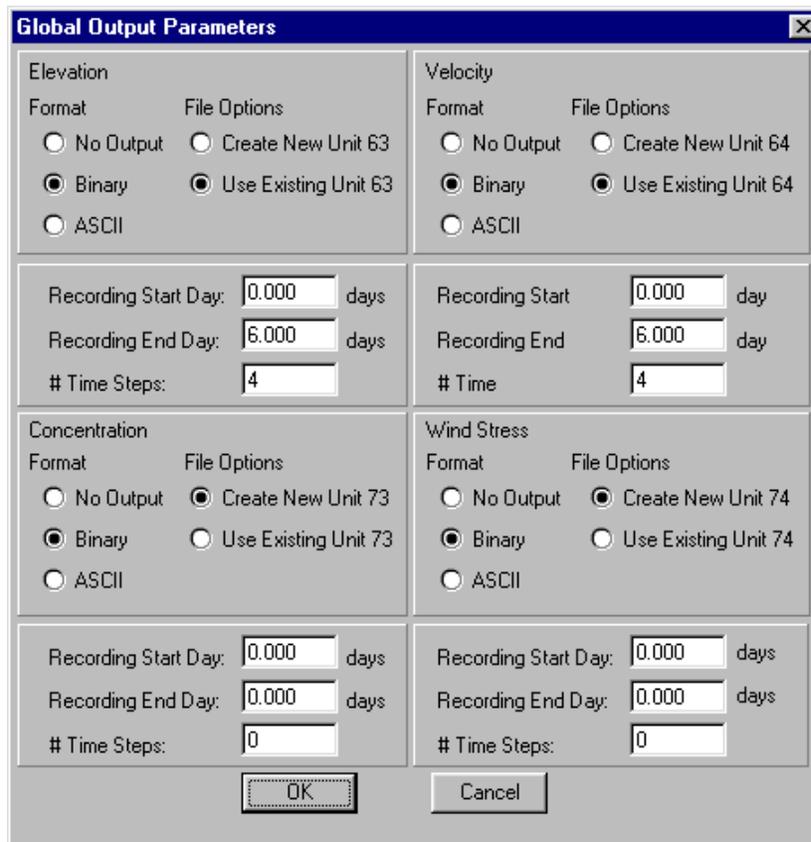
The dialog box, titled "Recording Station Output Parameters", is divided into three columns for Elevation, Velocity, and Concentration. Each column has a "Format" section with radio buttons for "No Output", "Binary", and "ASCII". The "File Options" section has radio buttons for "Create New Unit" (with unit numbers 61, 62, and 71 respectively) and "Use Existing Unit". Below these are input fields for "Recording Start Day", "Recording End Day", and "# Time Steps". The "Recording Start Day" and "Recording End Day" fields include a "days" label. At the bottom are "OK" and "Cancel" buttons.

Parameter	Format	File Options	Recording Start Day (days)	Recording End Day (days)	# Time Steps
Elevation	<input type="radio"/> No Output <input checked="" type="radio"/> Binary <input type="radio"/> ASCII	<input checked="" type="radio"/> Create New Unit 61 <input type="radio"/> Use Existing Unit 61	2.500	13.000	22
Velocity	<input type="radio"/> No Output <input checked="" type="radio"/> Binary <input type="radio"/> ASCII	<input checked="" type="radio"/> Create New Unit 62 <input type="radio"/> Use Existing Unit 62	2.500	13.000	22
Concentration	<input type="radio"/> No Output <input checked="" type="radio"/> Binary <input type="radio"/> ASCII	<input checked="" type="radio"/> Create New Unit 71 <input type="radio"/> Use Existing Unit 71	0.000	0.000	0

Figure 8.8 Recording Station Output Parameters dialog.

8.7.2 Global Output

The *Global Output* button brings up the *Global Output Parameters* dialog. The dialog controls the global output for elevation, velocity, concentration, and wind stress. The output will be formatted according to the *Format* option chosen. The *File Options* controls if the output is put into an existing file or a new file. The recording time section controls when the output will begin to be recorded and how often it will be recorded.



The **Global Output Parameters** dialog box is divided into four main sections for configuring output parameters:

- Elevation:**
 - Format: No Output, Binary, ASCII
 - File Options: Create New Unit 63, Use Existing Unit 63
- Velocity:**
 - Format: No Output, Binary, ASCII
 - File Options: Create New Unit 64, Use Existing Unit 64
- Concentration:**
 - Format: No Output, Binary, ASCII
 - File Options: Create New Unit 73, Use Existing Unit 73
- Wind Stress:**
 - Format: No Output, Binary, ASCII
 - File Options: Create New Unit 74, Use Existing Unit 74

Recording parameters for each section are as follows:

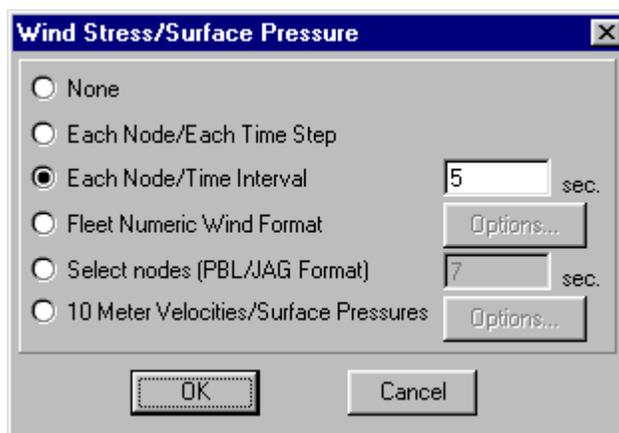
- Elevation/Velocity:** Recording Start Day: 0.000 days, Recording End Day: 6.000 days, # Time Steps: 4
- Concentration:** Recording Start Day: 0.000 days, Recording End Day: 0.000 days, # Time Steps: 0
- Wind Stress:** Recording Start Day: 0.000 days, Recording End Day: 0.000 days, # Time Steps: 0

Buttons: OK, Cancel

Figure 8.9 Global Output Parameters dialog.

8.7.3 Wind

The *Wind* button brings up the *Wind Stress/Surface Pressure* dialog. The dialog controls the reading of wind stress and surface pressures at nodes in the mesh.



The **Wind Stress/Surface Pressure** dialog box offers the following options:

- None
- Each Node/Each Time Step
- Each Node/Time Interval: 5 sec. (Options... button)
- Fleet Numeric Wind Format
- Select nodes (PBL/JAG Format): 7 sec. (Options... button)
- 10 Meter Velocities/Surface Pressures (Options... button)

Buttons: OK, Cancel

Figure 8.10 Wind Stress/Surface Pressure dialog.

Fleet Wind File Options

The *Fleet Numeric Wind Format Options* button brings up the *Fleet Wind File Options* dialog. The parameters of the fleet wind format numeric file can be set in this dialog.

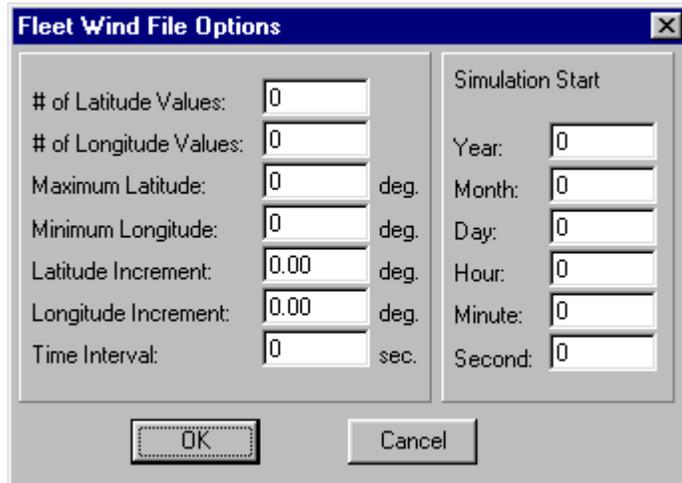


Figure 8.11 Fleet Wind File Options dialog.

Binary Wind File Options

The *10 Meter Velocities/Surface Pressures Options* button brings up the *Binary Wind File Options* dialog. The parameters of the wind file can be set in this dialog.

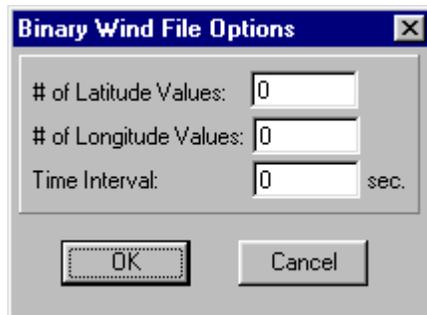


Figure 8.12 Binary Wind File Options dialog.

8.7.4 Time Control

The *Time Control* button brings up the *Time Control* dialog. The dialog controls the time settings for running ADCIRC.

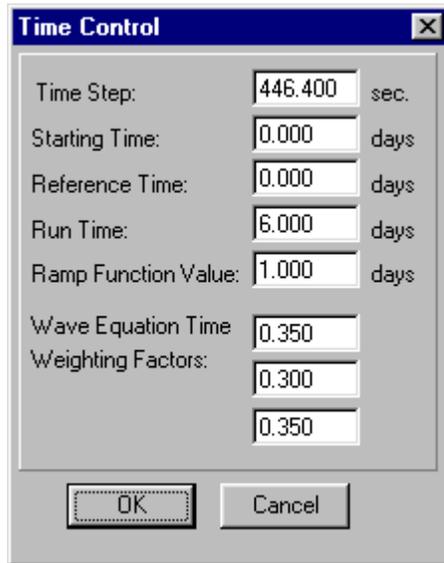


Figure 8.13 Time Control dialog.

8.7.5 Bottom Friction

The *Bottom Friction* button brings up the *Bottom Stress/Friction* dialog. The bottom friction can be set to constant or varying and linear, quadratic, or hybrid by choosing the options shown in Figure 8.14. If the *Hybrid* option is selected, the bottom friction can only be constant and the coefficient options can be set.

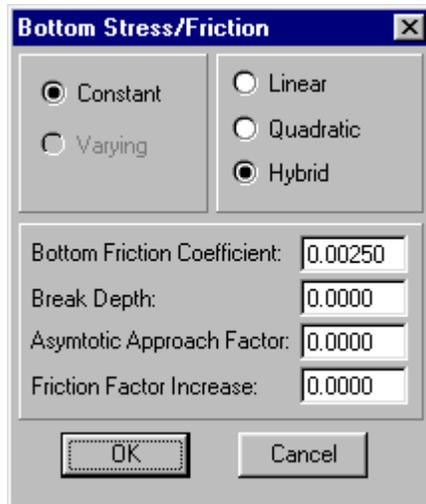


Figure 8.14 Bottom Stress/Friction dialog.

8.7.6 Tidal Forces

The *Tidal Forces* button brings up the *Tidal Functions* dialog. If the *Tidal Potential* is *Off*, the *Constituents* area is disabled. To create a tidal potential function, choose *On* and push the *New* button in the *Constituents* area. The parameters of the selected constituent can then be edited.

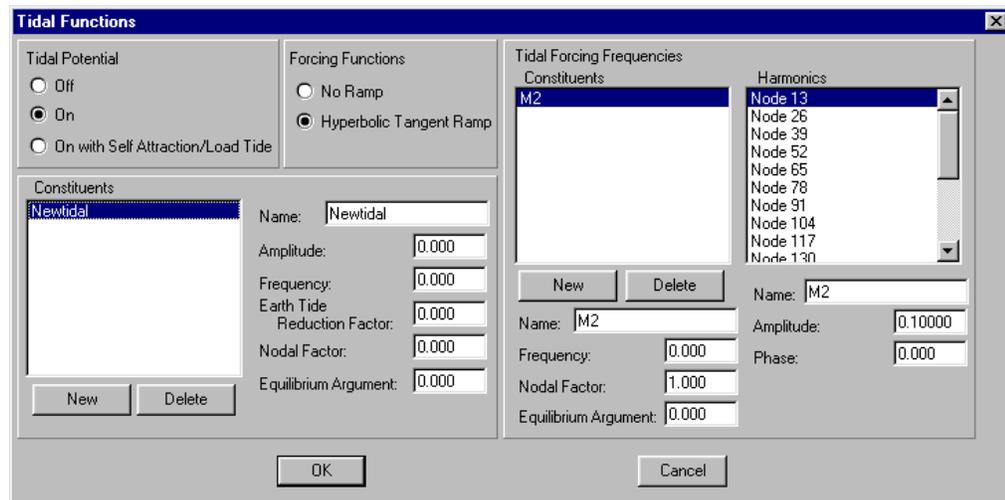


Figure 8.15 Tidal Functions dialog.

A constituent for the *Tidal Forcing Frequencies* can be created by pushing the *New* button. The parameters of the constituent can be edited. Nodes can be selected in the *Harmonics* section by clicking on the name. The amplitude and phase of the selected node can be edited in their respective fields.

8.7.7 Harmonic Analysis

The *Harmonic Analysis* button brings up the *Harmonic Analysis* dialog. Frequencies can be created by pushing the *New* button. A frequency can be selected by clicking on its name. The frequency, nodal factor, and equilibrium argument can be set for the selected frequency. The number of days after which the data starts to be harmonically analyzed and the timesteps can be set on the right side of the dialog. In the *Perform Harmonic Analysis* section, the location where the harmonic analysis will be performed can be set.

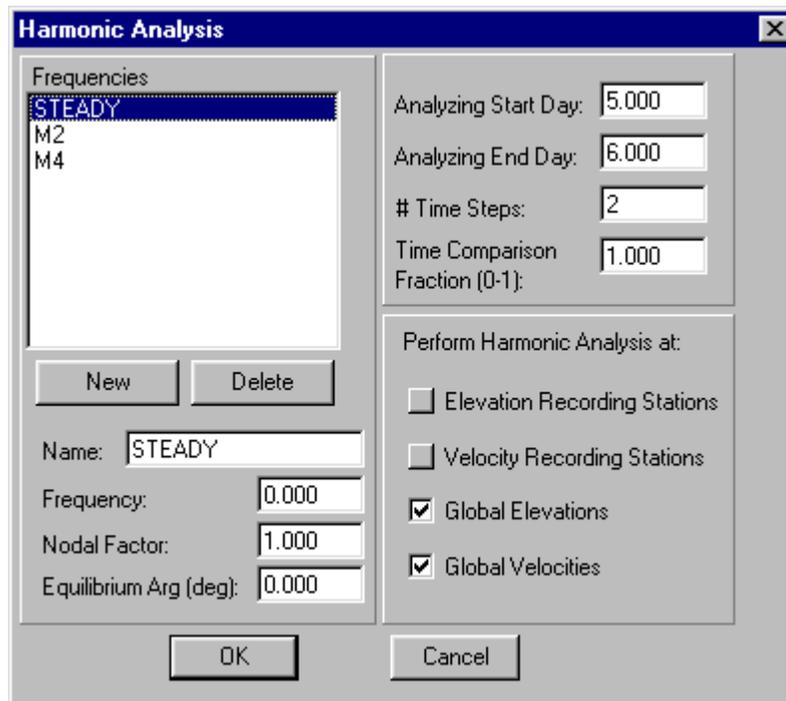


Figure 8.16 Harmonic Analysis dialog.

8.7.8 Solver

The *Solver* button brings up the *Solver Options* dialog. The *Absolute Convergence Criteria* and the *Maximum Number of Iterations* for each time step can be set in this dialog. The *Solver Type* allows the user to choose which type of solver will be used. The *Warning Messages* allows the user to set the error messages or information that will be generated.

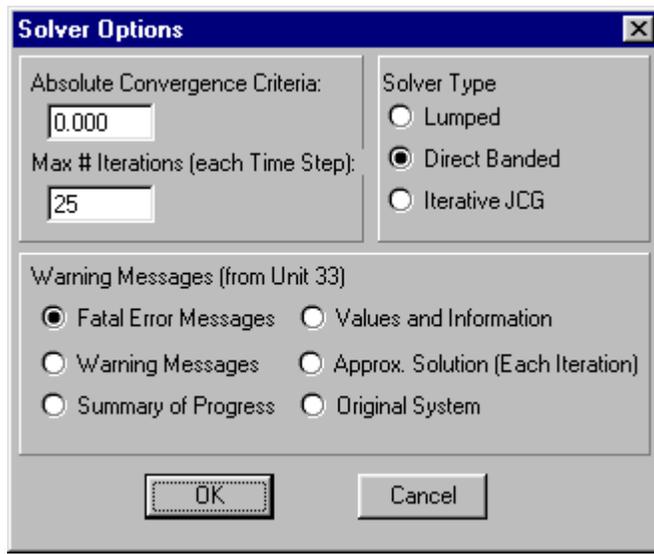


Figure 8.17 Solver Options dialog.

8.8 Run ADCIRC

After a simulation has been read or saved, *SMS* can launch the *ADCIRC* model. To run *ADCIRC*:

1. Select *ADCIRC / Run ADCIRC*. A dialog appears showing the engine that will execute as shown in Figure 5.16.
2. If you are given the message that the executable was not found, click the file browser icon and choose the correct version of the program that should run.
3. Click the *OK* button or press the *ENTER* key.

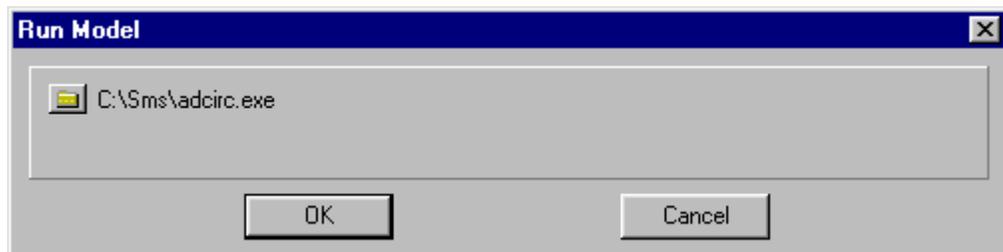


Figure 8.18 Run ADCIRC dialog.

A window will appear which displays information as *ADCIRC* runs.

CGWAVE Interface

CGWAVE (Panchang & Xu 1995) is a 2D finite element model based on the elliptic mild-slope wave equation. It is similar to wave models *HARBD* (Chen and Mei 1974) and *PHAROS* (Kostense et al. 1986). *CGWAVE* can simultaneously simulate the effects of refraction, diffraction, reflections by bathymetry and structures, dissipation due to friction and breaking, and nonlinear amplitude dispersion. The computational capabilities of *CGWAVE* model permit the modeling of large coastal regions. The governing equations of *CGWAVE* pass, in the limit, to the deep and shallow water equations, making this model applicable to a wide range of frequencies, including short wind waves, swell, and infra-gravity waves.

A mesh for use with *CGWAVE* must match one of three domain shapes including rectangular, semi-circular, and circular. *SMS* includes tools in the *Map Module* to create a network with a valid domain shape (see the tutorial for *CGWAVE*). Therefore, it is strongly recommended that the *Map Module* be the principal method for creation of new networks. The *Mesh Module* may be used to create a mesh as described in Chapter 4, however in the case of *CGWAVE* the *Mesh Module* primarily is useful for network editing and assigning model parameters required by *CGWAVE*. These model parameter commands are grouped in the *CGWAVE* menu and will be discussed later in this chapter.

Solutions generated by *CGWAVE* can be visualized in *SMS* in two ways. The most straightforward method is to import the solution file using the *Data Browser* and post-processing carried out as described in Chapter 3. This translator only creates wave amplitude, direction and phase functions. If additional functions are desired from the wave characteristics, they must be generated using a translating utility. An example of this is included in the *CGWAVE* reference manual. The translating utility creates generic data set files that can be imported using the *Data Browser*.

9.1 New Simulation

The *New Simulation* command in the *CGWAVE* menu deletes all of the *CGWAVE*-related data in *SMS*. This data includes the finite element mesh and boundary conditions, model control data, and any solutions that have been imported into *SMS*. This data is only removed from memory, not from the disk. Boundary conditions include the nodestring type and reflection coefficients. Model control data includes simulation titles, incident wave conditions, and solver type. Data such as scatter sets and feature objects are not removed from memory.

9.2 Open Simulation

The *Open Simulation* command in the *CGWAVE* menu reads in a data file that has been previously created and saved. The data file contains geometry and model parameters and usually has the “.dat” extension. Geometry includes the location of nodes that define element corners and the element connectivity of the nodes. Model parameters consist of boundary definitions, reflection coefficients, numerical controls, etc. Opening a new *CGWAVE* data file causes any existing data to be removed from memory.

9.3 Save Simulation

The *Save Simulation* command in the *CGWAVE* menu saves the data files required for executing the model. This includes the data file and a one-dimensional values file. The data file contains geometry and model parameters. The one-dimensional values file contains depths from the coastline out to open ocean. *CGWAVE* uses the file of one-dimensional values to generate boundary conditions on the open ocean boundaries. After saving a simulation, *CGWAVE* can be run directly from inside *SMS* as described in section 9.8.

9.4 Assign BC

The *CGWAVE Boundary Conditions* (Figure 9.1) is used to assign boundary conditions to individual nodestrings. Before assigning boundary conditions to nodestrings, at least one nodestring must be selected using the *Select Nodestrings* tool. To assign boundary conditions to the selected nodestring(s), select one of the *CGWAVE Type* options.

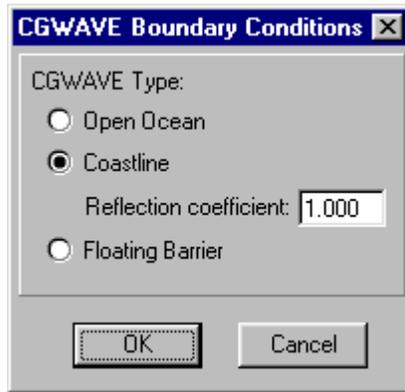


Figure 9.1 CGWAVE Boundary Conditions Dialog.

CGWAVE supports three types of boundary. *Open Ocean* boundaries delineate the region where waves will enter the domain. The attributes of the waves that enter the domain are defined as incident wave characteristics in the *Model Parameters* dialog (see Section 9.7.2). These values are propagated from an offshore location to the open ocean boundaries.

When assigning the *Coastline* condition, the *Reflection coefficient* should be set. The coastline reflection term reflects to what degree a section of coastline reflects incoming waves. Legal values vary from 0.0 for a gradual sandy incline to 1.0 for solid vertical rock wall.

When assigning the selected nodestring(s) to be the *Floating Barrier* type, the nodestring(s) must form a closed area. Elements inside of the *Floating Barrier* nodestrings will be treated as floating barriers when the simulation is saved.

9.5 Create Functions

The *Create Functions* dialog (Figure 9.2) is used to create useful function data sets (see Section 1.4) over the mesh geometry. Each selected function will be created when the *Ok* button is clicked. All of the available functions can be selected by pushing the *All On* button. All of the functions can be deselected by pushing *All Off*.

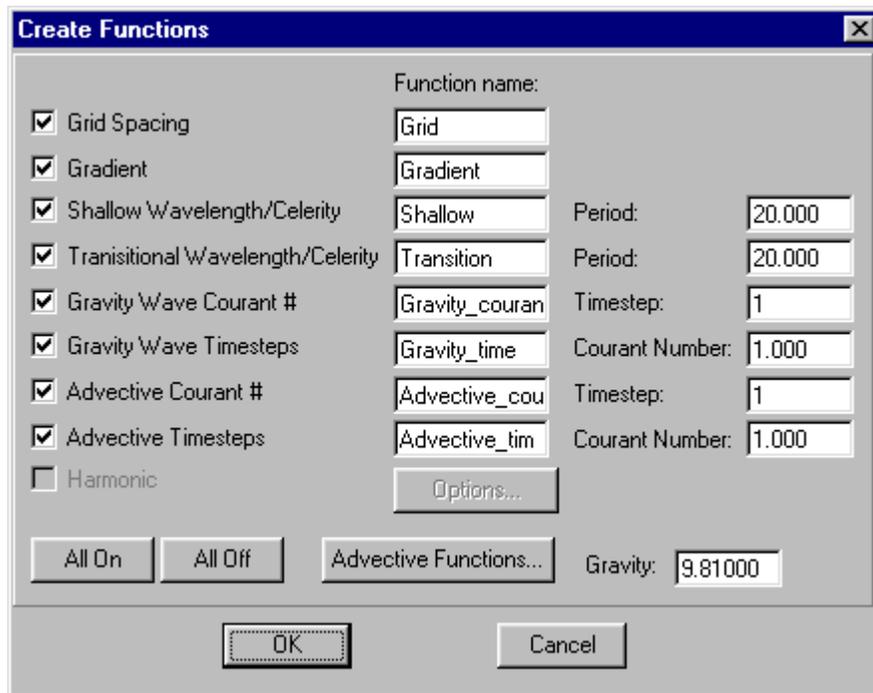


Figure 9.2 CGWAVE Create Functions Dialog.

The functions that can be created include:

- *Grid Spacing*. Creates a function that gives the average distance between a node and its neighbors.
- *Gradient*. Creates a function that gives the gradient at each node. The gradient is calculated as the inverse of the slope (run divided by the rise). This function is not computed for disjoint nodes.
- *Shallow Wavelength/Celerity*. Creates two functions that represent the celerity and wavelength at each node in shallow water. The celerity is calculated as: $Celerity = (Gravity * Nodal\ Elevation)^{0.5}$. The *Wavelength* is: $Wavelength = Period * Celerity$.
- *Transitional Wavelength/Celerity*. Creates two functions that represent the celerity and wavelength at each node for any depth. A recursive transitional function is used.
- *Gravity Wave Courant Number*. Creates a function that represents the courant number at each node for a user specified *Timestep* size. The equation is: $Courant\ Number = Timestep * (Gravity * Nodal\ Elevation)^{0.5} / Nodal\ Spacing$.
- *Gravity Wave Timesteps*. Creates a function that represents the gravity wave timestep required at each node to result in the user specified *Courant Number*.

SMS uses the equation presented above for the *Gravity Wave Courant Number*, solved for the timestep.

- *Advective Courant Number*. Creates a function that represents the courant number given the *Timestep* and a velocity function. The velocity function can be selected by clicking on the *Advective Functions...* button. This brings up a *Select Data Set* dialog that lists the vector functions currently in memory. The courant number is calculated as: $\text{Courant Number} = \text{Nodal Velocity Magnitude} * \text{Timestep} / \text{Nodal Spacing}$. This option is disabled if no vector functions exist.
- *Advective Timesteps*. Creates a function that represents the timestep required to result in the user specified *Courant Number* utilizing a velocity function. The velocity function can be selected as described above in the description of the *Advective Courant Number*. The equation is the same as for the *Advective Courant Number*, solved for the timestep. This option is disabled if no vector functions exist.
- *Harmonic*. This option is only available for the ADCIRC model.

9.6 Model Check

This is a future capability that will ensure the user has defined necessary parameters and satisfied conditions to run *CGWAVE*.

9.7 Model Control

The *CGWAVE* model requires several user-specified parameters to control the analysis. The *Model Control* command from the *CGWAVE* menu opens the *Model Control* dialog (Figure 9.3). This dialog contains parameters that control the execution of *CGWAVE*. The parameter description for each field is displayed in *SMS* using the interactive help messages.

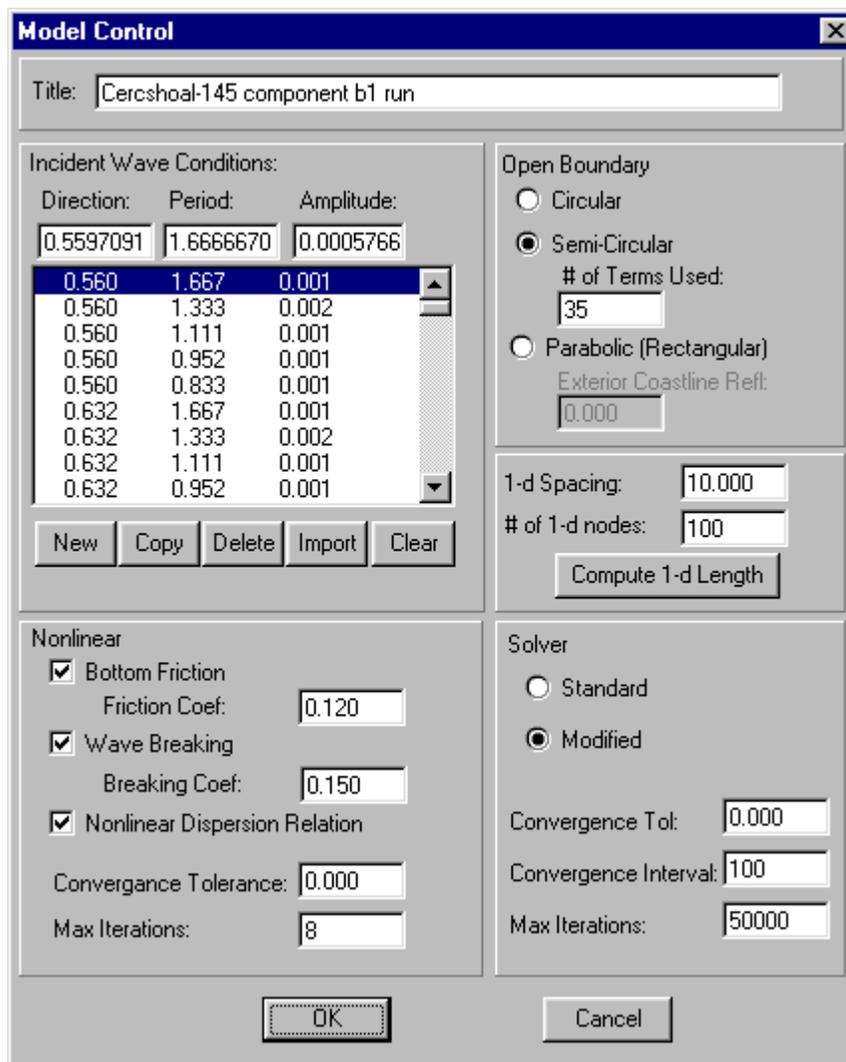


Figure 9.3 CGWAVE Model Control dialog.

9.7.1 Title

The *Title* section allows the user to specify a title for the project being modeled.

9.7.2 Incident Wave Conditions

The *Incident Wave Conditions* section allows the user to specify wave the directions, periods, and frequencies. These values are used as boundary conditions out to sea and are propagated to the open ocean boundaries on the domain using one-dimensional wave propagation. A set of values (one direction, frequency and period) can be selected by clicking in the window. The selected set can be via the edit boxes above the window. The buttons below the window can be used as follows:

New

The *New* button creates a new set below the selected set. The values are defaulted to zero.

Copy

The *Copy* button copies the selected set as a new set in the list. The new set is placed directly below the selected set.

Delete

The *Delete* button deletes the selected set. The rest of the sets are unaffected.

Import

The *Import* button allows the user to select a file containing *Incident Wave Conditions* data. *SMS* prompts the user to replace or append to existing data. See the *CGWAVE* documentation for the *Incident Wave Conditions* file format.

Clear

The *Clear* button deletes all of the *Incident Wave Conditions* data.

9.7.3 Open Boundary

The *Open Boundary* section allows the user to specify the open boundary type. This should match the type selected when generating the domain in the *Map Module*. If the mesh has been generated using other methods, the domain shape must match the *Open Boundary* type.

9.7.4 1-D

The *1-D* section allows the user to specify the number of nodes and the nodal spacing for the one-dimensional data that is used to distribute incident wave data to the open boundary. The file format is described in the *CGWAVE* user manual. The *Compute 1-d Length* button calculates the *1-d Spacing* variable such that the specified number of nodes will extend to the limits of the bathymetry data. This bathymetry data is defined by a scattered data set. Therefore, a scatter data set must exist in order for this button to be active. If no scatter data exists, the user must generate an appropriate file containing the one-dimensional depth values.

9.7.5 Solver

CGWAVE support two separate numerical solvers for robustness. Users select which solver to use via the *Solver* radio group. The *Standard* solver should be utilized first. If *CGWAVE* fails to converge, the *Modified* method can be utilized.

9.7.6 Iteration Control

CGWAVE allows the user to specify the maximum number of iteration to be performed by the numerical solver. This number is specified in the *Max Iterations* edit field. The solver will terminate after performing the maximum number of iterations or when the change in the solution is less than the convergence tolerance specified in the *Convergence Tol* edit field. It is recommended that the convergence value be between 1.0e-6 and 1.0e-9. *CGWAVE* will print a progress report on the tolerance at the user specified report interval entered in the *Convergence Interval* edit field.

9.7.7 Model Parameters

CGWAVE includes options to model bottom friction, wave breaking and nonlinear dispersion. These options may be enabled using the appropriate toggle box. See the *CGWAVE* user manual for more information about these parameters.

9.8 Run *CGWAVE*

After a simulation has been read or saved, *SMS* can launch the *CGWAVE* model. To run *CGWAVE*:

4. Select *CGWAVE / Run CGWAVE*. A dialog appears showing the engine that will execute as shown in Figure 9.4.
5. If a message that the executable was not found, or the path to the executable is not the desired version, the user may specify an executable to use. To do this, the user selects the file browser icon and selects the correct version of the program using the file browser.
6. Click the *OK* button or press the *ENTER* key to launch *CGWAVE*.

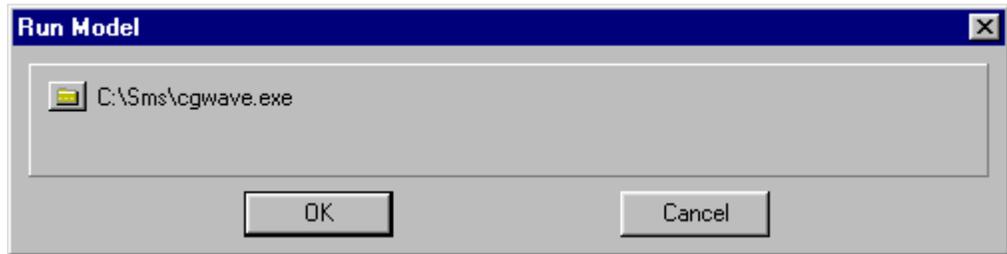


Figure 9.4 Run CGWAVE dialog.

A window appears which displays information as *CGWAVE* runs. You will be asked to enter the input filename (the “.dat” file) and the output wave potential filename.

FESWMS Interface

FESWMS is a hydrodynamic modeling code that supports both super and subcritical flow analyses, including area wetting and drying. It has been developed under funding by the U.S. Federal Highways Administration (FHWA) by Dr. Dave Froelich PE. *FESWMS* is specifically suited for modeling regions involving flow control structures, such as are encountered at the intersection of roadways and waterways. Specifically, the *FESWMS* model allows the user to include weirs, culverts, drop inlets, and bridge piers into a standard 2D finite element model. *SMS* provides graphical tools for defining these structures and controlling analysis using the *FESWMS* model. Both pre- and post-processing capabilities are included in the interface.

The *FESWMS* version 2.x package consists of three programs: DIN2DH, FLO2DH, and ANO2DH. Earlier versions of *FESWMS* referred to these programs as DINMOD, FLOMOD, and ANOMOD. DIN2DH and ANO2DH are non-interactive programs for mesh generation and plot generation, respectively. When using *SMS*, only the FLO2DH program is used. The FLO2DH program is the analysis engine of *FESWMS*. *SMS* supports version 1.x, 2.x and 3.x of the *FESWMS* analysis package.

This chapter describes the commands used to create and edit the *FESWMS* specific boundary conditions, flow control structures, run parameters, etc., included in the *FESWMS* menu. The commands for generating and editing the mesh are described in Chapter 4. Once the analysis is complete, post-processing of the analysis results is performed by importing the solution file into the *Data Browser*. (See Lesson 6 of the *SMS Tutorial* and the *FESWMS User Manual* for more about running FLO2DH).

10.1 Open Simulation

The *Open Simulation* command reads in a FLO2DH file, which contains a mesh that has been previously created and saved. File I/O for a *FESWMS* problem is controlled through a project file which typically has the file extension *.fil*. This file contains the names of ten other files that may or may not be required for an analysis. The order these filenames appear in the project depends on the version of *FESWMS* being used. The order for version 1.x and 2.x is:

FLOMOD (FESWMS version 1.x)	FLO2DH (FESWMS version 2.x)
Control data file - ".dat"	Control data file - ".dat"
Printed output file - ".pnt"	Grid network file - ".net"
Grid network file - ".net"	Initial flow file - ".ini"
Initial flow file - ".ini"	Boundary condition file - ".bnd"
Boundary condition file - ".bnd"	Wind data file - ".wnd"
Wind data file - ".wnd"	Printed output file - ".pnt"
Solution output file - ".out"	Solution output file - ".out"
Restart/recovery file - ".rsr"	Restart/recovery file - ".rsr"
Upper matrix decomposition file - ".upp"	upper matrix decomposition file - ".upp"
Lower matrix decomposition file - ".low"	lower matrix decomposition file - ".low"

The order for version 3.x may still be subject to change. Whether a particular file is required depends on the options specified by the user and stored in the control data file. *SMS* has been designed to easily control these options. As a new FLO2DH file is read, any previous mesh data is deleted from memory. If this data has not been saved since it was last edited, a warning message will appear. The name of the current FLO2DH file will be displayed at the top of the *Main Graphics Window*.

The files that are (or can be) used by FLO2DH or FLOMOD include:

- Control data file - ".dat" - contains run control parameters and data.
- Grid network file - ".net" - lists nodes and element connectivity's.
- Initial flow file - ".ini" - specifies initial flow conditions. Same format as a solution file created by *FESWMS*.
- Boundary condition file - ".bnd" - specifies the flows, water surface elevations etc. at mesh boundaries
- Wind data file - ".wnd" - specifies wind conditions.
- Printed output file - ".pnt" - specifies name of the file created by *FLO2DH*.
- Solution output file - ".out" - specifies name of solution file to be created. Contains output data at each iteration or time step as specified by the user.

Solution files contain velocities and water surface elevations at each node. For dynamic solutions derivatives of these quantities are also specified, and a solution at each time step is recorded.

- Restart/recovery file - ".rsr" - solution written at a specified interval for hotstarting future runs. (We recommend that this file not be used)
- upper matrix decomposition file - ".upp" - used to compute the solution.
- lower matrix decomposition file - ".low" - used to compute the solution.

The formats of these files are discussed in the *FESWMS Users Manual*. All input data may be contained in the control file. The optional files exist to allow the user to break up the data for multiple variations of a problem. The control file includes a card which specifies which of the other files are being utilized by *FLO2DH*. To specify the use of optional files, see the *FESWMS CONTROL* section later in this chapter.

10.2 Save Simulation

The *Save Simulation* command causes the *Save FESWMS* dialog to appear. This command will save a project file (extension .fil) as well as the other files the user has specified in the *FESWMS CONTROL* dialog.

10.3 Nodal Boundary Conditions

The *Assign BC* command is used to specify boundary conditions at selected nodes or nodestrings. Nodal boundary conditions are assigned if the *Select Node* tool has been selected. Flow rates and water surface elevations may only be specified at nodes on the mesh boundary. However, a water source or sink may be specified at any node in the mesh. Before choosing this menu item, at least one node must be selected. In the *FESWMS Nodal Boundary Conditions* dialog (see Figure 10.1), the user may select any combination of boundary conditions.

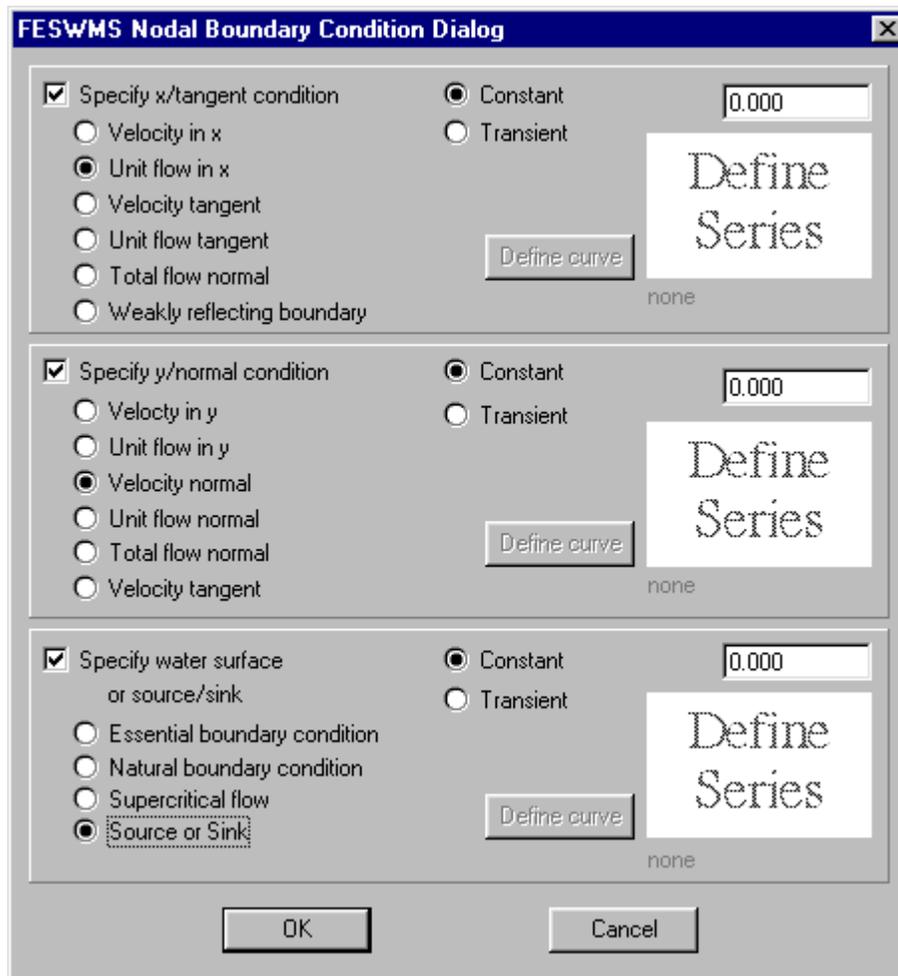


Figure 10.1 FESWMS Nodal Boundary Condition Dialog.

10.3.1 Specify x/Tangent Condition

The first type of nodal boundary condition is the velocity or flow in the x direction or tangent to the boundary. The options are as follows:

- Velocity/unit flow can be specified with respect to the positive x-axis.
- Velocity/unit flow can be specified *tangent* to the mesh boundary at the node.
- Total flow *normal* to the *open* boundary at that node can be specified.
- *Weakly reflecting boundary invariant* is a constant used to indicate the relationship of flow between outflow and tidal cycles.

10.3.2 Specify y/Normal Condition

The second type of nodal boundary condition is the velocity or flow in the y direction or normal to the boundary. The options are as follows:

- Velocity/unit flow can be specified with respect to the positive y-axis.
- Velocity/unit flow can be specified *tangent* to the mesh boundary at the node.
- Total flow *normal* to the *closed* boundary at that node can be specified.
- Velocity *tangent* to the *Open* boundary at that node can be specified.

10.3.3 Specify Water Surface or Source/Sink

If the selected node is on the mesh boundary, the water surface elevation or supercritical flow can be specified at the node. Otherwise, the node is on the interior of the mesh and only a source or sink is allowed. Water surface elevation can be specified as either an essential boundary condition or a natural boundary condition. If the water surface elevation is essential, *FESWMS* will not allow the value to fluctuate at all. If it is natural, small fluctuations are allowed. If supercritical flow is chosen, no water surface elevation value is entered. If a sink exists at the node, a negative value should be entered. Otherwise, a positive value should be entered.

10.4 Boundary Section

The *Assign BC* command is used to specify boundary conditions at selected nodes or nodestrings. Nodestring boundary conditions are assigned if the *Select Nodestring* tool has been selected. Boundary Sections are used for specifying the conditions existing at an open boundary. Inflow, outflow, and water surface elevation may be specified on a selected nodestring which lies on the mesh boundary. The conditions along the boundary are specified using the Boundary Section dialog (see Figure 10.2).

10.4.1 Flow

Flow defined with a nodestring at an open boundary is always considered to be perpendicular to the mesh boundary when using *FESWMS*. However, it can be defined as an inflow or an outflow. An inflow is defined by positive values, while an outflow is defined with negative values. *FESWMS* version 2.x allows the user to specify flow either as direct flow or as a flow across a weakly reflecting boundary for tidal

situations (see the FESWMS version 2.x reference manual). If a user specifies a weakly reflecting flow boundary, and saves the data as a FESWMS version 1.x file, the boundary condition is converted to a regular flow boundary condition.

10.4.2 Water Surface Elevation

As with nodal boundary conditions, water surface elevation defined at a nodestring is defined as either essential or natural. The water surface elevation option also allows for the definition of supercritical flow existing at the boundary, for which no values will be defined.

If either an essential (no fluctuation allowed) or natural (small fluctuations allowed) water surface elevation is chosen, the value can be constant or it can vary across the boundary. If the water surface elevation varies, initial and end values are specified. From these values the water surface elevation will be interpolated to each node of the nodestring.

Water surface elevation may also vary along a string or be supercritical.

10.4.3 Rating Curve/Friction Slope

The *Rating curves* and *Friction slope* items in the *FESWMS Nodestring Boundary Conditions* dialog should not be used for *FESWMS* version 1.x. A rating curve describes the relationship between water surface elevation and flow rate, and may be defined at a nodestring which has an essential or natural water surface elevation defined. This means that the water surface elevation at each node of the nodestring will depend on the flow at that node. Up to eight values can be entered for a specific rate curve. If the friction slope option is chosen, the water surface elevation applied to each node of the nodestring will be calculated using the slope-area method.

Figure 10.2 FESWMS Nodestring Boundary Condition Dialog.

10.5 Initial Conditions

Initial Boundary Conditions help the model to converge. Initial Boundary Conditions can be specified by the user, or a previous solution can be used as described in section 10.15.3. To set specific Initial Boundary conditions, the *Set Initial Conditions* dialog is used. The velocity in x, velocity in y, and the water surface elevation can all be set either as constant or with a pre-defined data set. When an initial conditions file is saved, it is automatically set as the default ini file in the *FESWMS Control* dialog.

10.6 Wind Conditions

Global wind conditions may be applied to the model though the *FESWMS Wind Conditions* dialog (see Figure 10.3). If a steady-state solution is being performed, only constant wind values can be entered. Time variant or constant values can be entered if

a time-dependent solution is to be performed. (See FESWMS Control, section 10.15 for more about changing the solution type).

Figure 10.3 FESWMS Wind Dialog.

The units of wind magnitude and air density depend on the units being used (SI or English). The wind direction is entered in degrees counter-clockwise from the positive x-axis. The three wind constants are used in computing the stress coefficient at the water's surface (see the *FESWMS* users manual).

As with nodal initial conditions, *SMS* is currently not supporting an interface for nodal wind conditions. However, a wind file may be created using the format described in the *FESWMS* users manual. As with the initial condition file, the wind file flag should be set in the model control file, and the correct filename should be specified in the *.fil* file.

10.7 Weirs

Weirs can be defined on the boundary or interior to the mesh. One or two nodes must be selected before defining a weir. If the water which flows over the weir does not return to the mesh, only the upstream node is needed. However, if the water is to return to the mesh, two nodes should be assigned to the weir.

The upstream node should have a higher water surface elevation than the downstream node, as water flows over a weir only in the direction from the upstream node to the

downstream node. Also, the weir's crest elevation should be lower than the upstream node's water surface elevation, or water will not be able to flow over the weir.

After a weir is defined, it is displayed according to the defined attributes (see page 10-19 to learn how to change the display). If two nodes are used to define a weir, a line is drawn in the weir attribute color from the upstream node to the downstream node.

10.8 Culverts

Culverts can be defined on the boundary or interior to the mesh. One or two nodes must be selected before trying to assign a culvert. If the water which flows through the culvert does not return to the mesh, only the upstream node is needed. However, if the water is to return to the mesh, two nodes should be assigned to the culvert.

The geometry of the culvert is defined by the cross sectional area and barrel length. Entrance loss, hydraulic radius, and Manning's roughness coefficient establish how well water flows into and through the culvert.

If the flap-gate option is chosen, the culvert will be treated as though only the upstream node was selected. In this case, water will leave the mesh but will not return.

10.9 Drop Inlets

Drop inlets can be defined on the boundary or interior to the mesh. One or two nodes must be selected before trying to assign a drop inlet. If the water which flows over and through the drop inlet does not return to the mesh, only the upstream node is needed. In this case, the hydraulic head elevation at the drop inlet spillway should be specified. If the water is to return to the mesh, two nodes must be assigned to the drop inlet.

The weir parameters are used for computing flow at the opening of the drop inlet if it is not fully submerged. If the entrance is submerged, entrance flow will be computed using the orifice parameters. Flow inside the drop inlet is computed using the conduit parameters. All parameters are necessary except for the hydraulic energy head.

10.10 Piers

To define piers, a mesh must already exist. In the *Piers* dialog, the pier coordinates, drag coefficient, and shape factors are defined. To add a pier, either click the *Add Pier* button or, if any field of the last pier is highlighted, hit the *DOWN ARROW* key. To delete a pier, select any cell on the pier's row and click the *Delete Pier* button.

10.11 Node Ceilings

Nodes can be assigned to have a ceiling value, or a value above which water flow is prohibited. This is useful for modeling river flow under a bridge. If the water surface is above the ceiling value at a specific node, pressure flow will result. Otherwise, flow at the node is modeled as free-surface flow.

To assign a ceiling value the user selects the nodes for which a ceiling is desired and selects the *Ceiling* command from the *FESWMS* menu. *SMS* will then prompt for the ceiling value.

SMS creates a data set for the ceiling elevations when a *FESWMS* file is input or the node ceiling values are edited. This data set can be selected through the data browser (see Section 3.1) and viewed as contours. This facilitates visualization of the ceiling values.

10.12 Flux String

A flux string is a nodestring at which continuity checks will be made during a solution run. Flux strings are generally placed at cross sections of the model. To define a flux string, at least one nodestring must be selected. Then the user selects the *FLUX String* command from the *FESWMS* menu. The selected nodestring(s) will be assigned as a flux string(s). The color of flux strings can be set in the *Continuity strings* option of *Nodestrings options* of the *Display Options* dialog.

10.13 Material Properties

FESWMS allows the user to set model specific material properties using the *FESWMS Material* dialog (see Figure 10.4). The user can specify the color, material name, and other general material parameters using the *General Materials* dialog.

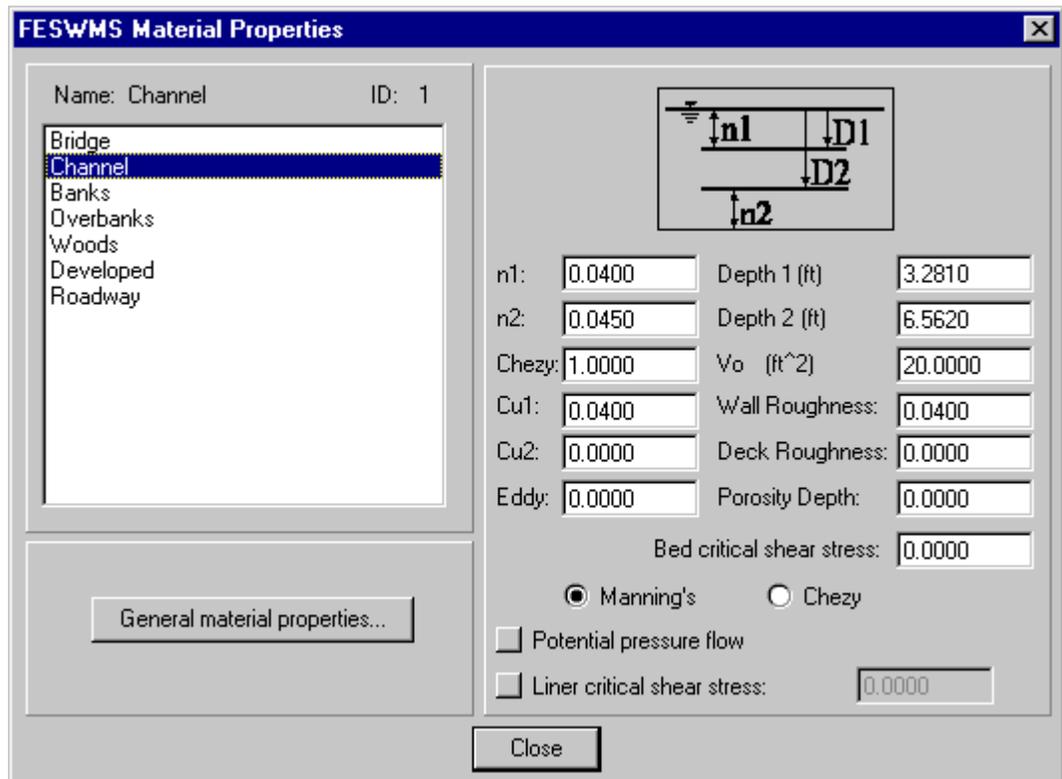


Figure 10.4 FESWMS Material Editor Dialog.

D1, n1, D2, n2

FESWMS supports Manning roughness coefficients that vary with depth. The first roughness coefficient (n_1) will be applied to all water depths less than the first depth (D_1). The second roughness coefficient (n_2) will be applied to all water depths greater than the second depth (D_2). If a value of zero is entered, n_1 will be applied to the entire system.

Other Coefficients

Other coefficients are applied to the entire model regardless of water depth. The Chezy discharge coefficient is used to calculate the bed friction coefficient. V_o and C_μ are used to calculate eddy viscosity.

General Material Properties

In the general *Materials* dialog, materials are created and deleted. Display colors of each material is set, as is the material name. After materials are created by using this option, they can be edited in the *FESWMS Material Editor*.

10.14 Model Check

The *FESWMS Model Checker* will check the mesh, making sure that all elements have material properties assigned and are properly defined, as well as checking items specific to *FESWMS*. By selecting the *Checker Options* button, items to be checked may be selected, both for *FESWMS* input data and for geometry data. If the *Check Boundary Conditions* option is chosen, the existence of boundary flow and water surface elevation will be checked. The *Check Water Surface Elevation* option will check if all defined water surface elevations are higher than the lowest nodal elevation.

After desired options are selected, press the *Run Check* button to check the current mesh. After a mesh has been checked, error messages may be saved by clicking on the *Save Messages* button. This dialog may be kept open while the mesh is edited, or it can be closed. (Section 5.7 discusses and illustrates the operation of the *Model Check*).

10.15 FESWMS Control

The user can specify what options will be used for the numerical analysis through the *FESWMS Control* dialog (see Figure 10.5).



Figure 10.5 FESWMS Control Dialog.

10.15.1 Project Title

The *Project Title* allows the user to name the problem under consideration. It is saved with the control data file, and is written at the top of the *Main Graphics Window*.

10.15.2 FESWMS Version

The *File Version* corresponds to the version of *FESWMS* that will be used for running a solution. Mesh data should be saved accordingly.

10.15.3 FLO2DH Input

The *FLO2DH Input* options tell SMS which files to use to compute a solution. The files created by SMS include the network file, boundary conditions file, a time file, and

a wind file. External files that can be used include a restart file, and an initial conditions file. SMS does not currently support sediment transport, or initial sediment files. The restart file can be created while the solution is being performed. In the event that the program is interrupted, this allows the solution to continue where it left off. To continue the solution later, the restart file needs to be specified in the input section of this dialog.

The initial conditions file can be created using the *create Initial Conditions* dialog see section 10.5 or a solution file can be used. If a model will not converge an initial conditions file will often help the model to converge. First a solution with identical geometry, but with an alternative flowrates, water surface elevations, or viscosities is computed. This solution file can be selected as the INI file in the *FESWMS Control Dialog*.

10.15.4 FLO2DH Output

The files that FLO2DH will give as solutions are set in the FLO2DH Output. LUD Matrix files are ??? If using FESWMS version is three, then scalar and vector output files can be specified. By using the options dialog the specific results that make up the scalar and vector solution files can be chosen. If the FESWMS version is one or two a general solutions file is automatically set as the default output.

A restart/recovery file can also be written while a solution is being performed. In the event that the program is interrupted, this allows the solution to continue where it left off. The solution is continued by using the restart/recovery file.

10.15.5 Units

The units for the model are specified as English or metric. Metric units are the default. If the units are changed, gravity constants and air densities are converted. However, initial conditions are not converted. It is a good idea to check all specified values if the units are changed after the values have been defined.

10.15.6 Solution Type

The solution type is specified as either steady state or dynamic. If a steady-state solution is to be performed, all boundary conditions such as wind, flow, and water surface elevation should be defined as constant. If a dynamic solution is to be performed, all boundary conditions may be either constant or time-dependent.

10.15.7 Slip Conditions

A slip condition will be specified for all closed boundaries during the run. If *Slip* is chosen, no shear stress will be applied to closed boundaries. *No Slip* indicates that the

shear stress at closed boundaries is so great that tangential velocity is zero. *Semi-slip* allows for slip at a closed boundary unless the flow is against a vertical wall. If the *semi-slip* option is chosen, a value should be entered in the *Vertical wall shear coefficient* in the *FESWMS Parameters* dialog.

10.15.8 Higher Order Integration

Low order integration is used as the default for all calculations unless otherwise specified. The *Curved* option tells *FESWMS* to use high-order integration only when dealing with curve-sided elements. The *All* option tells *FESWMS* to always use high-order integration. Using high-order integration gives more precise solutions, but also takes longer to perform the analysis.

10.15.9 Bottom Stresses

Either Manning's Equation or the Chezy Equation is specified for analyzing shear stresses at the water bed. The differences between these two algorithms are described in any basic fluid mechanics book.

10.15.10 Control Buttons

Wind Dialog

The *Wind* button opens up the *FESWMS Wind* dialog. (See Section 10.15.10 for information about this dialog).

FESWMS Parameters

The *Parameters* button causes the *FESWMS Parameters* dialog (see Figure 10.6) to appear. This dialog allows a user to edit optional run control parameters.

Parameter	Value
Water-surface elevation:	50.000
Average water density	1.937
Minimum continuity normal to be flagged:	3.403e+0
Vertical wall shear coefficient:	0.000
Unit flow convergence:	0.000
Water depth convergence	0.000
Number of ranked changes:	1
Default porosity depth	0.000
Element drying / wetting	<input type="checkbox"/>
Depth tolerance for drying (m):	0.500
Location:	
Latitude:	0.000
Longitude:	0.000
Angle:	0.000
FLO2DH Task:	
Check Network	<input type="checkbox"/>
Resequence	<input type="checkbox"/>
Coriolis effect	<input type="checkbox"/>
beta0:	1.000
cBETA:	0.000

Figure 10.6 FESWMS Parameters Dialog.

Optional parameters for *FESWMS* that are specified in the *FESWMS Parameters* dialog include:

Water-surface elevation

The default water-surface elevation (wse) entered will be assigned to all nodes which do not have an assigned boundary wse. The units of wse depend on the solution type and should be automatically converted if the unit base is changed. When using English units, enter a value in feet; for SI units, enter a value in meters.

Average water density

Average density of the water flowing through the mesh should be entered according to the solution units (English or SI). The current units are displayed next to the input field.

Minimum continuity normal to be flagged

The *Minimum continuity normal to be flagged* data entry allows for the marking of continuity norms greater than the value indicated. Any such continuity norms will be

marked with an asterisk in the printed output file. The default value is 3.4e38, meaning continuity norms will not be flagged in the file.

Vertical wall shear coefficient

If a semi-slip condition is specified and the shear at a vertical wall is the same as the bed shear, this value should be specified as zero. Otherwise, the shear coefficient value should be entered.

Element drying/wetting

The *Element drying / wetting* toggle is used to control how *FESWMS* handles changing water surface elevation. If this toggle is not selected, any element that is not fully submerged during a solution will be excluded from the analysis. If it is selected, the specified tolerance will be used to determine if a partially submerged element should or should not be included in the analysis. If the numerical model determines that an element has dried out, the flow that was passing through that element is passed over to elements that are still wet. Therefore, if too many elements get classified as dry in a single iteration, the water surface elevations in the remaining elements may go up enough to incorrectly make the dry elements wet again. This shifting back and forth leads to numerical instability in the solution. By specifying the *Depth Tolerance*, the elements will dry out in a more distributed fashion, minimizing the risk of instability.

LOCATION

Location is given by Latitude, Longitude, and angle. The angle is defined as the angle between the positive x-axis and true north. It should be given in degrees clockwise from true north. For example, if true north is along the positive y-axis, a value of 90 should be entered.

Effect of Coriolis force

If an angle other than 0 is entered, the effect of the Coriolis force will be considered. The angle is the average latitude in degrees of the model. A positive angle should be entered if the water body is in the Northern Hemisphere, and negative if it is in the Southern Hemisphere.

beta0 and cBETA

These coefficients are used for computing the momentum flux correction coefficients. Default values are $\text{Beta0} = 1.0$, $\text{cBETA} = 0.0$. The momentum flux correction coefficients are used to determine the change in water velocity with change in depth. The default values mean that such velocity changes are negligible, which is generally the case except in very shallow water.

FESWMS Iterations

The number of iterations used during the analysis is controlled via the *FESWMS Iterations* dialog (see Figure 10.7) which is invoked from the *FESWMS Control* dialog.



Figure 10.7 *FESWMS Iterations Dialog.*

The *Relaxation Factor* is used by *FESWMS* to scale the correction at the end of each Newton iteration. The default value is 1. If a model demonstrates slow convergence, it may be sped up by increasing this factor (up to 2). This should be used with caution however, as increasing the factor may lead to instability. Likewise, if a model diverges, the relaxation factor may be decreased to increase the numerical stability. A lower value will give a slower but more stable solution run.

The number of *iterations* is the number of Newton iterations performed unless the solution converges before the iterations are completed.

If a time dependent solution is being run, additional fields are provided to specify time dependent parameters. These are the starting time, run time, and time step size. A time integration factor (between 0 and 1) should also be specified. The default integration factor is 0.667.

FESWMS Print

The *Print* button at the lower right corner of the *FESWMS Control* dialog brings up the *FESWMS Print Options* dialog (see Figure 10.8).

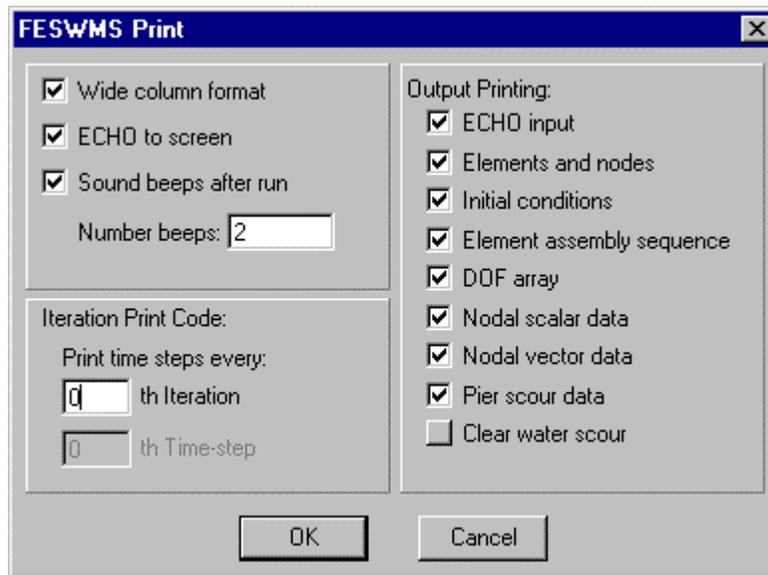


Figure 10.8 The FESWMS Print Dialog

Wide Column Format and ECHO to Screen

Wide column format means that the output will be printed in 132-column format. If this option is not selected, output will be printed in 80-column format. Echo to screen indicates that as a solution is run, messages will be written to the screen describing the status of the solution. Otherwise, such messages will not be seen.

Output Printing Options

When a solution is run, an output file is created. This output file contains control data, error messages, and solution results. Options in this section control other data which may or may not be printed. For example, if *ECHO Input* is selected, everything read from data records will be echo printed. If *Froude Number at Nodes* is selected, the Froude number for each node will be calculated and printed at each specified iteration/time step (see *Iteration Print Code* in the next paragraph).

Iteration print code

These fields specify how often the *Output Printing Options* will be written to the printed output file. For a Steady-state solution, only an iteration interval can be specified. For a Time Dependent solution, output printing is based on iteration intervals and specified time step values.

10.16 FESWMS Display Options

There are various boundary condition display options that can be set when dealing with the *FESWMS* model. As discussed previously, boundary conditions are assigned

to either nodes (section 10.3) or nodestrings (section 10.4). The display of these is accessed through the *Mesh Display Options* dialog (section 4.5).

10.16.1 Nodal Display Options

Next to the *Nodes* toggle box in the *Mesh Display Options* dialog, there is a button named *Options*. This button opens the *FESWMS Node Display Options* dialog (see Figure 10.9).

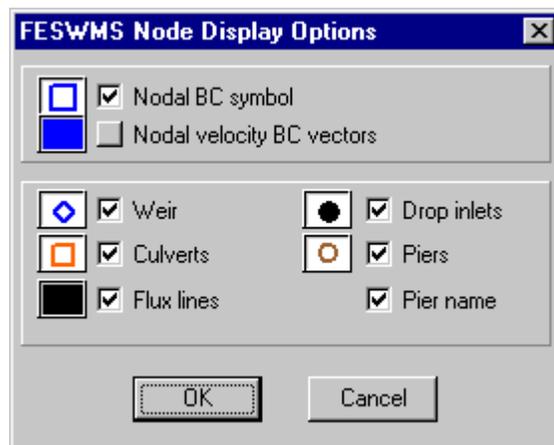


Figure 10.9 FESWMS Node Display Options Dialog.

The top section of this dialog contains the following options:

- *Nodal BC Symbol*. This controls the symbol drawn at all nodes where nodal boundary conditions have been defined, whether those boundary conditions be velocity, flow, head, source, or sink. Click the box to change the symbol attributes.
- *Nodal velocity BC vectors*. This controls the vector drawn at all nodes where velocity or flow boundary conditions have been defined. Click the box to change the line attributes.

The rest of this dialog contains symbols that are drawn when the specific entity is created. All of these except for the Piers are created at nodes. Piers are created anywhere in the mesh.

10.16.2 Nodestring Display Options

Next to the *Nodestring* toggle box in the *Mesh Display Options* dialog, there is a button named *Options*. This button opens the *FESWMS Nodestring Display Options* dialog (see Figure 10.10).

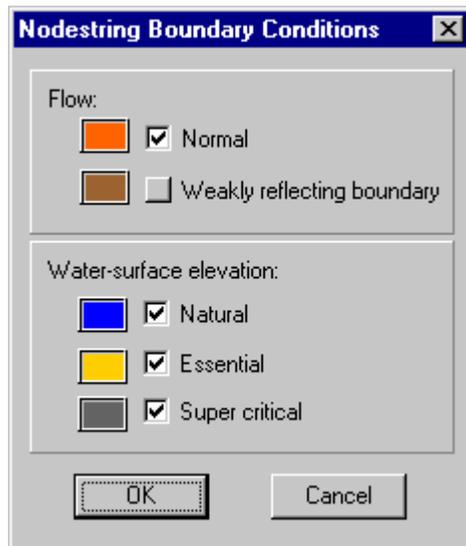


Figure 10.10 FESWMS Nodestring Display Options dialog.

This dialog contains the display settings for all nodestrings used with *FESWMS*.

River Module

The *River Module* is used to construct one-dimensional river profiles. Tools are provided in this module for creating a “tree” of data to describe the river being modeled. An example of a river tree is shown in Figure 11.1. Currently, the river model includes a model specific interface for the *WSPRO* model. An interface for the National Weather Service model *DAMBRK* is nearly complete. Interfaces for the USACE-WES model *UNET* and *HEC-RAS* are also in design.

SMS supports the creation of one-dimensional river models in two ways. The model may be created from a conceptual model consisting of a feature coverage(s) and bathymetric information, or defined interactively. Properties can be assigned sections or portions of sections to represent different bed conditions, or geometric characteristics. The user also interactively defines model parameters such as flow rates. This chapter explains the procedures of creating and editing the river tree. See Chapter 12 for an explanation of the *WSPRO* interface and Chapter 14 for a discussion on the use of a conceptual model.

Since the *River Window* is not utilized for two-dimensional modeling, it is not displayed by default. To open the river window choose *River Window* from the *Display* menu. It is also advisable to open the *Plot Window* when creating and editing the river tree by selecting *Plot Window* from the *Display* menu. Using the command line option `-dm river` starts *SMS* with the *River Window* and the *Plot Window* already open. See section 1.2 for more information on using command line arguments.

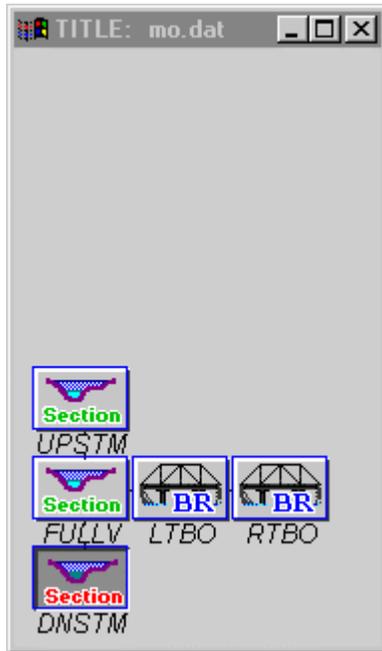


Figure 11.1 An example of the river window.

11.1 River Module Tools

The tool palette (Figure 11.2) for the *River Module* consists of general tools for the creation and editing of one-dimensional cross sections and tools specifically defined for the *WSPRO* model. The first three rows of tools are the general tools. These will be described in this chapter. See Chapter 12 for a description of the other tools.



Figure 11.2 River Tools

11.1.1 Creating Sections

A section in a one-dimensional model is a node at which the user specifies geometric and model parameters. The type of sections available depends on the one-dimensional model being used. Examples of sections include geometric cross sections, bridges, dams, and roadways. *SMS* provides two tools for creating sections that are considered to be generic. These include cross sections and bridges. To create a section, a user selects the appropriate tool and clicks in the river window at the desired position in the river tree. The new section is created at that position.



Create Cross Section

A cross section represents the geometric shape at a specific location in the one-dimensional model. A cross section's *Section Reference Distance* (SRD) positions the cross section relative to other cross sections in the model. In most cases this is the distance from one end of the model. For example, the SRD may be the distance upstream from an exit section. The exit section would have an SRD equal to zero. A cross section also requires geometrical points to define the shape. If a cross section already exists in *SMS*, a new geometrical shape is interpolated (or extrapolated) from the existing sections. For the first cross section, the shape is defaulted to a trapezoid defined by four geometry points.

After the user clicks in the *River Window*, *SMS* chooses a default value for the SRD from the any sections that already exist around the clicked location, and prompts for a new SRD. Once the SRD is defined, a section is created and an icon  representing that section appears.



Create Bridge Section

The *Create Bridge Section* tool is used to add a bridge section. A bridge section must be added to an existing cross section. To add a bridge section, select this tool and click

in the river window at the desired station. The bridge section  icon will appear next to the cross section. Each bridge must have an approach section and an exit section without bridges. This means that bridges cannot be attached to the first or last stations of the river tree.

11.1.2 Editing Sections

All sections have attributes that must be defined by the user. The generic attributes supported in the *River Module* include section geometry and material zones. A section is edited by first making it the active section. Whenever a section is created, it becomes the active section. Any section may be made the active section by selecting its icon in the *River Window* while a section-editing tool is the current tool. The active section's

icon is displayed as a depressed button in the *River Window*, in Figure 11.1 the section “DNSTM” is the active section.

Creating/Editing Geometry

Two-dimensional geometry points define section geometry. Default geometry points are created for all sections as the section is created. *SMS* also prevents the user from deleting all the geometry from a section. In some models, a section’s geometry may be defined by parametric definition. In these cases, there are no explicit geometry points. One example of this is the bridge section in *WSPRO*. A section’s geometry can be edited interactively or via a list of points.

The  create geometry point tool and  select geometry point tools allow the user to edit the section geometry interactively. While the create geometry point tool is selected, a click in the plot window adds a new point to the active section. While the select geometry point tool is selected, the user may drag existing nodes to new locations. The location of the selected node is displayed in the edit window.

In many situations, data describing the section has already been collected by a physical survey. The *Edit Geometry Points* dialog (Figure 11.3) allows the user to import such data from a tab-delimited file. The dialog also includes tools to create and edit the geometry of the section. This dialog is accessed through the model menu using the *List Geometry Points* item. The user may enter points by clicking the desired edit field and typing in the coordinate value. Successive values may be entered by tabbing through the edit fields.

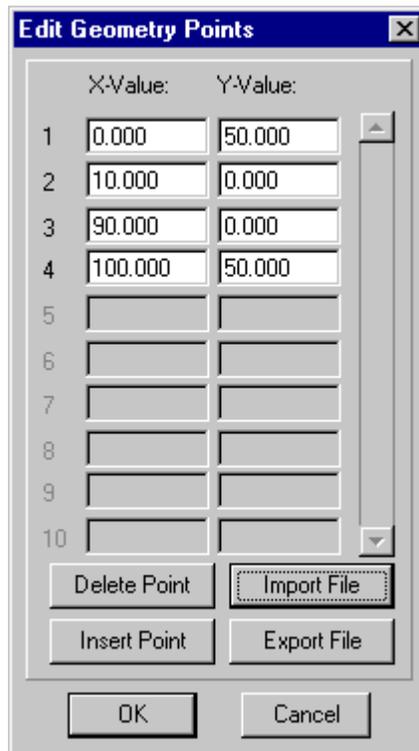


Figure 11.3 One-dimensional geometry point editor.

The *Delete Point* button deletes the currently selected point. However, *SMS* will not allow any deletions if less than four geometry points exist. The *Insert Point* button inserts a point before the current point. The new point is created at the midpoint between the current point and the point before the current point. If the current point is the first point, a new first point is extrapolated from the first two points in the section. Similarly, if a TAB is entered in the last edit field, a new point is extrapolated at the end of the section.

Creating/Editing Material Sub-areas

In one-dimensional models, parameters are lumped at nodes for an entire cross section. However, many models allow versatility in section by using section sub-areas. The most common use of these sub-areas is for specification of multiple roughness conditions. *SMS* allows the definition of these areas through the creation of SA (sub-area) points. A SA point consists of an X-value, which defines the location of the boundary, along with an identifier for each zone. The identifier defines a material, which contains the model specific parameters for that sub-area.

To create a SA point, the  *Create Sub-area Tool* must be active. Any click in the plot window adds a sub-area point to the current section. The location of the break is displayed in the *Edit Window* and may be entered exactly there. The  *Select Sub-area Tool* allows the user to drag an existing SA point to a new location, enter a

location for a selected point in the *Edit Window*, or delete a selected point. The  *Select Sub-area Tool* is used to select a sub-area and assign a material to that sub-area.

Translating Sections

The entire current section can be translated using the  *Translate Section Tool*. When this tool is selected, the user can drag the current section in the *Plot Window*, or specify a delta-X and delta-Y value in the *Edit Window* and translate all of the geometry and sub-area information associated with the current section.

11.2 River Plots

When a section is created, or an existing one-dimensional simulation is read into *SMS*, a check is performed to ensure that the user can view the data being modeled. This requires two separate plots (see Section 3.8.1), at least one *River Profile Plot* and one *River Cross Section Plot* must exist. If either does not exist, it will be created. These plots are only displayed if the *Plot Window* is visible.

11.2.1 River Profile Plots

A profile plot provides a side view of the one-dimensional model. When pre-processing, this includes only the thalweg. An example is shown in Figure 11.4. This gives an indication of slope.

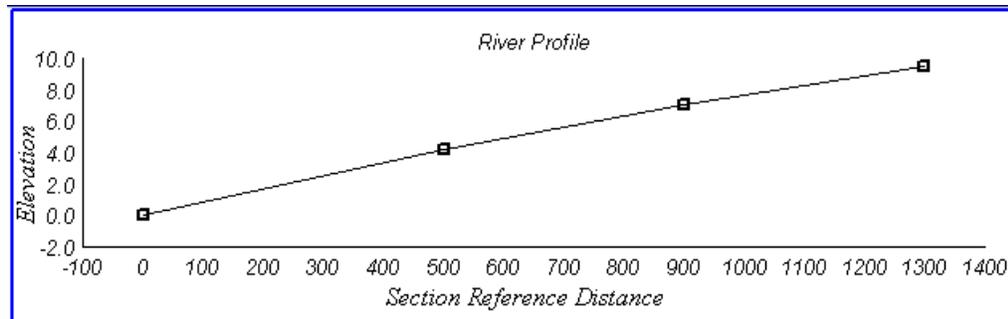


Figure 11.4 River Profile Plot showing thalweg

The main use of the river profile plot is to visualize the solution data generated by the numeric model. Once the solution is generated, it is imported into *SMS* through the *Data Browser* (Figure 11.5). The solution data consists of *Profiles* displayed in the upper left portion of the dialog. Each profile contains several functions that can be viewed over the profile or on the cross sections. The functions associated with all visible profiles are displayed in the *Function* window in the lower portion of the dialog. Each function can be turned on or off individually.

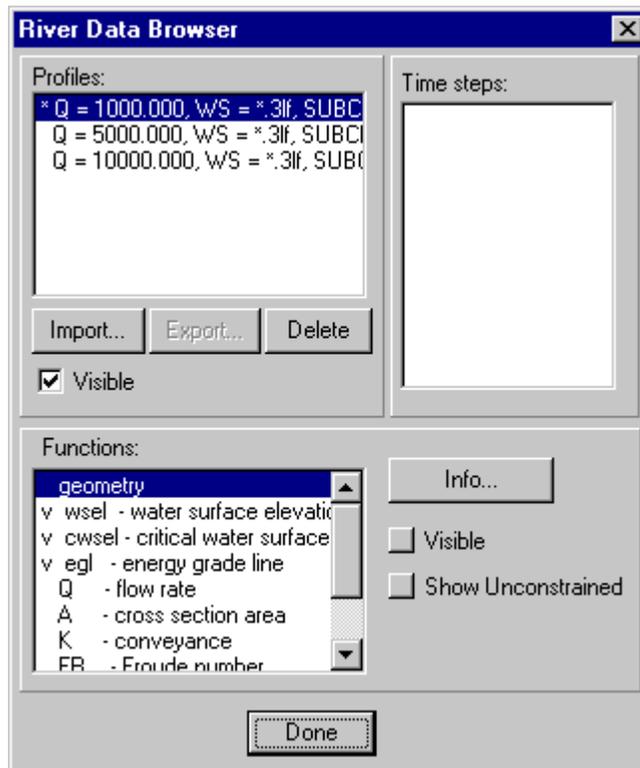


Figure 11.5 River Data Browser

The list of function is determined by the analysis code used. Common quantities include the water surface elevation, energy grade line, and critical water surface elevation. The *Data Browser* is invoked from the *Data Menu*.

Figure 11.6 displays a river profile plotting the solution of these common quantities through four cross sections.

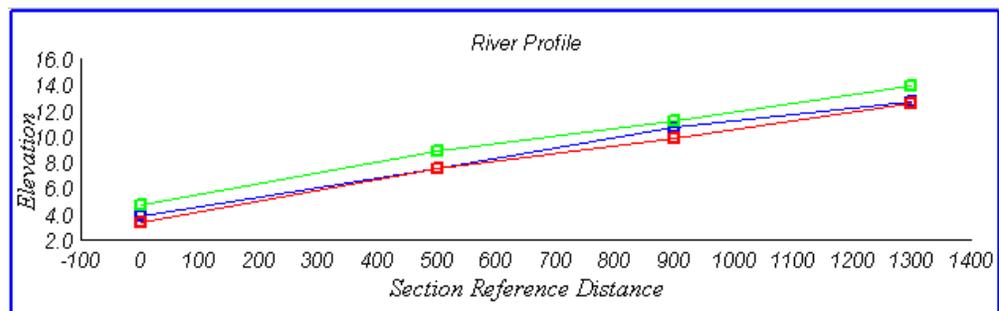


Figure 11.6 River profile displaying wsel, cwsel, and egl.

The user selects the colors and symbols of each function by selecting *Display Options*.

11.2.2 River Cross Section Plots

Cross section plots allow visualization of one or more sections. This plot is used to interact with the plot to define geometry and sub-areas. It is also useful for visualizing relationships between two cross sections or a section with a bridge opening. The geometry is displayed as coordinate pairs (x,y) connected into a section. Sub-area break points are displayed as carrots at the bottom of the plot, and the materials for each sub-area are displayed as a colored line along the X-axis of the plot. Figure 11.7 shows a sample cross section plot.

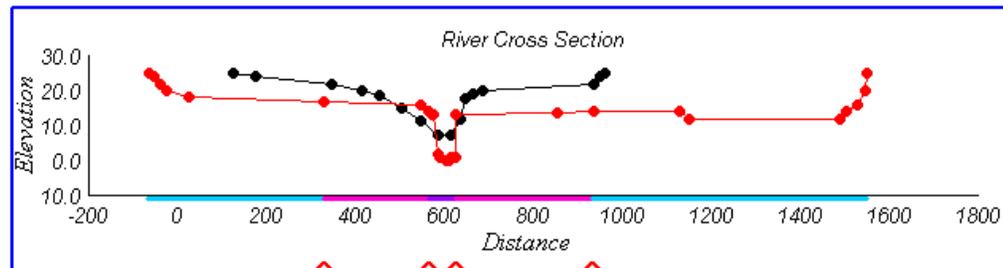


Figure 11.7 Cross section plot

The same functions that can be mapped to the profiles can be mapped to the cross sections in the cross section plot. The models compute a single value, so in these plots, the function is generally a horizontal line. This does facilitate visualization of the relative elevations and inundated area.

11.3 Select the Current Plot

The cross section and profile plots can be manipulated individually. Further, the user may wish to create multiple cross section plots. For example, one plot could be used to display all of the cross sections, while a second only displays the current plot. This can be accomplished by creating another plot of the appropriate type using the *Plot Options* command in the *Display* menu.

The user selects which sections are to be plotted in each plot using the  *Select Current Plot Tool*. While this tool is active, the current plot is selected by clicking on the plot in the *Plot Window*. The sections to be viewed in that plot are selected by clicking on them in the *River Window*. The shift key can be used to multiple select or deselect individual sections.

WSPRO Interface

WSPRO is a water-surface profile computation model that can be used to analyze one-dimensional, gradually varied, steady flow in open channels. *WSPRO* also can be used to analyze flow through bridges and culverts, embankment overflow, and multiple-opening stream crossings. *WSPRO* is supported and maintained by the U.S. Federal Highways Administration (*FHWA*).

The numerical calculations in *WSPRO* are based on the balance of energy from one cross section to the next. Each cross section consists of a geometric shape and parameters defining roughness. Additional information can be associated with sections to indicate sinuosity, contraction/expansion conditions, alignment, slope, etc. Structures such as bridges, culverts, roadways and guide banks can be simulated as sections associated with a specific cross section. Both supercritical and subcritical flow can be represented.

SMS provides graphical tools for defining and editing model parameters, profile control boundary conditions, sectional geometric data and data flow control. Tools include graphical creation/editing of the geometric properties of each section and dialog based input of all required and optional parameters. Plots of both cross sectional data and profile information are generated for visual feedback to the user (see Chapter 11). The cross sections themselves may be extracted from topographic and bathymetric scattered data. See section ??? for a description of this process.

This chapter describes the commands used to create and edit the *WSPRO* specific parameters included in the *WSPRO* menu. The commands for selecting sections and operating on the river model are described in Chapter 11. *SMS* also includes a command to launch the *WSPRO* analysis program. After the analysis is complete, *SMS* can import the solution file via the *Data Browser* dialog in the *River* module to allow

the user to view cross section and profile plots of all the data generated by *WSPRO*. The user should refer to Lessons 14 and 15 of the *SMS Tutorials* for examples of using *SMS* in conjunction with *WSPRO*. The *WSPRO User Manual* and the Bridge Waterways Analysis Model: Research Report (FHWA) discuss the theory of the numerical model in detail.

12.1 New Simulation

The *New Simulation* command in the *WSPRO* menu deletes the current river model including all of the *WSPRO* specific data. This data includes the run control data, the job parameters, and the solution data. Run control data contains the computation directions, profile discharges etc. Job parameters include output tables and tolerances. Solution data includes all variables computed for profile or cross section visualization that is not geometric. The *New Simulation* command also deletes the general river model data (geometric cross sections). To delete all the data currently in *SMS*, the user should select *New* from the *File* menu, which causes all existing data (geometry and model specific data) to be deleted from memory.

12.2 Open Simulation

The *Open Simulation* command in the *WSPRO* menu reads in a data file that has been previously created and saved. These files typically have the file extension “.dat”. The name of the current simulation is displayed at the top of the *River Window*. The data file contains both the geometric and model control data for a *WSPRO* analysis. Geometric data consists of both the definition of the shape of the section and the section reference distance locating sections in relation to each other. In *WSPRO*, the section reference distance must be positive and increases from the downstream end of the mode. (See the *WSPRO User Manual* for more information of data file formats). Opening a new simulation file causes all existing river data (geometry and model specific data) to be deleted from memory, however, data in other modules (such as two dimensional mesh data is not affected).

12.3 Save Simulation

The *Save Simulation* command in the *WSPRO* menu saves a data file so that it can be opened at a later time or used in an analysis.

12.4 Section Creation and Attributes

The tool palette for the river model when *WSPRO* is the current model is displayed in Figure 12.1. The first three rows of tools are generic in the *River* module. Their

application and use are described in Chapter 11. The remaining tools consist of geometric property tools in the first two columns and section creation tools in the third column.



Figure 12.1 River Module/WSPRO Tool Palette

Besides the generic cross-section and bridge section, WSPRO supports three other section types. The Culvert and Road sections are created using tools in the tool palette. Guidebank sections are associated with a specific bridge. Therefore they are created inside of the bridge attributes dialog.

12.4.1 Create Road Section

The *Create Road Section* tool is used to add a road section to a station. To add a road section, select this tool and click in the river window at the desired station. The road

section  icon will branch off the station and its section can be edited. Only one road section can be created at each station. After the road is created, its geometric section and flow obstruction attributes can be modified.

The purpose of defining the roadway is to allow weir flow over the road. Without a road section, WSPRO will allow water to back up behind a bridge indefinitely. The geometry of the road defines the profile shape seen by the water. This is not always just the road surface. Direction barriers and guide rails may need to be considered. To edit the attributes of the roadway, double click the icon or select the *Section Attributes* command from the *WSPRO* menu. This will bring up the *Road Grade Section* dialog (see Figure 12.2). In this dialog, the user can specify a section name, a section reference distance (which may be slightly higher than the station value), a surface type (for roughness computations), and a weir coefficient. A skew value may also be specified if the road crosses the flow in a direction other than perpendicular to flow. (See the *WSPRO* manual for a description of skew).

Road Grade Section

Road Section Parameters:

Name: XR2 Road type: Paved Rough

Location (SRD): 2000.000

Embankment top width: 0.000

Skew: 0.00

Specify unsubmerged weir flow:

Unsubmerged weir flow coefficient: 2.75

OK Cancel

Figure 12.2 Road Grade Section attributes dialog.

12.4.2 Create Culvert Section

The *Create Culvert Section* tool is used to add a culvert section to a station. To add a culvert section, select this tool and click in the river window at the desired station. The

culvert section  icon will branch off the station and its section can be edited. A culvert section consists of a user-specified number of barrels centered about a location on the cross section. Double clicking on the culvert icon or selecting the *Section Attributes* command from the *WSPRO* menu causes the *Culvert Section* dialog to appear (see Figure 12.3).

The upper left portion of the dialog allows the user to edit the location of the culvert. This consists of an SRD and a position in the cross section. The position includes a distance from the datum of the cross section and invert elevations for the culvert ends. Traditionally, the datum for a cross section has been the left bank. Therefore, the location of the culvert is labeled *Distance from left bank*. The datum actually can be anywhere in the section. *SMS* creates cross sections in the map module (see section [???](#)) for which the datum is the specified centerline. An entire *WSPRO* model must be based on the same vertical datum, so the invert elevations are referenced to the same datum as the section geometry. Also included in this section of the dialog are the section name, culvert length and number of barrels.

The user can override defaults for culvert coefficients in the lower left corner.

The geometry of the culvert is defined on the right side of the dialog. The user must select the culvert shape, material and inlet type as well as the size of the barrels. *SMS*

dims materials based on the selected shape and provides a list of inlet types based on both the shape and material specified.

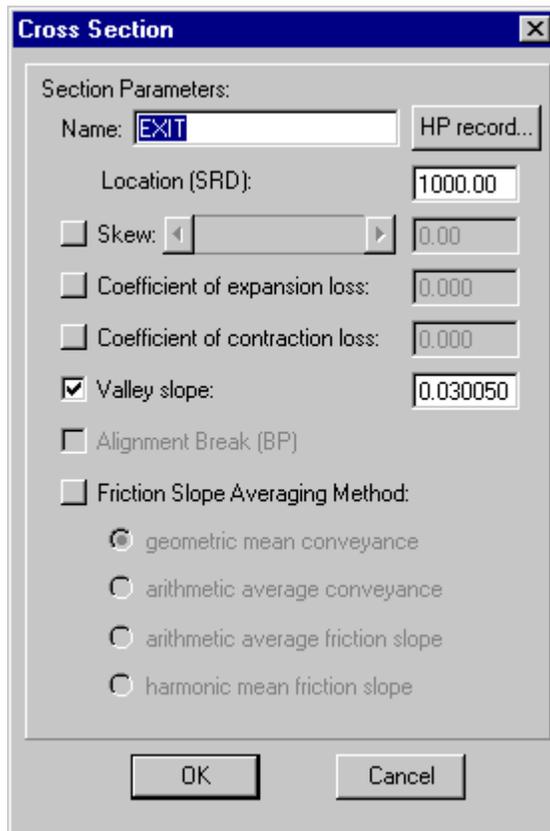
Figure 12.3 Culvert Section attributes dialog.

A single culvert section can exist at a station without any bridges. Culverts can also be used in the multiple opening calculations in *WSPRO*, but at least one bridge opening is required.

12.4.3 Cross Section Attributes

Cross sections are the most generic of section types for one-dimensional modeling. However, each numerical model supports its own set of attributes for a cross section. For *WSPRO*, a cross-section requires only a name and section reference distance (SRD). The SRD and the name can be specified at the top of the *Cross Section* dialog (see Figure 12.4). The user can access other optional attributes including skew, loss coefficients, valley slope and friction slope averaging method. Default values are filled into these attributes when they are turned on. Any unspecified attribute is entered as a blank in the *WSPRO* data file, causing *WSPRO* to assume a default value.

SMS creates a BP record for a cross section to align an approach section with a single opening bridge. When the section attributes are edited, *SMS* detects whether this option should apply and dims the option for sections which are not approaches, or which are approaches for multiple opening situations. Once the *user turns on the Alignment Break option*, the tool to edit the alignment points is undimmed for that section. (Note: *SMS* does not create BP records for section datum corrections, this can be achieved using the  Translate Section tool. *SMS* will read these records from existing files and translate the sections accordingly.)



The image shows a dialog box titled "Cross Section" with a close button (X) in the top right corner. The dialog is divided into a main area for "Section Parameters" and a bottom area with "OK" and "Cancel" buttons. The "Section Parameters" area contains the following fields and options:

- Name:** A text box containing "EXIT" and a button labeled "HP record..." to its right.
- Location (SRD):** A text box containing "1000.00".
- Skew:** A checkbox (unchecked) followed by a spinner box set to "0.00".
- Coefficient of expansion loss:** A checkbox (unchecked) followed by a text box set to "0.000".
- Coefficient of contraction loss:** A checkbox (unchecked) followed by a text box set to "0.000".
- Valley slope:** A checked checkbox followed by a text box set to "0.030050".
- Alignment Break (BP):** A checkbox (unchecked).
- Friction Slope Averaging Method:** A checkbox (unchecked) followed by four radio button options:
 - geometric mean conveyance
 - arithmetic average conveyance
 - arithmetic average friction slope
 - harmonic mean friction slope

Figure 12.4 Cross Section Attributes Dialog.

WSPRO also includes an option to create tables of cross sectional properties (HP record). For cross sections, the user defines the desired tables using the *HP Record* button in the upper right corner of the attribute dialog. This invokes the *WSPRO HP Tables* dialog (see Figure 12.5). Tables can be generated for the entire cross section or for sub areas. The user may also define one velocity conveyance table for each section. If the user desires additional velocity conveyance tables, the extra records must be managed by hand.

The image shows a dialog box titled "WSPRO HP Tables". It is divided into three main sections, each with a checked checkbox:

- Entire cross section:**
 - 453.000 Minimum elevation
 - 1.000 Elevation increment
 - 463.838 Maximum elevation
- Sub areas:**
 - 453.000 Minimum elevation
 - 1.000 Elevation increment
 - 463.838 Maximum elevation
- Velocity and conveyance:**
 - 5000.0 Discharge
 - 453.000 Minimum elevation
 - 1.000 Elevation increment
 - 463.838 Maximum elevation

At the bottom of the dialog, there is a "Reset Defaults" button, an "OK" button, and a "Cancel" button.

Figure 12.5 HP Record Dialog.

12.4.4 Bridge Section Attributes

Bridge sections are the second generic section type. That is because most one-dimensional models support bridges in one form or another. Notwithstanding being a generic type, bridge sections have many model specific attributes. In *WSPRO*, a bridge is defined as a hole. This represents the view the water has of the bridge. *WSPRO* supports a coordinate mode bridge in which the bridge opening is defined as a closed loop of geometry points or a component mode bridge in which the opening is defined by design components, which will be described later. Coordinate mode would commonly be used to analyze existing complex bridge openings. Component mode is easier to modify for considering multiple option in a design process.

The attributes of a selected bridge section are edited by double clicking on the section icon or selecting the *Section Attributes* command from the *WSPRO* menu. Either of these actions causes the *Bridge Section* Figure 12.6 dialog to appear.

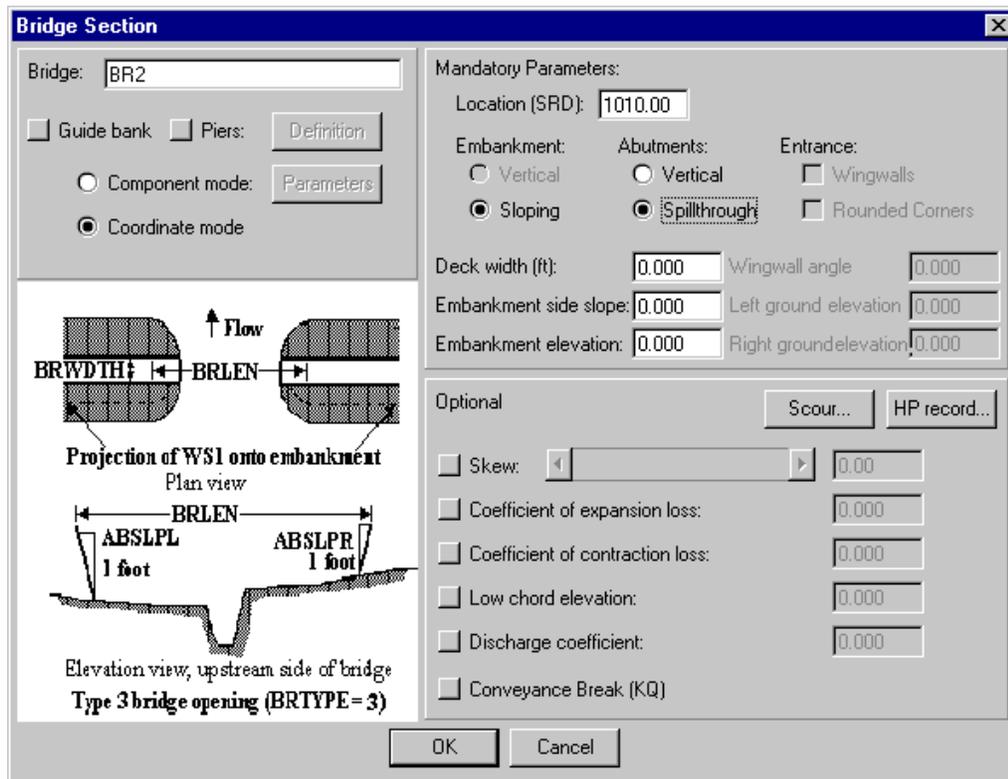


Figure 12.6 Bridge Section Attributes Dialog.

This dialog is divided into four parts.

The upper left portion allows the user to change the section name, add and edit piers, create guidebanks for the bridge opening, and specify the geometry mode for the bridge. If component mode is used, the parameters defining the bridge are specified using the *Bridge Component Mode Parameters* dialog (see Figure 12.7), which is invoked using the *Parameters* button. If the *Pier* toggle is selected, SMS will create a pier record for the bridge (PD record). The *Definition* button allows the user to define the widths of the piers and the number of piers at different elevation.

The upper right portion of the dialog includes the mandatory parameters for the bridge. These include the embankment type, the abutment type, the deck width and several slopes and angles that may or may not be required depending on the bridge type and mode. The elements that don't apply to the currently selected type are dimmed.

The lower left portion displays images from the *WSPRO* reference manual, which illustrate the parameters required for the currently selected bridge type.

The lower right portion includes optional parameters including skew, discharge and loss coefficients, cross sectional, and scour parameters. The *HP Record* button allows the user to specify the creation of a table of values for the bridge opening.

Bridge Components

In design or component mode, the bridge opening is defined with a flat bridge deck, vertical or sloped abutment walls and the geometry from the cross section under the bridge. The *Bridge Component Mode Parameters* dialog (see Figure 12.7) allows the user to specify the elevation of the low steel in the bridge deck and the decks slope. The right side of the dialog allows the user to specify the length of the opening. *WSPRO* also requires that the user position the bridge around control points. These points usually define the edge of the main channel and prevent the bridge abutments from occluding the channel. (See the *WSPRO* reference manual for a more detailed description of the design parameters for a bridge.)

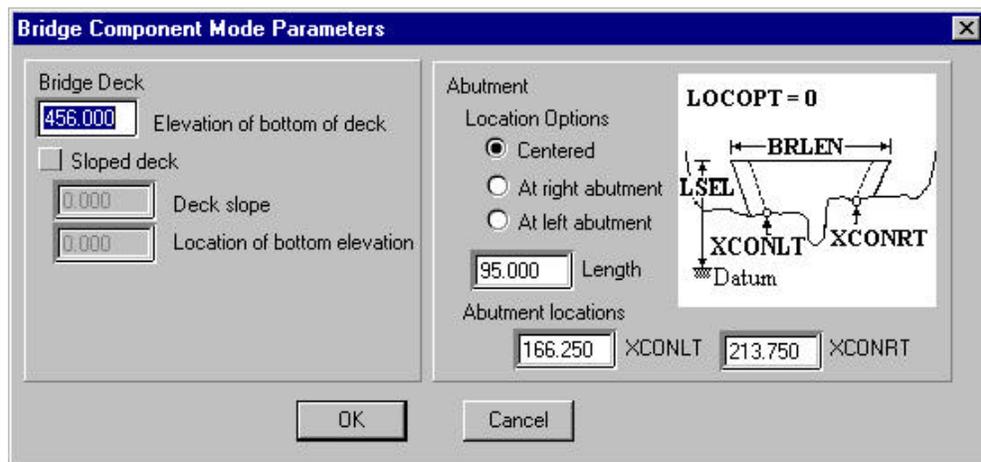


Figure 12.7 Bridge Component Mode Parameters Dialog.

Bridge Scour

WSPRO includes the capability to compute idealized scour around bridge abutments, in bridge openings and at bridge piers. The scour calculations to be performed at a bridge are defined using the *Bridge Scour Options* dialog (see Figure 12.8). The abutment scour option (DA record) reports the maximum scour depth at the toe of the abutments. The live-bed/clear-water contraction scour option (DC records) reports the maximum scour in a section under the bridge. Multiple DC records can be coded to allow multiple sets of parameters for sub regions. The pier scour option (DP records) reports the maximum scour at each pier. (Refer to the *WSPRO* reference manual for a description of the scour coefficients.)

Figure 12.8 Bridge Scour Options Dialog.

12.4.5 Guide Bank Section Attributes

Guide bank sections are bound to a specific bridge section. They are created by selecting the *Guide Bank* toggle in the *Bridge Section* dialog (see Figure 12.6). Once a guide bank exists, its parameters can be modified using the *Guide Bank Section* dialog (see Figure 12.9) accessed by double clicking on the section icon or selecting the *Section Attributes* command from the *WSPRO* menu. This dialog allows the user to define the name of the section, loss coefficients, skew, cross sectional tables, and the type of guide bank.

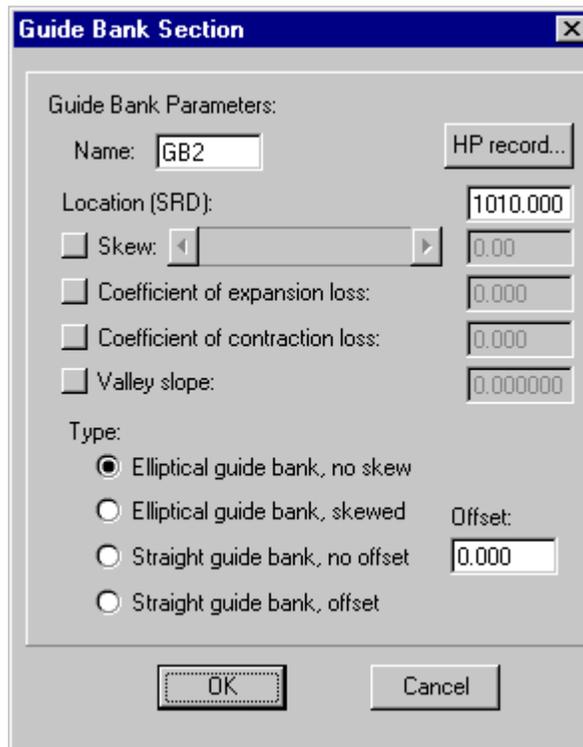


Figure 12.9 Guide Bank Section Attributes Dialog.

12.5 Roughness Parameters

WSPRO uses roughness parameters or Manning's n values to simulate the different types of bed conditions in the river. Using the sub area tools (see section 11.1.2), the user assigns materials to the different regions in a cross section. *The WSPRO Material Editor* (see Figure 12.10) allows the user to associate Manning's n values to the materials used by each section. Materials can use a single roughness value, or have a roughness that varies based on the depth of the flow. This simulates situations such as tall grass that resists flow (high roughness) until the flow reaches a depth at which the grass is pushed over and lies down. It could also be used to simulate a situation such as an orchard in which the effective roughness is low until the water surface elevation reaches the branches and foliage. At that point the effective roughness increases.

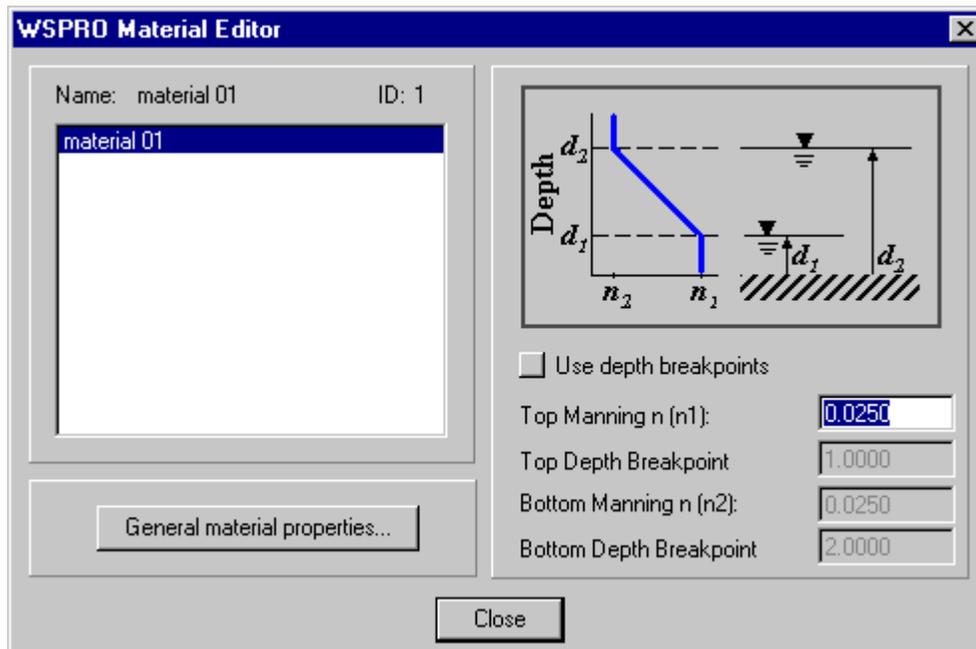
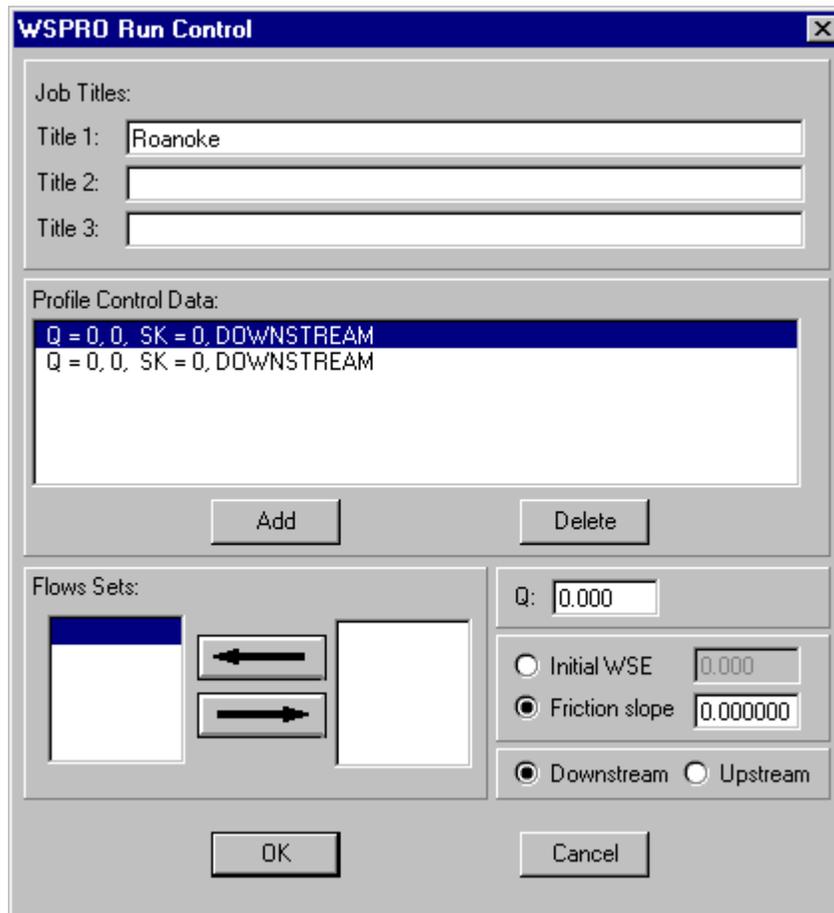


Figure 12.10 WSPRO Material Editor Dialog to specify Roughness Parameters.

12.6 WSPRO Run Control

WSPRO has the capability to compute multiple profiles at once. This allows the user to consider multiple storm events and/or boundary conditions and compare them. The user specifies the flow rates and either an initial water surface elevation or a friction slope for each profile. The *WSPRO Run Control* dialog (see Figure 12.11), which the user accesses in the *WSPRO* menu, allows the user to specify the flow rates, and boundary conditions as well as assign a title to the simulation.

Specifying different flow rates at different cross sections allows *WSPRO* to simulate tributaries entering a river reach. The user also controls the direction of computation. *Upstream* indicates that *WSPRO* will solve the step backwater equations to balance energy starting from the outflow section. This applies to subcritical flow regimes because subcritical flow at any point is affected or controlled by downstream conditions. Conversely, supercritical flow is controlled by upstream flow conditions, and therefore *Downstream* indicates energy will be balanced from the inflow boundary down to simulate supercritical flow. If energy cannot be balanced because of an erroneous assumption of flow regime, *WSPRO* defaults to critical depth.



The image shows a software dialog box titled "WSPRO Run Control". It is divided into several sections:

- Job Titles:** Three text input fields labeled "Title 1:", "Title 2:", and "Title 3:". The first field contains the text "Roanoke".
- Profile Control Data:** A list box containing two entries, both of which are "Q = 0, 0, SK = 0, DOWNSTREAM". The top entry is highlighted in blue.
- Buttons:** "Add" and "Delete" buttons are located below the list box.
- Flows Sets:** A section with two empty rectangular boxes. Between them are two arrow buttons: the top one points left and the bottom one points right.
- Parameters:** A "Q:" label followed by a text box containing "0.000". Below this are two radio button options: "Initial WSE" (with a text box containing "0.000") and "Friction slope" (with a text box containing "0.000000"). At the bottom of this section are two more radio buttons: "Downstream" (which is selected) and "Upstream".
- Final Buttons:** "OK" and "Cancel" buttons at the bottom of the dialog.

Figure 12.11 WSPRO Run Control Dialog

12.7 Job Parameters

The *Job Parameters* command allows the user to specify the optional parameters associated with a numerical analysis. These include computational step size and tolerances to be used during computation. The *WSPRO Job Parameters* dialog also allows the user to specify the units to be used in *WSPRO*, and to specify any output tables desired from the analysis.

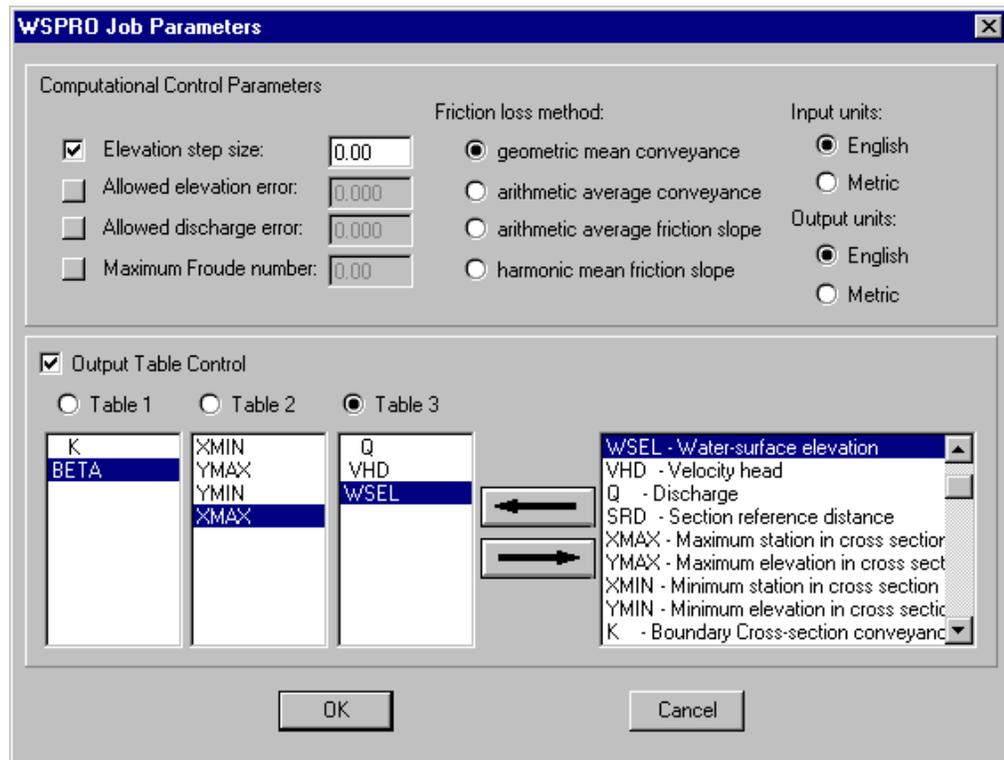


Figure 12.12 WSPRO Job Parameters Dialog

12.8 Model Check

A *Model Check* should be performed on all WSPRO models before attempting an analysis. The model check will perform a basic check to insure that all of the needed information to run the analysis is present. The *Model Check* command in the WSPRO menu causes the *Model Check* dialog to appear.

Selecting the *Checker Options* button will cause the *WSPRO Model Checking Options* dialog to appear. This dialog lists the checks that may be performed during the model checking procedure. By default all supported checks are enabled. The checks include:

- *Check Profiles*. This check assures that at least one profile has been specified and that the appropriate number of discharges and boundary conditions are specified for each profile.
- *Check SRD Values*. This option checks the SRD of each section in the model. SRD values should increase monotonically from the exit to the most upstream section. There are some exceptions for multiple opening situations. This option also checks the placement of bridge approach and exit sections to assure correct placement.

- *Check GR Points.* Cross sections geometry should proceed from left to right. Coordinate mode bridge openings should proceed across the bottom of the opening, then change direction one time and define the bottom of the bridge deck.

After running the model check, messages are generated to aid in the correction of the problems. This information may be saved to a text log file. To do this, click the *Save Messages* button and choose a file to save the information in. To close the *Model Checker*, click the *Done* button.

12.9 Run WSPRO

Once the data for an analysis has been defined, *WSPRO* may be invoked by selecting the *Run WSPRO* menu item. This command checks the status of the model in *SMS*. If edits have been made, the user is prompted to save his files before running. It then launches *WSPRO* using the data files loaded into *SMS*.

Scatter Point Module

The *Scatter Point Module* is used to interpolate from groups of scattered data points to the other data types (i.e., meshes and grids). *SMS* supports three interpolation schemes including linear, natural neighbor and inverse distance weighted.

Interpolation is useful for setting up input data for analysis codes. Generally, the data gathered from a site to be modeled varies in density. Generating a finite element mesh directly from these points would result in a very low quality mesh. Further this data does not lie in a grid for use as a finite difference grid. Interpolation allows the gathered data points to be used as background information. The user may then generate a base mesh or grid in the *Mesh Module*, the *Grid Module* or the *Map Module* (see Chapter 14). The only consideration of bathymetry for such a mesh or grid would be the definition of element edges along geometric or property features. The actual bathymetry comes from the scattered data. *SMS* interpolates this data to the created mesh or grid points.

Interpolation may also be used to create data sets for one mesh from data related to another mesh of the same region. For example, a user may have a mesh of a river reach for which analysis has been performed. If a bridge is to be added to the reach, the mesh topology changes. The data from the first mesh can be converted to a scattered data set and then interpolated to the second mesh. This data may be used as initial conditions for the second mesh, or compared to results of analysis run on the second mesh using the *Data Calculator*.

A third purpose of interpolation is to create additional data sets from either observed, or calculated data.

13.1 Scatter Point Sets

Points from which values are interpolated from are called scatter points. A group of scatter points is called a scatter point set. Each of the scatter points is defined by an xy coordinate pair.

Each scatter point set has a list of functional scalar data sets (vector data sets are not currently supported for scatter point groups). Each data set are samples of values that represent observed or computed values. These values can be interpolated to a mesh or grid.

Multiple scatter point sets can exist in memory. One of the scatter point sets is always designated as the "active" scatter point set. This active scatter point set can be changed to any set in memory. This is done by selecting the *Select Data Sets* tool, selecting the icon for the desired set, and selecting the *Make Set Active* command from the *Interpolation* menu. The active set is also changed internally when interpolation is performed as part of automatic mesh generation (see Chapter 14).

13.2 Inputting Sets

Scatter point sets can be created by converting from other data types (i.e., meshes, grids). For example, if a finite element mesh is converted to a scatter point set (see Section 4.4.1), each of the nodes in the mesh become a scatter point and each of the scalar data sets associated with the mesh are copied to the data set list for the new scatter point set.

Scatter point sets can also be input from a text file. The file formats for scatter point sets are described in Section 17.2. Two types of file formats can be used to import scatter point sets: the XY format and the XYD format. With the XY format, only the xy coordinates for each point are contained in the file. Data values associated with each point must then be input through the *Data Browser* using a data set file. With the XYD format, the xy coordinates and one or more scalar values may be defined for each scatter point. When this type of file is imported, a data set is automatically created for each set of scalar values.

13.3 Saving Sets

Scatter point sets may be saved to a text file using the *Save* command from the *File* Menu. When a scatter point file is saved, the user must specify in the *Save* dialog whether the XY or XYD format will be used. If the XY format is chosen, only the xy coordinates of the scatter points are saved to the file. Any data sets associated with the scatter points must be saved using the *Export* option in the *Data Browser*. If the XYD format is chosen, the xy coordinates and the data sets are saved to a single file. However, since the XYD format is only used for steady-state data, only the current

time step of each data set is saved. To save a complete dynamic data set, the XY format must be used in conjunction with the *Export* option in the *Data Browser*

13.4 Tool Palette

The following tools are active in the dynamic portion of the *Tool Palette* whenever the *Scatter Point Module* is active.

13.4.1 Select Scatter Point

The *Select Scatter Point* tool is used to select individual scatter points for editing using the *Edit Window*.

13.4.2 Select Scatter Point Set

The *Select Scatter Point Set* tool is used to select entire scatter point sets for deletion or to designate the active scatter point set. When this tool is active, an icon appears at the centroid of the set for each of the scatter point sets. A scatter point set is selected by selecting the icon for the set.

A selected scatter point set can be made the active set by double clicking on the icon for the set or by selecting the *Make Active* command from the *Interpolate* menu.

13.5 Display Options

A scatter point set is displayed by drawing a symbol for each of the scatter points. The display options control the appearance of the symbol and the labels. Each scatter point set has its own display options. Selecting the *Display Opts* command from the *Display* menu allows the user to set the symbol, color and display flags for the active data set.

The scatter point display options are as follows:

- The *Scatter Point Numbers* item is used to display the scatter point ID number next to each scatter point.
- The *Scatter Point Symbols* item is used to display a symbol at the location of each scatter point. The button to the left of the item is used to bring up a dialog listing the available symbols. The color of each of the scatter points in a set may be changed from this dialog also.

The *Scatter Point* display options apply to the active scatter point set only. To change the symbols for a scatter point set other than the active set, the set must first be made

active. This is done by double clicking on the set with the *Select Scatter Point Set* tool, or by selecting the set and selecting the *Make Active* command from the *Interpolation* menu prior to bringing up the *Display Options* dialog.

13.6 Scatter Point Conversion

Scatter point sets can be used to create other types of objects. When the new object is created, all data sets associated with the scatter point set are copied to the new object.

13.6.1 Scatter Points -> Mesh Nodes

The *Scatter Points -> Mesh Nodes* command creates a set of finite element nodes. These nodes can be triangulated by selecting the *Triangulate* command from the *Mesh Module*.

13.7 Interpolation

13.7.1 Interpolation From Scatter Point Sets

Interpolation can be invoked explicitly or implicitly. The implicit invocation is part of the automatic mesh or grid generation and is discussed as part of Chapter 14. Explicit interpolation occurs when the user selects the *... to Mesh* or *... to Grid* from the *Interpolate* menu. These commands require that at least one scatter point set exist with at least one function associated. A mesh or grid must also exist in order for the associated interpolation command to be available.

13.7.2 Interpolation Options

When the user selects either interpolation command, the interpolation option dialog (see Figure 13.1) appears labeled to indicate what interpolation is being performed. The user selects the appropriate options and once the OK button is selected, the interpolation procedure is performed.

Interpolation results in a new data set being created for the mesh or grid. For explicit interpolation, the user must define a name for that new data set. There is also an option to map this new data set as the elevations or bathymetry for the mesh or grid.

The user must also select an interpolation method. Since no interpolation scheme is superior in all cases, *SMS* supports three interpolation techniques. Many other methods are possible, however, since surface water modeling requires a fairly rich data set, the more simplistic interpolation methods are more applicable. The user selects a

current method that is used for all interpolation until the user selects another method. The methods supported for interpolation are *linear*, *inverse distance weighted*, and *natural neighbor*.

Based on the selected method, the user also selects extrapolation options and truncation values for interpolation. When interpolating a set of values, it is sometimes useful to limit the interpolated values to a specific range. For example, when interpolating contaminant concentrations, a negative value of concentration is meaningless. However, many interpolation schemes will produce negative values even if all of the scatter points have positive data values. This occurs in areas where the trend in the data is toward a zero value. The interpolation may extend the trend beyond a zero value into the negative range. In such cases it is useful to limit the minimum interpolated value to zero. Interpolated values can be limited to a given range by selecting the *Truncate values* option in the interpolation options dialog and entering a minimum and maximum interpolation value.

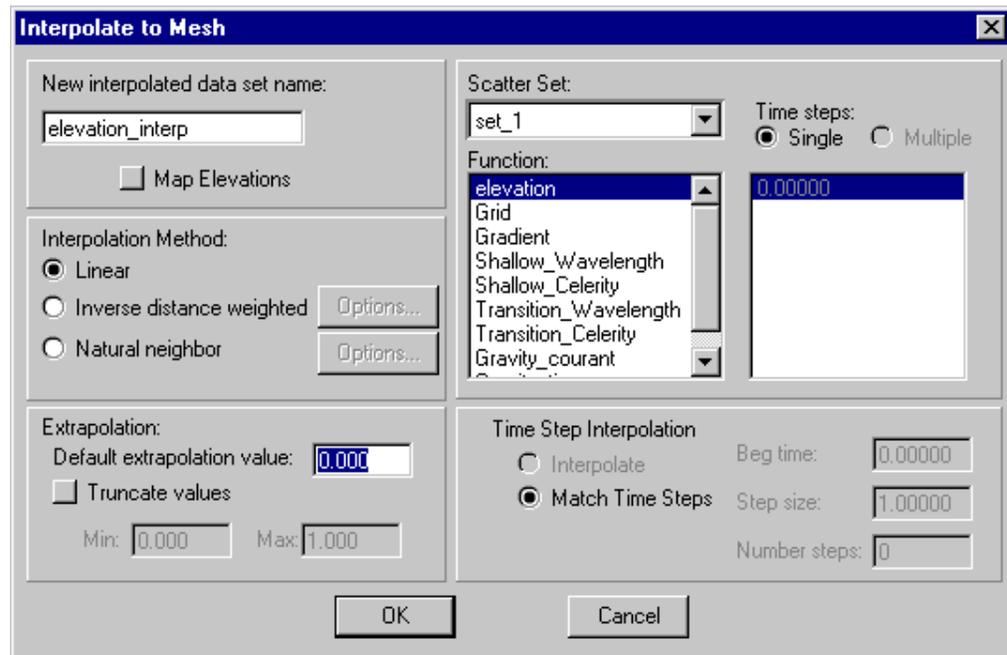


Figure 13.1 The Interpolation Options Dialog.

On the right side of the interpolations options dialog, *SMS* provides the user with another opportunity to select the desired scatter point set and function to interpolate from. By default this is set to be the active scatter point set and the current function. The dialog also includes tools to select the timesteps to create in the new data set.

13.7.3 Linear Interpolation

If the *linear* interpolation scheme is selected, the scatter points are first triangulated to form a temporary triangular network. If the surface is assumed to vary linearly across

each triangle, the network describes a piecewise linear surface that interpolates the scatter points. The equation of the plane defined by the three vertices of a triangle is as follows:

$$Ax + By + Cz + D = 0 \dots\dots\dots (13.1)$$

Where A, B, C, and D are computed from the coordinates of the three vertices (x_1, y_1, z_1) , (x_2, y_2, z_2) , & (x_3, y_3, z_3) :

$$A = y_1(z_2 - z_3) + y_2(z_3 - z_1) + y_3(z_1 - z_2) \dots\dots\dots (13.2)$$

$$B = z_1(x_2 - x_3) + z_2(x_3 - x_1) + z_3(x_1 - x_2) \dots\dots\dots (13.3)$$

$$C = x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2) \dots\dots\dots (13.4)$$

$$D = -Ax_1 - By_1 - Cz_1 \dots\dots\dots (13.5)$$

The plane equation can also be written as:

$$z = f(x,y) = -\frac{A}{C} x - \frac{B}{C} y - \frac{D}{C} \dots\dots\dots (13.6)$$

Which is the form of the plane equation used to compute the elevation at any point on the triangle.

Since a triangular network only covers the convex hull of a scatter point set, extrapolation beyond the convex hull is not possible with the linear interpolation scheme. Any points outside the convex hull of the scatter point set are assigned the *Default extrapolation value* specified in the *Interpolation Options* dialog.

13.7.4 Inverse Distance Weighted Interpolation

One of the most commonly used techniques for interpolation of scatter points is *inverse distance weighted (IDW)* interpolation. Inverse distance weighted methods are based on the assumption that the interpolating surface should be influenced most by the nearby points and less by the more distant points. The interpolating surface is a weighted average of the scatter points and the weight assigned to each scatter point diminishes as the distance from the interpolation point to the scatter point increases.

While there are constant, linear and quadratic inverse distance weighted interpolation schemes, *SMS* supports only the constant method due to the richness of the required data sets used in surface water modeling. Several options are available for inverse distance weighted interpolation. The options are specified using the *IDW Options* dialog (see Figure 13.2).

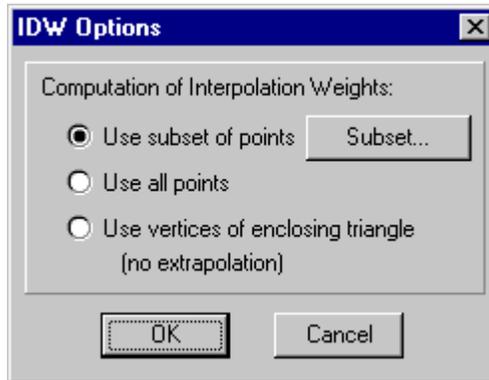


Figure 13.2 The IDW Interpolation Options Dialog.

Shepard's Method

The constant form of inverse distance weighted interpolation is sometimes called "Shepard's method" (Shepard 1968). The equation used is as follows:

$$F(x,y) = \sum_{i=1}^n w_i f_i \dots\dots\dots (13.7)$$

Where n is the number of scatter points in the set, f_i are the prescribed function values at the scatter points (e.g. the data set values), and w_i are the weight functions assigned to each scatter point. The classical form of the weight function is:

$$w_i = \frac{h_i^{-p}}{\sum_{j=1}^n h_j^{-p}} \dots\dots\dots (13.8)$$

Where p is an arbitrary positive real number called the power parameter (typically, p=2) and h_i is the distance from the scatter point to the interpolation point or:

$$h_i = \sqrt{(x-x_i)^2 + (y-y_i)^2} \dots\dots\dots (13.9)$$

Where (x,y) are the coordinates of the interpolation point and (x_i,y_i) are the coordinates of each scatter point. The weight function varies from a value of unity at the scatter point to a value approaching zero as the distance from the scatter point increases. The weight functions are normalized so that the weights sum to unity.

The effect of the weight function is that the surface will interpolate each scatter point and be influenced most strongly between scatter points by the points closest to the point being interpolated.

Although equation 13.8 is typically used for the weight function in inverse distance weighted interpolation, the following equation is used in *SMS*:

$$w_i = \frac{\left[\frac{R-h_i}{Rh_i} \right]^2}{n} \dots\dots\dots (13.10)$$

$$\sum_{j=1} \left[\frac{R-h_j}{Rh_j} \right]^2$$

Where h_i is the distance from the interpolation point to scatter point i , R is the distance from the interpolation point to the most distant scatter point, and n is the total number of scatter points. This equation has been found to give superior results to equation 13.8 (Franke & Nielson, 1980).

The weight function is a function of Euclidean distance and is radically symmetric about each scatter point. As a result, when interpolating elevation, the interpolating surface is somewhat symmetric about each point and tends toward the mean elevation of the scatter points between the scatter points. Shepard's method has been used extensively because it is very simple.

Interpolation Subsets

In the *IDW Options* dialog shown in Figure 13.2, an option is available for using a subset of the scatter points (as opposed to all of the available scatter points) in the computation of the nodal function coefficients and in the computation of the interpolation weights. Using a subset of the scatter points drops distant points from consideration since they are unlikely to have a large influence on the nodal function or on the interpolation weights. Using a subset can speed up the computations since fewer points are involved.

If the *Use subset of points* option is chosen, the *Subsets* button can be used to bring up the *Subset Definition* dialog shown in Figure 13.3. Two options are available for defining which points are included in the subset. In one case, only the nearest N points are used. In the other case, only the nearest N points in each quadrant are used (see Figure 13.4). This approach may give better results if the scatter points tend to be clustered.

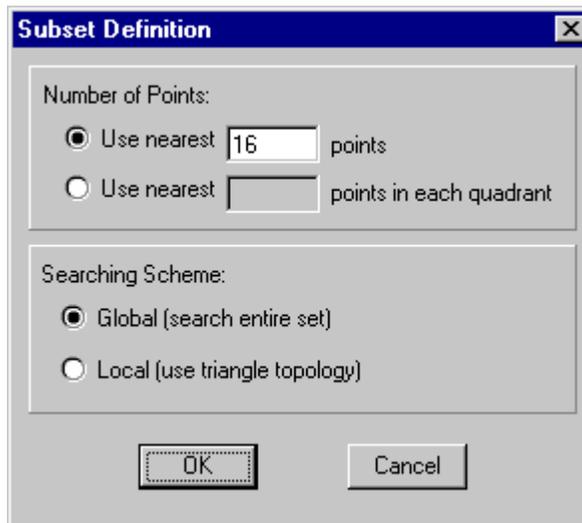


Figure 13.3 The Subset Definition Dialog.

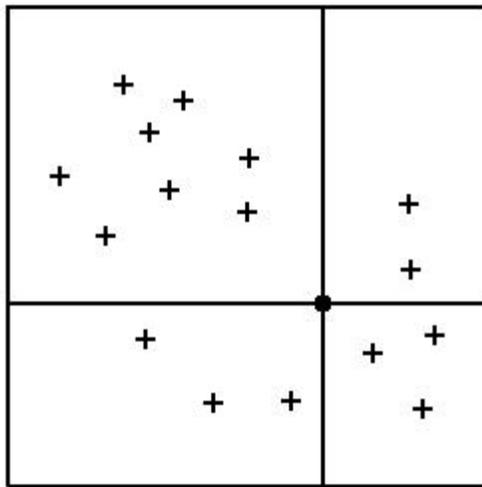


Figure 13.4 The Four Quadrants Surrounding an Interpolation Point.

If a subset of the scatter point set is being used for interpolation, a scheme must be used to find the nearest N points. Two methods for finding a subset are provided in the *Subset Definition* dialog: the global method and the local method. With the global method, each of the scatter points in the set are searched for each interpolation point to determine which N points are nearest the interpolation point. This technique is fast for small scatter point sets but may be slow for large sets.

With the local scheme, the scatter points are triangulated to form a temporary triangular network before the interpolation process begins. To compute the nearest N points, the triangle containing the interpolation point is found and the triangle topology is then used to sweep out from the interpolation point in a systematic fashion until the N nearest points are found. The local scheme is typically much faster than the global scheme for large scatter point sets.

Local Weighting Method

As mentioned above, it is possible to localize the search for the nearest N scatter points to the interpolation point using the topology of a triangular network constructed from the scatter points. Yet another scheme is available for making the interpolation process a local scheme by taking advantage of network topology (Franke & Nielson, 1980). With this technique, the subset of points used for interpolation consists of the three vertices of the triangle containing the interpolation point. The weight function or blending function assigned to each scatter point is a cubic S-shaped function (see Figure 13.5(a)). The fact that the slope of the weight function tends to unity at its limits ensures that the slope of the interpolating surface will be continuous across triangle boundaries.

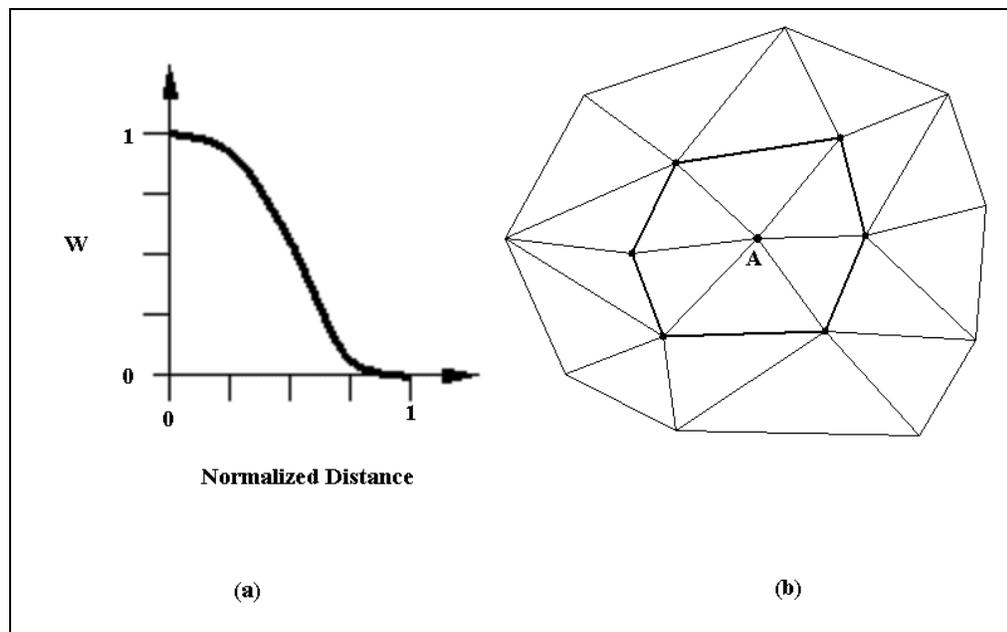


Figure 13.5 (a) S-Shaped Weight Function (b) Delauney Point Group for Point A.

The influence of the weight function extends over the limits of the Delauney point group of the scatter point. The Delauney point group is the "natural neighbors" of the scatter point. The perimeter of the group is made up of the outer edges of the triangles that are connected to the scatter point, as shown in Figure 13.5b. The weight function varies from a weight of unity at the scatter point to zero at the perimeter of the group. For every interpolation point in the interior of a triangle, there are three nonzero weight functions (the weight functions of the three vertices of the triangle). For a triangle T with vertices i, j, & k, the weights for each vertex are determined as follows:

$$W_i(x,y) = b_i^2(3 - 2b_i) + 3 \frac{b_i^2 b_j b_k}{b_i b_j + b_i b_k + b_j b_k} \left\{ b_j \left[\frac{\|e_i\|^2 + \|e_k\|^2 - \|e_j\|^2}{\|e_k\|^2} \right] + b_k \left[\frac{\|e_i\|^2 + \|e_j\|^2 - \|e_k\|^2}{\|e_j\|^2} \right] \right\} \dots\dots\dots(13.11)$$

Where $\|e_i\|$ is the length of the edge opposite vertex i , and b_i, b_j, b_k are the area coordinates of the point (x,y) with respect to triangle T . Area coordinates are coordinates that describe the position of a point within the interior of a triangle relative to the vertices of the triangle. The coordinates are based solely on the geometry of the triangle. Area coordinates are sometimes called "barycentric coordinates." The relative magnitude of the coordinates corresponds to area ratios, as shown in Figure 13.6

The xy coordinates of the interior point can be written in terms of the xy coordinates of the vertices using the area coordinates as follows:

$$x = b_i x_i + b_j x_j + b_k x_k \dots\dots\dots (13.12)$$

$$y = b_i y_i + b_j y_j + b_k y_k \dots\dots\dots (13.13)$$

$$1.0 = b_i + b_j + b_k \dots\dots\dots (13.14)$$

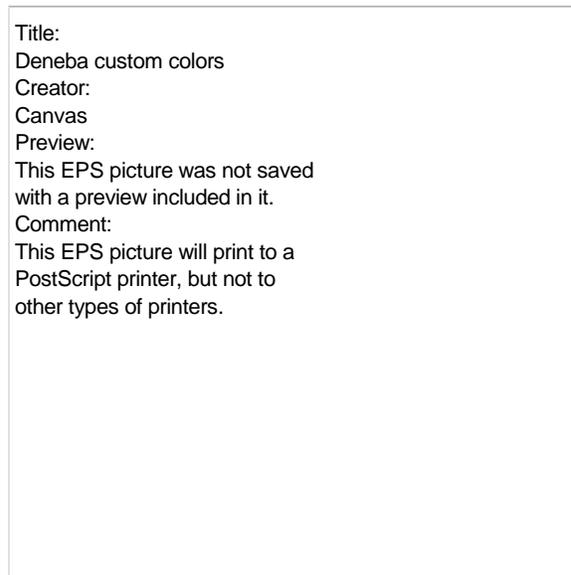


Figure 13.6 Barycentric Coordinate for a Point in a Triangle.

Solving the above equations for $b_i, b_j,$ and b_k yields:

$$b_i = \frac{1}{2A} [(x_j y_k - x_k y_j) + (y_j - y_k)x + (x_k - x_j)y] \dots\dots\dots (13.15)$$

$$b_j = \frac{1}{2A} [(x_k y_i - x_i y_k) + (y_k - y_i)x + (x_i - x_k)y] \dots\dots\dots (13.16)$$

$$b_k = \frac{1}{2A} [(x_i y_j - x_j y_i) + (y_i - y_j)x + (x_j - x_i)y] \dots\dots\dots (13.17)$$

$$A = \frac{1}{2}(x_i y_j + x_j y_k + x_k y_i - y_i x_j - y_j x_k - y_k x_i) \dots\dots\dots (13.18)$$

Using the weight functions defined above, the interpolating surface at points inside a triangle is computed as:

$$F(x,y) = W_i(x,y)Q_i(x,y) + W_j(x,y)Q_j(x,y) + W_k(x,y)Q_k(x,y).....(13.19)$$

where W_i , W_j , and W_k are the weight functions and Q_i , Q_j , and Q_k are the nodal functions for the three vertices of the triangle.

IDW interpolation using local weights as defined in equation 13.11 can be selected in the *Weighting Method* section of the *IDW Interpolation Options* dialog. This type of weighting is significantly faster than the standard weighting method, particularly if the number of scatter points is large.

13.7.5 Natural Neighbor Interpolation

Natural neighbor interpolation is also supported in *SMS*. Natural neighbor interpolation has many positive features. It can be used for both interpolation and extrapolation, and it behaves very well with clustered scatter points. Sibson (1981) first introduced natural neighbor interpolation. A more detailed description of natural neighbor interpolation in multiple dimensions can be found in Owen (1992).

The basic equation used in natural neighbor interpolation is identical to the one used in IDW interpolation. As with IDW interpolation, the nodal functions can be constants, gradient planes, or quadratics. *SMS* supports constant nodal functions. The difference between the IDW interpolation and natural neighbor interpolation is the method used to compute the weights and the method used to select the subset of scatter points used for interpolation.

Natural neighbor interpolation is based on the Thiessen polygon network of the scatter point set. The Thiessen polygon network can be constructed from the Delauney triangulation of a scatter point set (see ???). A Delauney triangulation is a triangulated irregular network that has been constructed so that the Delauney criterion has been satisfied.

There is one Thiessen polygon in the network for each scatter point. The polygon for a scatter point encloses the area that is closer to the scatter point than any other scatter point. The polygons in the interior of the scatter point set are closed polygons and the polygons on the convex hull of the set are open polygons.

Each Thiessen polygon is constructed using the circumcircles of the triangles resulting from a Delauney triangulation of the scatter points. The vertices of the Thiessen polygons correspond to the centroids of the circumcircles of the triangles.

Local Coordinates

The weights used in natural neighbor interpolation are based on the concept of local coordinates. Local coordinates define the "neighborliness" or amount of influence any

scatter point will have on the computed value at the interpolation point. This neighborliness is entirely dependent on the area of influence or Thiessen polygons of the surrounding scatter points.

To define the local coordinates for the interpolation point, P_n , the area of all Thiessen polygons in the network must be known. Temporarily inserting P_n into the TIN will cause the TIN and the corresponding Thiessen network to change, resulting in new Thiessen areas for the polygons in the neighborhood of P_n .

Title:
Deneba custom colors
Creator:
Canvas
Preview:
This EPS picture was not saved
with a preview included in it.
Comment:
This EPS picture will print to a
PostScript printer, but not to
other types of printers.

Figure 13.7 Delauney Triangulation and Corresponding Thiessen Polygon Network

The concept of local coordinates is shown graphically in Figure 13.8. Points 1-10 are scatter points and P_n is a point where some value associated with points 1-10 is to be interpolated. The dashed lines show the edges of the Thiessen network before P_n is temporarily inserted into the TIN and the solid lines show the edges of the Thiessen network after P_n is inserted.

Only those scatter points whose Thiessen polygons have been altered by the temporary insertion of P_n are included in the subset of scatter points used to interpolate a value at P_n . In this case, only points 1, 4, 5, 6, & 9 are used. The local coordinate for each of these points with respect to P_n is defined as the area shared by the Thiessen polygon defined by point P_n and the Thiessen polygon defined by each point before point P_n was added. The greater the common area, the larger the resulting local coordinate, and the larger the influence or weight the scatter point has on the interpolated value at P_n .

If we define $\kappa(n)$ as the Thiessen polygon area of P_n and $\kappa_m(n)$ as the difference in the Thiessen polygon area of a neighboring scatter point, P_m , before and after P_n is inserted, then the local coordinate $\lambda_m(n)$ is defined as:

$$\lambda_m(n) = \frac{\kappa_m(n)}{\kappa(n)} \dots\dots\dots(13.20)$$

The local coordinate $\lambda_m(n)$ varies between zero and unity. If P_n is at precisely the same location as P_m , then the Thiessen polygon areas for P_n and P_m are identical and $\lambda_m(n)$ has a value of unity. In general, the greater the relative distance P_m is from P_n , the smaller its influence on the final interpolated value.

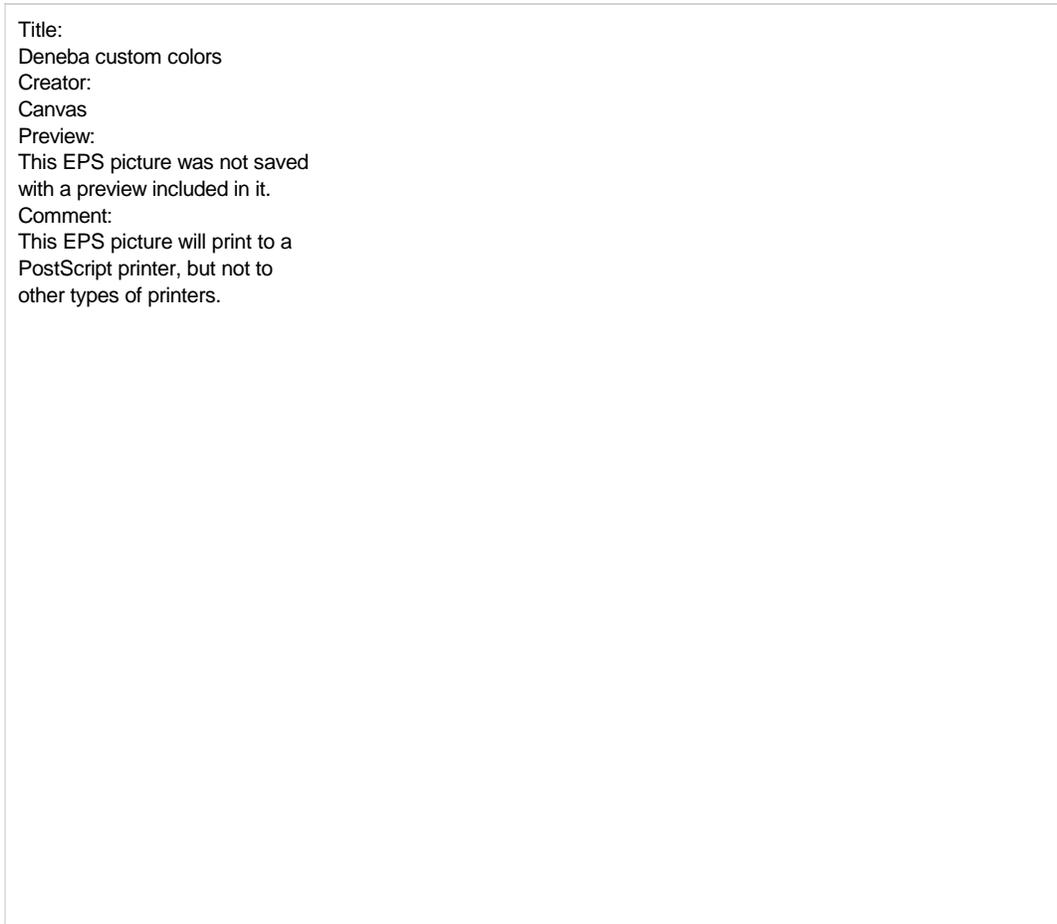


Figure 13.8 Overlapping Thiessen Polygon Areas Used to Compute Local Coordinates.

The weights used in natural neighbor interpolation are computed by normalizing the local coordinates so that they sum to one:

$$w_m(n) = \frac{\lambda_m(n)}{\sum_{i=1}^p \lambda_i(n)} \dots\dots\dots (13.21)$$

where $w_m(n)$ is the weight of scatter point P_m with respect to the interpolation point P_n , and p is the number of points in the neighborhood of P_n with non-zero local coordinates.

Bounding Window

As shown in Figure 13.7, the Thiessen polygons for scatter points on the perimeter of the TIN are open-ended polygons. Since such polygons have an infinite area, they cannot be used directly for natural neighbor interpolation. In order to make the area of these polygons finite, a bounding box or window is superimposed on the scatter point set (see Figure 13.9). Polygons for points on the exterior of the scatter point set are clipped or truncated to the bounding window.

Title:
Deneba custom colors
Creator:
Canvas
Preview:
This EPS picture was not saved
with a preview included in it.
Comment:
This EPS picture will print to a
PostScript printer, but not to
other types of printers.

Figure 13.9 Bounding Window Used in Natural Neighbor Interpolation.

The user specifies the size of the bounding window in the *Natural Neighbor Options* dialog (see Figure 13.10).



Figure 13.10 The Natural Neighbor Options Dialog.

With finite Thiessen polygon areas on the perimeter of the scatter point set, it is possible to perform extrapolation (estimate values for interpolation points outside the convex hull of the scatter point set) as well as interpolation. However, the values computed by extrapolation are somewhat influenced by the relative size of the bounding window with respect to the size of the scatter point set. The larger the bounding window, the greater the influence of the perimeter scatter points on the extrapolated values. If the bounding window is extremely large, only the points on the convex hull of the scatter point set will influence the extrapolated values. If the bounding window is not significantly larger than the scatter point set, the extrapolated values will be influenced by interior scatter points in the neighborhood of the interpolation point in addition to perimeter scatter points near the interpolation point. The relative size of the bounding window can be altered using the *Natural Neighbor Interpolation Options* dialog. In addition, the dialog can be used to turn off the extrapolation option entirely.

Map Module

The *Map* module provides a suite of tools for defining conceptual models in a GIS format, adding annotation to a plot, displaying digital background maps, and displaying CAD drawings.

Four types of objects are supported in the *Map* module: feature objects, drawing objects, digital images, and DXF files.

Feature objects are used to provide GIS capabilities within *SMS*. Feature objects include points, arcs, and polygons. Feature objects can be grouped into layers or *coverages*. One or more coverages can be constructed to represent a conceptual model of a physical system involving surface water, which is to be modeled. *SMS* generates a numeric model (mesh or grid) based on this conceptual model.

Drawing objects provide a simple method for adding annotation to a plot. Drawing objects include text, arrows, lines, rectangles, and ovals.

Images are scanned maps or aerial photos imported in the form of TIFF files. Images are displayed in the background for on-screen digitizing, model placement, or simply to enhance the display of a model.

DXF files are CAD drawings that can be imported into *SMS* and displayed in the *Graphics Window* to assist in model placement or simply to enhance the display of a model. Polylines from the imported DXF files can also be converted into feature objects for use in mesh generation.

14.1 Feature Objects

Feature objects in *SMS* have been patterned after Geographic Information Systems (GIS) objects and include points, nodes, arcs, and polygons. Feature objects can be grouped together into coverages. Each coverage defines a particular set of information. The primary use of feature objects is to construct finite element meshes, finite difference grids, or one-dimensional cross sections. The area included by the polygons defines the domain of the mesh, grid, or limit the extents of cross sections. Each polygon represents a material zone or element type. Special points can be identified in the interior of the domain as areas of particular interest. Boundary parameters such as flow and head values can also be assigned to points or arcs. *SMS* then generates a mesh points by filling in the interior of the polygonal zones with nodes and elements, a grid by filling in the interior of the polygonal zones with cells, or a one-dimensional model by automatically generating cross sections.

14.1.1 Feature Object Types

The definition of feature objects in *SMS* follows that used by typical GIS software that supports vector data. The basic object types are points, nodes, vertices, arcs, and polygons. The relationship between these objects is illustrated in Figure 14.1.

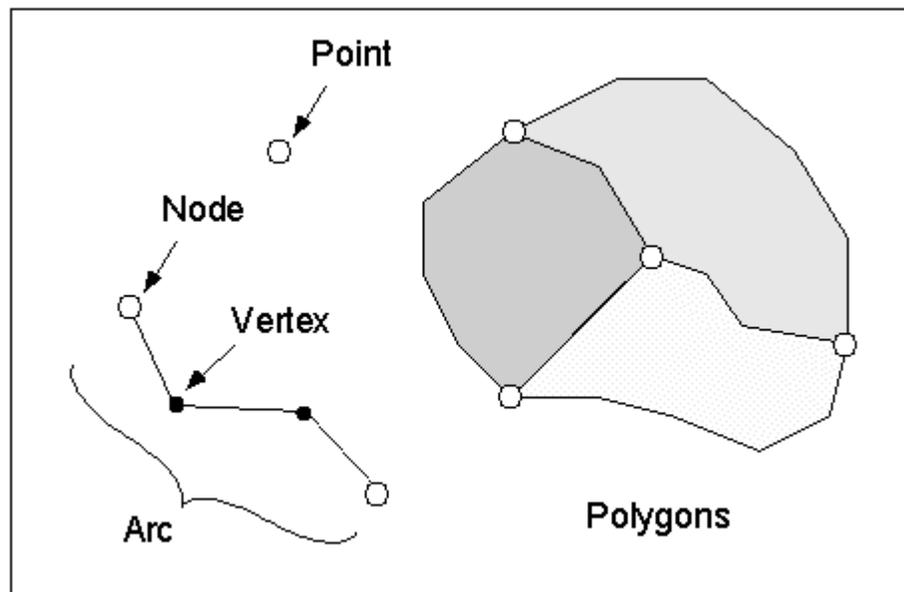


Figure 14.1 Feature object types.

Points

Points are XY locations that are not attached to an arc. Points have unique ids and can be assigned attributes such as a source or sink. Points are often used to refine a mesh in an area of interest. Points are also used when importing a set of XY locations for the purpose of creating arcs or polygons.

Arcs

Arcs are sequences of line segments or edges, which are grouped together as a single "polyline" entity. Arcs have unique ids and can be assigned attributes such as specified head. Arcs are grouped together to form polygons or are used independently to represent geometrical features such as ridges or channels. The two end points of an arc are called "nodes" and the intermediate points are called "vertices".

Nodes

Nodes define the beginning and ending XY locations of an arc. Nodes have unique ids and can be assigned attributes.

Vertices

Vertices are XY locations along arcs in between the beginning and ending nodes. They are used solely to define the geometry of the arcs. Vertices do not have ids or attributes.

Polygons

Polygons are a group of connected arcs that form a closed loop. A polygon consists of one or more arcs. If two polygons are adjacent, the arc(s) forming the boundary between the polygons is shared (not duplicated). Polygons may not overlap. However, a polygon can have a hole(s) defined by having a set of closed arcs defining interior polygon(s). An example of a hole is shown in Figure 14.2. In this case, four arcs define two polygons. Polygon A is made up of arcs 1, 2, 3 and 4, whereas polygon B is defined by a single arc (arc 2). For polygon A, arcs 1, 3, and 4 define the exterior boundary whereas arc 2 defines a hole.

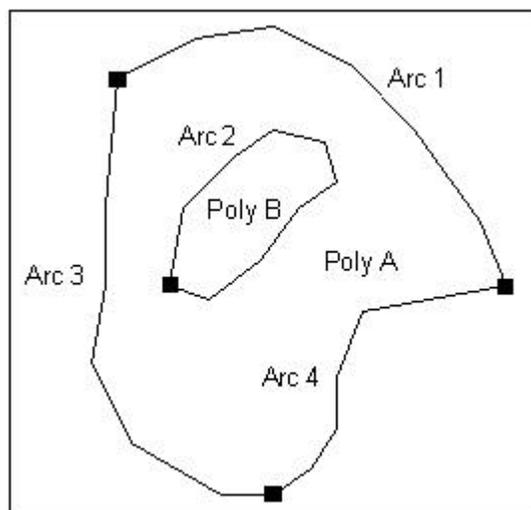


Figure 14.2 Polygon with Holes.

Polygons have unique ids and can be assigned attributes. Polygons are used to represent material zones such as main channel, overbank flood plain, lakes, etc.

14.1.2 Feature Object Tools

Several tools are provided in the *Tool Palette* for creating and editing feature objects. These tools are located in the dynamic portion of the *Tool Palette* and are only available when the *Map* module is active. The tools are as follows:

Select Point/Node

The *Select Point/Node* tool is used to select existing points or nodes. A selected point/node can be deleted, moved to a new location, or operated on by one of the commands in the *Feature Objects* menu. The coordinates of selected points/nodes can be edited using the *Edit Window*. Double clicking on a point or node with this tool brings up the *Point* or *Node Attribute* dialog.

Select Vertex

The *Select Vertex* tool is used to select vertices on an arc. Once selected, a vertex can be deleted, moved to a new location, or operated on by one of the commands in the *Feature Objects* menu. The coordinates of a selected vertex can be edited using the *Edit Window*.

Select Arc

The *Select Arc* tool is used to select arcs for operations such as deletion, redistribution of vertices, or building polygons. Double clicking on an arc with this tool brings up the *Arc Attributes* dialog.

Create Point

The *Create Point* tool is used to interactively create new points. A new point is created for each location the cursor is clicked on in the *Graphics Window*. Once the point is created, it can be repositioned or otherwise edited with the *Select Point/Node* tool.

Create Vertex

The *Create Vertex* tool is used to interactively create new vertices along an existing arc. This is typically done to add more detail to the arc. A new vertex is created for each location the cursor is clicked on in the *Graphics Window*, that it is within a given pixel tolerance of an existing arc. Once the vertex is created, it can be repositioned with the *Select Vertex* tool.

 **Create Arc**

The *Create Arc* tool is used to interactively create new arcs. An arc is created by clicking once on the location where the arc is to begin, clicking once to define the location of each of the vertices in the interior of the arc, and double-clicking at the location of the end node of the arc.

As arcs are created, it is often necessary for the beginning or ending node of the arc to coincide with an existing node. If you click on an existing node or point (within a given pixel tolerance) when beginning or ending an arc, that node is used to define the arc node as opposed to creating a new node. Also, if you click on a vertex or segment of another arc while creating an arc, that vertex is converted to a node or a new node is created and the node is used in the new arc.

While creating an arc, it is not uncommon to make a mistake by clicking on the wrong location. In such cases, hitting the *BACKSPACE* key backs up the arc by one vertex. The *ESCAPE* key can also be used to abort the entire arc creation process at any time.

 **Select Feature Arc Group**

The *Select Feature Arc Group* tool is used to select one or more previously created arc groups to assign attributes to that group or groups. An arc group is created by selecting each of the arcs to be associated with the group and issuing the *Feature Objects/Create Arc Group* command. Arc groups are used to distribute a single attribute over more than one arc. An example of their use would include an inflow region to a mesh that is comprised of several polygons. All of the arcs on the inflow boundary should be used to construct an arc group and the boundary condition attribute assigned to the group rather than individual arcs.

 **Select Polygon**

The *Select Polygon* tool is used to select previously created polygons for operations such as deletion, assigning attributes, etc. A polygon is selected by clicking anywhere in the interior of the polygon. Double clicking on a polygon with this tool brings up the *Polygon Attributes* dialog.

14.1.3 Feature Object Commands

SMS includes several commands to operate with feature objects. These are found in the *Feature Objects* menu and include the following.

Build Polygon

While the other feature objects can be constructed with tools in the *Tool Palette*, polygons are constructed with the *Build Polygon* command. Since arcs define polygons, the first step in constructing a polygon is to create the arcs forming the

boundary of the polygon. Once the arcs are created, they should be selected with the *Select Arc* tool, and the *Build Polygon* command should be selected from the *Feature Objects* menu.

The *Build Polygon* command can be used to construct one polygon at a time or to construct several polygons at once. If the selected arcs form a single loop, only one polygon is created. If the arcs form multiple loops, a polygon is created for each unique (non-overlapping) loop. If no arcs are selected, all of the currently defined arcs in the active coverage are used to create polygons. This process preserves existing valid polygons.

Clean

The *Clean* command is used to fix errors in feature object data. Specifically, it prompts for a snapping tolerance and minimum dangling arc length, and then uses these parameters to do the following:

1. A check is made to see if any nodes are within the specified tolerance of other nodes. If so, the nodes are snapped together. (see Figure 14.3 A)
2. A check is made to see if any arcs intersect. If so, a node is created at the intersection and the arcs are split. (see Figure 14.3 B)
3. A check is made for dangling arcs (arcs with one end not connected to another arc) with a minimum length. If any are found they are deleted. (see Figure 14.3 C)

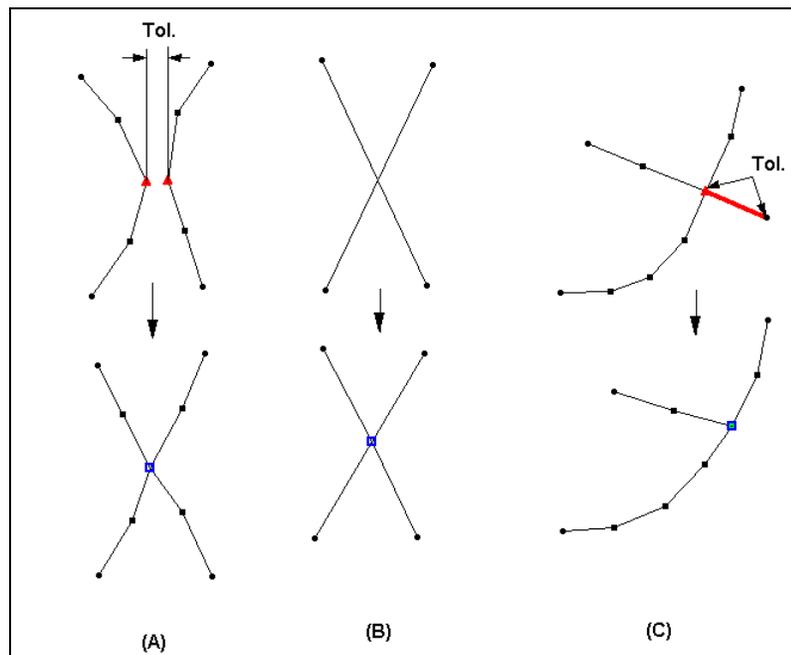


Figure 14.3 Cleaning Arcs. (A) snapping (B) intersecting (C) dangling arcs

All objects in the active coverage are "cleaned".

Vertex <-> Node

In some cases, it is necessary or desirable to split an arc into two arcs. This can be accomplished using the *Vertex <-> Node* command. Before selecting this command, a vertex on the arc at the location where the arc is to be split should be selected. The selected vertex is converted to a node and the arc is split in two.

The *Vertex <-> Node* command can also be used to combine two adjacent arcs into a single arc. This is accomplished by converting the node joining the two arcs into a vertex. Two arcs can only be merged if no other arcs are connected to the node separating the arcs. Otherwise, the node must be preserved to define the junction between the branching arcs.

Redistribute Vertices

The primary function of the vertices of an arc is to define the geometry of the arc. If the arcs are to be used for automatic mesh generation, the spacing of the vertices is important. The spacing of the vertices defines the density of the elements in the resulting mesh. Each edge defined by a pair of vertices becomes the edge of an element. The mesh gradation is controlled by defining closely spaced vertices in regions where the mesh is to be dense and widely spaced vertices in regions where the mesh is to be coarse.

When spacing vertices along arcs, the *Redistribute* command in the *Feature Objects* menu can be used to automatically create a new set of vertices along a selected set of arcs at either a higher or lower density. The desired arc(s) should be selected prior to selecting the *Redistribute* command. The *Redistribute* command brings up the dialog shown in Figure 14.4.

The current status of the selected arc(s) is given at the top of the dialog. This includes the number of segments and spacing of those segments. When multiple arcs are selected, the current status is a combination of all selected arcs. However, the parameters set in this dialog apply to each arc individually. Therefore if multiple arcs were selected, each arc would reflect the options selected in this dialog.

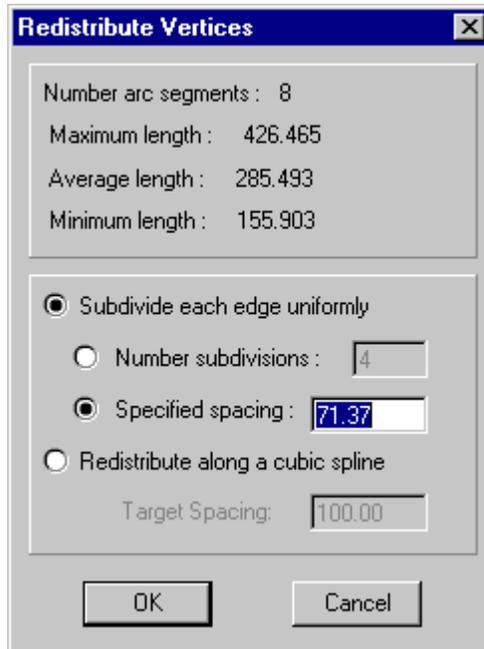


Figure 14.4 Redistribute Vertices Dialog.

The following options are available for redistributing vertices:

Linear Interpolation

If the *Linear interpolation* options is specified, then either a number of intervals or a target spacing can be given to determine how points are redistributed along the selected arcs. In either case, the new vertices are positioned along a linear interpolation of the original arc. The arc may change shape due to the fact that original vertices are removed as the new vertices are created. This may round corners from the arc.

Spline Interpolation

If the *Spline interpolation* option is specified, vertices are redistributed along a series of cubic splines defined by the original vertices of the selected arcs.

The difference between the linear and spline interpolation methods is illustrated in Figure 14.5.

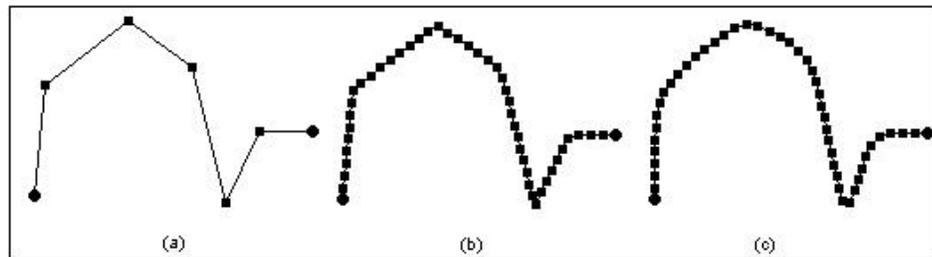


Figure 14.5 Redistributing Vertices. (a) Original Arc (b) Linear (c) Spline.

Transform

Conceptual models may be constructed in sections as data is gathered. When combining sections, it may be necessary to transform one section to make the datum align with the other section. The transform command allow selected GIS entities or all entities in a coverage to be scaled, rotated or translated to accomplish this datum correction.

14.1.4 Coverages

As mentioned before, feature objects are grouped into coverages. A coverage is similar to a layer in a CAD drawing. Each coverage represents a particular type of information. For example, one coverage could be used to define one dimensional model features such as river centerlines and cross-section locations. Another coverage could be used to define two-dimensional material zones. These objects could not be included in a single coverage since polygons within a coverage are not allowed to overlap and material regions will cover one dimensional geometry features.

Other coverages may exist to define attributes such as boundary conditions for a specific model or to define entities for model verification (see Chapter 15).

When *SMS* is first launched, a default coverage is created. Any feature objects created are added to this coverage. When multiple coverages are created, one coverage is designated the "active" coverage. New feature objects are always added to the active coverage and only objects in the active coverage can be edited.

Each coverage has a coverage type associated with it. The coverage type controls what attributes are assigned to the objects within the coverage. For example, with a *TABS* type of coverage, materials can be assigned to polygons, element sizes can be assigned to points, and *RMA2* boundary conditions can be assigned to nodes and arcs.

Options related to coverages are controlled with the *Coverages* dialog (Figure 14.6). This dialog is accessed through the *Coverages* command in the *Feature Objects* menu.



Figure 14.6 The Coverages Dialog.

The currently defined coverages are listed in the text box in the upper left corner of the dialog. One of the coverages in the list is always highlighted. The name and other attributes associated with the highlighted coverage are edited with the other fields in the dialog. The coverage that is marked with an “A” is the active coverage.

Name

Each coverage has a name that is used in the list to identify the coverage. The name of a coverage is edited by selecting the coverage and editing the name in the *Name* field.

Default Elevation

Coverages are essentially two-dimensional entities. All objects within a coverage are displayed in the same XY plane. The default elevation field can be used to define the Z elevation of the XY plane containing the coverage. Since an underlying scatter set almost always becomes the source for the elevation of numerical models, this value does not have much significance.

Creating/Deleting Coverages

A new coverage is created by selecting the *New* button. This adds a new empty coverage to the list. Another way to create a new coverage is with the *Copy* button. This creates a new coverage with the same set of feature objects as the selected coverage. This is useful when two coverages share the same boundary or zones as previously discussed. An existing coverage can be deleted by highlighting the coverage and selecting the *Delete* button.

Setting the Active Coverage

A coverage is selected to be the active coverage by selecting the *Active* button with the desired coverage highlighted. In previous versions of SMS, highlighting the coverage and leaving the dialog specified the active coverage. This does not change the active coverage in this version.

Display Color

When several coverages are present, the display of coverages can become confusing. Each of the feature objects in a coverage has a set of display options (color, line style, etc.) that can be edited in the *Display Options* dialog. However, these colors are only used to display the objects in the active coverage. All of the objects in the inactive coverages are displayed using the same color. By default, this color is a light gray color. The inactive coverage color can be edited in the *Display Options* dialog.

Visibility

In some cases it is useful to hide some or all of the coverages. Each coverage has a visible flag that can be edited. Only visible coverages are displayed. The flag for a coverage is edited by highlighting the coverage and selecting the visible toggle in the lower left corner of the *Coverages* dialog. Coverages which are visible have a small "v" displayed before the coverage name. The visible flag for all coverages can be edited at once using the *Hide All* and *Show All* buttons.

Attributes Sets

Each coverage is assigned a coverage type which controls which set of attributes are associated with the coverage. The appropriate attribute set for a coverage depends on the intended use of the coverage.

Currently SMS supports coverages for each supported numerical model. For example, if *FESWMS* is to be run as the numerical model, a coverage with type of *FESWMS* should be used. This coverage includes generic two-dimensional mesh attributes such as element type, but also supports attributes specific to *FESWMS* such as weakly reflecting boundaries and essential water surface elevation. Refer to the overview tutorial for an example of the attributes and commands associated with a specific coverage.

Coverages also exist for one-dimensional cross section and profiles (WSPRO) and observation tools.

Coverage Type

The coverage type assigned to a coverage defines what the data in the coverage will be used for. SMS supports four categories of coverages. These include two-dimensional finite element coverages, one-dimensional river model coverages, land use coverages, and observation coverages. Two-dimensional finite element coverages are associated with a specific numerical engine such as RMA2 or FLO2DH. Boundary conditions

specific to these models can be assigned to the feature objects of these coverages and SMS will generate finite element networks for these models. One-dimensional river model coverages are associated with a specific numeric engine such as WSPRO. Features created in this coverage represent one-dimensional entities such as cross sections and centerlines. These coverages can be used in conjunction with land use coverages, which define properties associated with geometric areas of the coverage. Material types are assigned to polygons in the land use coverage, and then mapped to the appropriate portion of the one-dimensional cross sections. Chapter 15 describes the application and features of observation coverages.

14.1.5 Feature Object Attributes

Attributes of feature objects are assigned and edited by selecting the objects and selecting the *Attributes* command from the *Feature Objects* menu, or by double clicking on the object. Depending on the type of the object selected, this brings up the *Point/Node Attributes* dialog, the *Arc Attributes* dialog, or the *Polygon Attributes* dialog. The contents of the dialogs (the attribute options) depend on the attribute set assigned to the active coverage. This section will use the two-dimensional mesh coverage as an example of the dialogs. Creating a new coverage or assigning an active coverage is described in Section 14.1.4.

Several of the coverage types supported in SMS are associated with a specific numerical model. SMS currently includes interface to six different finite element engines. These engines all have similarities, which are exploited, using similar tools to edit coverages of these types.

Point/Node Attributes

The *Point/Node Attributes* dialog for a coverage for a two-dimensional finite element model is shown in Figure 14.7.

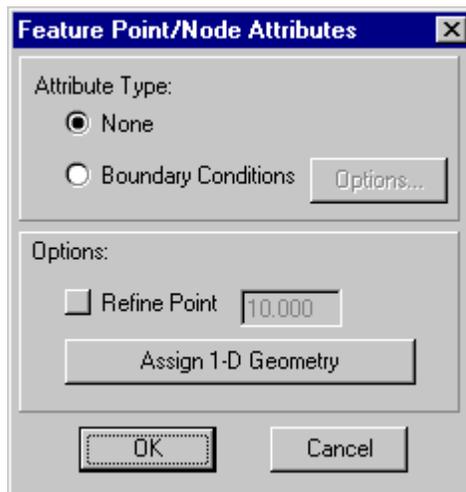


Figure 14.7. The Point/Node Attributes Dialog.

Nodes or points can be assigned boundary conditions (steady state or transient based on the coverage setting) which are defined by the numeric model. To see what these boundary conditions are, see the chapters in this document for the appropriate model.

The attributes assigned to feature points and nodes will be saved as nodal boundary conditions when the feature objects are converted to a two dimensional mesh. The attributes that can be assigned to feature nodes and points are as follows:

Refine Point

A point can be made a refine point by clicking on the Refine Point button. A refine attribute is assigned to points or nodes to control the grid density around a point when a mesh is constructed. The element size can be entered into the edit field to the right of the button.

Arc Attributes Dialog

The *Arc Attributes* dialog is shown in Figure 14.8. This dialog is used to input and edit attributes assigned to arcs. The most significant function of this dialog is to specify which type of attribute is assigned to the arc.

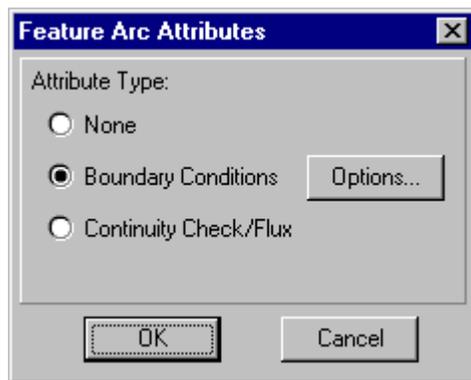


Figure 14.8. The Arc Attributes Dialog

An arc can be purely geometrical, it may be a continuity (flux) check point, or it may have a boundary condition associated with it. The boundary conditions supported depend on the type of model being generated. If the coverage is for *TABS*, the nodestring boundary conditions are associated with arcs and arc groups. Clicking the *Options* button would invoke the *RMA2 Nodestring Boundary Conditions* dialog. For a description of the boundary conditions for each model, refer to the appropriate chapter in this manual.

Polygon Attributes Dialog

The *Polygon Attributes* dialog is shown in Figure 14.9.

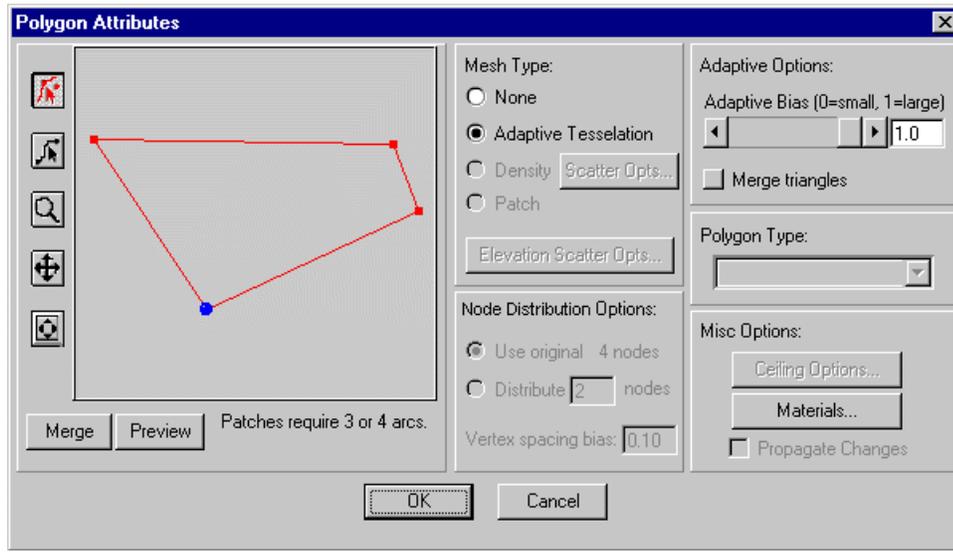


Figure 14.9. The Polygon Attributes Dialog

This dialog is used to set the attributes for feature polygons. Attributes that can be specified for each polygon include:

Mesh Type

The mesh type defines how elements will be generated to fill the polygon when the numeric model network is generated. There are four mesh types that can be assigned to each polygon. They are as follows:

NONE

A polygon can be assigned a mesh type of none by clicking on the *None* button. This results in a hole in the finite element network.

Adaptive Tesselation

A polygon can be assigned a mesh type of Adaptive Tesselation by clicking on the *Adaptive Tesselation* button. A sample mesh created using the adaptive tessellation technique is shown in Figure 14.10.

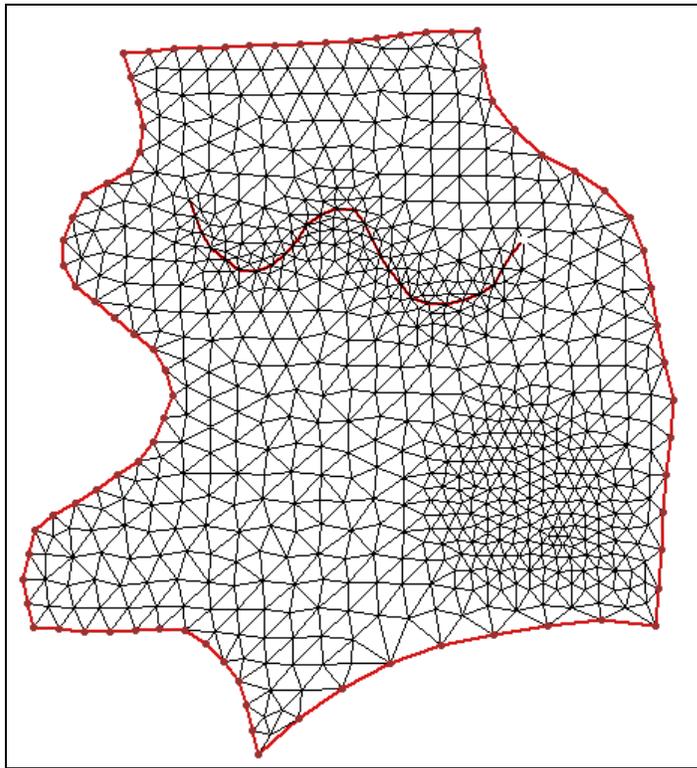


Figure 14.10 Sample Mesh Created Using Adaptive Tessellation Method.

When using adaptive tessellation, *SMS* uses the existing node spacing on the feature objects defining the input polygon to determine the element sizes on the interior. It should be noted that the characteristics of the input polygon include the boundary, interior arcs and refine points. If the input polygon has varying node densities along its perimeter, *SMS* attempts to create a smooth element size transition between these areas of differing densities. By altering the size bias in the *Adaptive Options* section of the *Feature Polygon Attributes* dialog, the user can indicate whether *SMS* should favor the creation of large or small elements. Decreasing the bias will result in smaller elements; increasing the bias will result in larger elements. In either case, the elements in the interior of the mesh will honor the arc edges and the element sizes specified at nodes. The bias simply controls the element sizes in the transition region.

Density

Density meshing is a variation of adaptive tessellation. The difference is that instead of element size being controlled by the vertex spacing on the polygon boundary, the user provides a target size function. This function is defined at scatter point locations. The vertices are distributed along the polygon boundary to attempt to match the target size in each location, and then points are distributed on the interior to match the target size there.

This process requires that at least one scatter point set exists, and that a set has at least one function. Examples of target size functions include projected water depths, and gradient of bathymetry. Size based on depth would result in small elements in shallow areas and large elements in deep water. Size based on gradient would result in small elements where the depth is changing quickly and large elements where the bathymetry is flat.

Patch

Patching is a meshing process described in Sections 4.7.4 and 4.7.5. The technique is based on Coons patches. Patches must have three sides for a triangular patch or four sides for a rectangular patch. Both can be defined from the *Polygon Attributes Dialog*. If a polygon cannot be patched, a help string under the *preview window* explains what needs to be changed. Modifying an existing polygon to have three or four sides is described in the following section. The goal of patches is to align the elements inside the patch to the boundary. This is applicable to areas of highly ordered flow such as in a channel.

Elevation Scatter Opts

Features objects do not have elevation or bathymetry data associated with them intrinsically. Elevations may be assigned through the scatter point module after the network is generated, but this requires the entire mesh to use the same source of bathymetry. One way to specify a source of bathymetry for a single polygon is to define the scatter set, and select that set using the *Elevation Scatter Opts* button.

Merge triangles

When the meshing type results in triangular elements, the user has the option to tell *SMS* to merge triangles into quads. This is just like the *Merge Triangle* command in the mesh module described in Section 4.7.1.

Polygon Material or Type

Depending on the coverage type, polygons can be assigned to either a material or a polygon type. The polygon type is selected from a list of types supported by the numeric model. ADCIRC and CGWAVE use this method. Models such as FLO2DH and RMA2 allow the definition of material types. These are specified using the *Materials* button

Node Distribution Options

In the *Node Distribution Options* section of the *Polygon Attributes* dialog, the can redistribute the nodes along selected arcs, or use the original vertex distribution. The *Vertex spacing bias* allows the newly spaced vertices to be distributed in a non-uniform manner. The user specified the relationship between the first space and the last. For example, if a value of 2 is entered, the last space will be twice as large as the

first. This is dependent on the orientation of the arc and experimentation may be required.

Graphical Tools

The *Polygon Attributes* dialog includes a preview window on the left side. This window shows the arcs and nodes of the selected polygon and allows the user to interact with that definition. The *Preview* button generates the elements that will be created for the polygon. It is recommended that the preview is used with the patch and adaptive tessellation options only due to the time required performing density meshing. There are several tools for modifying the existing polygon. Zooming, panning and framing work in the preview window just as they normally would in the graphics window of *SMS*. They are used to facilitate the selection tools, which are:

Select Node 

While this tool is selected, the user selects nodes in the preview window. Once selected the arcs connected to that node can be merged using the *Merge* button. Arcs may be merged to create three or four sides to the polygon for patching.

Select Arc 

While this tool is selected, the user selects arcs in the preview window. Once selected the node distribution options become active.

Ceiling

A polygon can be assigned a ceiling value, or a value above which water flow is prohibited for a FESWMS type coverage. The ceiling will be assigned to each mesh node after meshing has taken place.

To assign a ceiling value, the user can either enter four points or the *a*, *b*, *c*, and *d* values of a plane equation. The ceiling value for each node will be interpolated and assigned after meshing.

14.1.6 Constructing Numeric Models

The purpose for constructing conceptual models is to allow the automatic generation of the numerical model. This may be a finite element mesh with boundary conditions, material parameters and model control, or a set of one-dimensional cross sections. If the resulting network or set of cross sections have attributes that are not desired, it may be edited directly using the appropriate module in *SMS*, or the conceptual model may be modified and a new network generated. The generation is invoked using the *Map->2D Mesh* or *Map->River* commands in the feature objects menu.

Map -> 2D Mesh

Although several options are provided in the *2D Mesh* (Chapter 4) module for automated mesh generation, the Map->2D Mesh method is by far the most powerful method for generating a mesh. The result of constructing a mesh from feature objects is illustrated in Figure 14.11.

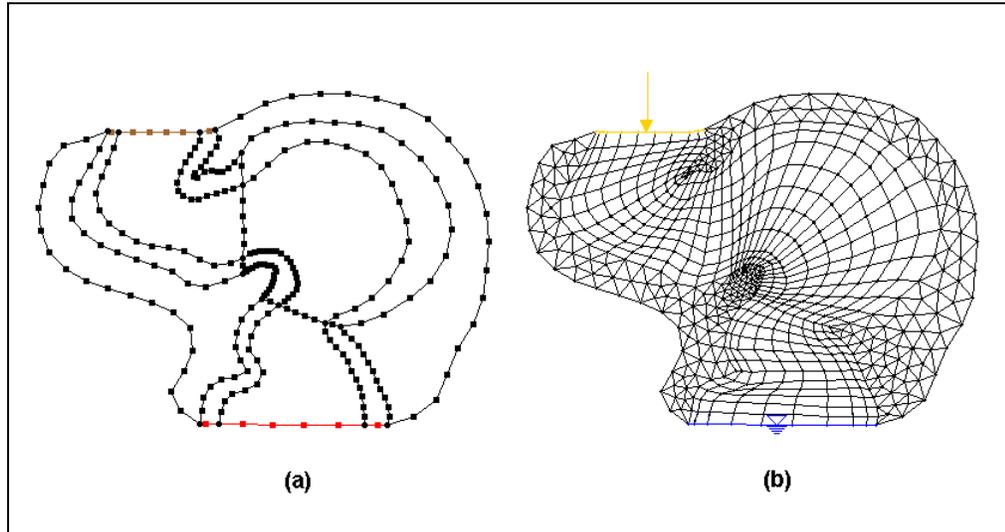


Figure 14.11 Mesh Generation with Feature Objects. (a) Feature Objects (b) Resulting Mesh.

The types of feature objects that are used to guide the construction of the mesh are shown in Figure 14.11a and the resulting mesh is shown in Figure 14.11b. This example shows the use of adaptive tessellation as well as patches. Boundary conditions are assigned to the arc groups and propagated to the network. The domain of the network matches the polygons, and attributes such as material zones are specified in the conceptual model and propagated to the network too.

Map->2D Mesh can be applied to selected polygons or the entire coverage. When the *Map->2D Mesh* command is selected, the dialog shown in Figure 14.12 appears.

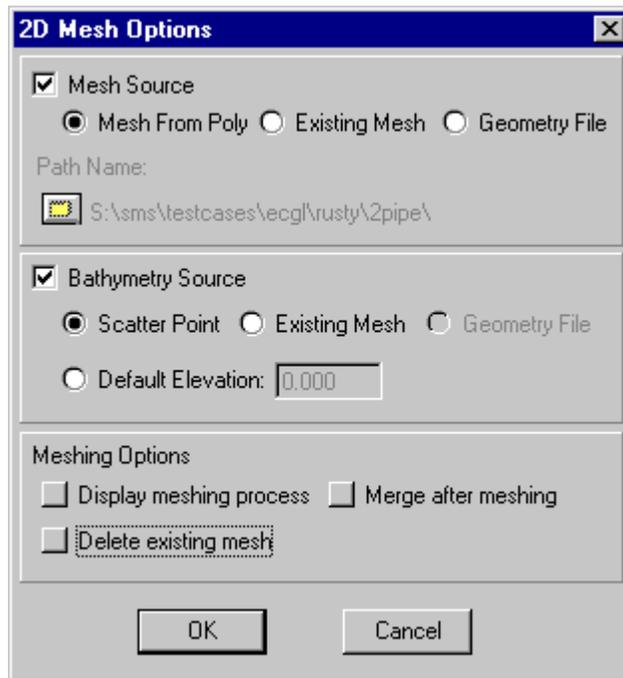


Figure 14.12 The 2D Mesh Options Dialog.

The result of the operation is a finite element mesh. If the *Delete existing mesh* toggle is selected, any exiting network is deleted. This toggle should not be selected if one or more polygons are selected and they are to be merged with the existing mesh. The newly created mesh consists of nodes, elements with material attributes, and node strings with boundary conditions specified. The elements can be either linear or quadratic based on the current element setting in the mesh module. They can also be converted after the mesh is generated if needed.

If the *Merge After Meshing* option is used, SMS merges a portion of the triangular elements into quadrilateral elements following the meshing process. This tends to reduce the total number of elements in the mesh. The elements can also be merged after the meshing process is completed using the *Merge Elements* command in the *2D Mesh* module. This command and the element merging process are described in more detail in Section 4.7.9.

If the *Display meshing progress* option is selected, each step of the meshing process is displayed. This tends to slow down the meshing significantly.

Mesh Source

The mesh source allows the user to specify where the elements and nodes will come from. The options include using the polygon attributes, the existing mesh clipped to the polygons, or another geometry file clipped to the polygons. Generally the polygon attributes should be used.

Bathymetry Source

The bathymetry source option allows the user to specify where bathymetry will come from. Options include a scatter set, the existing mesh, or a constant value. The scatter set causes the bathymetry to be generated from the polygon attributes.

Map -> River

The *Map->River* command tells SMS to construct cross sections at the specified arc locations. The bathymetry is extracted from the underlying scatter point set or node network. An area property coverage (land usage) can be used to define sub-sections where roughness or other property changes in the cross section.

14.1.7 Display Options

The *Display Options* command in the *Feature Objects* menu controls the display of coverages and feature objects. The options that appear in the *Display Options* dialog depend on the type of the active coverage. A sample dialog, used for two-dimensional mesh coverages is shown in Figure 14.13.

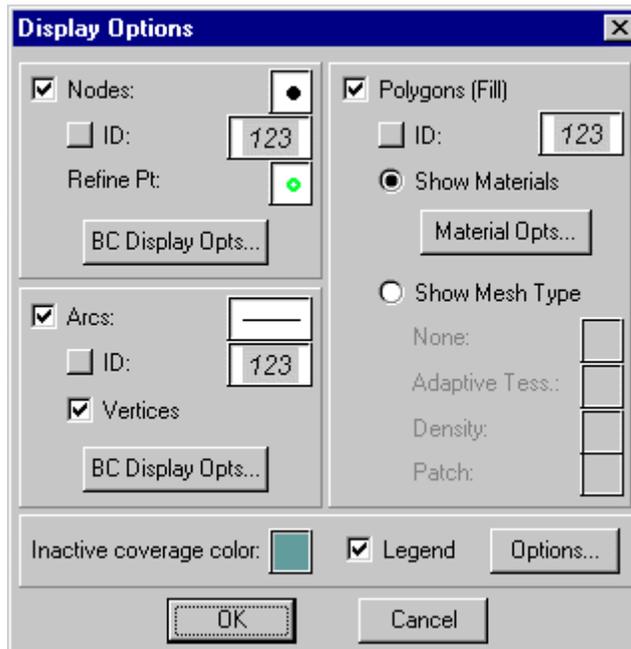


Figure 14.13 The Feature Object Display Options Dialog.

ID

If this option is selected, the id of each of the feature objects is displayed next to the object. The graphical attributes of the text used to display the ids are edited by clicking on the fields on the left side of each id toggle box. ID's are only displayed if the corresponding feature object is displayed.

Nodes

This option is used to display nodes. The graphical attributes of the nodes (symbol, color, size, etc.) are edited by clicking on the fields in the node section of the dialog.

Arcs

This option is used to display arcs. The graphical attributes of the arcs (color, line style, thickness, etc.) are edited by clicking on the fields in the arc section of the dialog.

Vertices

This option is used to display the vertices of arcs. A small dot is placed on the arcs at the location of each of the vertices. The color of the vertices is the same as the color of arcs. Vertices are only displayed when arcs are displayed.

Polygons

If this option is selected, polygons are displayed filled. The graphical attributes of the polygons (fill color) are edited using the fields in the polygon section of the dialog. The polygon fill color may represent either the mesh type or material property.

Inactive Color

The inactive color is used to display all of the objects in inactive coverages.

Legend

The legend item can be used to display a legend listing each of the feature object types being displayed. The legend shows what graphical attributes (symbol, line style, fill color and pattern) are being used to display each feature object type.

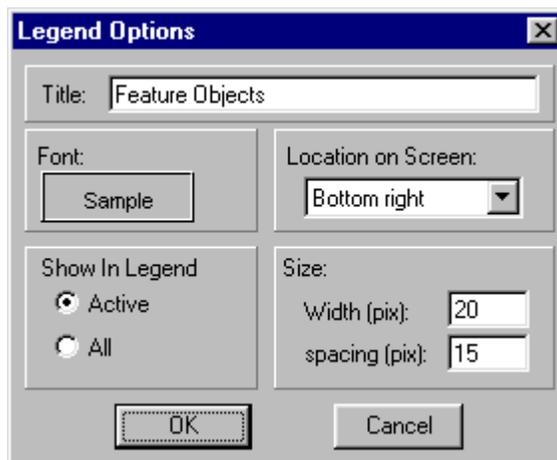


Figure 14.14 Feature Legend Options Dialog.

Summary

Polygons, arcs and isolated points may all have only one type of attribute. Nodes, however, may be connected to multiple arcs of different types. Because of this, nodes may have more than one type of attribute. Nodes are limited though, to having only the same types of attributes that the attached arcs have.

14.2 Drawing Objects

The drawing objects in the *Map* module provide a set of tools for adding simple graphics and annotation to a plot. These tools are not intended to be a full-featured drawing package as would be found in products like *AutoCAD* or *Corel Draw*. However, they can be very useful for adding titles, arrows, and other annotation to a plot so that the plot can be directly included in a project report without the need to import the plot into an external drawing package prior to report generation.

The types of drawing objects that can be created are text, lines (including arrows), rectangles, and ovals. Drawing objects are created and edited using tools in the *Tool Palette*. Drawing objects are saved in the Map file along with feature objects.

14.2.1 Drawing Object Tools

The following drawing object tools are in the dynamic portion of the *Tool Palette* when the *Map* module is activated. Only one tool is active at any given time.



Create Text Tool

The *Create Text* tool is used to create a single line text string. The location clicked on defines where the text string will be placed. After clicking on a location, the *Text Attributes* dialog appears allowing you to enter the text string and choose the font, color, etc.



Create Rectangle

The *Create Rectangle* tool is used to create wire frame or filled rectangles. Rectangles can be used to represent buildings, frame a series of text strings, etc.. Rectangles are created with this tool by dragging a rectangle with the mouse at the location on the screen where you wish to place the rectangle.



Create Oval

The *Create Oval* tool can be used to create wire frame or filled ovals. Ovals are created by dragging a rectangle with the mouse at the location on the screen where you

wish to place the oval. The rectangle width and height determine the major and minor axes of the oval.



Create Line

The *Create Line* tool can be used to create single line segments or polylines (a series of connected segments). An arrowhead can be placed on either end of the line. Lines are typically used in conjunction with text strings to highlight key features in a plot. A line is created by clicking on a series of points on the screen with the mouse and double-clicking to end to end the line. The color, line style, and arrowhead options of a line are edited with the *Attributes* dialog described below.



Select Drawing Objects

The *Select Drawing Objects* tool is used to select previously created text, rectangles, ovals, and lines. Once selected, a drawing object can be moved to another location by clicking on the object and dragging it to a new location. Lines, rectangles, and ovals can be resized by dragging the handles that appear on the corners or ends of the object when the object is selected. The *Select Drawing Objects* tool is also used to edit the graphical attributes as described in the following section.

14.2.2 Display Attributes

Each type of drawing object has a set of graphical attributes that can be edited by selecting the object with the *Select Drawing Objects* tool and selecting the *Attributes* command in the *Drawing Objects* menu. The attributes can also be edited by double-clicking on an object.

Text Attributes

If a text object is selected, the *Attributes* command in the *Drawing Objects* menu brings up the dialog shown in Figure 14.15. The dialog can be used to change the font, the color, or the text string itself. An option is also provided to fill a rectangle just containing the text with a user-specified color. This option can be useful to help the text stand out from the objects being drawn behind the text.

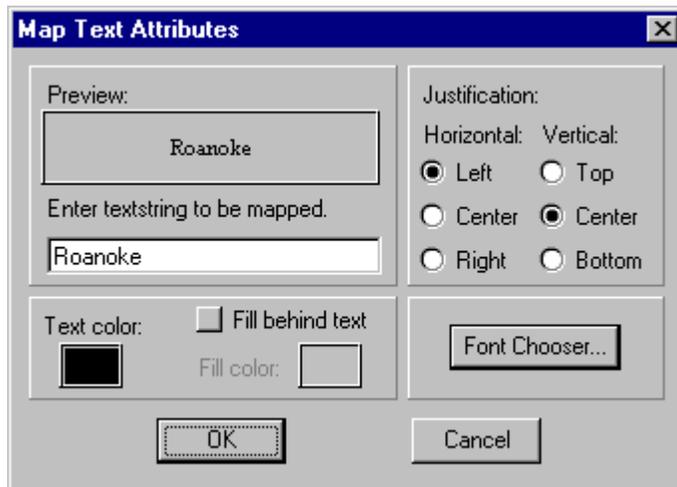


Figure 14.15 The Text Attributes Dialog.

Rectangle and Oval Attributes

The attributes for both rectangles and ovals can be edited with the *Rectangle/Oval Attributes* dialog shown in Figure 14.16. Rectangle and oval attributes include line style, line color, and line width. An option can also be set to either draw only the outline of the rectangle or oval (no fill) or fill the object with a user-specified fill pattern and color.

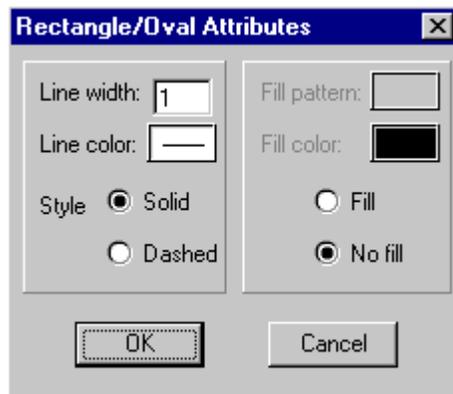


Figure 14.16 The Rectangle and Oval Attributes Dialog.

Line Attributes

The attributes for lines are edited with the *Line Attributes* dialog shown in Figure 14.17. The line attributes include line color, line width, and line style. The arrowheads associated with a line can also be edited. The length and width of the arrowhead can be defined along with the placement of the arrowheads. The arrowheads can be placed at the beginning of the line, the end of the line, or at both ends of the line.

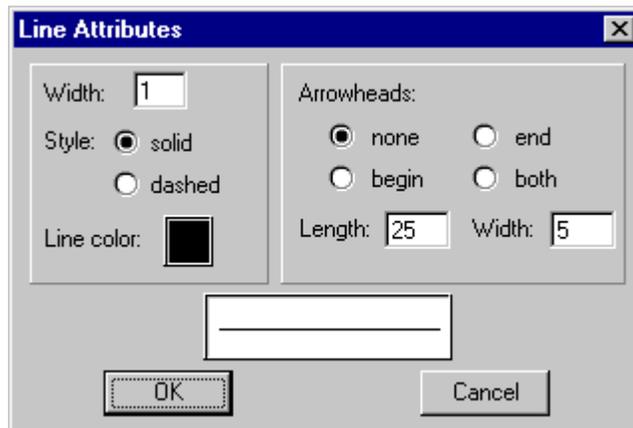


Figure 14.17 The Line Attributes Dialog.

Default Attributes

When a new object is created, it inherits the default attributes for that object type. The default attributes are defined by selecting one of the drawing object tools (line, rectangle, oval or text) and selecting the *Attributes* command in the *Drawing Objects* menu.

14.2.3 Display Options

The *Display Options* command in the *Drawing Objects* menu brings up the dialog shown in Figure 14.18. The dialog is used to toggle the display of text, lines, rectangles, and ovals. Turning off the toggle for an item disables the display of the item but does not delete the item.

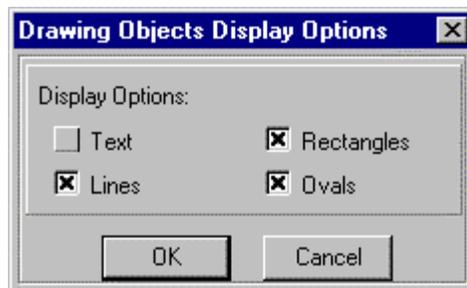


Figure 14.18 The Drawing Object Display Options Dialog.

14.2.4 Drawing Order

The order in which drawing objects are displayed becomes important whenever a rectangle or oval is displayed in the color fill mode. The order of drawing objects can be controlled using the *Move to Front*, *Move to Back*, *Shuffle Up*, and *Shuffle Down* commands.

Move to Front

The *Move to Front* command causes the selected drawing object to be drawn last. In other words it appears on top or in front of all other drawing objects.

Move to Back

The *Move to Back* command causes the selected drawing object to be drawn first. In other words it appears at the bottom or in back of all other drawing objects.

Shuffle Up

The *Shuffle Up* command causes the selected drawing object to be displayed one object later than it is currently displayed. This causes it to appear in front of the object which is currently being displayed just ahead of it.

Shuffle Down

The *Shuffle Down* command causes the selected drawing object to be displayed one object sooner than it is currently displayed. This causes it to appear in back of the object which is currently being displayed just behind it.

14.3 Images

An image is a digital image such as a scanned map or aerial photo. A common format for saving such images is the TIFF (Tags Image File Format) format. TIFF images can be imported to SMS and displayed in the background to aid in the placement of objects as they are being constructed or simply to enhance a plot.

14.3.1 Importing an Image

The first step in using a new digital image for either background display or for texture mapping is to import the image. This is accomplished by selecting the *Import* command in the *File* menu. This will open the *Select Import Format* dialog, where you will choose *TIF/GIF*. This brings up the *File Browser* which is used to select the TIFF file. The selected TIFF file must be in the "packbits" compressed format. If your image is not in this format, you will need to convert it using an image processing program such as *XV* (UNIX) or *Paint Shop Pro* (PC).

After selecting the TIFF file in the *File Browser*, the *Register Image* dialog always appears. This dialog is described in the following section.

14.3.2 Registering an Image

Before an image can be used for background display or for texture mapping, the image must be "registered". Registering an image involves identifying three points on the image corresponding to locations with known real world (XY) coordinates. Once these points are identified, they are used by *SMS* to stretch or map the image to the proper location when it is drawn with the other objects in *SMS* in the *Graphics Window*. If an image is not registered properly, any objects which are created using the background image as a guide will have the wrong coordinates.

An image is registered using the dialog shown in Figure 14.19. This dialog is used to register a new image as it is being imported. It appears automatically after the *Import* command is selected and a new TIFF file is chosen. It can also be accessed using the *Register* command to change the registration of a previously imported image.

The *Lat/Lon* calculator buttons can be used to convert a latitude-longitude pair into equivalent UTM coordinates.

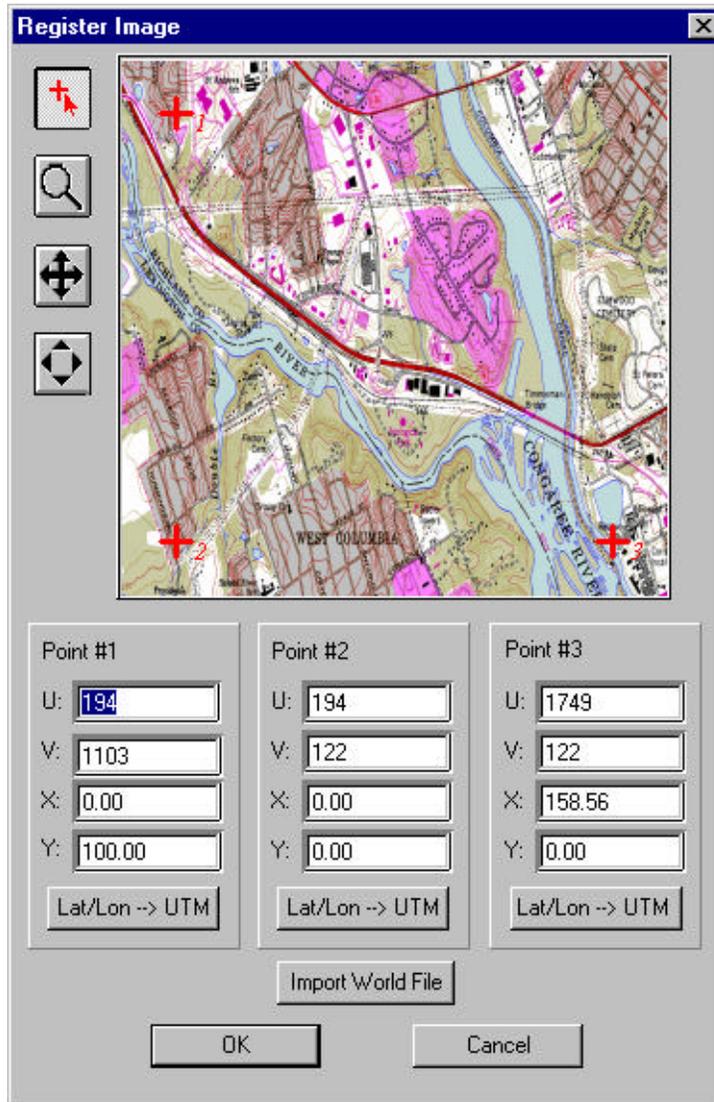


Figure 14.19 The Register Image Dialog.

The main feature of the *Register Image* dialog is a large window in which the image is displayed. Three points (shown by "+" symbols) are also displayed in the window. These points are used to identify locations with known real world coordinates. The real world coordinates (X,Y) and image coordinates (U,V) of the three registration points are listed in edit fields below the image. The points are moved to the desired locations on the image by dragging the points using the tools to the left of the image (described below). Once the points are located, the real world coordinates can be entered in the edit fields shown below the image.

The following tools can be used to help position the registration points:

 **Select Point Tool**

The *Select Point* tool is used to select and drag register points to a location on the map for which real coordinates are known so that they can be entered in the corresponding XY edit fields.

 **Zoom Tool**

In some cases, it is useful to magnify a portion of the image so that a registration point can be placed with more accuracy. The *Zoom Tool* is used to zoom in a portion of the image.

 **Pan Tool**

After zooming in on a portion of the image, the *Pan Tool* is used to pan the image vertically or horizontally.

 **Frame Macro**

The *Frame Macro* is used to automatically center the entire image within the drawing window of the dialog after panning and zooming in on a specific location.

14.3.3 Resampling an Image

Once an image is registered, it is positioned and displayed in the *Graphics Window* such that the entire image is visible. As the image is first drawn, the image goes through a process called "sampling". Sampling is a process of converting the image from its actual resolution to the screen resolution. TIFF images typically have a much higher resolution (pixels per inch) than the computer screen. During the sampling process, each pixel on the screen is assigned a color by evaluating the pixels of the TIFF image at the same location. This process can take a few seconds or a few moments, depending on the size of the image and speed of your computer. Once the process is complete, the image is drawn to the screen.

Since the sampling process can take a significant amount of time, it is not convenient to wait for the image to be resampled each time the screen is refreshed or each time the image is panned or zoomed. Thus, to speed up the image display, when the image is first sampled *SMS* stores a bitmap copy of the sampled image at the screen resolution. This bitmap image is then used to refresh the display of the image.

In many cases, the region of interest in an image corresponds to a small sub-region of the image. In such cases, it is possible to focus on the region of interest by zooming in with the *Zoom* tool. However, after zooming in, *SMS* initially displays the image by stretching the bits of the sampled bitmap described above. This results in a grainy image, the graininess depending on the level of magnification. The magnified image can be restored to a high resolution, sharp image by selecting the *Resample* command

from the *Image* menu. The *Resample* command repeats the sampling process described above to generate a screen resolution bitmap image of the currently visible region using the underlying high resolution TIFF image. Of course, since the TIFF image has a limited level of resolution, zooming in too far will result in a situation where the resolution of the screen exceeds the resolution of the TIFF image. In such cases, the *Resample* command is ineffective in increasing the clarity of the image.

After zooming in and resampling an image, it may be necessary to pan the image or zoom back out. When doing so, it will be discovered that the entire image is no longer visible. Only the portion of the TIFF image that was visible when the last *Resample* command was issued is visible. To make a different portion of the image visible, pan or zoom to where the desired region fits in the *Graphics Window* and select the *Resample* command again.

14.3.4 Fit Entire Image

As described in the previous paragraph, after zooming in on a small sub-region of the image and selecting the *Resample* command, it may be necessary to zoom back out and resample either the entire image or a different sub-region of the image. The *Fit Entire Image* command is provided to assist this process. When the *Fit Entire Image* command is selected, the visible region is zoomed out so that the entire TIFF image just fits within the *Graphics Window* and the boundary of the TIFF image is shown in red. At this point, the entire image can be resampled or a new sub-region can be zoomed and resampled.

14.3.5 Deleting Images

The *Delete* command in the *Images* menu is used to delete the current image.

14.3.6 Exporting the Resampled Region

Since TIFF images often have extremely high resolutions, they can require significant memory. For example, a digitized version of a USGS quad sheet can require as much as 40 MB in uncompressed form and 6 MB in compressed form. Fortunately, *SMS* works directly with TIFF images in the compressed form so there is no need to uncompress the entire image. However, memory may still be a concern. Even though *SMS* always works with a sampled bitmap which has a resolution no greater than required by the screen, the entire compressed image must be loaded into RAM whenever the image is resampled.

In many cases, only a portion or sub-region of a large image is needed for a modeling study. If so, the memory requirements can be reduced and the importing and resampling speed can be increased by clipping out the region of the TIFF required for the study prior to importing the image to *SMS*. This can be accomplished by reading the image into a graphics program such as *XV* or *Paint Shop Pro*, clipping out the

region of interest, and saving the clipped region to a separate file. This clipped region can then be imported to *SMS*.

Another option for saving a TIFF image is to use the *Export* command in the *File* menu. This command allows you to save the original TIFF image corresponding to the currently resampled TIFF file. When the *Export* command is selected, the dialog shown in Figure 14.20 appears.

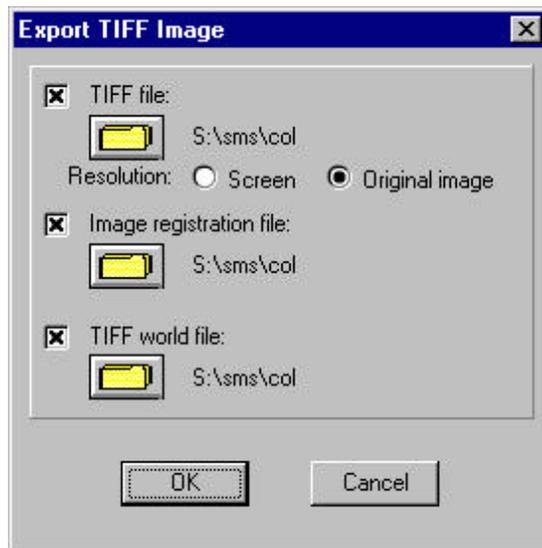


Figure 14.20 The Export Region Dialog.

The TIFF file item is used to designate the name of the new TIFF file that will be created containing the current resampled region. The *Image registration file* option is used to designate the name of the image file corresponding to the resampled image. The image file will include the name of the new TIFF file.

Two options are available for determining the resolution of the new TIFF file. If the *Screen* option is chosen, the TIFF image will be saved at a resolution which matches the screen resolution. If the original TIFF image has a high resolution, this can significantly reduce the memory required to store the resampled region. However, if this option is chosen, once the image is read back in, you will not be able to zoom in and resample the image. If the *Original image* option is chosen, the resampled region is saved using the pixel density of the original TIFF image. This allows you to zoom in on the image and use the *Resample* command after the image is read back into memory.

The last option is to save a *TIFF world file*. This file stores information on the register points. In the register image dialog this file can be imported to automatically assign values to the register points of that image.

14.3.7 Export TIFF vs. Save Image

There are two options for saving TIFF images or files related to TIFF images. A summary of the two options is provided here to help reduce confusion concerning the differences between the two commands.

Export TIFF

When the *Export* command is selected from the *File* menu, one of the options listed in the resulting *Export* dialog is a TIFF file. This option is used to save a TIFF image of whatever is currently being displayed in the *Graphics Window*.

Save Image

The *Save* command in the *File* menu can be used to save an image file. An image file contains the name of the file containing the TIFF image, the registration points, and the bounds of the currently resampled region. Once the file is saved, an image can be restored by reading the image file using the *Open* command in the *File* menu. *SMS* reads the image file, opens the TIFF file, and registers and resamples the image. This makes it possible to restore an image without having to repeat the registration and resampling process.

14.4 DXF Files

In many modeling studies, drawings of the site being modeled are generated in a CAD package such as *AutoCAD*. These drawings can be exported from the CAD package in the DXF format. DXF stands for "Drawing Exchange Format" and is supported by most CAD programs. DXF files can be imported to *SMS* and displayed in the *Graphics Window* to assist in model placement or simply to enhance the display of a model.

14.4.1 Importing DXF Files

DXF files are imported to *SMS* using the *Import* command in the *DXF* menu. Once a file is read, the objects in the file are displayed in the *Graphics Window*. *SMS* currently supports the R12-R14 (Release 12-14) versions of DXF files

14.4.2 Display Options

The objects in a DXF file are organized into layers. The display of layers in a DXF drawing is controlled using the *Display Options* command in the *DXF* menu. This command brings up the dialog shown in Figure 14.21.

In the *DXF Display Options* dialog, the color and style of DXF objects are determined from the application that originally created the drawing file. These options can be changed for individual layers for each entity type in that layer. In addition, the visibility of each layer can be specified. The names of the layers in the drawing are shown in the box at the top of the dialog. An 'v' appears to the left of the names of the visible layers. The visibility of a layer is toggled on or off by selecting the *Visible* toggle while the layer name is selected. All of the layers can be made visible with the *Show All* button. Likewise, all of the layers can be hidden with the *Hide All* button. The *Delete* button deletes the selected layer.

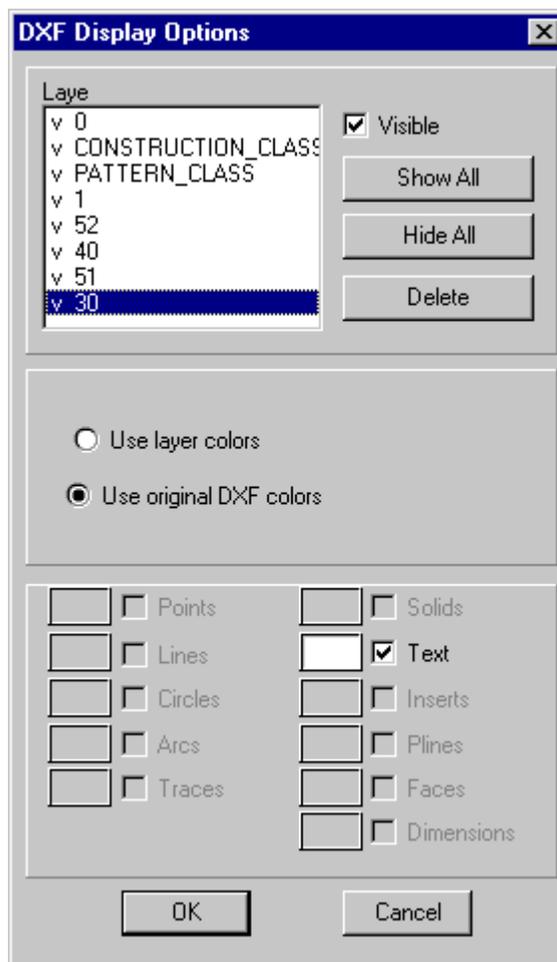


Figure 14.21 The DXF Display Options Dialog.

14.4.3 DXF -> Feature Objects

A set of DXF objects which have been imported to *SMS* can be converted to feature objects by selecting the *DXF -> Feature Objects* command in the *DXF* menu. DXF points are turned into points, DXF lines and polylines are turned into arcs, and DXF polygons are turned into polygons. The feature objects are added to the active coverage. Once converted, the feature objects can be used to build conceptual models.

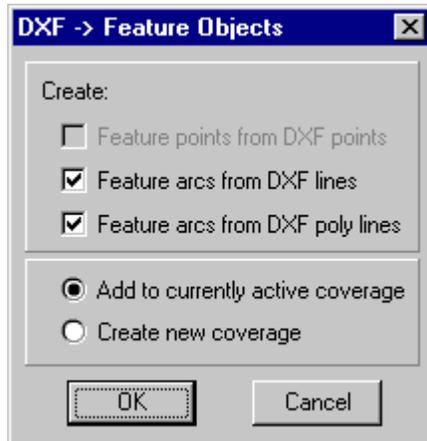


Figure 14.22 DXF to Feature Objects Conversion Dialog.

14.4.4 DXF -> Scatter Points

A set of DXF objects which have been imported to *SMS* can be converted to a scatter point set by selecting the *DXF -> Scatter* command in the *DXF* menu. DXF points, line end points, polyline points, and face vertices are turned into scatter point in a set. Once converted, the scatter set can be used as bathymetric background data or density meshing size data.

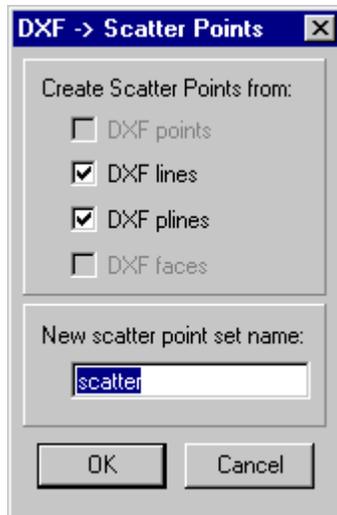


Figure 14.23 DXF to Scatter Point Set Conversion Dialog.

14.4.5 Deleting DXF Files

The *Delete* command is used to delete all DXF objects.

14.5 Reading and Saving Map Files

Most of the objects created in the *Map* module can be saved to a map file. Map files are saved by selecting the *Save* command in the *File* menu and selecting the *Map file* option. Feature objects, drawing objects, and the image file are saved in the map file. A map file can be read back into *SMS* using the *Open* command in the *File* menu.

Observation Tools

The *Map Module* of *SMS* includes a special coverage type called an observation coverage. Observation coverages have two principal purposes. The first is to aid in the process known as model verification (sometimes called calibration). The second is to allow the user to extract two dimensional data from a three dimensional data set for clarity.

Model verification is the process of modifying the input parameters to a numerical model until the output from the model matches an observed set of data. *SMS* includes a suite of tools to assist in the process of verifying a surface water model. One or more observed values (typically head or velocity) can be defined at observation points. With each value a confidence can be assigned. When a computed solution is imported to *SMS*, the residual error is plotted on a calibration target at each point and a variety of plots can be generated showing verification statistics. The verification tools can be used with any of the models in *SMS*.

Observation coverages are also useful for extracting data for two dimensional plots from three dimensional data. Such plots include profile and cross section plots and are useful for visualization of complex data.

Before reading this chapter, the basic tools and commands associated with feature objects and coverages should be understood. These topics are described in Chapter 14.

15.1 Overview of Verification Process

Two types of objects can be created in the observation coverage: points and arcs. Points represent locations in the field where some value has been observed. In most cases, the points will correspond to high water marks or velocity gages, and the value will be the elevation of the water surface (the head) or velocity. However, the verification tools are designed in a general fashion and the observed value can be anything (concentration, salinity, etc.) Each observed value can have a confidence interval or calibration target assigned to it.

When you have entered a set of observation points, *SMS* automatically interpolates the computed solution to the observation points when you read in a solution to a numerical surface water simulation. A symbol may be displayed next to each point, which graphically displays the error (computed-observed) superimposed on a plot of the calibration target (Figure 15.1). The size of the target is based on the confidence interval or the standard deviation. In addition to the symbols next to the observation points, you can choose to display any of a number of statistical plots.

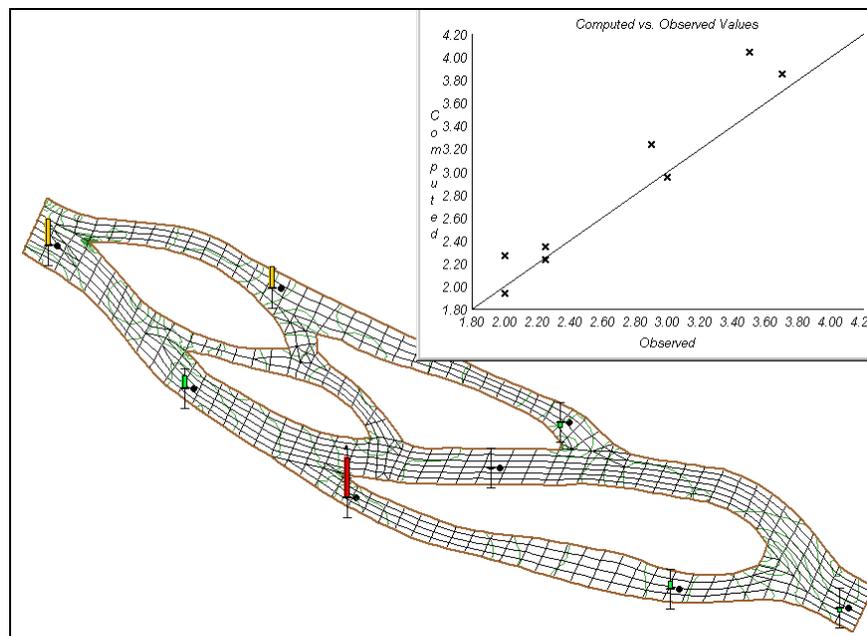


Figure 15.1 Calibration Targets and Plots of Calibration Statistics.

15.2 Observation Coverage

Observation points and profile arcs are managed in the *Map* module using an *Observation* coverage. A new coverage is marked as an *Observation* coverage by selecting the *Observation* option in the *Coverage Type* section of the *Coverages* dialog. Before creating any objects in the coverage, some general options should be defined for the coverage by selecting the *Options* button next to the *Observation*

option in the *Coverages* dialog. The *Options* button brings up the dialog shown in Figure 15.2.



Figure 15.2 The Observations Coverage Options Dialog.

Most of the options in this dialog are associated with observation points. The options are as follows:

15.2.1 Interpolation Method

The *Interpolation Method* controls the manner in which data set values are interpolated from the mesh to the observation points. Two options are available:

Interpolate From Neighbor Nodes/Cells

If the *Interpolate from neighbor nodes/cells* option is selected, the computed values at the neighboring set of nodes or cells are interpolated to the location of the observation points. Specifically, when interpolation is performed in a 2D mesh, the element containing the observation point is found and a value is computed at the point using the data set values at the nodes of the element.

Interpolation Equation

When interpolating from a set of nodes or cells to an observation point, the following inverse distance weighted interpolation equation is used:

$$F(x, y) = \sum_{i=1}^n w_i f_i \dots\dots\dots (15.1)$$

where n is the number of data sets values at the nodes, cell centers, or cell corners, f_i are the data set values, and w_i are the weights assigned to each of the values. The weights are computed as:

$$w_i = \frac{h_i^{-2}}{\sum_{j=1}^n h_j^{-2}} \dots\dots\dots(15.2)$$

where h_i is the distance from the cell or node to the observation point or:

$$h_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} \dots\dots\dots(15.3)$$

This method tends to give greater weight to the values which are closest to the observation point.

Use Values From Nearest Node/Cell

If the *Use value(s) from nearest node/cell* option is selected, the element node, grid cell, or grid corner closest to the observation point is found and that value is assigned to the observation point as the computed value.

15.2.2 Measurement Type List

If field observations are to be associated with the points in an observation coverage, a list of measurement types must be created for the coverage. This list is used to associate a name with each measurement type and to classify the measurement type as steady state or transient.

15.2.3 New/Delete/Name

A new measurement type is created by selecting the *New* button. An existing measurement type is removed from the list by selecting the measurement type in the list and selecting the *Delete* button. The name of a measurement type can be edited by selecting the measurement type and editing the name in the *Name* field.

15.2.4 Steady State vs. Transient

If a single value is measured at each observation point, the *Steady state* option should be selected for the measurement type. If a series of values are measured over a period of time, the *Transient* option should be chosen. The option chosen here controls how the observed values are entered at each observation point at a later point in time.

15.3 Observation Points

Once an observation coverage has been created and the general options for the coverage have been established as described in the previous section, the next step is to define a series of observation points. Observation points can be used for two purposes:

- (1) They may be independent locations without physically corresponding points used simply to interpolate and plot a computed value (steady state) or time series (transient) at the point.
- (2) They can be associated with a physical location at which observed values have been gathered and are then used to plot calibration error representing the difference between the computed and the observed values.

15.3.1 Creating Observation Points

Observation points are created using the *Create Point* tool in the *Map* module. The xyz coordinates of the new point can be edited using the *Edit Window*. The point can also be repositioned simply by dragging the point. If the active coverage is an observation coverage, all points created in the coverage are observation points by default. Once a point is created, the attributes associated with the point can be edited by double clicking on the points with the *Select Point/Node* tool or by selecting the point and selecting the *Attributes* command in the *Feature Objects* menu. This brings up the *Observation Point Attributes* dialog shown in Figure 15.3.

The screenshot shows the "Observation Point Attributes" dialog box. The title bar is blue with the text "Observation Point Attributes" and a close button. The dialog is divided into several sections. At the top, there is a "Name:" label followed by a text box containing "Main Channel - Near inflow" and a "Color:" label followed by a red color swatch. Below this, there are two columns. The left column has a "Measurement type:" label, a list box containing "WSE (observed)", an "Edit..." button, and a checked checkbox labeled "observed". The right column has an "Observed Value:" label, a text box containing "51.35", an "Edit..." button, and a small empty square box. At the bottom, there are two radio buttons: "Confidence interval" (selected) and "Standard deviation". Below the "Confidence interval" radio button are two text boxes: "Interval:" with "1.00" and "Conf. (%):" with "95". Below the "Standard deviation" radio button is a text box: "Std. dev.:" with "0.51". At the very bottom are "OK" and "Cancel" buttons.

Figure 15.3 The Observation Point Attributes Dialog.

The items in the dialog are as follows:

Name

Each observation point can be assigned a user-defined name. This name can be plotted next to the point to aid in identifying the points.

Color

A color is associated with each point that is used when plotting the points.

Measurement Type

The measurement types defined in the *Observation Coverage Options* dialog are listed in the center of the *Attributes* dialog. A value is defined for each observation type by selecting the measurement type in the list, selecting the *Observed* toggle, and entering the observed value.

Edit Button

The *Edit* button in the *Measurement Type* section is used to bring up the *Observation Coverage Options* dialog (Figure 15.2). This dialog can be used to add or delete measurement types.

Observed Toggle

If the *Observed* toggle is not selected for a particular measurement type, the observation point will simply be used to interpolate and plot the computed data set values at the point or to plot a time series at the point. If this toggle is selected, an observed value can be entered and the calibration error (computed vs. observed) can be plotted at the point.

Observed Value

If the *Observed* toggle is selected, the observed value associated with the point can be defined using either an edit field or a button that brings up the *XY Series Editor*. If the measurement type is defined to be steady state, the edit field is used. If the measurement type is defined to be transient, the *XY Series Editor* is used.

Confidence Interval

With each observed value, a calibration target must be defined. The calibration target is defined using either a confidence interval or a standard deviation. If the *Confidence interval* option is selected, an interval and a confidence must be defined. For example, for head an interval of 1.5 with a 85% confidence indicates that the measurement of the observed head has an error of +/- 1.5 length units with 85% confidence. The confidence intervals must always be entered in increments of 5% up to 95%. Above 95%, the values of 96%, 98%, and 99% may also be used. If a confidence interval is

defined that does not correspond to one of these values, *SMS* will automatically round to the nearest interval.

Standard Deviation

The *Standard deviation* option can also be used to define a confidence interval. The standard deviation represents the standard deviation of the error involved in measuring the observed value.

If a calibration target is entered initially as a confidence interval and the *Standard deviation* option is subsequently selected, *SMS* automatically converts the confidence interval to a standard deviation or vice versa. This conversion is accomplished using a standard normal distribution. This is based on the 68-95-99.7 rule, meaning that 68% of the data fall within one standard deviation of the mean, 95% fall within two standard deviations of the mean, and 99.7% fall within three standard deviations of the mean. The conversion is accomplished using a z statistic as follows:

$$sd = CI/z \dots\dots\dots (15.4)$$

or

$$CI = sd * z \dots\dots\dots (15.5)$$

where sd = standard deviation, CI = confidence interval, and z = z statistic. Some of the z statistics based on common confidences are shown in Table 15.1. More details can be found in (Moore, 1995) or any standard statistics textbook.

Confidence	Z Statistic
99	2.576
98	2.326
96	2.054
95	1.960
90	1.645
...	

Table 15.1 Confidence vs. Z Statistic.

15.3.2 Importing Observation Points

As described in the previous section, observation points can be created one at a time using the *Create Point* tool and the point attributes can be entered using the Attributes dialog. However, for sites with large numbers of points, this type of entry can become tedious and time consuming. In some cases, it is more efficient to organize the observation point data in a spreadsheet, export the spreadsheet as a text file, and import the points using the *Observation Points* option in the *Import* dialog accessed through the *File* menu. If an observation coverage already exists and is the active coverage, the points are added to the coverage. Otherwise, a new one is created. Once the file is imported, the *Observation Coverage Options* dialog appears and the general options for the coverage can be established.

The text file should be formatted as shown in Figure 15.4. A sample file is shown in Figure 15.5.

```
NODATA value
"id" "name" "x" "y" "z" "mtype1" "int" "conf" "mtype2"...
id "name" x y z val1 int2 conf1 val2...
.
.
```

Figure 15.4 Observation Point Import File Format.

```
NODATA -999
"id" "name" "x" "y" "z" "head" "int" "conf"
1 "OBS Q5" 234.3 44.2 323.2 567.5 1.2 0.95
2 "OBS Q6" 833.3 842.3 320.2 555.3 1.4 0.90
3 "OBS Q8" 855.3 898.3 322.2 -999 0 0
.
.
```

Figure 15.5 Sample Observation Point Import File.

The file can be delimited using spaces, tabs, or commas. The items in the text file are as follows:

NODATA Record

The NODATA record is an **optional** record used to list a key value that is used to flag values which were not observed. For example, if a NODATA value of -999 is listed and one of the observed values in a measurement type column has a value of -999, the value is assumed to be not observed at that point. This is equivalent to turning off the *Observed* toggle in the *Observation Point Attributes* dialog.

ID Column

The id column is an **optional** column used to list the id of each point. If this column is not present, SMS will assign ids to the points beginning at one and increasing sequentially. If the column is present, it must begin with a column header of "id" (case insensitive). The double quotes on the column header are optional.

Name Column

The name column is an **optional** column used to list the name of each point. If the column is missing, SMS will assign a default name to each point. For example, point #1 would be named "point_1" by default. If the column is present, it must begin with a column header of "name" (case insensitive). The quotes on the column header are optional. The quotes on the point names are also optional, but must be used if the name contains spaces.

X, Y, and Z Columns

The x, y, and z columns are **required** columns and should be used to list the coordinates of each point. The quotes on the column header are optional.

Observed Values Column

After the z column, a column of observed values may be listed. The column header provides the name of the measurement type. The values should then be listed in the column. The NODATA value may be used to signify an unobserved value as explained above. Quotes must be used on the column header if it contains spaces but are optional otherwise.

Interval and Confidence or Standard Deviation Columns

After the observed values column, the calibration target should be specified using one or two columns. If the interval and confidence option is used, two columns should be listed. The column header for the first column must begin with the letters "Int" (case insensitive) and the column should contain the interval. The header for the second column must begin with the letters "con" and the column should contain the confidence values (0.0 - 1.0). If the standard deviation option is chosen, only one column should be listed. The column header must begin with the letters "St" (case insensitive) and the standard deviation values should be listed. In each case, the quotes on the column names are optional. The quotes should be used if the names contain spaces.

Multiple Measurement Types

If multiple measurement types exist, they should be listed in subsequent columns using the value-interval-confidence or value-std. deviation sequence described above.

15.3.3 Calibration Targets

If an observed value has been defined at an observation point, the points can be used to plot calibration error on a "calibration target". The components of a calibration target are illustrated in Figure 15.6. The center of the target corresponds to the observed value. The top of the target corresponds to the observed value plus the interval and the bottom corresponds to the observed value minus the interval. The colored bar represents the error. If the bar lies entirely within the target, the color bar is drawn in green. If the bar is outside the target, but the error is less than 200%, the bar is drawn in yellow. If the error is greater than 200%, the bar is drawn in red. The display options related to calibration targets are described in Section 15.5.

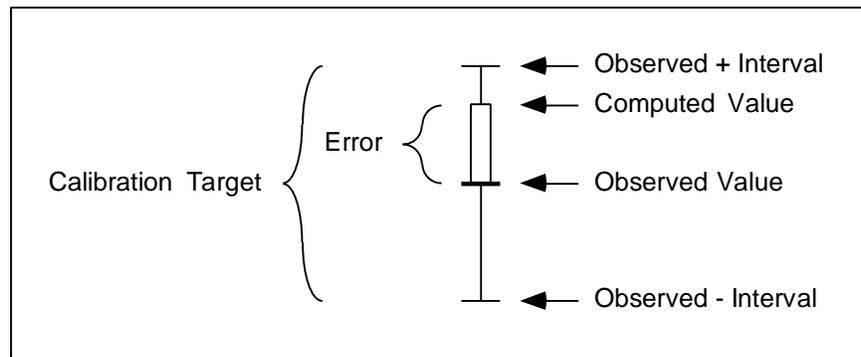


Figure 15.6 Calibration Target.

15.3.4 Point Error Statistics

While the calibration target is a useful qualitative indicator of calibration error, in some cases it is useful to see the exact number. The numerical values associated with an observation point can be viewed simply by selecting the point. The observed value, the computed values, and the calibration error of selected observation points are displayed in the *Help Window*.

15.4 Arcs

In addition to observation points, an observation coverage may also contain arcs. Arcs are used to generate profile plots illustrating the variation of a data set along a 2D mesh or a 2D grid as shown in Figure 15.7. The steps involved in generating a profile plot are described on page 15-21. The only attribute associated with an arc in an observation coverage is a color. The color can be defined by double clicking on the arc or by selecting the arc and selecting the *Attributes* command in the *Feature Object* menu.

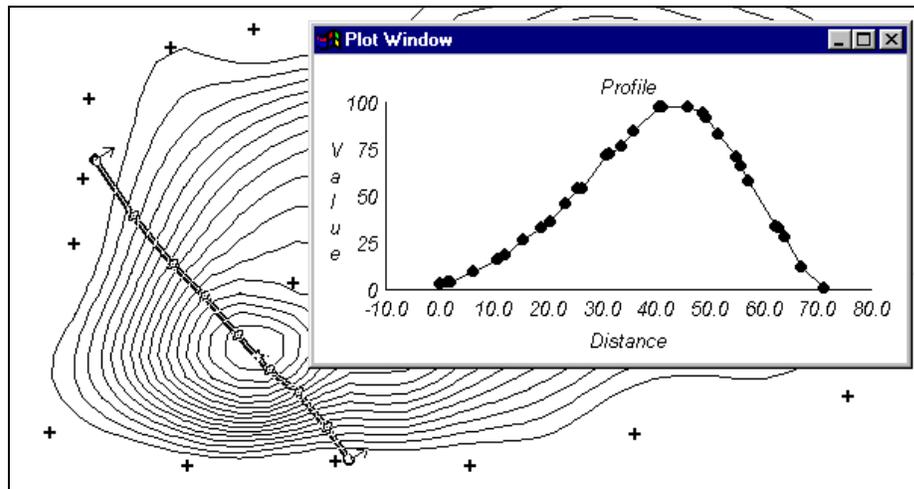


Figure 15.7 Sample Arc Profile Plot.

15.5 Display Options

A variety of display options are available for observation coverages. When the observation coverage is the active coverage, the dialog shown in Figure 15.8 is displayed when the *Display Options* command in the *Display* menu is selected.

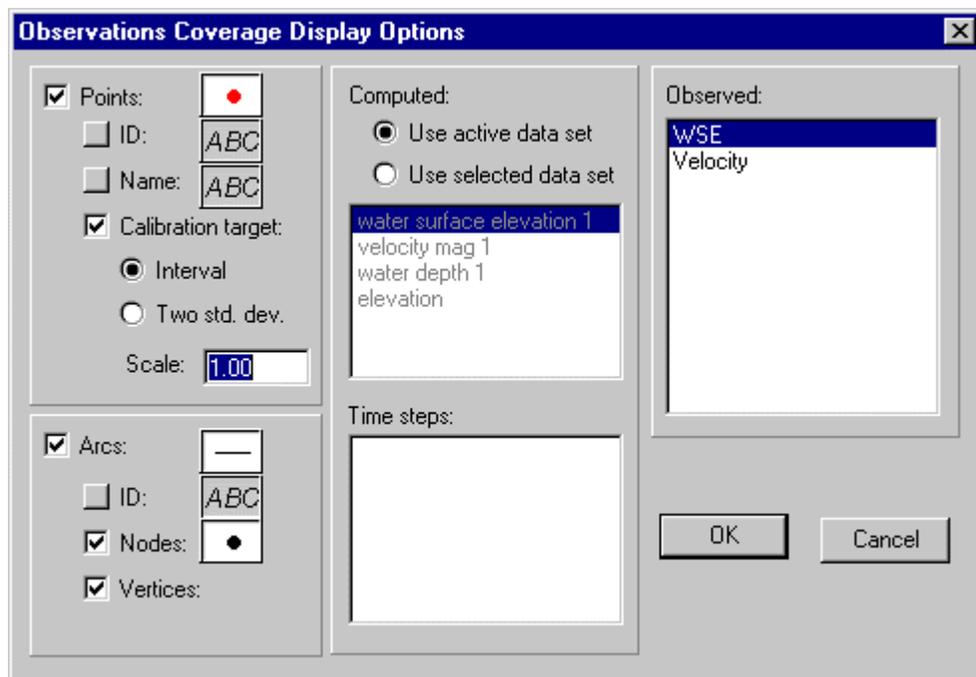


Figure 15.8 The Observation Coverage Display Options Dialog.

The options in the *Display Options* dialog are for controlling the display of observation points with calibration targets and arcs. The options for general calibration statistics plots are described in Section 15.6.

15.5.1 Points

The options for displaying points are as follows:

ID

If this option is selected, the id of each point is displayed next to the point. The graphical attributes of the text are edited using the window on the right side of the option.

Name

If this option is selected, the name of the point (defined in the *Point Attributes* dialog) is displayed next to the point. The graphical attributes of the text are edited using the window on the right side of the option.

Symbol

If this option is selected, a symbol is displayed at the location of each point. The graphical attributes of the symbol are edited using the window on the right side of the option.

Calibration Target

If this option is selected, a calibration target as depicted in Figure 15.6 is displayed next to each point. The calibration target is only displayed for points where the *Observed* toggle has been turned on in the *Attributes* dialog. Furthermore, the error bar is only superimposed on the plot when a data set (computed solution) is in memory.

Two options are available for determining the relative size of the calibration target. If the *Interval* option is chosen, the target is drawn such that the height of the target is equal to twice the confidence interval (+ interval on top, - interval on bottom). If the *Two Std. Dev.* option is chosen, the target is drawn such that the height of the target is equal to four times the confidence interval (+ 2 X std. dev. on top, - 2 X std. dev. on bottom). Conversion between the standard deviation and the confidence interval is performed automatically using standard probability distribution tables.

The overall size of the calibration targets is determined using a default value. The overall size can be increased or decreased using the *Scale* option. For example, a scale factor of 1.5 draws all targets 50% larger than the default size.

Computed

The options in the *Computed* section are used to determine which data set is used to generate the "computed" value when plotting the error on the calibration target. The default is to use the active data set. However, in some cases it is useful to explicitly select a data set from the list of current data sets.

Observed

The options in the *Observed* section are used to select which of the currently defined measurement types is used as the "observed" value when computing the error for plotting on the calibration target.

15.5.2 Arcs

The edges of the arcs are displayed if the arcs are displayed. The graphical attributes of the edges can be edited using the window on the right side of the display arcs toggle. Other options for displaying arcs are as follows:

ID

If this option is selected, the id of each arc is displayed next to the arc. The graphical attributes of the text are edited using the window on the right side of the option.

Nodes

This option is used to display the nodes (end points) of the arcs. The graphical attributes of the nodes can be edited using the window on the right side of the option.

Vertices

This option is used to display the vertices of arcs. A small dot is placed on the arcs at the location of each of the vertices. The graphical attributes of the vertices can be edited using the window on the right side of the option.

15.6 Plotting Options

In addition to the calibration targets described above, a large number of plots can be generated to illustrate various general calibration statistics (error vs. time, error vs. simulation, etc.). These plots are drawn in a special window called the *Plot Window*.

15.6.1 Open/Close Plot Window

The *Plot Window* can be opened and closed using the *Open/Close Plot Window* command in the *Display* menu. The contents of the window depend on the plotting options that are selected using the *Obs. Plot Options* command described below.

The *Plot Window* is a special type of window that acts as a "floating" window. It always displays on top of the other windows.

15.6.2 Plot Options Dialog

The plot options can be defined by selecting the *Plot Options* command from the *Display* menu. This command brings up the dialog shown in Figure 15.9.

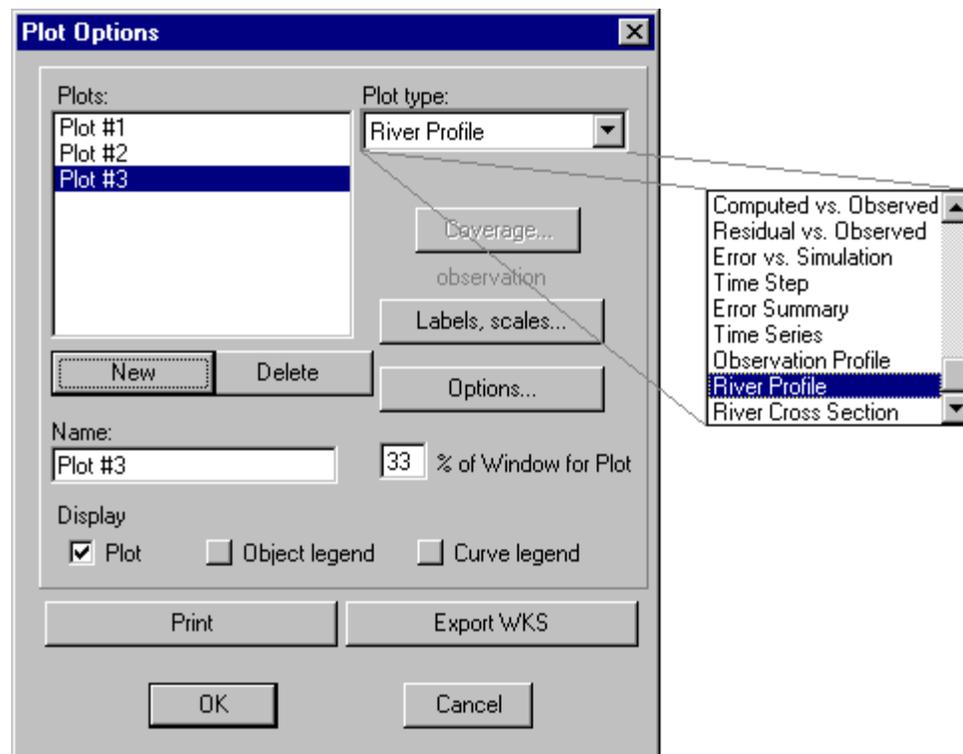


Figure 15.9 The Plot Options Dialog.

This dialog controls which plots are generated in the *Plot Window*. The options for observation plots are as follows:

Plot List

The list of currently defined plots are shown in the *Plots* section. A new plot is created by selecting the *New* button. An existing plot is deleted by selecting the plot and selecting the *Delete* button. The name of the plot can be edited by selecting the plot and using the *Name* edit field.

By default, all of the plots in the plot list are displayed in the *Plot Window*. However, selected plots can be hidden temporarily without deleting the plot using the *Display Plot* toggle.

Type of Plot

The plot type for the selected plot can be edited using the option buttons to the right of the plot list. The options specific to each plot type can be edited by selecting the *Options* button. The available observation plot types are as follows:

Computed vs. Observed

The *Computed vs. observed* option creates a plot of symbols illustrating the relationship of the computed vs. observed values for each observation point (Figure 15.10). If the computed value is equal to the observed value at a point, the symbol for the point plots precisely on the diagonal line.

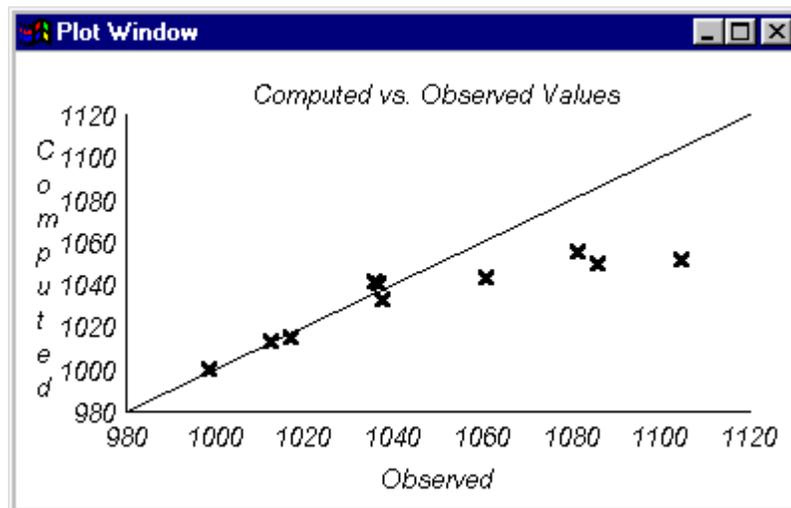


Figure 15.10 Sample Computed vs. Observed Plot.

The *Options* dialog for the *Computed vs. observed* plot is shown in Figure 15.11. The dialog is used to select which data set is to be used for the computed data and which of the measurement types is to be used for the observed data.

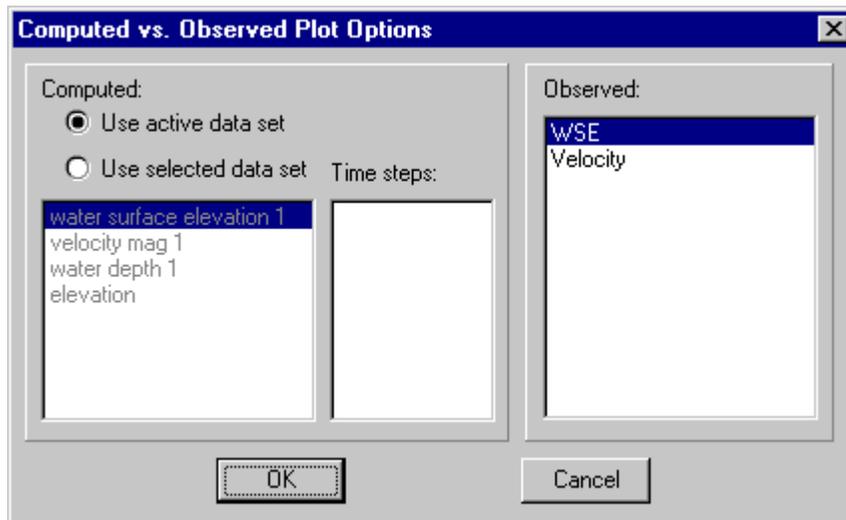


Figure 15.11 The Computed vs. Observed Plot Options Dialog.

Residual vs. Observed

The *Residual vs. observed* option creates a plot of symbols representing the error or residual (computed-observed) vs. observed values for each observation point (Figure 15.12). If the computed value is equal to the observed value at a point, the symbol for the point plots precisely on the horizontal line. The *Options* dialog for the *Residual vs. observed* plot is identical to the one shown in Figure 15.11.

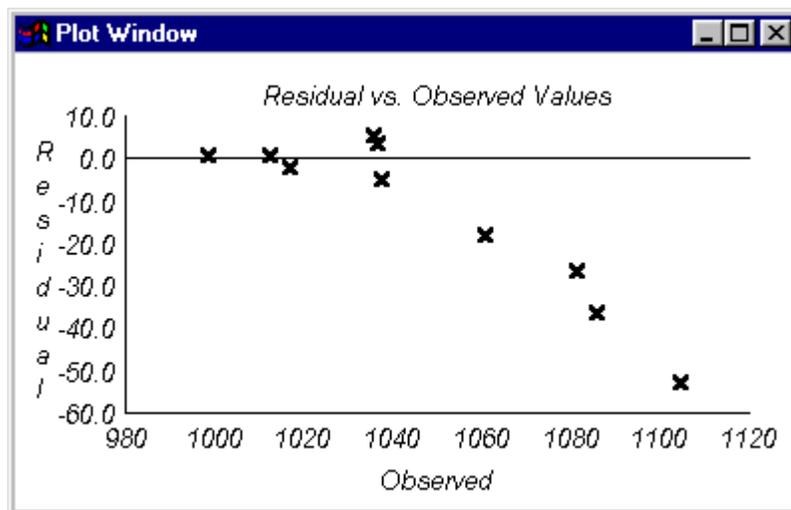


Figure 15.12 Sample Residual vs. Observed Plot.

Error vs. Simulation

In a typical calibration exercise, the parameters defining the model are iteratively changed by the user. Following each change, a new solution is computed, imported to *SMS*, and the calibration error is recomputed. This process is repeated until calibration is achieved. The *Error vs. Simulation* plot is a valuable diagnostic tool to

assist in this iterative process. The *Error vs. Simulation* plot displays the total calibration error vs. simulation (Figure 15.13). By examining this plot, it can be immediately determined if the most recent change to the model increased or decreased the total error. Plotting the trend in the error aids in determining which changes have the most positive effect on the model.

If the solutions used to generate the *Error vs. Simulation* plot are transient, the average error is used. In other words, the total error is computed for each time step, and the time step errors are averaged to compute the total error for the data set.

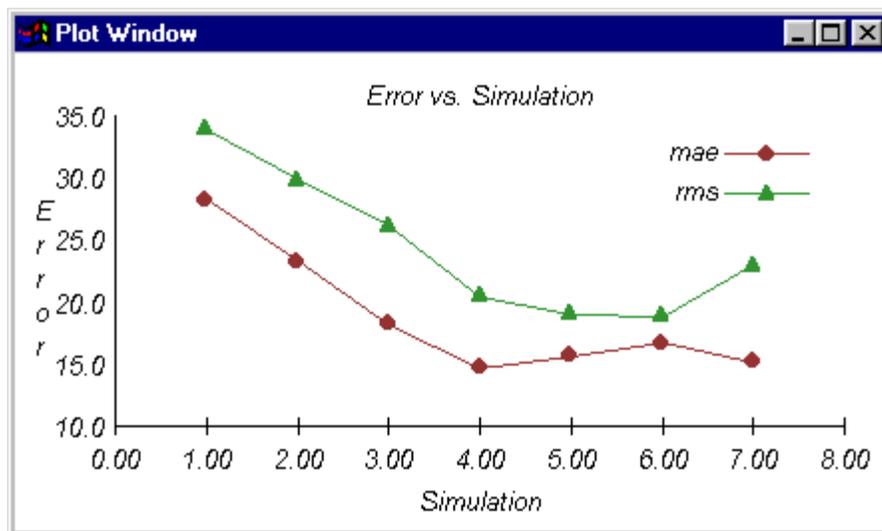


Figure 15.13 Sample Error vs. Simulation Plot.

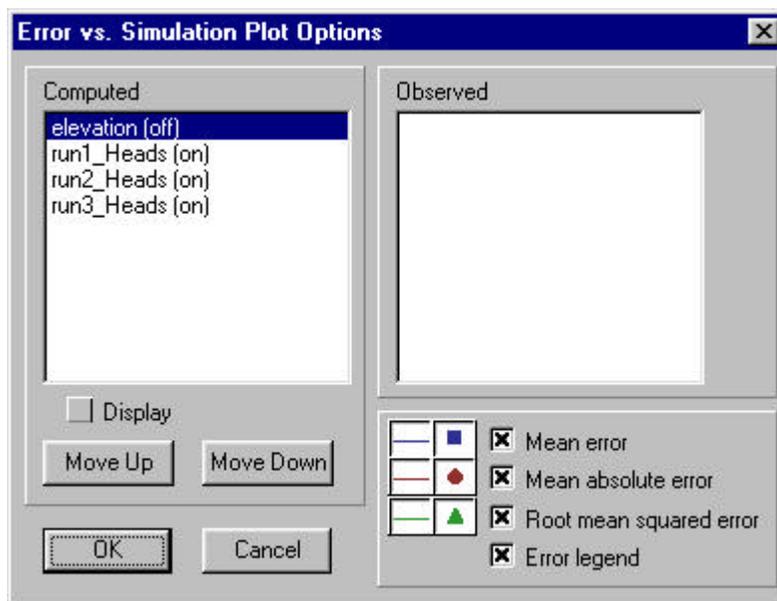


Figure 15.14 Error vs. Simulation Plot Options Dialog.

The *Options* dialog for the *Error vs. Simulation* plot is shown in Figure 15.14. The items in the *Computed* section are used to indicate which of the current data sets are plotted and the order in which they are plotted. By default, they are plotted in the order that they are generated. The ordering of the data sets can be altered by selecting data sets and selecting the *Move Up* and *Move Down* buttons. The *Observed* section is used to indicate which of the measurement types is to be used as the observed value for computing the error.

Three types of errors are available for plotting: mean error, mean absolute error, and root mean squared error. The mean error is defined as:

$$ME = \frac{1}{n} \sum_{i=1}^n (h_c - h_o)_i \dots\dots\dots(15.6)$$

where n is the number of observation points, h_c is the computed value, and h_o is the observed value. The mean absolute error is defined as:

$$MAE = \frac{1}{n} \sum_{i=1}^n |(h_c - h_o)_i| \dots\dots\dots(15.7)$$

The root mean squared error is defined as:

$$RMS = \sqrt{\frac{1}{n} \sum_{i=1}^n (h_c - h_o)_i^2} \dots\dots\dots(15.8)$$

Any combination of the three error norms can be plotted. An error legend may also be plotted.

Error vs. Time Step

For a transient simulation, it is often useful to view the change in the error vs. time as a simulation proceeds. This can be accomplished with the *Error vs. Time Step* plot option (Figure 15.15).

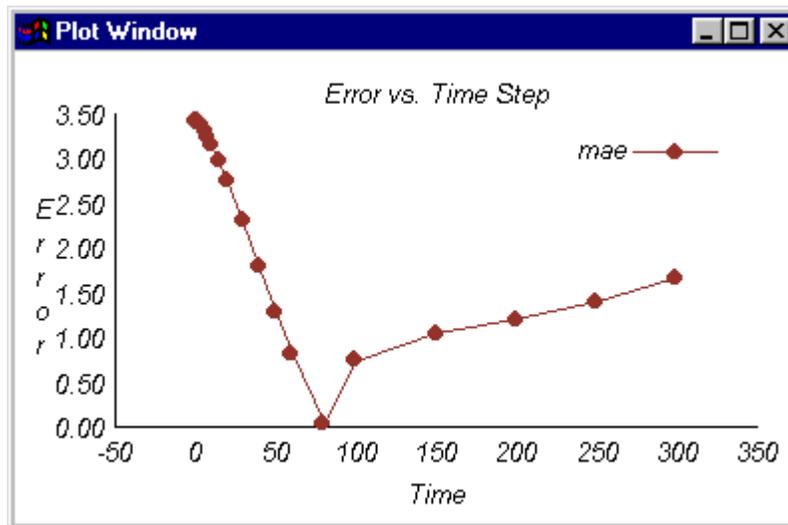


Figure 15.15 Sample Error vs. Time Step Plot.

The *Options* dialog for the *Error vs. Time Step* plot is similar to the *Error vs. Simulation* dialog shown in Figure 15.14 except that only one data set can be selected. Only the transient data sets are listed in the dialog.

Error Summary

The *Error Summary* plot option can be used to display a text listing of the total error as shown in Figure 15.16. The *Options* dialog for the *Error Summary* plot is shown in Figure 15.17. The dialog is used to select which data set is to be used for the computed data and which of the measurement types is to be used for the observed data. For the computed data, either a single time step or all time steps may be selected. If the *All time steps* option is selected, the average of the total error for each time step is plotted.

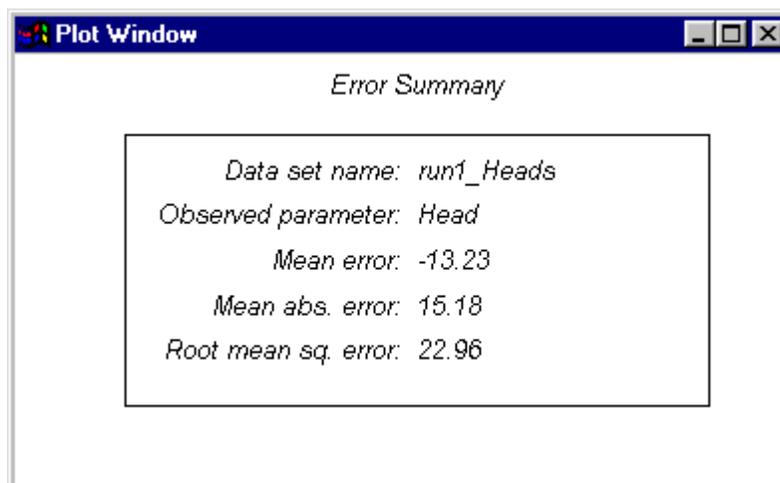


Figure 15.16 Sample Error Summary Plot.

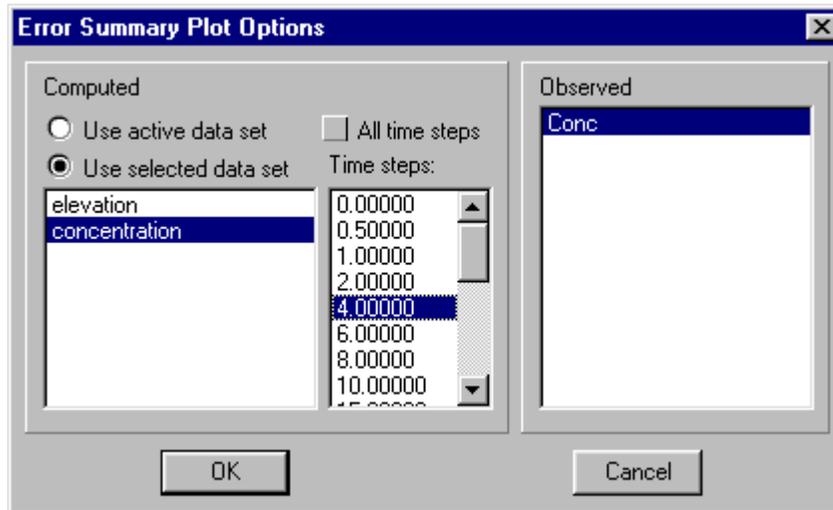


Figure 15.17 The Error Summary Plot Options Dialog.

Time Series

The *Time Series* plot option can be used to plot the change in a key value (such as head) vs. time for a transient solution (Figure 15.18). If a set of observed values has been defined for the observation point over the time domain of the solution, a curve of the observed values is plotted with a band representing the calibration target. Calibration is achieved when the computed curves lies within the band of the calibration target over the duration of the simulation. This option can also be used simply to plot the computed value vs. time without comparison to observed values.

Once a time series plot is created, the plot is empty until a point is selected. Once a point is selected, the time series for the point is displayed in the plot. Multiple time series can be displayed simply by selecting multiple points.

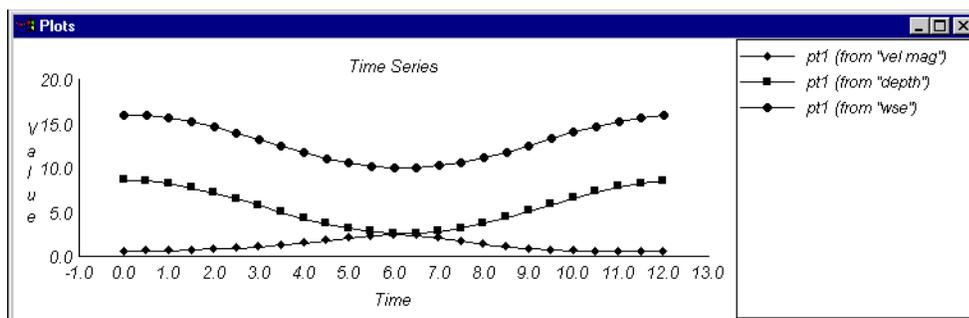


Figure 15.18 Sample Time Series Plot.

The *Options* dialog for the *Time Series* plot is shown in Figure 15.19. The options in the *Computed* section are used to select which data sets are to be plotted. More than one data set may be plotted at once by turning on the *Display* toggle for multiple data sets. The options on the right are used to select which of the measurement types is to be used for the observed data plot. In some cases, one may decide to plot none of the

observed measurement types (i.e., plot the computed values only). The calibration target can also be turned on or off.

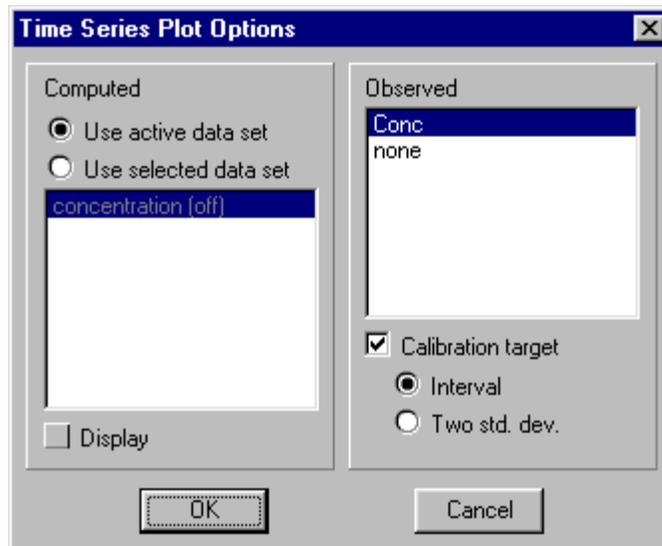


Figure 15.19 The Time Series Plot Options Dialog.

Profile

A *Profile* plot is used to display the variation of a data set associated with a 2D mesh or 2D grid along an arc. A sample profile plot is shown in Figure 15.7. Once the plot is created, the plot is empty until an arc is selected. Once an arc is selected, the variation of the selected data set along the length of the arc is displayed.

When the arc is selected, two small arrows appear at the ends of the arc indicating the default viewing direction for the profile plot. To view the profile in the opposite direction, select the arc and select the *Reverse Arc Direction* command in the *Feature Objects* menu.

XY Series Editor

The *XY Series Editor* is a special dialog that is used to generate and edit curves defined by a list of x and y coordinates. The curve can be created and edited by directly editing the xy coordinates using a spreadsheet-like list of the coordinates. The curve can also be generated and edited graphically. An entire list of curves can be generated and edited with the Editor, and curves can be imported from and exported to text files for future use.

The *XY Series Editor* is used in several places in *SMS*. It was designed to be general in nature so that it could be used anywhere that a curve or function needs to be defined. In some cases, the x values of the curve must correspond to a pre-defined set of values. For example, the x values may correspond to a set of time steps whose interval is established in a separate dialog. In such cases, the x fields cannot be edited but the y values associated with the pre-defined x values can be edited. In other cases, there is no limit on the number of x values or on the x spacing and both the x and y values can be edited.

The *XY Series Editor* is shown in Figure 16.1. Each component of the dialog is described below.

16.1 XY Series List

At the bottom of the dialog in the center, there is a list of xy series. One of the items in the list is active and highlighted at all times. The xy values of the active series are shown in the spreadsheet on the left side of the dialog and the curve is shown graphically in the upper right portion of the dialog. The name associated with the active series can be edited using the edit field to the right of the xy series list.

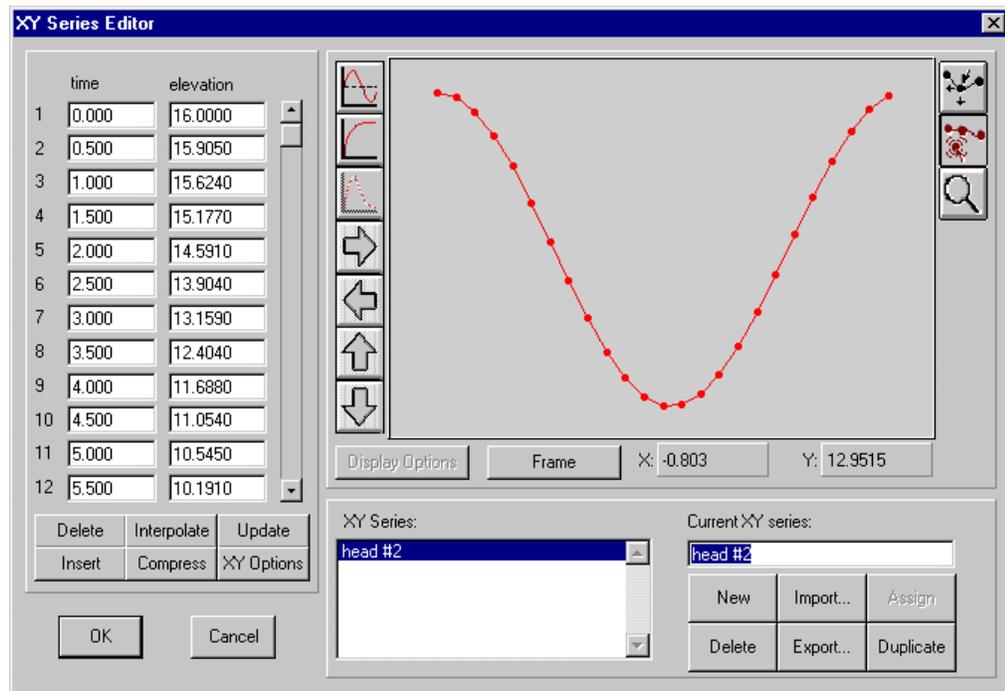


Figure 16.1 The XY Series Editor.

A new xy series can be created and added to the xy series list by selecting the *New* button. An existing series can be copied to create a new series by selecting the *Duplicate* button. This option is useful when two series need to be the same except for slight differences. An existing series can be deleted from the list by highlighting the series and selecting the *Delete* button to the right of the xy series list. A set of series can be read from a file by selecting the *Import* button. Likewise, the entire list of series can be saved to a file using the *Export* button. The file format used to save xy series is described in section 17.19.

16.2 The XY Edit Fields

The two vertical columns on the left side of the xy series dialog are for editing the values of the highlighted series. A title at the top of each column specifies what the points represent. The *TAB* key can be used to move the cursor through the edit fields.

If the number of points in the series is greater than the number of pairs of fields in the columns, the scroll bar to the right of the columns can be used to scroll through the entire range of the xy series.

The buttons below the xy edit fields are used to manipulate the values in the edit fields. A description of each button follows.

16.2.1 Delete

The *Delete* button deletes the contents of the edit field that the cursor is located in. If the x field is static, only the y field is allowed to be deleted. Otherwise, both fields are deleted.

16.2.2 Interpolate

The *Interpolate* button causes any blank fields in the xy series to be filled in by linear interpolation between the closest non-blank fields above and below the blank fields.

16.2.3 Update

The *Update* button redraws the xy series curve in the plot window using the current values in the edit fields.

16.2.4 Insert

The *Insert* button adds a new point to the xy series list by adding a pair of blank fields just above the field containing the cursor.

16.2.5 Compress

The *Compress* button reduces the length of the xy series by removing all points whose edit fields are blank.

16.2.6 XY Options

The *XY Options* button brings up the *XY Series Options* dialog shown in Figure 16.2. The top group of controls, labeled "time" in the figure is provided for manipulation of the x series. The lower group labeled "elevation" is provided for the y series. The titles such as "time" and "elevation" change based on what is being edited. The groups can be used to generate or replace the values in the x series, the y series, or both. If the check box just below the x title ("time") is selected, a beginning value, and increment, and a percent change can be input for the x range. These values are applied when the *OK* button is selected. All of the x values are replaced by a new series generated with the specified parameters. Likewise, if the box beneath the y title ("elevation") is selected, the values in the y series can be generated or redefined. The number of new values generated is specified at the bottom of the dialog. If the check boxes by the titles are not selected, the xy values are unaltered when the *OK* button is selected to exit the dialog.

The *XY Series Options* dialog can also be used to define whether the x or y series values should be interpreted as either absolute or relative (delta). If the delta option is chosen, the values beyond the initial value are interpreted as offsets from the previous value. For example, the x series (0.0, 1.0, 1.0, 1.5, 1.5) would actually represent the values (0.0, 1.0, 2.0, 3.5, 5.0).

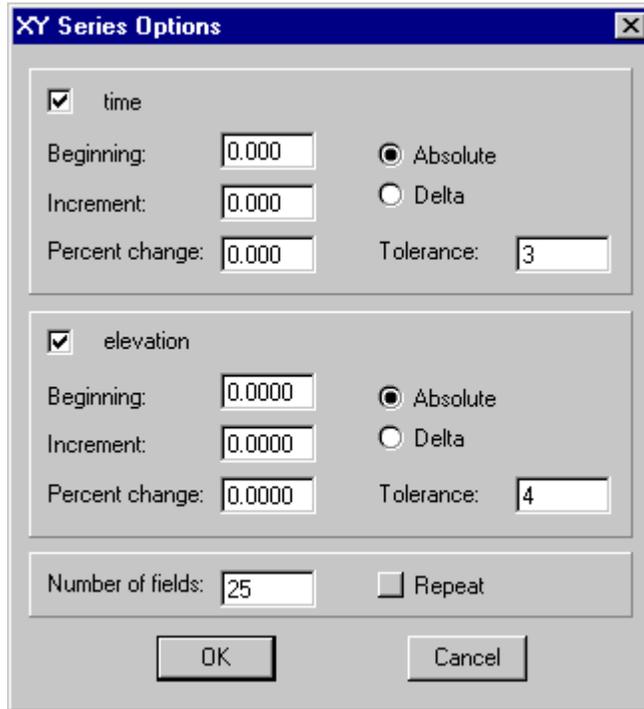


Figure 16.2 *XY Series Options Dialog.*

The *Tolerance* field controls how many decimal places are used to display the numbers in the edit fields of the *XY Series Editor*.

The *repeat* toggle is used to define a cyclic series. If the *repeat* toggle is selected, the series is assumed to repeat indefinitely. This information is saved to a file when the series is exported.

16.3 The XY Series Plot

The window in the upper right hand corner of the *XY Series Editor* is used to plot the series corresponding to the xy values in the edit fields. As each value in the edit fields is edited, the corresponding point on the series is adjusted instantaneously. The plot provides an immediate visual feedback to the user, which is helpful in detecting erroneous input values.

16.3.1 The Plot Tools

The *Plot Window* can also be used to edit the xy series graphically. The following tools (found on the upper right side of the plot) are used for graphical editing:



The Select Point Tool

The *Select Point* tool is used to graphically change the xy values of a point by clicking then dragging it to a new location. The tool can also be used to select points for deletion. A set of points can be selected by clicking on the points with the *SHIFT* key depressed or by dragging a box around a set of points. The selected points can then be deleted by selecting the Delete button beneath the xy edit fields.



The Create Point Tool

The *Create Point* tool is used to add new points to a series by clicking in the plot window at the location of the new point.



The Zoom Tool

The *Zoom* tool is used to zoom in on a region of the current series. Clicking on a point zooms the view by a factor of two around the point. Dragging a rectangle alters the mapping so that the region in the rectangle fills the plot window. Holding the *SHIFT* key down while clicking in the plot window causes the view to be enlarged by a factor of two around the point clicked.

16.3.2 The Plot Macros

The *Arrow* buttons to the lower left of the plot window in the *XY Series Editor* are used to scroll the view in the plot window up, down, left, or right. After altering the view using either the *Arrow* buttons or the *Zoom* tool, the series can be centered in the plot window by selecting the *Frame* button beneath the plot window.

The buttons to the upper left of the plot window are used to quickly create a series using analytic functions. Each button brings up a dialog that allows the parameters of a function to be specified from which a series is created. Figure 16.3 is an example of how to model a tidal curve using a sine curve.

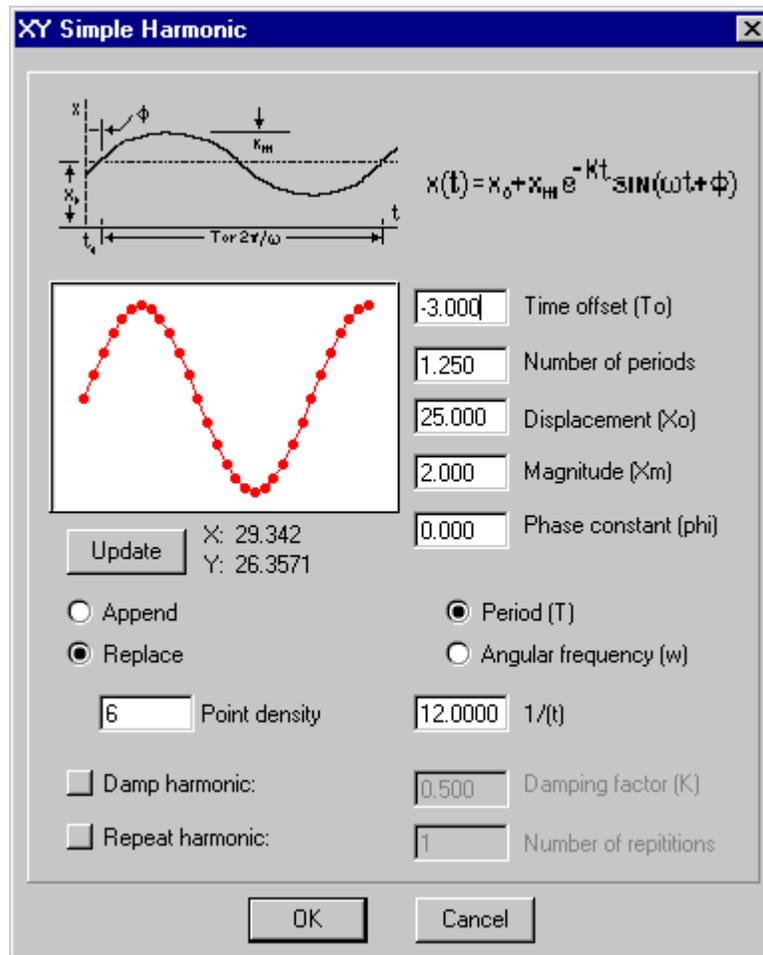


Figure 16.3 XY Simple Harmonic Motion Dialog.

For this example, the high tide is 27 feet and the low tide is 23 feet. The time between high tide and low tide is 6 hours so the period (T) is 12 hours. The displacement (X_o) is the average tide. The magnitude (X_m) is the difference of the high tide and the average tide. The time offset (T_o) is -3.0, which will put high tide at t=0. With the number of periods at 1.25, the curve will display from t=-3 hours to t=12. The point density is set at 6 so that there is a point every half-hour.

File Formats

This chapter contains the file formats for generic files used by *SMS*. A generic file is defined as any file that was not formatted for a specific numeric model. Model specific files such as those used by *FESWMS* and *RMA2* are documented in their respective reference documentation.

Most of the generic files used by *SMS* use a modified form of the *HEC* style card type format. With this format, the different components of the file are grouped into logical groups called "cards." The first component of each card is a short name that serves as the card identifier. The remaining fields on the line contain the information associated with the card. In some cases, such as lists, a card can use multiple lines.

While card style input makes the file slightly more verbose, there are many advantages associated with the card type approach to formatting files. Some of the advantages are:

1. Card identifiers make the file easier to read. Each input line has a label, which helps to identify the data on the line.
2. The card names are useful as text strings for searching in a large file. All input lines of a particular type can be located quickly in a large input file.
3. In many cases, Cards allow the data to be input in any order (i.e., the order that the cards appear in the file is usually not important).
4. Cards make it easy to modify a file format. New card types can be added without invalidating older files. New files have additional data in the new cards. The new card must be optional (which is typically the case for new cards) for old files to remain compatible. If an old card type is no longer used, the card can simply be ignored without causing input errors.

17.1 2D-Mesh Files

Two-dimensional finite element meshes can be stored in 2D-mesh files. This file is an option for storing the geometric data representing the mesh. The *SMS* element types are discussed in Chapter 4. The file format for 2D-mesh files is shown in Figure 17.1.

```
MESH2D /* File type identifier */
E3T id n1 n2 n3 mat /* 3 node triangle */
E6T id n1 n2 n3 n4 n5 n6 mat /* 6 node triangle */
E4Q id n1 n2 n3 n4 mat /* 4 node quad */
E8Q id n1 n2 n3 n4 n5 n6 n7 n8 mat /* Three node triangle */
ND id x y z /* Nodal coordinates */
```

Figure 17.1 2D-Mesh File Format.

The card types used in the 2D-mesh file are as follows:

<i>Card Type</i>	MESH2D
<i>Description</i>	File type identifier. Must be on first line of file. No fields.
<i>Required</i>	YES

<i>Card Type</i>	E3T		
<i>Description</i>	Defines a three node (linear) triangular element.		
<i>Required</i>	NO		
<i>Format</i>	E3T id n1 n2 n3 mat		
<i>Sample</i>	E3T 283 13 32 27 4		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID of the element.
2-4	n1-n3	+	The nodal indices of the element ordered counterclockwise.
5	mat	+	The material ID for the element.

<i>Card Type</i>	E6T		
<i>Description</i>	Defines a six node (quadratic) triangular element.		
<i>Required</i>	NO		
<i>Format</i>	E6T id n1 n2 n3 n4 n5 n6 mat		
<i>Sample</i>	E6T 283 13 32 27 22 25 30 4		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID of the element.
2-7	n1-n6	+	The nodal indices of the element ordered counterclockwise starting at a corner node.
8	mat	+	The material ID for the element.

<i>Card Type</i>	E4Q		
<i>Description</i>	Defines a four node (linear) quadrilateral element.		
<i>Required</i>	NO		
<i>Format</i>	E4Q id n1 n2 n3 n4 mat		
<i>Sample</i>	E4Q 283 13 32 27 30 4		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID of the element.
2-5	n1-n4	+	The nodal indices of the element ordered counterclockwise.
6	mat	+	The material ID for the element.

<i>Card Type</i>	E8Q		
<i>Description</i>	Defines an eight node (quadratic) quadrilateral element.		
<i>Required</i>	NO		
<i>Format</i>	E8T id n1 n2 n3 n4 n5 n6 n7 n8 mat		
<i>Sample</i>	E8T 283 13 32 27 22 25 30 29 31 4		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID of the element.
2-9	n1-n8	+	The nodal indices of the element ordered counterclockwise starting at a corner node.
10	mat	+	The material ID for the element.

<i>Card Type</i>	ND		
<i>Description</i>	Defines the coordinates of a node.		
<i>Required</i>	NO		
<i>Format</i>	ND id x y z		
<i>Sample</i>	ND 84 120.4 380.3 5632.0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID of the node.
2-4	x, y, z	±	The nodal coordinates.

17.2 2D Scatter Point Files

Two-dimensional scatter point sets are stored in 2D scatter point files. *SMS* supports two formats of scatter point files. The first was originally part of version 4.x of *SMS*. This format allows all the scatter points and their functions to be included in a single file. The second format includes only the scatter point locations and requires that functional information be defined in a separate data set file. However, multiple scatter point sets can be stored in a single file. In either case, an XY coordinate pair defines each point in a scatter point set.

The format of a older type of 2D scatter point file is shown in Figure 17.2. In this format, there are two types of scatter point cards files: XY and XYD. With the XY card, only the XY coordinates are defined. It is assumed that any data sets that will be associated with the scatter point set will be imported from a data set file after the scatter point file is imported.

With the XYD card, one or more data values may be associated with each scatter point in the file. These values are converted to data sets on input. This approach makes it easy to import a scatter point set with scalar values for each point.

```

SCAT2D /* File type identifier */
DELEV elev1 /* Default elevation */
XY np "name" /* # pts and name of scatter point set */
x1 Y1 /* Point coordinatess, one per line */
x2 Y2
.
.
xnp Ynp
DELEV elev2 /* Default elevation */
XYD np "name" nds nameds1 nameds2 . . . namedsnds
/* # pts, name of scatter point set,
# data sets, name of data sets */
x1 Y1 d1,1 d1,2 . . . d1,nds /* Point coordinatess and data values */
x2 Y2 d2,1 d2,2 . . . d2,nds
.
.
xnp Ynp dnp,1 dnp,2 . . . dnp,nds

```

Figure 17.2 SMS V4.x 2D Scatter Point File Format.

The cards used in this format include:

Card Type	SCAT2D
Description	File type identifier. Must be on first line of file. No fields.
Required	YES

<i>Card Type</i>	DELEV		
<i>Description</i>	Defines the default elevation for the scatter point set.		
<i>Required</i>	NO		
<i>Format</i>	DELEV e1		
<i>Sample</i>	DELEV 9.0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	e1	±	The default elevation for the following scatter points. Remains as default until new DELEV card is encountered.

<i>Card Type</i>	XY		
<i>Description</i>	Defines a scatter point set.		
<i>Required</i>	NO		
<i>Format</i>	XY np name x ₁ y ₁ x ₂ y ₂ . . x _{np} y _{np}		
<i>Sample</i>	XY 4 sample 12.3 34.5 52.2 23.5 63.2 27.4 91.1 29.3		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	np	+	The number of scatter points in the scatter point set.
2	name	str	The name of the scatter point set.
3-4	x,y	±	The coordinates of the points.
Repeat fields 3-4 np times			

<i>Card Type</i>	XYD		
<i>Description</i>	Defines a scatter point set and data sets.		
<i>Required</i>	NO		
<i>Format</i>	XYD np name nds nameds1 nameds2 . . . namedsnds x ₁ y ₁ d _{1,1} d _{1,2} . . . d _{1,nds} x ₂ y ₂ d _{2,1} d _{2,2} . . . d _{2,nds} . . x _{np} y _{np} d _{np,1} d _{np,2} . . . d _{np,nds}		
<i>Sample</i>	XYD 2 sample 1 elevation 12.3 34.5 10.2 91.1 29.3 13.7		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	np	+	The number of scatter points in the scatter point set.
2	name	str	The name of the scatter point set.
3	nds	+	The number of data sets in the scatter point set
4-(nds+3)	nameds	str	The names of the data sets
nds+4 - nds+5	x,y	±	The coordinates of the points.
nds+6 - (2*nds)+5	d	±	The data value at the points.
Repeat last (nds+2) fields np times			

This older format of scatter point file is supported for input, but is not exported. The newer format was adopted to allow time variant data sets to be associated with scattered data points as well as to organize data sets by allowing the user to assign an ID to the scattered data set. The most common usage of the older format would be to import survey data, which will be used as a source of bathymetry. A list of three dimensional survey points can easily be converted to scatter point format by adding the “SCAT2D” identifier and the “XYD” record.

The newer 2D scatter point file format is shown in Figure 17.3.

```

SCAT2D /* File type identifier */
BEGSET /* Beginning of cards for scatter point set */
NAME "name" /* Name of scatter point set */
ID id /* ID of scatter point set */
DELEV elevl /* Default elevation */
IXY np /* Number of points in set, begin point listing */
id1 x1 y1 /* Point id and coordinates, one per line */
id2 x2 y2
.
.
idnp xnp ynp
ENDSET /* End of cards for scatter point set */
/* Repeat point set cards as many times as necessary */

```

Figure 17.3 Version 5.x 2D Scatter Point File Format.

The “SCAT2D” and “DELEV” cards remain unchanged from the older format. New cards are included to further organize the data. A sample file is shown in Figure 17.4

```

SCAT2D
BEGSET
NAME "lakes"
ID 8493

```

```

DELEV 0.0000000000e+00
IXY 25
1 1.4700000000e+02 3.9000000000e+02
2 8.8200000000e+02 9.4900000000e+02
.
.
24 1.7300000000e+02 7.0100000000e+02
25 5.3900000000e+02 8.9800000000e+02
ENDSET
    
```

Figure 17.4 Sample 2D Scatter Point File.

The additional cards used in the 2D scatter point file are as follows:

<i>Card Type</i>	BEGSET
<i>Description</i>	Identifies the beginning of a scatter point set. No fields.
<i>Required</i>	NO

<i>Card Type</i>	NAME		
<i>Description</i>	Defines the name for the following scatter point set.		
<i>Required</i>	NO		
<i>Format</i>	NAME "name"		
<i>Sample</i>	NAME "st mary"		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	name	str	The name for the following scatter points. Remains as default until new NAME card is encountered.

<i>Card Type</i>	ID		
<i>Description</i>	Defines the ID for the scatter point set.		
<i>Required</i>	YES		
<i>Format</i>	ID id		
<i>Sample</i>	ID 43098		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID for the following scatter point set.

<i>Card Type</i>	IXY		
<i>Description</i>	Defines a scatter point set.		
<i>Required</i>	YES		
<i>Format</i>	IXY np id ₁ x ₁ y ₁ id ₂ x ₂ y ₂ . . id _{np} x _{np} y _{np}		
<i>Sample</i>	IXY 4 1 12.3 34.5 2 52.2 23.5 3 63.2 27.4 4 91.1 29.3		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	np	+	The number of scatter points in the scatter point set.
2	id	+	The ids of the points.
3-4	x,y	±	The coordinates of the points.
Repeat fields 2-4 np times			

<i>Card Type</i>	ENDSET		
<i>Description</i>	Identifies the end of a scatter point set. No fields.		
<i>Required</i>	NO		

17.3 ASCII Data Set Files

Data sets can be stored to either ASCII or binary files. Multiple data sets can be stored in a single file and both scalar and vector data sets can be saved to the same file. The ASCII data set format is shown in Figure 17.5. A sample data set file is shown in Figure 17.6.

```

DATASET          /* File type identifier */
OBJTYPE type     /* Type of object data set is associated with */
BEGSCL           /* Beginning of scalar data set */
OBJID id         /* Object id */
ND numdata       /* Number of data values */
NC numcells      /* Number of cells or elements */
NAME "name"      /* Data set name */
TS istat time    /* Time step of the following data. */
stat1           /* Status flags */
stat2
.
.
statnumcells
val1            /* Scalar data values */
val2
.
.
valnumdata
/* Repeat TS card for each time step */
ENDDS          /* End of data set */
BEGVEC         /* Beginning of vector dataset */
VECTYPE type   /* Vector at node/gridnode or element/cell */
OBJID id       /* Object id */
ND numdata     /* Number of data values */
NC numcells    /* Number of cells or elements */
NAME "name"    /* Data set name */
TS istat time  /* Time step of the following data. */
stat1         /* Status flags */
stat2
.
.
statnumcells
vx1 vy1
vx2 vy2
.
.
Vnumdata Vnumdata Vnumdata
/* Repeat TS card for each time step */
ENDDS          /* End of data set */
/* Repeat BEGSCL and BEGVEC sequences for each data set */

```

Figure 17.5 ASCII Data Set File Format.

For scalar data set files, one value is listed per vertex, cell, node, or scatter point. For vector data set files, one set of XY vector components is listed per vertex, cell, node, or scatter point. If necessary, a set of status flags can be included in the file. If the status flag is false (0), the corresponding item (node, cell, etc.) is inactive. If status flags are not included in the file, it is assumed that all items are active.

```

DATASET
OBJTYPE grid2d
BEGSCL
OBJID 27211
ND 8
NC 8
NAME "sediment transport"
TS 1 1.00000000e+00
0
0
0
1
1
1
1
1
0
0.00000000e+00
0.00000000e+00
0.00000000e+00
3.24000000e+00
4.39000000e+00
2.96000000e+00
7.48000000e+00
0.00000000e+00
ENDDS
BEGVEC
VECTYPE 0
OBJID 27211
ND 8
NC 8
NAME "velocity"
TS 1 5.00000000e+00
0
0
0
1
1
1
1
1
0
1.60000000e+01 1.60000000e+01
6.40000000e+01 6.40000000e+01
1.44000000e+02 1.44000000e+02
1.96000000e+02 1.96000000e+02
2.25000000e+02 2.25000000e+02
9.21600000e+03 9.21600000e+03
9.60400000e+03 9.60400000e+03
9.80100000e+03 9.80100000e+03
ENDDS

```

Figure 17.6 Sample ASCII Data Set File.

The card types used in the scalar data set file format are as follows:

Card Type	DATASET
Description	File type identifier. Must be on first line of file. No fields.
Required	YES

<i>Card Type</i>	OBJTYPE		
<i>Description</i>	Identifies the type of objects that the data sets in the file are associated with.		
<i>Required</i>	YES. If card does not exist, the file can only be read through the Data Browser. The data sets would then be assigned to the objects corresponding to the active module.		
<i>Format</i>	OBJTYPE type		
<i>Sample</i>	OBJTYPE tin		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	type	tin	TINs
		mesh2d	2D meshes
		scat2d	2D scatter points

<i>Card Type</i>	BEGSCL
<i>Description</i>	Scalar data set file identifier. Marks beginning of scalar data set. No fields.
<i>Required</i>	YES

<i>Card Type</i>	BEGVEC
<i>Description</i>	Vector data set file identifier. Marks beginning of vector data set. No fields.
<i>Required</i>	YES

<i>Card Type</i>	VECTYPE			
<i>Card ID</i>	150			
<i>Description</i>	Identifies the type of vector data that will be read and where to apply it.			
<i>Required</i>	This card is only required if the vector data is associated with elements/cells. If this card is not present, it is assumed that the data are associated with nodes/gridnodes.			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	type	4 byte int	0	The vectors will be applied to the nodes/gridnodes.
			1	The vectors will be applied to the elements/cells.

<i>Card Type</i>	ND		
<i>Description</i>	The number of data values that will be listed per time step. This number should correspond to the total number of vertices, nodes, cells centers (cell-centered grid), cell corners (mesh-centered grid), maximum node id (meshes) or scatter points.		
<i>Required</i>	YES		
<i>Format</i>	ND numdata		
<i>Sample</i>	ND 10098		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	numdata	+	The number of items. At each time step, numdata values are printed.

<i>Card Type</i>	NC		
<i>Description</i>	This number should correspond to the maximum element id (meshes) or the number of cells (grids).		
<i>Required</i>	YES		
<i>Format</i>	NC numcells		
<i>Sample</i>	NC 3982		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	numcells	+	The number of elements or cells.

<i>Card Type</i>	NAME		
<i>Description</i>	The name of the data set.		
<i>Required</i>	YES		
<i>Format</i>	NAME "name"		
<i>Sample</i>	NAME "Total head"		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	"name"	str	The name of the dataset in double quotes.

<i>Card Type</i>	TS		
<i>Description</i>	Marks the beginning of a new time step, indicates if stat flags are given, and defines the time step value, status flags, and scalar data values for each item.		
<i>Required</i>	YES		
<i>Format</i>	<pre> TS istat time stat1 stat2 . . stat numcells val1 val2 . . valnumdata </pre>		
<i>Sample</i>	<pre> TS 1 12.5 0 1 1 1 34.5 74.3 58.4 72.9 </pre>		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	istat	0 1	Use status flags from previous time step. For first time step, this indicates that all cells are active. Status flags will be listed.
2	time	+	The time step value. If only one time step exists, time is not required
2 - (n+1)	stat	0,1	The status of each item. If active, stat=1. If inactive stat=0. Omitted if i=0 on STAT card.
(n+2) - (2n+1)	val	±	The scalar data values of each item.

17.4 ASCII Scalar Data Set Files (version 4)

SMS is backward compatible, supporting data set files used in version 4.x. Version 4 ASCII data set formats support a single data set which may be steady state or transient. The format is shown in Figure 17.7. One scalar value is listed per vertex, cell, node, or scatter point. If necessary, a set of status flags can be included in the file. If the status flag is false (0), the corresponding item (node, cell, etc.) is inactive. If status flags are not included in the file, it is assumed that all items are active.

```

SCALAR /* File type identifier */
ND n /* Number of data values */
STAT i /* Status flags will be included in file */
TS time /* Time step of the following data. */
stat1 /* Status flags */
stat2
.
.
statn
val1 /* Scalar data values */
val2
.
.
valn
/* Repeat TS card for each time step */

```

Figure 17.7 ASCII Scalar Data Set File Format.

The card types used in the scalar data set file format are as follows:

Card Type	SCALAR
Description	File type identifier. Must be on first line of file. No fields.
Required	YES

Card Type	ND		
Description	Defines the number of data values per time step. This number should correspond to the total number of vertices, nodes, cells, or scatter points in the object the data set is imported to.		
Required	YES		
Format	ND n		
Sample	ND 4		
Field	Variable	Value	Description
1	n	+	The number of data values per time step.

<i>Card Type</i>	STAT		
<i>Description</i>	Specified whether or not status flags will be included in the file. If not, all items are assumed to be active.		
<i>Required</i>	YES		
<i>Format</i>	STAT i		
<i>Sample</i>	STAT 0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	i	0,1	If status flags are to be included in the file, i=1. If status flags are not to be included in file, i=0.

<i>Card Type</i>	TS		
<i>Description</i>	Defines a set of scalar values associated with a time step.		
<i>Required</i>	YES		
<i>Format</i>	TS time stat ₁ stat ₂ . stat _n val ₁ val ₂ . val _n		
<i>Sample</i>	TS 12.5 0 1 1 1 34.5 74.3 48.3 72.9		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	time	±	The time step value. Ignored if there is only one time step.
2 - (n+1)	stat	0,1	The item status. If active, stat=1. If inactive stat=0. Omitted if i=0 on STAT card.
(n+2) - (2n+1)	val	±	The scalar data values of each item.

17.5 ASCII Vector Data Set Files (version 4)

SMS is backward compatible, supporting data set files used in version 4.x. Version 4 ASCII data set formats support a single data set which may be steady state or transient. The format is shown in Figure 17.8. One set of xy vector components is listed per vertex, cell, node, or scatter point. If necessary, a set of status flags can be included in the file. If the status flag is false (0), the corresponding item (node, cell, etc.) is inactive. If status flags are not included in the file, it is assumed that all items are active.

```

VECTOR /* File type identifier */
ND n /* Number of data values */
STAT i /* Status flags will be included in file */
TS time /* Time step of the following data. */
stat1 /* Status flags */
stat2
.
.
statn
Vx,1 Vy,1 /* Vector data values */
Vx,2 Vy,2
.
.
Vx,n Vy,n Vz,n
/* Repeat TS card for each time step */
    
```

Figure 17.8 ASCII Vector Data Set File Format.

The card types used in the vector data set file format are as follows:

<i>Card Type</i>	VECTOR
<i>Description</i>	File type identifier. Must be on first line of file. No fields.
<i>Required</i>	YES

<i>Card Type</i>	ND		
<i>Description</i>	Defines the number of data values per time step. This number should correspond to the total number of vertices, nodes, cells, or scatter points in the object the data set is imported to.		
<i>Required</i>	YES		
<i>Format</i>	ND n		
<i>Sample</i>	ND 4		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	n	+	The number of data values per time step.

<i>Card Type</i>	STAT		
<i>Description</i>	Specified whether or not status flags will be included in the file. If not, all items are assumed to be active.		
<i>Required</i>	YES		
<i>Format</i>	STAT I		
<i>Sample</i>	STAT 0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	i	0,1	If status flags are to be included in the file, i=1. If status flags are not to be included in file, i=0.

<i>Card Type</i>	TS		
<i>Description</i>	Defines a set of vector values associated with a time step.		
<i>Required</i>	YES		
<i>Format</i>	TS time stat ₁ stat ₂ . stat _n v _{x,1} v _{y,1} v _{x,2} v _{y,2} . . v _{x,n} v _{y,n}		
<i>Sample</i>	TS 12.5 0 1 1 1 3.4 -5.6 0.0 6.2 -8.9 0.0 -2.9 6.2 0.0 -1.2 -3.4 0.0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	time	±	The time step value. Ignored if there is only one time step.
2 - (n+1)	stat	0,1	The item status. If active, stat=1. If inactive stat=0. Omitted if i=0 on STAT card.
(n+2) - (4n+1)	v _x , v _y , v _z	±	The xyz components of each vector.

17.6 Binary Data Set Files

Data sets can be stored to either ASCII or binary files. Compared to ASCII files, binary files require less memory and can be imported to *SMS* more quickly. The disadvantages of binary files are that they are not as portable and they cannot be viewed with a text editor.

The binary data set file format is shown in Figure 17.9. The binary format is patterned after the ASCII format in that the data are grouped into "cards". However, the cards are identified by a number rather than a card title.

Card	Item	Size	Description
	version	4 byte integer	The SMS binary data set file format version. value = 3000.
100	objecttype	4 byte integer	Identifies the type of objects that the data sets in the file are associated with. Options are as follows: 1 TINs 3 2D meshes 5 2D scatter points
110	SFLT	4 byte integer	The number of bytes that will be used in the remainder of the file for each floating point value (4, 8, or 16).
120	SFLG	4 byte integer	The number of bytes that will be used in the remainder of the file for status flags.
130 or 140	BEGSCL or BEGVEC		Marks the beginning of a set of cards defining a scalar or vector data set.
150	VECTYPE	4 byte integer	(0 or 1) In the case of vector data set files, indicates whether the vectors will be applied at the nodes/gridnodes or the elements/cells.
160	OBJID	4 byte integer	The id of the associated object. Value is ignored for grids and meshes.
170	NUMDATA	4 byte integer	The number of data values that will be listed per time step. This number should correspond to the number of vertices, nodes, cell centers (cell-centered grid), cell corners (mesh-centered grid) or scatter points.
180	NUMCELLS	4 byte integer	This number should correspond to the number of elements (meshes) or the number of cells (mesh-centered grids). Value is ignored for other object types.
190	NAME	40 bytes	The name of the dataset. Use one character per byte. Mark the end of the string with the '\0' character.
200	TS		Marks the beginning of a time step.
	ISTAT	SFLG integer	(0 or 1) Indicates whether or not status flags will be included in the file.
	TIME	SFLT real	Time corresponding to the time step.
	statflag1	SFLG integer	Status flag (0 or 1) for node 1
	statflag2	SFLG integer	Status flag (0 or 1) for node 2
		
	val1	SFLT real	Scalar value for item 1
	val2	SFLT real	Scalar value for item 2
		
			Repeat card 200 for each timestep in the data set.
210	ENDDS		Signal the end of a set of cards defining a data set.

Figure 17.9 The Binary Scalar or Vector Data Set File Format.

The cards in the binary data set file are as follows:

<i>Card Type</i>	VERSION
<i>Card ID</i>	3000
<i>Description</i>	File type identifier. No fields.
<i>Required</i>	YES

<i>Card Type</i>	OBJTYPE			
<i>Card ID</i>	100			
<i>Description</i>	Identifies the type of objects that the data sets in the file are associated with.			
<i>Required</i>	YES. If card does not exist, the file can only be read through the Data Browser. The data sets would then be assigned to the objects corresponding to the active module.			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	id	4 byte int	1 3 5	TINs 2D meshes 2D scatter points

<i>Card Type</i>	SFLT			
<i>Card ID</i>	110			
<i>Description</i>	Identifies the number of bytes that will be used in the remainder of the file for each floating point value (4, 8, or 16).			
<i>Required</i>	YES			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	sizefloat	4 byte int	4, 8, or 16	Number of bytes

<i>Card Type</i>	SFLG			
<i>Card ID</i>	120			
<i>Description</i>	Identifies the number of bytes that will be used in the remainder of the file for status flags (1, 2, or 4).			
<i>Required</i>	YES			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	sizeflag	4 byte int	1, 2, or 4	Number of bytes

<i>Card Type</i>	BEGSCL			
<i>Card ID</i>	130			
<i>Description</i>	Marks the beginning of a set of cards defining a scalar data set.			
<i>Required</i>	YES			

<i>Card Type</i>	BEGVEC			
<i>Card ID</i>	140			
<i>Description</i>	Marks the beginning of a set of cards defining a vector data set.			
<i>Required</i>	YES			

<i>Card Type</i>	VECTYPE			
<i>Card ID</i>	150			
<i>Description</i>	Identifies the type of vector data that will be read and where to apply it.			
<i>Required</i>	This card is only required if the vector data is associated with elements/cells. If this card is not present, it is assumed that the data are associated with nodes/gridnodes.			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	type	4 byte int	0 1	The vectors will be applied to the nodes/gridnodes. The vectors will be applied to the elements/cells.

<i>Card Type</i>	OBJID			
<i>Card ID</i>	160			
<i>Description</i>	The id of the associated object.			
<i>Required</i>	This card is required in the case of TINs, 2D scatter points, and 3D scatter points. With each of these objects, multiple objects may be defined at once. Hence the id is necessary to relate the data set to the proper object.			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	id	4 byte int	+	The id of the object.

<i>Card Type</i>	NUMDATA			
<i>Card ID</i>	170			
<i>Description</i>	The number of data values that will be listed per time step. This number should correspond to the number of vertices, nodes, cell centers (cell-centered grid), cell corners (mesh-centered grid), maximum node id (meshes) or scatter points.			
<i>Required</i>	YES			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	numdata	4 byte int	+	The number of items. At each timestep, numdata are listed.

<i>Card Type</i>	NUMCELLS			
<i>Card ID</i>	180			
<i>Description</i>	This number should correspond to the element id (meshes) or the number of cells (grids).			
<i>Required</i>	YES			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	numcells	4 byte int	+	The number of elements or cells.

<i>Card Type</i>	NAME			
<i>Card ID</i>	190			
<i>Description</i>	The name of the data set.			
<i>Required</i>	YES			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	name	40 bytes	str	The name of the data set. Use one character per byte. Mark the end of the string with the '\0' character.

<i>Card Type</i>	TS			
<i>Card ID</i>	200			
<i>Description</i>	Defines the set of scalar values associated with a timestep. Should be repeated for each time step.			
<i>Required</i>	YES			
<i>Field</i>	<i>Variable</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
1	istat	SFLG int	0 1	Use status flags from previous time step. For the first time step, this value indicates that all cells are active. Status flags will be listed.
2	time	SFLT int	+	The time step value. This number is ignored if there is only one time step.
	stat	SFLG int	0 1	Inactive Active One status flag should be listed for each cell or element. These flags are included only when istat = 1.
	val	SFLT real	±	The scalar values

<i>Card Type</i>	ENDDS	
<i>Card ID</i>	210	
<i>Description</i>	Signals the end of a set of cards defining a data set	
<i>Required</i>	YES	

17.7 Binary Scalar Data Set Files (version 4)

SMS is backward compatible, supporting data set files used in version 4.x. Version 4 binary data set formats support a single data set which may be steady state or transient. The binary data set file format is shown in Figure 17.10. One scalar value is listed per vertex, cell, node, or scatter point. If necessary, a set of status flags can be included in the file. If the status flag is false (0), the corresponding item (node, cell, etc.) is inactive. If status flags are not included in the file, it is assumed that all items are active.

Item	Size	Description
Version	4 byte integer	Version = 1000 for scalar file
N	4 byte integer	Number of items (cells, nodes, etc.)
status data	4 byte integer	(0 or 1) Indicates whether or not status flags will be included in the file.
SFLT	4 byte integer	The number of bytes that will be used in the remainder of the file for each floating point value (4, 8, or 16).
SFLG	4 byte integer	The number of bytes that will be used in the remainder of the file for each status flag (1, 2, or 4).
time step I	SFLT real	Time corresponding to time step I
Statflag1	SFLG integer	Status flag (0 or 1) for node 1
Statflag2	SFLG integer	Status flag (0 or 1) for node 2
Statflag3	SFLG integer	Status flag (0 or 1) for node 3
.		
.		
val1	SFLT real	Scalar value for item 1
val2	SFLT real	Scalar value for item 2
val3	SFLT real	Scalar value for item 3
.		
.		
Repeat time step group for each time step.		

Figure 17.10 The Binary Scalar Data Set File Format.

The following sample code illustrates how binary scalar files are written using FORTRAN:

```

OPEN(30, FILE='results.bin', FORM='UNFORMATTED', STATUS='UNKNOWN')
WRITE(30) IVERSION, NNP, ISTAT, ISFLT, ISFLG
DO 10 I=1, NTS
  WRITE(30) TIME(I)
  IF(ISTAT .NE. 0)THEN
    WRITE(30)(IFLG(I,J), J=1, NNP)
  END IF
  WRITE(30)(VALUE(I,J), J=1, NNP)
10 CONTINUE

```

Explanation of variables:

Variable	Description
IVERSION	Version = 1000 for scalar file
NNP	Number of items (cells, nodes, etc.)
ISTAT	(0 or 1) Indicates whether or not status flags will be included in the file.
ISFLT	The number of bytes that will be used in the remainder of the file for each floating point value. (4, 8 or 16)
ISFLG	The number of bytes that will be used in the remainder of the file for each status flag. (1, 2 or 4)
NTS	The number of time steps (separate sets of data corresponding to the vertex, cell, node, or scatter point group)
TIME(I)	Time corresponding to time step I
IFLG(I,J)	Status flag (0 or 1) for node J of time step I
VALUE(I,J)	Scalar value for item J of time step I

Note: Most FORTRAN compilers use the convention of writing four byte block size markers before and after each set of data written to a binary (UNFORMATTED) file. Block size markers indicate the size of the data that comes next in the file defined by individual WRITE statements. *SMS* expects that the data has been written to the binary file with the same configuration of WRITE statements shown in the previous example. *SMS* can also read a binary file without block size markers (standard C convention).

17.8 Binary Vector Data Set Files (version 4)

SMS is backward compatible, supporting data set files used in version 4.x. Version 4 binary data set formats support a single data set which may be steady state or transient. The binary data set file format is shown in Figure 17.11. One set of vector values is listed per vertex, cell, node, or scatter point. If necessary, a set of status flags can be included in the file. If the status flag is false (0), the corresponding item (node, cell, etc.) is inactive. If status flags are not included in the file, it is assumed that all items are active.

Item	Size	Description
Version	4 byte integer	Version = 2000 for vector file.
N	4 byte integer	Number of items (nodes, cells, etc.)
status data	4 byte integer	(0 or 1) Indicates whether or not status flags will be included in the file.
SFLT	4 byte integer	The number of bytes that will be used in the remainder of the file for each floating point value (4, 8, or 16).
SFLG	4 byte integer	The number of bytes that will be used in the remainder of the file for each status flag (1, 2, or 4).
time step I	SFLT real	Time corresponding to time step I
Statflag1	SFLG integer	Status flag (0 or 1) for item 1
Statflag2	SFLG integer	Status flag (0 or 1) for item 2
Statflag3	SFLG integer	Status flag (0 or 1) for item 3
.		
.		
vx1	SFLT real	X comp. of vector. for item 1
vy1	SFLT real	Y comp. of vector for item 1
.		
.		
Repeat time step group for each time step.		

Figure 17.11 The Binary Vector Data Set File Format.

The following sample code illustrates how binary vector files are written using FORTRAN:

```

OPEN(30, FILE='vector.bin', FORM='UNFORMATTED', STATUS='UNKNOWN')
WRITE(30) IVERSION, NNP, ISTAT, ISFLT, ISFLG
DO 10 I=1, NTS
  WRITE(30) TIME(I)
  IF(ISTAT .NE. 0)THEN
    WRITE(30)(IFLG(I,J), J=1, NNP)
  END IF
  WRITE(30) (X(I,J), Y(I,J), J=1, NNP)
10 CONTINUE

```

Explanation of variables:

Variable	Description
IVERSION	Version = 2000 for vector file
NNP	Number of items (cells, nodes, etc.)
ISTAT	(0 or 1) Indicates whether or not status flags will be included in the file.
ISFLT	The number of bytes that will be used in the remainder of the file for each floating point value. (4, 8 or 16)
ISFLG	The number of bytes that will be used in the remainder of the file for each status flag. (1, 2 or 4)
NTS	The number of time steps (separate sets of data corresponding to the vertex, cell, node, or scatter point group)
TIME(I)	Time corresponding to time step I
IFLG(I,J)	Status flag (0 or 1) for node J of time step I
X(I,J)	X comp. of vector. for item J of time step I
Y(I,J)	Y comp. of vector. for item J of time step I

Note: Most FORTRAN compilers use the convention of writing four byte block size markers before and after each set of data written to a binary (UNFORMATTED) file. Block size markers indicate the size of the data that comes next in the file defined by individual WRITE statements. SMS expects that the data has been written to the binary file with the same configuration of WRITE statements shown in the example. SMS will also read a binary file without block size markers (standard C convention).

17.9 Boundary ID Files

Boundary Id files are used to store boundary data. They contain information about the nodestrings and the node ids. An example of the Boundary Id file is shown in Figure 17.12

```
BOUNDARY          /* File type identifier */
2 Number of nodestrings
17 Number of nodes in nodestring
13      /* Node Ids */
26
39
52
.
.
.
.
/* EOF */
```

Figure 17.12 Sample Boundary ID File

17.10 Boundary XY Files

Boundary XY files are similar to boundary id files except they contain the node xy locations rather than the ids. An example of the Boundary XY file is shown in Figure 17.13

```
BOUNDARY          /* File type identifier */
2 Number of nodestrings
17 Number of nodes in nodestring
500000.000000 0.000000          /* Node XY values */
497592.400000 49008.600000
490392.600000 97545.200000
.
.
.
/* EOF */
```

Figure 17.13 Sample Boundary XY File

17.11 Coastline Files

Coastline files contain data that define coastlines used in the ADCIRC and CGWAVE coverages. When a Coastline file is read in SMS, feature arcs are created at the elevation indicated in the file. An example of a coastline file is shown in Figure 17.14.

```
COAST /* file type identifier */
1 /* Number of coastlines */
1309 0 /* Number of segments in coastline and Elevation */
-7794.9054 3396.0346 /* XY Points */
-7822.6129 3391.8341
-7852.6508 3386.68
.
.
.
/* EOF */
```

Figure 17.14 Sample Coastline File.

17.12 Drogue Files

Drogue files contain particle/path data. Drogue plots are generated by the ADCIRC model. An example of a Drogue file is shown in Figure 17.15

```

ACE/vis drogue path file      /* Title */
5                             /* Number of Time Steps */
7200.0000      199           /* Current Time Step and Number of Particles */
-0.766993322E+02  0.346589454E+02  1      /* xy values and id */
-0.766986001E+02  0.346616775E+02  2
.
.
.
/* EOF */

```

Figure 17.15 Sample Drogue File

17.13 Map Files

Map files are used to store feature object and drawing object data. Feature objects include points, nodes, vertices, arcs and polygons. Drawing objects include rectangles, ovals, lines, and text. The map file also includes the grid frame. The map file format is shown in Figure 17.16. Figure 17.16 does not include the cards defining feature object attributes. These cards are described below. A sample map file is shown in Figure 17.17.

```

MAP                             /* file type identifier */
BEGCOV                          /* beginning of a new coverage */
COVNAME "name"                  /* coverage name */
COVATTS attstypetype           /* coverage attributes type */
POINT                           /* new point identifier */
XY x y                          /* xy coordinates of the point*/
ID id                           /* id of the point*/
END                             /* end of data for point*/
NODE                            /* new node identifier */
XY x y                          /* xy coordinates of node*/
ID id                           /* id of node */
END                             /* end of data for node */
ARC                             /* new arc identifier */
ID id                           /* id of arc */
NODES id1 id2                  /* ids of beginning and ending nodes for arc */
ARCVERTICES n                  /* number of vertices between nodes for arc */
x1 y1                          /* xy location of vertices */
x2 y2
xn yn
ARCBIAS Value                   /* bias value for meshing */
END                             /* end of data for arc 1 */
POLYGON                        /* new polygon identifier */
ID id                           /* id of polygon*/
ARCS n                          /* number of boundary arcs for polygon */
id1                             /* ids of boundary arcs */
id2
.
idn
HARCS n /* number of hole-arcs for polygon */
id1 /* ids of hole-arcs */
id2
.
idn
END /* end of data for polygon */
ENDCOV /* end of coverage data */
RECT /* new rectangle identifier */
C1 x1 y1 z1 /* xyz coordinates of corner 1 */

```

```

C2 x2 y2 z2 /* xyz coordinates of corner 2 */
C3 x3 y3 z3 /* xyz coordinates of corner 3 */
C4 x4 y4 z4 /* xyz coordinates of corner 4 */
THICK width /* line thickness of rectangle border */
STYLE style /* line style of rectangle */
LINECOL r g b /* red green blue components of line color */
FILLCOL r g b /* red green blue components of fill color */
FILLPAT pattern /* fill pattern of rectangle */
THETA theta /* Viewing angle rect. was created in */
ALPHA alpha /* Viewing angle rect. was created in */
END /* end of rectangle data */
OVAL /* new oval identifier */
. /* same cards as rectangle */
.
END /* end of rectangle data */
LINE /* new line identifier */
VERTS n /* number of points in the line */
PT x1 y1 z1 /* xyz coordinates of point 1 */
PT x2 y2 z2 /* xyz coordinates of point 2 */
.
.
PT xn yn zn /* xyz coordinates of point n */
THICK width /* line thickness */
STYLE style /* line style */
LINECOL r g b /* red green blue components of line color */
ARRHED type /* arrow head flag */
HEDWID width /* arrow head base width */
HEDLEN length /* arrow head length */
END /* end of line data */
TEXT /* new text string identifier */
STRING "str" /* text string */
LOCAL x y z /* xyz location of text */
PCFONT "name" /* pc font name */
UNIXFONT "name" /* unix font name */
COLOR r g b /* red green blue components of text */
END /* end of text string data */
GRIDFRAME x y z dx dy dz  $\theta$  /* dimensions of gridframe */

```

Figure 17.16. Map File Format.

```

MAP
LEND
BEGCOV
COVNAME "default coverage"
COVATTS 2DMESH
NODE
XY 174.112149532710280 483.364485981308410
ID 1
GWNCARD 0.000 0.000 0.000 0.000
END
ARC
ID 1
NODES          1          1
ARCVERTICES 2
132.99065 133.83178
630.56075 429.90654
ARCBIAS 1.000000
END
POLYGON
ARCS 0
PATCHPTS 0 0 0 0
HARCS 1
1
ID 0
ADAPTESS 1.000000
END
POLYGON
ARCS 1
1
PATCHPTS 0 0 0 0

```

```

ID 1
ADAPTESS 1.000000
CELLING 1 0.577 0.577 0.577 0.577
MAT
MN      1                mat_1
MC      1                255    0    204
MS      1                5
END
ENDCOV
    
```

Figure 17.17 Sample Map File.

The format of the cards in the map file are given below:

17.13.1 General

The following card is used to define general map data.

<i>Card Type</i>	MAP
<i>Description</i>	File type identifier. Must be on first line of file. No fields.
<i>Required</i>	YES

17.13.2 Feature Objects

The following cards are used to define coverages, polygons, arcs, and points.

<i>Card Type</i>	BEGCOV
<i>Description</i>	Beginning of a series of cards defining a coverage.
<i>Required</i>	YES

<i>Card Type</i>	COVNAME		
<i>Description</i>	Coverage name.		
<i>Required</i>	NO		
<i>Format</i>	COVNAME "name"		
<i>Sample</i>	COVNAME "general"		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	name	str	Coverage name.

<i>Card Type</i>	COVATTS		
<i>Description</i>	Coverage model attributes set.		
<i>Required</i>	YES		
<i>Format</i>	COVATTS type		
<i>Sample</i>	COVATTS 2DMESH		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	type	2DGRID	2D grid.
		2DMESH	2D mesh

<i>Card Type</i>	ENDCOV
<i>Description</i>	End of cards defining a coverage.
<i>Required</i>	YES

<i>Card Type</i>	POINT
<i>Description</i>	Beginning of a series of cards defining a point.
<i>Required</i>	NO

<i>Card Type</i>	XY		
<i>Description</i>	xy coordinates of a point or node.		
<i>Required</i>	YES		
<i>Format</i>	XY x y		
<i>Sample</i>	XY 10.0 20.0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-2	x, y	±	xy coordinates of the point or node.

<i>Card Type</i>	ID		
<i>Description</i>	id of a feature object.		
<i>Required</i>	YES		
<i>Format</i>	ID id		
<i>Sample</i>	ID 10		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	id of the feature object

<i>Card Type</i>	GWN		
<i>Description</i>	1-D element paramaters		
<i>Required</i>	NO		
<i>Format</i>	GWNCARD W ₀ W _S S _L S _R		
<i>Sample</i>	GWNCARD 1.000 2.000 3.000 4.000		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-4	W ₀ , W _S , S _L , S _R	±	1-D element paramaters

<i>Card Type</i>	END
<i>Description</i>	End of a series of cards defining a feature object or drawing object.
<i>Required</i>	YES

<i>Card Type</i>	NODE
<i>Description</i>	Beginning of a set of cards defining a node.
<i>Required</i>	NO

<i>Card Type</i>	ARC
<i>Description</i>	Beginning of a set of cards defining an arc.
<i>Required</i>	NO

<i>Card Type</i>	NODES		
<i>Description</i>	Beginning and ending nodes of arc.		
<i>Required</i>	YES		
<i>Format</i>	NODES n1 n2		
<i>Sample</i>	NODES 10 15		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-2	n1, n2	+	id of beginning and ending nodes of an arc.

<i>Card Type</i>	ARCVERTICES		
<i>Description</i>	Vertices between nodes of an arc identifier.		
<i>Required</i>	Only if the arc has vertices		
<i>Format</i>	ARCVERTICES nvert x1 y1 x2 y2 . . . xn yn		
<i>Sample</i>	ARCVERTICES 3 5.0 10.0 12.0 8.0 2.5 7.6		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	nvert	+	Number of intermediate vertices
2-3	x, y	±	Vertex coordinates. Fields 2-3 repeated for each vertex

<i>Card Type</i>	POLYGON		
<i>Description</i>	Polygon Type identifier.		
<i>Required</i>	NO		

<i>Card Type</i>	ARCS		
<i>Description</i>	Defines the arcs forming the outer boundary of a polygon. There should only be one card of this type for each polygon, except for the universal polygon.		
<i>Required</i>	YES		
<i>Format</i>	ARCS count		
<i>Sample</i>	ARCS 2 10 12		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	count	+	Number of arcs in the polygon.
2	id	+	Arc id. Field 2 is repeated for each arc in the polygon. The arcs should be listed in clockwise order.

<i>Card Type</i>	HARCS		
<i>Description</i>	Defines the arcs bounding the holes in the polygon. There should be one card of this type for each hole in the polygon.		
<i>Required</i>	NO.		
<i>Format</i>	HARCS count		
<i>Sample</i>	HARCS 2 10 12		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	count	+	Number of hole-arcs in the polygon.
2	id	+	Arc id. Field 2 is repeated for each arc in the polygon. The arcs should be listed in counter-clockwise order.

17.13.3 Feature Object Attributes

The following cards are used to specify the attributes associated with points, arcs, and polygons. They should be placed in the file between the beginning and ending cards for the associated objects.

Attributes For Points and Nodes:

<i>Card Type</i>	SPECHHEAD		
<i>Description</i>	Specified Head head attribute.		
<i>Required</i>	YES if a feature object is of Specified Head type.		
<i>Format</i>	SPECHHEAD constflag value/id		
<i>Sample</i>	SPECHHEAD 1 138.6		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	constflag	0,1	The type code: 0 = Transient attribute represented by xy series. 1 = Constant attribute represented by a single value.
2	value/id	±/+	The constant value or the id of an xy series.

<i>Card Type</i>	SPECXVEL		
<i>Description</i>	Specified Velocity, X velocity component		
<i>Required</i>	YES if a feature point or node is of Specified Velocity type.		
<i>Format</i>	SPECXVEL constflag perp-to-boundary value/id		
<i>Sample</i>	SPECXVEL 1 0 138.6		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	constflag	0,1	The type code: 0 = Constant attribute represented by a single value. 1 = Transient attribute represented by xy series.
2	perp-to-boundary	0,1	1 = Nodal Velocity will be perpendicular to boundary.
3	value/id	±/+	The constant value or the id of an xy series.

Card Type	SPECYVEL		
Description	Specified Velocity, Y velocity component		
Required	YES if a feature point or node is of Specified Velocity type.		
Format	SPECYVEL constflag value/id		
Sample	SPECYVEL 1 152.6		
Field	Variable	Value	Description
1	constflag	0,1	The type code: 0 = Constant attribute represented by a single value. 1 = Transient attribute represented by xy series.
2	perp-to-boundary	0,1	1 = Nodal Velocity will be perpendicular to boundary.
3	value/id	±/±	The constant value or the id of an xy series.

Card Type	REFINE		
Description	es the size of the refine point.		
Required	NO		
Format	REFINE value		
Sample	REFINE 100		
Field	Variable	Value	Description
1	Value	+	The constant value which specifies the size of surrounding elements after meshing.

Arcs:

Card Type	SPECHELEV		
Description	Specified Head elevation attribute		
Required	YES if a refine arc is of Specified Head type.		
Format	SPECHELEV constflag value/Id type total-flow		
Sample	SPECHELEV 0 132 0 1		
Field	Variable	Value	Description
1	constflag	0,1	The type code: 0 = Transient attribute represented by xy series. 1 = Constant attribute represented by a single value.
2	value/id	±/±	The constant value or the id of an xy series.
3	type	0,1	The type code: 0 = Essential Specified Head Arc. 1 = Natural Specified Head Arc.
4	total-flow	0,1	The type code: 0 = Arc is not total flow arc. 1 = Arc is total flow arc.

Card Type	SPECCONC		
Description	Specified Concentration, Concentration Attribute.		
Required	YES if a feature arc is of Specified Concentration type.		
Format	SPECCONC constflag value/id		
Sample	SPECCONC 0 1.32		
Field	Variable	Value	Description
1	constflag	0,1	The type code: 0 = Transient attribute represented by xy series. 1 = Constant attribute represented by a single value.
2	value/id	±/±	The constant value or the id of an xy series.

<i>Card Type</i>	SPECFLOW		
<i>Description</i>	Specified Flow, flow attribute.		
<i>Required</i>	YES if a feature arc is of Specified Flow type.		
<i>Format</i>	SPECFLOW constflag value/id total-flow		
<i>Sample</i>	SPECFLOW 0 1.32 1		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	constflag	0,1	The type code: 0 = Transient attribute represented by xy series. 1 = Constant attribute represented by a single value.
2	value/id	±/+	The constant value or the id of an xy series.
3	total-flow	0,1	The type code: 0 = Arc is not total flow arc. 1 = Arc is total flow arc.

<i>Card Type</i>	FLUX		
<i>Description</i>	Flux arc attribute.		
<i>Required</i>	YES if a feature arc is of Flux type.		
<i>Format</i>	FLUX total-flow		
<i>Sample</i>	FLUX 1		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	total-flow	0,1	The type code: 0 = Arc is not total flow arc. 1 = Arc is total flow arc.

Polygons:

<i>Card Type</i>	ADAPTESS		
<i>Description</i>	Adaptive Tessellation type attributes.		
<i>Required</i>	YES if a feature polygon is of adaptive tessellation type.		
<i>Format</i>	ADAPTESS bias		
<i>Sample</i>	ADAPTESS 1.0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	bias	+	Bias for meshing.

<i>Card Type</i>	PATCH		
<i>Description</i>	Patch type attributes.		
<i>Required</i>	YES if a feature polygon is of patch type.		
<i>Format</i>	PATCH bias		
<i>Sample</i>	PATCH 1.0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	bias	+	Bias for meshing.

<i>Card Type</i>	CEILING		
<i>Description</i>	Four points defining a ceiling.		
<i>Required</i>	NO		
<i>Format</i>	CEILING flag a b c d		
<i>Sample</i>	CEILING 1 0.577 0.577 0.577 0.577		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	flag	0,1	The type code: 0 = Ceiling is off. 1 = Ceiling is on.
2-5	a,b,c,d	±	Four values a,b,c,d for use in plane equation $ax + by + cz + d = 0$ to define plane of ceiling..

<i>Card Type</i>	MAT		
<i>Description</i>	Beginning of the polygon material values		
<i>Required</i>	NO		

<i>Card Type</i>	MN		
<i>Description</i>	Material name and id.		
<i>Required</i>	YES if the MAT card is used.		
<i>Format</i>	MN id name		
<i>Sample</i>	MN 1 material_1		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	id of the material.
2	name	+	name of the material.

<i>Card Type</i>	MC		
<i>Description</i>	Material colors.		
<i>Required</i>	YES if the MAT card is used.		
<i>Format</i>	MN id r g b		
<i>Sample</i>	MN 1 255 255 255		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	id of the material.
2-4	r,g,b	+	Red, Green, Blue color components.

<i>Card Type</i>	MS		
<i>Description</i>	Material pattern.		
<i>Required</i>	YES if the MAT card is used.		
<i>Format</i>	MN id pattern		
<i>Sample</i>	MN 1 5		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	id of the material.
2	pattern	+	Pattern index.

17.13.4 Drawing Objects

The following cards are used to define rectangles, ovals, lines, and text strings.

<i>Card Type</i>	RECT
<i>Description</i>	The beginning of a set of cards defining a rectangle.
<i>Required</i>	NO

<i>Card Type</i>	OVAL
<i>Description</i>	The beginning of a set of cards defining an oval.
<i>Required</i>	NO

<i>Card Type</i>	C# (C1, C2, C3, C4, for the four corner points)		
<i>Description</i>	Corner point identifiers of rectangle or oval.		
<i>Required</i>	YES if a rectangle or oval has been defined.		
<i>Format</i>	C1 x y z		
<i>Sample</i>	C1 10 10 0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-3	x, y, z	±	Coordinates of corner point.

<i>Card Type</i>	THICK		
<i>Description</i>	Line thickness identifier.		
<i>Required</i>	YES if a line, rectangle or oval has been defined.		
<i>Format</i>	THICK width		
<i>Sample</i>	THICK 1		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	width	+	Line thickness in pixels.

<i>Card Type</i>	STYLE		
<i>Description</i>	Line style identifier.		
<i>Required</i>	YES if a line, rectangle or oval has been defined.		
<i>Format</i>	STYLE style		
<i>Sample</i>	STYLE 0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	style	+	0 - Solid line style. 1 - Dashed line style.

<i>Card Type</i>	LINECOL		
<i>Description</i>	Line color identifier.		
<i>Required</i>	YES if a line, rectangle or oval has been defined.		
<i>Format</i>	LINECOL r g b		
<i>Sample</i>	LINECOL 255 255 255		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-3	r, g, b	0-255	Red, green and blue color components

<i>Card Type</i>	FILLCOL		
<i>Description</i>	Polygon fill color identifier.		
<i>Required</i>	YES if a rectangle or oval has been defined.		
<i>Format</i>	FILLCOL r g b		
<i>Sample</i>	FILLCOL 255 255 255		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-3	r, g, b	0-255	Red, green, blue color components

<i>Card Type</i>	FILLPAT		
<i>Description</i>	Polygon fill pattern identifier.		
<i>Required</i>	YES if a rectangle or oval has been defined.		
<i>Format</i>	FILLPAT pattern		
<i>Sample</i>	FILLPAT 0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	pattern	+	Pattern index.

<i>Card Type</i>	LINE		
<i>Description</i>	Beginning of a set of cards defining a line object.		
<i>Required</i>	NO		

<i>Card Type</i>	VERTS		
<i>Description</i>	Number of points in a line.		
<i>Required</i>	YES if a line has been defined.		
<i>Format</i>	VERTS count		
<i>Sample</i>	VERTS 3		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	count	+	Number of points in the line.

<i>Card Type</i>	PT		
<i>Description</i>	Defines a point on a line object		
<i>Required</i>	YES if a line has been defined.		
<i>Format</i>	PT x y z		
<i>Sample</i>	PT 213.2. 523.2 0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-3	x y z	±	Coordinates of the point.

<i>Card Type</i>	ARRHED		
<i>Description</i>	Arrow head type identifier.		
<i>Required</i>	YES if a line has been defined.		
<i>Format</i>	ARRHED style		
<i>Sample</i>	ARRHED 0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	style	0	No arrow head.
		1	Arrow head at beginning of line.
		2	Arrow head at end of line.
		3	Arrow heads at both ends of line.

<i>Card Type</i>	HEDWID		
<i>Description</i>	Arrow head base width identifier.		
<i>Required</i>	YES if a line has been defined.		
<i>Format</i>	HEDWID width		
<i>Sample</i>	HEDWID 10		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	width	+	Width of the base of the arrow head in pixels.

<i>Card Type</i>	HEDLEN		
<i>Description</i>	Arrow head length identifier.		
<i>Required</i>	YES if a line has been defined.		
<i>Format</i>	HEDLEN length		
<i>Sample</i>	HEDLEN 25		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	length	+	Length of the arrow head in pixels.

<i>Card Type</i>	TEXT		
<i>Description</i>	Beginning of a set of cards defining a text object.		
<i>Required</i>	NO		

<i>Card Type</i>	STRING		
<i>Description</i>	Text string identifier.		
<i>Required</i>	YES if a text string has been defined.		
<i>Format</i>	STRING "string"		
<i>Sample</i>	STRING "map title"		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	string	str	Text string.

<i>Card Type</i>	LOCAL		
<i>Description</i>	Text string location identifier.		
<i>Required</i>	YES if a text string has been defined.		
<i>Format</i>	LOCAL x y		
<i>Sample</i>	LOCAL 100 200		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-2	x, y	±	Coordinates of the beginning of the text string.

<i>Card Type</i>	PCFONT		
<i>Description</i>	PC font identifier.		
<i>Required</i>	YES if a text string has been defined.		
<i>Format</i>	PCFONT id		
<i>Sample</i>	PCFONT 2		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The id of the PC font.

<i>Card Type</i>	UNIXFONT		
<i>Description</i>	UNIX font identifier.		
<i>Required</i>	YES if a text string has been defined.		
<i>Format</i>	UNIXFONT id		
<i>Sample</i>	UNIXFONT 2		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The id of the UNIX font.

17.14 Image Files

Image files are used in conjunction with TIFF files which have been previously imported to *SMS* and registered. They include the name of the TIFF file, the registration points, and the bounds of the clipping window. The format of the image file is shown in Figure 18.50 and a sample image file is shown in Figure 18.51.

```
IMAGE /* File type identifier */
TIFF "filename" /* Indicates the name of the tiff file used */
IMREGPTS
PT1 u1 v1 x1 y1
PT2 u2 v2 x2 y2
PT3 u2 v2 x2 y2
CLIPPOINT
x1 x2
y1 y2
```

Figure 17.18 The Image File Format.

```
IMAGE
TIFF "easttex.tif"
IMREGPTS
PT1 117 797 0.000000 10000.000000
PT2 117 88 0.000000 0.000000
PT3 1053 88 13220.000000 0.000000
CLIPPOINT
-1082.059503 13885.402536
-992.568818 8457.158566
```

Figure 17.19 Sample Image File.

The card types used in the Image file format are as follows:

<i>Card Type</i>	IMAGE		
<i>Description</i>	File type identifier. Must be on first line of file. No fields.		
<i>Required</i>	YES		

<i>Card Type</i>	TIFF		
<i>Description</i>	Defines the name of the TIFF file to be displayed as an image.		
<i>Required</i>	YES		
<i>Format</i>	TIFF "filename"		
<i>Sample</i>	TIFF "easttex.tif"		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	filename	str	The name of the TIFF file.

<i>Card Type</i>	PT1, PT2, PT3		
<i>Description</i>	The three registration points used to define locations on a given image.		
<i>Required</i>	YES		
<i>Format</i>	PT1 tx1 ty1 wx1 wy1 PT2 tx2 ty2 wx2 wy2 PT3 tx3 ty3 wx3 wy3		
<i>Sample</i>	PT1 117 797 0.000000 10000.000000 PT2 117 88 0.000000 0.000000 PT3 1053 88 13220.0 0.000000		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-2	tx ty	±	Texture map coordinates.
3-4	wx wy	±	World coordinates.

<i>Card Type</i>	CLIPPOINTS		
<i>Description</i>	Defines the coordinates of the area in the TIFF file to be displayed as the image. (The area clipped and displayed from the TIFF file.)		
<i>Required</i>	YES		
<i>Format</i>	CLIPPOINTS xmin xmax ymin ymax		
<i>Sample</i>	CLIPPOINTS -628.990382 14338.471657 -857.665608 8354.617436		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1-2	xmin xmax	±	Min and max values in the x direction.
3-4	ymin ymax	±	Min and max values in the y direction.

17.15 Material Files

Each element of a 2D mesh has an assigned material ID. Specific material properties are related to the analysis models, and are stored in the analysis files. However, general material properties, such as color, are not stored in these files. Therefore, they are stored in the material file. A material ID represents an index to a global list of materials. The material file associates general attributes such as a name, color, and pattern with each of the materials. The format for a material file is shown in Figure 17.20.

```

MAT      /* File type identifier */
MN id name          /* Material name */
MC id red green blue /* Material color */
MS id stippleid     /* Material stipple (fill pattern) */

```

Figure 17.20 Material File Format.

Each card in the material file represents an attribute for a material. The attribute cards can be repeated as many times as necessary to define each material being used. The cards used in the material file are as follows:

<i>Card Type</i>	MAT
<i>Description</i>	File type identifier. Must be on first line of file. No fields.
<i>Required</i>	YES

<i>Card Type</i>	MN		
<i>Description</i>	Identifies a name to be associated with the material.		
<i>Required</i>	NO		
<i>Format</i>	MN id name		
<i>Sample</i>	MN 5 bedrock		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID of the material.
2	name	str	The name of the material.

<i>Card Type</i>	MC		
<i>Description</i>	Identifies a color to be associated with the material.		
<i>Required</i>	NO		
<i>Format</i>	MN id red green blue		
<i>Sample</i>	MN 5 124 67 245		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID of the material.
2	red	0-255	The value of the red component of the color.
3	green	0-255	The value of the green component of the color.
4	blue	0-255	The value of the blue component of the color.

<i>Card Type</i>	MS		
<i>Description</i>	Identifies a stipple (fill pattern) to be associated with the material. This stipple is used whenever an object is being drawn using color filled polygons.		
<i>Required</i>	NO		
<i>Format</i>	MN id stippleid		
<i>Sample</i>	MN 5 13		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID of the material.
2	stippleid	+	The ID of the stipple.

17.16 Mesh From Polygon Files

Mesh from polygon files are used with the *Mesh From Poly* command. It is accessed from the *Elements* menu while in the *Mesh Module*. The format for an adaptive tessellation file is identical to that of a polygon file, except that additional feature information describing channel or ridge locations may be included. These include *refine points* and *breaklines*.

An adaptive tessellation file contains one or more POLY cards as shown in Figure 17.21. The vertices of the polygon are listed in counter-clockwise order. In some cases, it is necessary to define a polygon with holes in the interior using a polygon file. In such cases, the boundary of the polygon should be defined with a POLY card with the vertices in counter-clockwise order. Each of the interior holes should then be listed with separate POLY cards with the vertices listed in clockwise order.

A refine point causes *SMS* to insert a hard nodal point into the mesh. When the adaptive tessellation is run, *SMS* creates six small triangles surrounding the refine point. Each of the triangles will have edge lengths corresponding to a specified value. A smooth element size transition will be defined as the mesh moves out from the refine point. The smaller the edge length, the smaller the elements surrounding the refine point. A breaklinepolygon causes *SMS* to force a breakline through the mesh. Element edges will conform to the breakline.

```
POLY np /* Beg. of polygon loop */
x1 y1 z1          /* Vertices are listed, ccw order */
x2 y2 z2
.
xnp Ynp znp
BLINE np          /* Beg. break line definition */
x1 y1 z1          /* vertices of the breakline */
x2 y2 z2
.
xnp Ynp znp
REFPT x y z edgelen /* refine point definition */
```

Figure 17.21 Adaptive Tessellation File Format.

The format of the adaptive tessellation file is as follows:

Card Type	POLY		
Description	Defines one complete loop of a polygon.		
Required	YES		
Format	POLY np x1 y1 z1 x2 y2 z2 . . xnp ynp znp		
Sample	POLY 4 0.0 0.0 0.0 10.0 0.0 0.0 10.0 10.0 0.0 0.0 10.0 0.0		
Field	Variable	Value	Description
1	np	+	The number of points in the polygon loop.
2-4	x,y,z	±	Vertex coordinates. Repeat for each vertex. List in counter-clockwise order for the outer polygon boundary. List in clockwise order for holes in the interior of a polygon.

Card Type	REFPT		
Description	Defines a refine point within a 2D mesh		
Required	NO		
Format	REFPT x y z edgelen		
Sample	REFPT 10.0 25.0 13.5 0.25		
Field	Variable	Value	Description
1-3	x,y,z	±	Location of the refine point within the boundary polygon.
4	edgelen	±	length of one of the edges of the triangles that will immediately surround the refine point

Card Type	BLINE		
Description	Defines a feature line or "breakline" within a 2D mesh		
Required	NO		
Format	BLINE np x1 y1 z1 x2 y2 z2 . . xnp ynp znp		
Sample	BLINE 4 1.0 0.0 0.0 5.0 1.0 2.0 10.0 5.0 2.5 12.0 10.0 3.0		
Field	Variable	Value	Description
1	np	+	The number of vertices in the break line.
2-4	x,y,z	±	Vertex coordinates. Repeat for each vertex. The breakline should not cross itself.

17.17 SMS Super Files

A *SMS* super file is a file which contains a list of other files. Each of the files in the list must be one of the basic *SMS* file types (2D meshes, 2D scatter points, materials, TINs). If a super file is selected using the *Open* command in the *File* menu, each of the files listed in the super file are opened and imported. This makes it possible to quickly read in several files without having to identify each file individually in the file browser.

The file format for a super file is shown in Figure 17.22. The first line in the file is the *SUPER* card, which identifies the file as a super file. Each of the other cards shown are optional. Each of the file cards has a card identifier representing the type of file. The identifier is followed by a file name. The file name should be a complete path if the file is not in the same directory as the super file. Any suffix may be used for the file name. A sample super file is shown in Figure 17.23.

```

SUPER                               /* File type identifier */
MAT filename                         /* Material File */
SCAT2D filename                      /* 2D scatter point file */
MAP filename                         /* Map file */
MESH2D filename                      /* 2D Mesh file */
DATA filename                        /* Dataset File */
STNGS filename                       /* Settings (*.ini) File */
IMAGE filename                       /* Image file */

```

Figure 17.22 Super File Format.

```

SUPER
MAT c:\SMS\DATA\SITE1\site1.mat
SCAT2D c:\SMS\DATA\SITE1\site1.xyf

```

Figure 17.23 Sample Super File.

17.18 TIN Files

TIN files store triangulated irregular network data. Multiple TINs can be stored to a single file. The TIN file format is shown in Figure 17.24. SMS can import TINs, converting them to triangular elements.

```
TIN      /* File type identifier */
BEGT    /* Beginning of TIN group */
TNAM name          /* Name of TIN */
MAT id /* TIN material id */
VERT nv /* Beg. of vertices */
x1 y1 z1 lf1          /* Vertex coords. */
x2 y2 z2 lf2
.
.
.
xnv Ynv znv lfnv
TRI nt /* Beg. of triangles */
v11 v12 v13          /* Triangle vertices */
v21 v22 v23
.
.
.
Vnt1 Vnt2 Vnt3
ENDT    /* End of TIN group */
/* Repeat TIN group for other TINs */
```

Figure 17.24 TIN File Format.

The cards used in the TIN file are as follows:

<i>Card Type</i>	TIN
<i>Description</i>	File type identifier. Must be on first line of file. No fields.
<i>Required</i>	YES

<i>Card Type</i>	BEGT
<i>Description</i>	Marks the beginning of a group of cards describing a TIN. There should be a corresponding ENDT card at a latter point in the file. No fields.
<i>Required</i>	YES

<i>Card Type</i>	TNAM		
<i>Description</i>	Provides a name to be associated with the TIN.		
<i>Required</i>	NO		
<i>Format</i>	TNAM name		
<i>Sample</i>	TNAM bathymetry		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	name	str	The name of the TIN.

<i>Card Type</i>	MAT		
<i>Description</i>	Associates a material id with the TIN. This is typically the id of the material which is below the TIN.		
<i>Required</i>	NO		
<i>Format</i>	MAT id		
<i>Sample</i>	MAT 3		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The material ID.

<i>Card Type</i>	VERT		
<i>Description</i>	Lists the vertices in the TIN		
<i>Required</i>	YES		
<i>Format</i>	VERT nv x1 y1 z1 lf1 x2 y2 z2 lf2 . . x _{nv} y _{nv} z _{nv} lf _{nv}		
<i>Sample</i>	VERT 4 0.0 3.1 7.8 0 5.3 8.7 4.0 1 2.4 4.4 9.0 1 3.9 1.2 3.6 0		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	nv	+	The number of vertices in the TIN.
2-4	x,y,z	±	Coordinates of vertex.
5	lf	0,1	Locked / unlocked flag for vertex (optional). 0=unlocked, 1=locked. Repeat fields 2-5 nv times.

<i>Card Type</i>	TRI		
<i>Description</i>	Lists the triangles in the TIN		
<i>Required</i>	NO		
<i>Format</i>	TRI nt v11 v12 v13 v21 v23 v23 . . v _{nt1} v _{nt2} v _{nt3}		
<i>Sample</i>	TRI 4 5 1 4 4 1 2 4 2 3 5 4 3		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	nt	+	The number of triangles in the TIN.
2-4	v1,v2,v3	+	Vertices of triangle listed in a counter-clockwise order. Repeat nt times.

<i>Card Type</i>	ENDT		
<i>Description</i>	Marks the end of a group of cards describing a TIN. There should be a corresponding BEGT card at a previous point in the file. No fields.		
<i>Required</i>	YES		

17.19 XY Series Files

The *XY Series Editor* described in Chapter **Error! Reference source not found.** is used in several places in *SMS*, including the definition of time dependent boundary conditions and rating curves. The *XY Series Editor* is a general purpose editor for entering curves or pairs of lists of data. The *XY Series Editor* allows a curve to be imported from a file, created and edited graphically, or created and edited using two columns of edit fields in a spreadsheet-like interface.

XY series files can be used to input a set of curves into the *XY Series Editor*. XY series files are also used to export curves generated within the Editor for future use.

The format of the XY series file is shown in Figure 17.25. Curves are defined in an XY Series File using one of three types of cards: XY1, XY2, or XY3. With the XY1 card, both the x and y values are listed for each point on the curve. There is no limit to the spacing or interval used between subsequent x values. The XY2 card is identical to the XY1 card except that the number of points and the x values are assumed to be static and cannot be altered by the user. With the XY3 card, the x values are defined by a beginning x value, an initial increment in x, and a per cent change in x per increment. Only the y values are explicitly listed.

```

XY1 id n dx dy rep begc name          /* XY Series vers. #1 */
x1 y1 /* XY values */
x2 y2
.
xn Yn
XY2 id n dx dy rep begc name          /* XY Series vers. #2 */
x1 y1 /* XY values */
x2 y2
.
xn Yn
XY3 id n x1 incx pcx dx dy rep begc name /* XY Series vers. #2 */
y1 /* Y values */
y2
.
.
yn

```

Figure 17.25 The XY Series File Format.

The card types used in the XY series file format are as follows:

Card Type	XY1		
Description	Defines a curve with a list of XY values. Any number of points and any x spacing between points may be used.		
Required	NO		
Format	XY1 id n dx dy rep begc name x1 y1 x2 y2 . . xn Yn		
Sample	XY1 1 5 0 0 0 0 head 0.0 0.0 1.0 2.0 2.5 7.0 3.0 8.0 4.5 9.5		
Field	Variable	Value	Description
1	id	+	The ID of the XY series.
2	n	+	The number of point in the series.
3	dx	0,1	A flag defining whether the x values listed are to be interpreted as incremental (dx=1) or absolute (dx=0).
4	dy	0,1	A flag defining whether the y values listed are to be interpreted as incremental (dy=1) or absolute (dy=0).
5	rep	0,1	A flag defining whether the xy series is to be interpreted as cyclic (repeating).
6	begc	±	The x value in the series where the cyclic portion of the curve begins. Value is ignored if rep=0.
7	name	str	The name of the series.
8-9	x,y	±	The xy values of the points defining the curve. Repeat n times.

Card Type	XY2		
Description	Defines a curve with a list of XY values. This card is identical to the XY1 card except that the number of points and the x values are assumed to be static and cannot be altered by the user.		

<i>Card Type</i>	XY3		
<i>Description</i>	Defines a curve with a list of Y values. The x values are defined by a beginning value, an increment, and a bias.		
<i>Required</i>	NO		
<i>Format</i>	XY3 id n x1 incx biasx dx dy rep begc name y1 y2 . . yn		
<i>Sample</i>	XY3 1 10 0 1 0 0 0 0 0 head 0.0 2.0 7.0 8.0 9.5 9.1		
<i>Field</i>	<i>Variable</i>	<i>Value</i>	<i>Description</i>
1	id	+	The ID of the XY series.
2	n	+	The number of point in the series.
3	x1	±	The first x value.
4	incx	±	The increment in x used to compute the next x value.
5	pcx	+	The per cent change in x used to compute subsequent x values. Expressed as a decimal, i.e., 0.05 = 5%.
6	dx	0,1	A flag defining whether the x values listed are to be interpreted as incremental (dx=1) or absolute (dx=0).
7	dy	0,1	A flag defining whether the y values listed are to be interpreted as incremental (dy=1) or absolute (dy=0).
8	rep	0,1	A flag defining whether the xy series is to be interpreted as cyclic (repeating).
9	begc	±	The x value in the series where the cyclic portion of the curve begins. Value is ignored if rep=0.
10	name	str	The name of the series.
11	y	±	The y values of the points defining the curve. Repeat n times.

17.20 XYZ Files

XYZ points can be imported from an ASCII file and converted to nodes. This provides a convenient way to import a set of points for mesh construction operations. The format of the XYZ file is shown in Figure 17.26.

```
XYZ      (The first line should contain the word "XYZ")
x1 y1 z1      (Listing of XYZ data point coordinates)
x2 y2 z2
x3 y3 z3
.
.
.
xn yn zn
```

Figure 17.26 XYZ File Format.

References

Clough, R. W., and J. L. Tocher, 1965, Finite element stiffness matrices for analysis of plates in bending, *Proc. Conf. Matrix Methods in Structural Mechanics, Wright-Patterson A.F.B., Ohio, Air Force Flight Dynamics Lab., Research and Technology Division, Air Force Systems Command, The Air Force Institute of Technology, Air University*, pp. 515-545.

Davis, J.C., 1986, *Statistics and Data Analysis in Geology*, John Wiley & Sons, New York, 550 p.

Deutsch, C.V., & A.G. Journel, 1992, *GSLIB: Geostatistical Software Library and User's Guide*, Oxford University Press, New York, 340 p.

Franke, R. & G. Nielson, 1980, "Smooth interpolation of large sets of scattered data," *International Journal for Numerical Methods in Engineering*, Vol. 15, pp. 1691-1704.

Franke, R., 1982, "Scattered data interpolation: tests of some methods," *Mathematics of Computation*, Vol. 38, No. 157, pp. 181-200.

Heine, G. W., 1986, "A controlled study of some two-dimensional interpolation methods," *COGS Computer Contributions*, Vol. 2, No. 2, pp. 60-72.

Jones, N. L., 1990, *Solid Modeling of Earth Masses for Applications in Geotechnical Engineering*, Ph.D. Dissertation, The University of Texas at Austin, 324 p.

Journel, A.G., & Huijbregts, C.J., 1978, *Mining geostatistics*. Academic Press, New York, NY.

Lam, N.S., 1983, "Spatial interpolation methods: a review," *The American Cartographer*, Vol. 10, No. 2, pp. 129-149.

Lancaster, Peter and Kestutis Salkauskas, 1986, *Curve and Surface Fitting*, Academic Press, London, 280 pp.

McDonald, M.G., & A.W. Harbaugh, 1988, *A modular three-dimensional finite-difference ground-water flow model*, Techniques of Water Resources Investigations 06-A1, United States Geological Survey.

Olea, R.A., 1974, "Optimal contour mapping using universal kriging." *J. Geophys. Res.*, Vol. 79, No. 5, pp. 695-702.

Owen, S.J., 1992, *An implementation of natural neighbor interpolation in three dimensions*, Master's Thesis, Brigham Young University, 119 p.

Philip, G.M., & D.F. Watson, 1986, "Comment on 'comparing splines and kriging,'" *Computers and Geosciences*, Vol. 12, No. 2, pp. 243-245.

Royle, A. G., F. L. Clausen, & P. Frederiksen, 1981, "Practical universal kriging and automatic contouring," *Geo-Processing*, Vol. 1, No. 4, pp. 377-394.

Shearman, J.O., Kirby, W.H., Schneider, V.R., and Flippo H.N., 1986, "Bridge Waterways Analysis Model: Research Report", *Report No. FHWA/RD-86/108*, US DOT, FHWA.

Shepard, D., 1968, "A two dimensional interpolation function for irregularly spaced data," *Proc. 23rd National Conference of the ACM*, pp. 517-523.

Sibson, R., 1981, "A brief description of natural neighbor interpolation," *Interpreting Multivariate Data*, John Wiley & Sons, New York, pp. 21-36.

Watson, D. F. and G. M. Philip, 1985, A refinement of inverse distance weighted interpolation, *Geo-Processing*, Vol., 2, No. 4, pp. 315-327.

WES, 1994, *FEMWATER Reference Manual*, U.S. Army Engineer Waterways Experiment Station.

Yeh, G.T., S.S. Hansen, B. Lester, R. Strobl, J. Scarbrough, 1992, *3DFEMWATER/3DLEWASTE: Numerical Codes for Delineating Wellhead Protection Areas in Agricultural Regions Based on the Assimilative Capacity Criterion*, U.S. Environmental Protection Agency.

Zheng, C., 1990, "MT3D: A Modular Three-Dimensional Transport Model for Simulation of Advection, Dispersion and Chemical Reactions of Contaminants in Groundwater Systems." S.S. Papadopoulos & Associates, Inc.

Index

- 1D, 12-1
- 2D Boundary Fitted Grid
 - module, 1-6
- 2D mesh
 - file, 2-10
 - import, 2-13
- 2D Mesh
 - module, 1-6
- 2D scatter point
 - file, 2-9, 3-4, 17-4
 - XY format, 2-9
 - XYD format, 2-9
- 2D Scatter Point
 - module, 1-6
- active
 - scatter point set, 13-2, 13-3
- ADCIRC, 8-15
 - boundary conditions, 8-3
 - create functions, 8-5
 - global output, 8-9
 - model control, 8-7
 - new, 8-1
 - normal flow, 8-4
 - open simulation, 8-2
 - save simulation, 8-2
 - station output, 8-8
 - tides, 8-13
 - time control, 8-12
 - wind, 8-10
- animation, 3-4, 3-13, 3-16
 - flow trace, 3-16
 - time, 3-15
- assign boundary conditions, 5-3
- background, 2-22
- band width, 4-34
- barycentric, 13-12
- boundary condition
 - delete, 5-6
 - HIVEL, 7-5
 - node, 5-3
 - nodestring, 5-5, 9-2
- Boundary Condition
 - SED2D-WES, 6-9
- boundary conditions
 - ADCIRC, 8-3
- boundary conditions, assign, 5-3
- breakline, 4-30, 17-45
- bridge sections
 - WSPRO, 12-7
- card, 17-1
 - \$L, 5-14, 9-7
 - \$M, 5-17
 - FT, 5-16
 - SI, 5-16, 9-8
 - T1-3, 5-13, 8-8, 9-6
- CGWAVE
 - create functions, 9-3
 - model control, 9-5
 - new, 9-2
 - open geometry, 9-2
 - save geometry, 9-2
 - solver, 9-8
 - title, 9-6
- circumcircle, 4-25
- coastline files, 2-13
- color

- background, 2-22
- contour, 3-10
 - HSV, 3-11
 - intensity, 3-11
 - ramp, 3-10
 - scatter point, 13-4
- continuity string, 5-6
- contour
 - animation, 3-15
 - colors, 3-10, 3-11
 - label, 4-8
 - label tool, 3-12
 - labels, 3-11
 - legend, 3-9
 - spline, 3-10
 - values, 3-9
- control structure
 - RMA2, 5-22
- convert
 - scatter point, 13-4
- coordinates, 2-8
 - barycentric, 13-12
 - local, 13-14
 - snapping to a grid, 2-22
 - xy series, 16-1
- copyright, i
- create
 - element, 4-4
 - node, 4-2
 - nodestring, 4-3
 - pier, 4-8
- create functions
 - ADCIRC, 8-5
 - CGWAVE, 9-3
- DA records, 12-9
- data, 1-7, 10-10
 - browser, 3-2
 - film loops, 3-13
 - menu, 3-1
 - vectors, 3-7
- data set
 - active, 3-4
 - animation, 3-16
 - calculator, 3-4, 3-5
 - contours, 3-8
 - deleting, 3-5
 - elevation, 3-4
 - export, 3-4
 - file, 2-10, 3-3
 - file (ASCII), 17-9
 - file (binary), 17-17
 - import, 3-3
 - info, 3-5
 - legend, 3-9
 - maximum, 3-5
 - mean, 3-5
 - minimum, 3-5
 - name, 3-5
 - SED2D-WES, 6-12
 - standard deviation, 3-5
 - statistics, 3-5
 - time step, 3-4
- data sets, 1-7
- DC records, 12-9
- delete, 2-7, 2-18
 - boundary condition, 5-6
 - confirm, 2-19
 - data set, 3-5
 - element, 4-6
- demo, 2-17
- diffusion coefficients
 - SED2D-WES, 6-7
- disjoint
 - node, 4-18, 4-29
- display
 - background, 2-22
 - contour, 3-8
 - frame image, 2-21
 - grid, 2-22
 - refresh, 2-20
 - vectors, 3-7
 - window bounds, 2-21
- display options
 - mesh, 2-21, 4-12
 - RMA2, 5-25
 - scatter points, 13-3
- drogue files, 2-13
- duplicate
 - node, 4-18
- eddy viscosity, 5-9
- EDIT MENU, 2-18
- element, 4-22
 - concave, 4-7
 - create, 4-4
 - deletion, 4-6
 - junction, 5-22
 - line, 5-20
 - transition, 5-21
 - material, 4-34
 - material type, 4-6
 - merge/split, 4-7
 - options, 4-22
 - patch, rectangular, 4-26
 - patch, triangular, 4-27

- refine, 4-24
- relax, 4-32
- renumber, 4-33
- select, 4-6
- swap edges, 4-6
- triangular, 4-6
- elements
 - dry, 5-18
- elevation, 3-4
 - map, 4-11
- enabling SMS, 2-18
- environment
 - save, 2-15
- exit, 2-18
- export
 - data set, 3-4
 - file, 2-14
 - scatter points, 13-3
 - xy series, 16-2
- extrapolation, 13-7
 - default value, 13-7
 - inverse distance weighted, 13-7
 - natural neighbor, 13-17
- feature
 - from material, 4-10
- FESWMS, *10-1*
 - append, 2-12
 - Boundary Conditions, 10-3
 - Boundary Section, 10-5
 - Ceiling, 10-10
 - Control, 10-12
 - Culvert, 10-9
 - Drop Inlet, 10-9
 - Flux String, 10-10
 - initial boundary conditions, *10-7*
 - Material Properties, 10-11
 - Model Check, 10-12
 - open simulation, *10-2*
 - Pier, 10-10
 - print, *10-18*
 - save, *10-3*
 - Weir, 10-9
 - Wind Conditions, 10-8
- FHWA, 12-1
- file
 - 2D mesh, 2-10
 - 2D scatter point, 2-9, 3-4, 17-4
 - binary, 3-4
 - cards, 17-1
 - coastline, 2-13
 - data set, 2-10, 3-3
 - data set (ASCII), 17-9
 - data set (binary), 17-17
 - drogue, 2-13
 - DXF, 2-14
 - export, 2-14
 - FESWMS, 3-3
 - film loop, 3-17
 - HIVEL2D, 3-3
 - image, 2-10, 17-40
 - import, 2-11
 - map, 2-10, 17-28
 - material, 2-10, 2-19, 17-43
 - menu, 2-9
 - mesh from polygon, 17-45
 - RMA2, 3-3
 - scatter point, 13-2
 - settings, 2-10
 - shapefiles, 2-13
 - simulation, 2-9
 - types, 2-9
 - view data, 2-22
 - xy series, 17-50
 - XYZ, 17-53
- File
 - SED2D-WES, 6-1
- film loop, 3-4, 3-16
 - data set, 3-14
 - dialog, 3-13
 - file, 3-17
 - flow trace, 3-16
 - image size, 3-14
 - playback, 3-17
 - saving, 3-17
 - time, 3-15
- film loops, 3-13
- find
 - node, 4-17
- format
 - reference manual, 1-8
- frame image, 2-21
- fringe
 - colors, 3-11
 - legend, 3-9
- front width, 4-34
- GC String, 5-6
- geometry
 - append, 2-12
 - open, 5-1, 9-2
 - save, 5-2, 9-2
- get info, 2-15
- GFGEN
 - append, 2-12
 - run, 5-24

- GIS
 - from material, 4-10
- global output
 - ADCIRC, 8-9
- Graphics Window, 2-4
- grid, 2-22
 - snap to, 2-22
- help, 2-8
- HIVEL
 - boundary condition
 - delete, 7-6
 - boundary condition, 7-5
 - card
 - grav, 7-8
 - mcon, 7-8
 - pgwc, 7-8
 - SI, 7-8
 - step, 7-9
 - T1-T3, 7-7
 - time, 7-9
 - turb, 7-8
 - hot start, 7-3
 - materials, 7-10
 - model check, 7-10
 - new, 7-2
 - open, 7-2
 - run control, 7-7
 - save, 7-2
 - title, 7-7
 - units control, 7-8
- HP records, 12-6
- IDW. (see inverse distance weighted)
- image
 - file, 2-10
- image file, 17-40
- import
 - xy series, 16-2
- import, 2-11
 - file, 2-11
- index
 - material, 2-19
 - node, 4-17, 4-29
 - scatter point, 13-4
- initial boundary conditions
 - FESWMS, 10-7
- initial concentration
 - SED2D-WES, 6-7
- interpolation
 - cluster, 13-9
 - global, 13-10
 - IDW. (see inverse distance weighted)
 - inverse distance weighted. (see inverse distance weighted)
 - linear, 13-6
 - local, 13-10
 - local coordinates, 13-14
 - natural neighbor. (see natural neighbor)
 - scatter point, 13-4
 - Shepard's method. (see inverse distance weighted)
 - subsets, 13-9
 - truncation, 13-5
- inverse distance weighted, 13-13
 - barycentric weights, 13-12
 - interpolation subsets, 13-9
 - local weighting method, 13-10
 - local/global, 13-10
 - weights, 13-7
- iterations, 5-14
- junction
 - element, 5-22
- labels
 - plots, 3-19
- legend, 3-9
 - plots, 3-19
- line elements, 5-20
- lock
 - node, 4-19
- manning's n, 5-7
- map
 - elevation, 4-11
 - file, 2-10, 17-28
 - module, 1-6
- marsh
 - porosity, 5-10
 - RMA2, 5-19
- material
 - dialog, 2-19
 - file, 2-10, 2-19, 17-43
 - mesh, 4-34
- materials, 2-19
 - HIVEL, 7-10
 - RMA2, 5-7
- maximum, 3-8
- mean, 3-5
- menu
 - bar
 - general, 2-9
 - data, 3-1
 - display, 2-20, 2-22
 - edit, 2-19
 - file, 2-9
 - node, 4-2

- merge, 4-7
 - triangles, 4-30
- mesh, 4-1
 - create, 4-2
 - creating, 4-24
 - display options, 2-21, 4-12
 - generation, 4-6, 4-7
 - material, 4-34
 - quality, 4-14
- mesh from polygon
 - file, 17-45
- mesh generation
 - triangulation, 4-24
- minimum, 3-9
- model check
 - HIVEL, 7-10
 - RMA2, 5-10
 - WSPRO, 12-14
- Model Check
 - SED2D-WES, 6-12
- model control
 - ADCIRC, 8-7
 - CGWAVE, 9-5
- module, 1-5
 - 2D Boundary Fitted Grid, 1-6
 - 2D Mesh, 1-6
 - 2D Scatter Point, 1-6
 - map, 1-6
 - mesh, 4-1
 - river, 1-7, 11-1
 - scatter point, 13-1
 - selecting a new, 2-5
- MS Windows, 1-8
- natural neighbor
 - bounding window, 13-16
 - extrapolation, 13-17
 - local coordinates, 13-14
 - weights, 13-16
- new, 2-10, 12-2
 - ADCIRC, 8-1
 - CGWAVE, 9-2
 - HIVEL, 7-2
 - WSPRO, 12-2
- New
 - SED2D-WES, 6-2
- node
 - boundary condition, 4-22
 - boundary condition, 5-3
 - create, 4-2
 - disjoint, 4-18
 - duplicate, 4-18
 - find, 4-17
 - interpolation, 4-17
 - interpolation options, 4-16
 - lock, 4-19
 - midside, 4-5
 - operations, 4-15
 - options, 4-20
 - renumber, 4-33
 - select, 4-2
 - transform, 4-19
- nodestring
 - boundary condition, 5-5, 9-2
 - continuity string, 5-6
 - create, 4-3
 - select, 4-4
 - smooth, 4-32
- normal flow
 - ADCIRC, 8-4
- open, 2-11
 - ADCIRC simulation, 8-2
 - geometry, 5-1, 9-2
 - HIVEL, 7-2
 - SED2D-WES, 6-2
 - WSPRO, 12-2
- open simulation
 - FESWMS, 10-2
- orientation
 - paper, 2-16
- OVERVIEW, 1-1
- page layout, 2-16, 2-17
- page size, 2-16
- palette
 - dynamic, 2-7
 - macro, 2-7
 - module, 2-5
 - static, 2-6
- pan, 2-6
- patch
 - Coon's patch, 4-26, 4-28
 - elements, 4-26
 - triangular, 4-27
- pier
 - create, 4-8
 - select, 4-8
- pixel map, 3-13
- plot
 - window, 2-23
- plots
 - labels, 3-19
 - legend, 3-19
 - print, 3-19
- polygon, 2-19
 - Thiessen, 13-13

- porosity
 - marsh, 5-10
 - RMA2, 5-19
- postscript
 - setup, 2-15
- print
 - plots, 3-19
- Print
 - SED2D-WES, 6-11
- printer
 - postscript, 2-15
 - setup, 2-15
- printing, 2-15
 - page layout, 2-16, 2-17
 - postscript, 2-16
 - setup, 2-15, 2-16
- quit, 2-18
- reference
 - format, 1-8
- references, 18-1
- refine
 - elements, 4-24
- refresh, 2-20
- register, 2-18
- relax, 4-32
 - elements, 4-23
- renumber, 4-33
- river, *11-1*
 - module, 1-7
 - plots, 11-6
 - sections, 11-3
 - window, 12-2
- RMA2, 5-1
 - files, 5-13
 - geometry, 5-18
 - iterations, 5-14
 - machine type, 5-17
 - materials, 5-7
 - model check, 5-10
 - open geometry, 5-1
 - other options control, 5-16
 - run, 5-24
 - save geometry, 5-2
 - simulation file, 5-1
 - time control, 5-15
 - title, 5-13
 - units control, 5-16
- roughness, 12-11
 - by depth, 5-8
- run control
 - HIVEL, 7-7
 - RMA2, 5-12
- save, 2-11
 - ADCIRC simulation, 8-2
 - environment, 2-15
 - FESWMS, *10-3*
 - geometry, 5-2, 9-2
 - HIVEL, 7-2
 - SED2D-WES, 6-2
 - WSPRO, *12-2*
- scalar
 - displaying, 2-8
- scatter point, *13-1*
 - active, 13-2
 - color, 13-4
 - convert, 4-9, 13-4
 - create, 13-2
 - data set, 13-2
 - display options, 13-3
 - editing, 13-3
 - file, 13-2
 - icon, 13-3
 - interpolation, *13-4*
 - numbers, 13-4
 - read, 13-2
 - saving to a file, 13-2
 - selecting, 13-3
 - set, 13-3
 - symbols, 13-4
- scour, 12-9
- sections
 - WSPRO, *12-3*
- SED2D-WES, *6-1*
 - BC Concentrations, 6-9
 - Boundary Condition, 6-9
 - data sets, 6-12
 - diffusion coefficients, 6-7
 - File, 6-1
 - Global Parameters, 6-2
 - initial concentration, 6-7
 - Model Check, 6-12
 - Model Control, 6-10
 - new, 6-2
 - open, 6-2
 - Print Control, 6-11
 - save, 6-2
- select
 - all, 2-19
 - by material type, *2-19*
 - element, 4-6
 - node, 4-2
 - nodestring, 4-4
 - pier, 4-8
 - scatter point, 13-3

- scatter point set, 13-3
 - with poly, 2-19
 - xy series point, 16-5
- settings
 - file, 2-10
- shading
 - color fill contours, 3-10
- shapefiles, 2-13
- simulation
 - file, 2-9
- smoth
 - nodestring, 4-32
- snap
 - to grid, 2-22
- solver
 - CGWAVE, 9-8
- spline, 3-10
- split, 4-7
- standard deviation, 3-5
- station output
 - ADCIRC, 8-8
- super file
 - SMS, 17-47
- symbol
 - scatter point, 13-4
- tension, 3-10
- Thiessen polygon, 13-13
- tides
 - ADCIRC, 8-13
- TIFF
 - import, 2-12
- time, 5-15
 - animation, 3-15
- time control
 - ADCIRC, 8-12
- TIN, 2-14
 - import, 2-12
- tool
 - general palette, 2-4
 - mesh palette, 4-2
 - scatter point palette, 13-3
- transform
 - nodes, 4-19
- transition
 - element, 5-21
- triangles
 - merge, 4-30
 - thin, 4-29
- triangulated irregular network, 2-14
- triangulation
 - Delauney, 13-13
 - mesh, 4-24
- truncation
 - interpolation, 13-5
- units, 5-16
 - HIVEL, 7-8
- Unix, 1-8
- vector
 - animation, 3-15
- velocity, 3-3
- viscosity
 - eddy, 5-9
- water surface elevation, 3-3
- WES, 7-1
- wind
 - ADCIRC, 8-10
- window
 - bounds, 2-21
 - edit, 2-1, 2-8
 - film loop, 3-14
 - help, 2-8
 - plot, 2-23
 - tool palette, 2-1
- Window
 - river, 12-2
- WSPRO, 12-1
 - bridge sections, 12-7
 - model check, 12-14
 - new, 12-2
 - open, 12-2
 - parameters, 12-12
 - records
 - DA, 12-9
 - DC, 12-9
 - HP, 12-6
 - roughness, 12-11
 - save, 12-2
 - scour, 12-9
 - section
 - cross sections, 12-5
 - guide bank, 12-11
 - sections, 12-3
- xy series
 - active, 16-2
 - compress, 16-3
 - creating point, 16-5
 - delete, 16-2
 - duplicate, 16-2
 - editing, 16-3
 - export, 16-2
 - file, 17-50
 - frame, 16-6
 - import, 16-2
 - insert, 16-3

interpolate, 16-3
list, 16-1
pan, 16-6
plot, 16-5
selectingpoint, 16-5
update, 16-3

xy options, 16-4
zoom, 16-5
XYZ
file, 17-53
import, 2-12
zoom, 2-6