

**CRWR Online Report 96-2**

**Modeling of Dissolved Oxygen in the Houston Ship Channel using WASP5 and  
Geographic Information Systems**

**by**

**Jennifer Benaman, B.S.C.E., M.S.C.E.  
Graduate Research Assistant**

**and**

**Neal E. Armstrong, Ph.D. and David R. Maidment, Ph.D.  
Co-Principal Investigators**

**August 1996**

**CENTER FOR RESEARCH IN WATER RESOURCES**

**Bureau of Engineering Research • The University of Texas at Austin  
J.J. Pickle Research Campus • Austin, TX 78712-4497**

**This document is available online via the World Wide Web at  
<http://www.ce.utexas.edu/centers/crwr/reports/online.html>**

## **Acknowledgments**

The authors would like to thank Dr. George Ward for his valuable technical assistance throughout this research. The advice of Dr. Edward Holley concerning the modeling of the Houston Ship Channel was appreciated. Ms. Karen Visnovsky, Mr. Larry Koenig, Mr. Richard Kiesling, and Mr. Paul LaWare are acknowledged for their help in the initial data collection. Mr. Tim Plover and Mrs. Laurie Wood assisted in the various computer problems and challenges that arose during this project. In addition, Mr. Robert Ambrose with the USEPA was helpful in debugging WASP5 input data. Dr. Maidment's research team: Ferdinand Hellweger, Pawel Mizgalewicz, Francisco Olivera, Seann Reed, Bill Saunders, and Zichuan Ye, are thanked for their help, humor, and support during the work.

This research was partially funded by the National Science Foundation under a Graduate Research Fellowship. Without this financial support, this research would not have been possible.

August 9, 1996

**Modeling of Dissolved Oxygen in the Houston Ship Channel using WASP5 and  
Geographic Information Systems**

by

Jennifer Benaman, M.S.E.

The University of Texas at Austin, 1996

CO-SUPERVISORS: Neal E. Armstrong and David R. Maidment

The goal of this research was the establishment of a connection of an existing water quality model to Geographic Information Systems (GIS). The water quality model investigated was the Water Quality Analysis Simulation Program (WASP5), while the actual linkage was performed using object-oriented and FORTRAN programming. The study area chosen to implement this GIS/WASP5 interface was the Upper Houston Ship Channel, which has a history of poor water quality caused by industrial inputs and non-point source loadings from the city of Houston. Initially, the information required for a simple steady state BOD/DO model in the Upper Houston Ship Channel was digitally represented in GIS. In addition, the non-point source loadings into the Houston Ship Channel were determined using a grid-based model developed in GIS. Afterwards, scripts were written in the ArcView programming language, Avenue, in order for ArcView to perform the following tasks: 1) write the input file information, 2) format the input information into the proper WASP5 file, 3) execute the WASP5 subprogram for BOD/DO modeling and model calibration, 4) process the WASP5 output, and 5) assist the user in output visualization. This connection is limited by model complexity, computer speed, and Avenue programming constraints.

## Table of Contents

Section	Page No.
List of Tables	ix
List of Figures	x
List of Procedures	xiii
1 Introduction	1
1.1 Background	1
1.2 Need for Study	4
1.3 Objective	8
1.4 Scope	9
1.5 Contribution to Knowledge	12
2 Literature Review	13
2.1 Introduction	13
2.2 Environmental Modeling using GIS	13
2.3 Previous GIS/WASP Model Connections	20
3 Data Sets	23
3.1 Data Formats	23
3.2 Obtaining a Digital Database	24
3.2.1 Map Projection	25
3.2.2 Digital Line Graphs	25
3.2.3 Digital Elevation Models	26
3.2.4 Land Use/Land Cover Files	29
3.2.5 Precipitation Data	37
3.2.6 USGS Gauge Stations and Flow Data	37
3.2.7 Tables and Coverages	39
4 Methodology	42
4.1 Study Area	43
4.2 Channel Segmentation	45
4.2.1 Previous Segmentation in the Houston Ship Channel	45
4.2.2 Segmentation Chosen	50
4.2.3 Segment Terminology in this Research	50
4.2.4 Segmentation in GIS	52
4.3 Non-point Source Loads	57
4.4 Point Source Loads	80
4.5 WASP5 Model Development	83
4.5.1 Introduction	83
4.5.2 Model Constants	85
4.5.3 Main Segment Characteristics	87
4.5.4 Boundary Segment Characteristics	92
4.5.5 Flow and Baseflow	95
4.5.6 Constituent Loading	99
4.5.7 Model Calibration	99
4.5.8 Model Runs	100



## Table of Contents (cont.)

Section		Page No.
	4.6 GIS/WASP5 Connection	102
	4.6.1 Introduction	102
	4.6.2 Programming Involved	102
	4.6.3 Menu Overview	104
	4.6.4 Tables Needed for Connection	107
	4.6.5 Coverages Needed for Connection	114
	4.6.6 Input File Creation and Modification	116
	4.6.7 Running the Model from ArcView	131
	4.6.8 Output File Processing	132
	4.6.9 Help File Creation	141
	4.6.10 Setting up the WASP5/ArcView Connection	141
5	Results and Discussion	144
	5.1 Rainfall/Runoff and Rainfall/Flow Correlation	144
	5.2 Watershed Delineation	148
	5.3 Non-point and Point Source Loadings	149
	5.4 Model Calibration	152
	5.5 BOD/DO Model Results	155
	5.6 Model Sensitivity	159
	5.7 WASP5/GIS Connection	162
	5.8 Output Presentation	167
6	Conclusions and Recommendations	173
Appendices		
A	Data Dictionary	178
B	Projection File: geoalb.prj	190
C	Programs, Macros, and Procedures	191
	1. Procedure and Macro for Importing Digital Line Graphs from CD ROM to Arc/Info	192
	2. Procedure to Obtain USGS Land Use/Land Cover Files from USEPA Ftp Site	194
	3. FORTRAN Program Used to Estimate Baseflow from Measured Flow	195
	4. FORTRAN Programs Used to Format Text Files for WASP5 Input File	196
	5. FORTRAN Programs Used to Format WASP5 Output for Avenue Processing	208
D	WASP5 Input Files	211
	1. Input File for WASP5 for Model Calibration, Created by Avenue Formatting Program (calgen.for)	212
	2. Input File for WASP5 BOD/DO Model, Created by Avenue and FORTRAN Formatting Program (outgen.for)	213

## Table of Contents (cont.)

Section		Page No.
E	Examples of Free Form Texts Files Generated by Avenue	216
F	Avenue Scripts Used for ArcView/WASP5 Connection	221
G	Help File for ArcView/WASP5 Connection	245
H	Houston Ship Channel Waster Quality Data	273
I	List of Acronyms and Nomenclature	276
	Bibliography	280
	Vita	285

## List of Tables

Table		Page No.
3-1	Digital Data used for GIS Database Development	23
3-2	Internet Addresses for Data Sources	25
3-3	USGS Albers Equal Area Projection Parameters	25
3-4	Anderson Land Classification System and Corresponding Newell, <i>et al.</i> (1992) Classifications.	30
3-5	USGS Subwatersheds which Correspond to the Upper Houston Ship Channel Study Area	31
3-6	USGS Gauge Stations used in Methodology	39
4-1	General Characteristics of the Main Segments used for the Modeling Effort	50
4-2	Start and End Points for Model Segmentation.	56
4-3	Delineated Areas of Segment Subwatersheds	65
4-4	USGS Watershed Areas Compared to Delineated Watershed Areas	67
4-5	Streamflow Gauges with Runoff, Precipitation, and Relative Runoff Coefficient	70
4-6	Runoff Coefficients Used for Relative Measure of Watershed Urbanization	73
4-7	Estimated Mean Concentration Values Used for Non-Point Source Loading	78
4-8	Non-point Source BOD Loading into Each Segment	80
4-9	Summary of BOD Discharged into Each Segment, from the Available Source Data	81
4-10	Physical Characteristics and Model Parameters for Upper HSC Main Segmentation	88
4-11	Hydrographic Segments Corresponding to Main Segmentation	89
4-12	Physical Characteristics and Model Parameters for Upper HSC Boundary Segmentation	93
4-13	Final Flow and Runoff Values for Main Segment	98
4-14	Final Steady State Loading Values for BOD	99
4-15	Cases Applied for BOD/DO Model in the Upper Houston Ship Channel	100
4-16	Scripts and Programs Created for ArcView/WASP5 Connection	105
4-17	Detailed Description of <i>BOD/DO Input Blocks</i> Menu Options	108
5-1	Comparison of Calculated to Measured Flow for Various Tributaries at Different Years	148
5-2	Comparison of BOD Non-Point Source Loading Results by USGS Watershed	150

## List of Figures

Figure		Page No.
1-1	Galveston Bay and the Houston Ship Channel.	5
2-1	Levels of GIS/model integration.	14
3-1	1:100,000 scale Digital Line Graph of the hydrography of the Galveston Bay area.	27
3-2	Digital Elevation Model (DEM) showing location of USGS quads.	28
3-3	USGS subwatershed area and delineated watershed for the Upper Houston Ship Channel.	32
3-4	1990 land use coverage with delineated watershed boundary.	33
3-5	1980 land use coverage that was used to fill in the missing areas not covered by the 1990 land use data.	35
3-6	Final land use coverage, consisting of joined 1980 and 1990 land use data.	36
3-7	Precipitation grid for January, averaged over the years 1961 - 1990.	38
3-8	USGS Flowing gauging stations used in study.	40
4-1	Study area, which includes the Upper Houston Ship Channel from the Turning Basin to the San Jacinto Monument and all land draining into this reach.	44
4-2	Single line representation of the Upper Houston Ship Channel used in GIS.	46
4-3	Upper Houston Ship Channel showing related TWC (now TNRCC) segments.	47
4-4	Ward and Armstrong (1992) hydrographic segmentation over study area.	48
4-5	Original segmentation used by 1971 Tracor modeling effort.	49
4-6	Final segmentation used, as depicted in GIS.	51
4-7	Arcs representing boundary segments for the water quality model.	53
4-8	Upper Houston Ship Channel showing start and end points used for the modeled reach.	55
4-9	Eight direction pour-point model used in Grid for watershed delineation.	59
4-10	Delineated watershed and streams from original DEM.	61
4-11	Comparison of streams delineated from "burned" DEM grid and 1:100,000 DLG of hydrography.	63
4-12	Segmentation with corresponding watersheds.	64
4-13	Study area showing USGS streamflow gauges.	66
4-14	Flow record for gauge 8073600, shows the estimated baseflow along with the measured streamflow.	68

<b>List of Figures (cont.)</b>		
<b>Figure</b>		<b>Page No.</b>
4-15	Streamflow records for nine gauges used in rainfall/runoff correlation.	71
4-16	Relation between ratio of mean annual runoff divided by mean annual precipitation to runoff coefficients for the same area, based on land use and standardized runoff coefficient table.	74
4-17	Final runoff distribution over watershed area.	76
4-18	Estimated Mean Concentrations for BOD distributed over the watershed area.	77
4-19	Final biochemical oxygen demand (BOD) non-point source loading distribution.	79
4-20	Point source dischargers along the Upper Houston Ship Channel.	82
4-21	Original graph from Espey, <i>et al.</i> (1971) showing dispersion (i.e. exchange) coefficient versus net freshwater flow.	90
4-22	Schematic of final segmentation used in Upper HSC model.	94
4-23	Relation between ratio of mean annual flow divided by mean annual precipitation to runoff coefficients for the same area, based on land use and standardized runoff coefficient table.	97
4-24	Concept behind GIS/WASP5 model connection.	103
4-25	Customized menus created to run Avenue scripts.	106
4-26	Flow accumulation tables needed in model connection.	110
4-27	<i>Boundary Segments</i> and <i>Main Segment Parameters</i> tables as they appear in ArcView.	112
4-28	<i>BOD Point Source Loading</i> table used for model connection.	113
4-29	ArcView view, entitled "Segmentation".	115
4-30	ArcView setup for model connection.	118
4-31	Examples of query boxes created by Avenue and shown during input file creation through ArcView.	119
4-32	Examples of various message boxes created by Avenue.	120
4-33	ArcView menu showing option to <i>Start Editing</i> .	122
4-34	The two dbf files which are created during the BOD/DO input block execution.	127
4-35	Query box showing the list of tables to choose, from which new charts and themes are created, and displaying the output options available to the user.	134
4-36	Table (dbf format) which is created during the output file processing of the WASP5 output.	135
4-37	Result of second choice in output file presentation options: chart of concentration vs. time for a given segment and chosen table.	137
4-38	Query box and chart for third output presentation option: concentration vs. segment number at a chosen time.	138

<b>List of Figures (cont.)</b>		
<b>Figure</b>		<b>Page No.</b>
4-39	New theme created by the fourth output presentation option.	139
4-40	Themes created by the fifth output viewing option.	140
4-41	Windows help document run by winhelp.exe and created to assist a user in the execution of the WASP5/ArcView connection.	142
5-1	Comparison of calculated runoff depths with measured runoff depths at each USGS gauge station.	145
5-2	Direct comparison of calculated runoff depths at same location as measured depths for each gauge.	146
5-3	Model calibration results.	153
5-4	GIS representation of model calibration results.	154
5-5	BOD and DO modeling results for average year and dry weather conditions.	156
5-6	GIS representation of DO modeling results for the Upper HSC.	157
5-7	GIS representation of BOD modeling results for the Upper HSC.	158
5-8	Comparison of Case I and Case III for BOD/DO Model	160
5-9	Comparison of Case I and Case IV for BOD/DO Model	161
5-10	One-to-one relation which exists between the main segmentation theme and its corresponding parameters table.	169
5-11	Layout showing the charting capabilities of ArcView.	170
5-12	Four views created from “movie” display option in the output presentation process from the GIS/WASP5 connection.	171
5-13	Layout illustrating the multiple display capabilities of ArcView.	172

<b>List of Procedures</b>		
<b>Procedure</b>		<b>Page No.</b>
3-1	Commands used to import USGS Digital Elevation Models into Arc/Info format.	29
3-2	Process used in Arc/Info to obtain a full coverage of the land use/land cover.	34
4-1	Process used to determine flowlength values for the start and end points of the segmentation.	54
4-2	Commands used to segment the single line representation of the Houston Ship Channel into eight arcs.	56
4-3	Procedure used to delineate a watershed from a DEM for a given outlet.	60
4-4	Procedure for "burning in" the DLG streams and delineating the corresponding watershed.	62
4-5	Commands used to develop segment subwatersheds.	65
4-6	Procedure used to determine the flow accumulation values of precipitation at each gauge station.	72
4-7	Procedure used to distribute average annual runoff over the entire watershed area.	75
4-8	Steps taken to establish BOD NPS loading over watershed area and into each segment.	80
4-9	Steps taken to determine total flow (mm/yr/hect) into each main segment.	96

# Chapter 1

## Introduction

In today's environmental engineering practice, many computer models have been developed which explain chemical and biological processes in lakes, estuaries, and rivers. In addition, advancements have been made in determining the pollutants which enter those water bodies. However, the connection between the spatially represented land surface pollutants to the in-stream processes has been relatively weak. Most water quality models concentrate just on modeling the system once the pollutants have reached the receiving waters, while many pollutant models never route the loadings into the water bodies to examine their effects. This lack of continuity presents a need of a method in order to link the spatially-based pollutant source characterization with the water quality modeling of the receiving waters. A link would provide an easier way to examine the cause and effect relationship which exists between these two areas. To establish such a connection, a system which can allow spatial representation of parameters, while still having the ability to retain tabular or one dimensional information is needed. With this type of system, the spatial modeling of the land surface to determine such parameters as non-point source pollutant loadings would be possible, along with the storage and manipulation of water quality modeling data. One computer programming and modeling environment which displays this relational capability is a Geographic Information System.

### 1.1 BACKGROUND

A Geographic Information System (GIS) is a computer system capable of spatially representing data on the land surface and linking additional data related to this spatial depiction, through tables and charts. In addition, GIS is used in the area of environmental modeling, by providing ease and accuracy in surface terrain representation, watershed delineation, precipitation data compilation, non-point source pollutant loading calculation, and other concepts related to environmental processes (Maidment, 1993; Saunders, 1996; Newell, *et al.*, 1992; Maidment, *et al.*, 1996). Consequently, GIS has emerged as more than just a viewing interface for engineers and city planners; it is a powerful modeling tool which



can help provide the knowledge necessary to make sound engineering and management decisions. Numerous studies have utilized GIS for non-point source loading assessments, groundwater and surface water modeling predictions, agricultural chemical concentration estimations, and water balance forecasting (Saunders, 1996, Newell, *et al.*, 1992; Maidment, *et al.*, 1996, Mizgalewicz, 1996; Ye, 1996). Additional research has been dedicated to illustrating the strength of GIS as a management tool and decision support system in environmental areas such as permitting and watershed management (Furst, *et al.*, 1993; Chen, *et al.*, 1995; Sternberg, 1996)

Besides developing models within GIS, other studies have focused on the feasibility of connecting current environmental models to this information system, in order to make data manipulation and output presentation easier. Recent research has linked environmental numerical models to GIS to improve the pre- and post-processing of the model data (Biesheuvel and Hemker, 1993; Brown, *et al.*, 1996; Rindahl, 1996; Stuart and Stocks, 1993). Particular attention has been given to water quality and quantity models, such as the Hydrologic Simulation Program FORTRAN (HSPF), the Water Quality Analysis Simulation Program (WASP), and various models from the Agricultural Research Service (ARS) (Al-Abed and Whiteley, 1995; DePinto, *et al.*, 1994; Geter, *et al.*, 1995).

The majority of these past studies have mainly used two different GIS computer software systems: Arc/Info and Geographic Resource Analysis System (GRASS). Both of these programs have become widely accepted as the operating software programs for the rapidly growing GIS market. The Environmental Research Systems Institute, Inc. (ESRI), which is the creator of Arc/Info, also has a counterpart to Arc/Info, called ArcView. Until recently, ArcView, with its user-friendly Windows-based atmosphere, was mainly used as a display tool for viewing GIS coverages and related tables. Arc/Info, on the other hand, was the "number-crunching" software which created the coverages and tables and performed the environmental modeling. Arc/Info is also the software which many of the aforementioned studies utilized for their GIS/numerical model link. However, new versions of ArcView have made it possible to perform modeling and the model linkage within this Windows-based

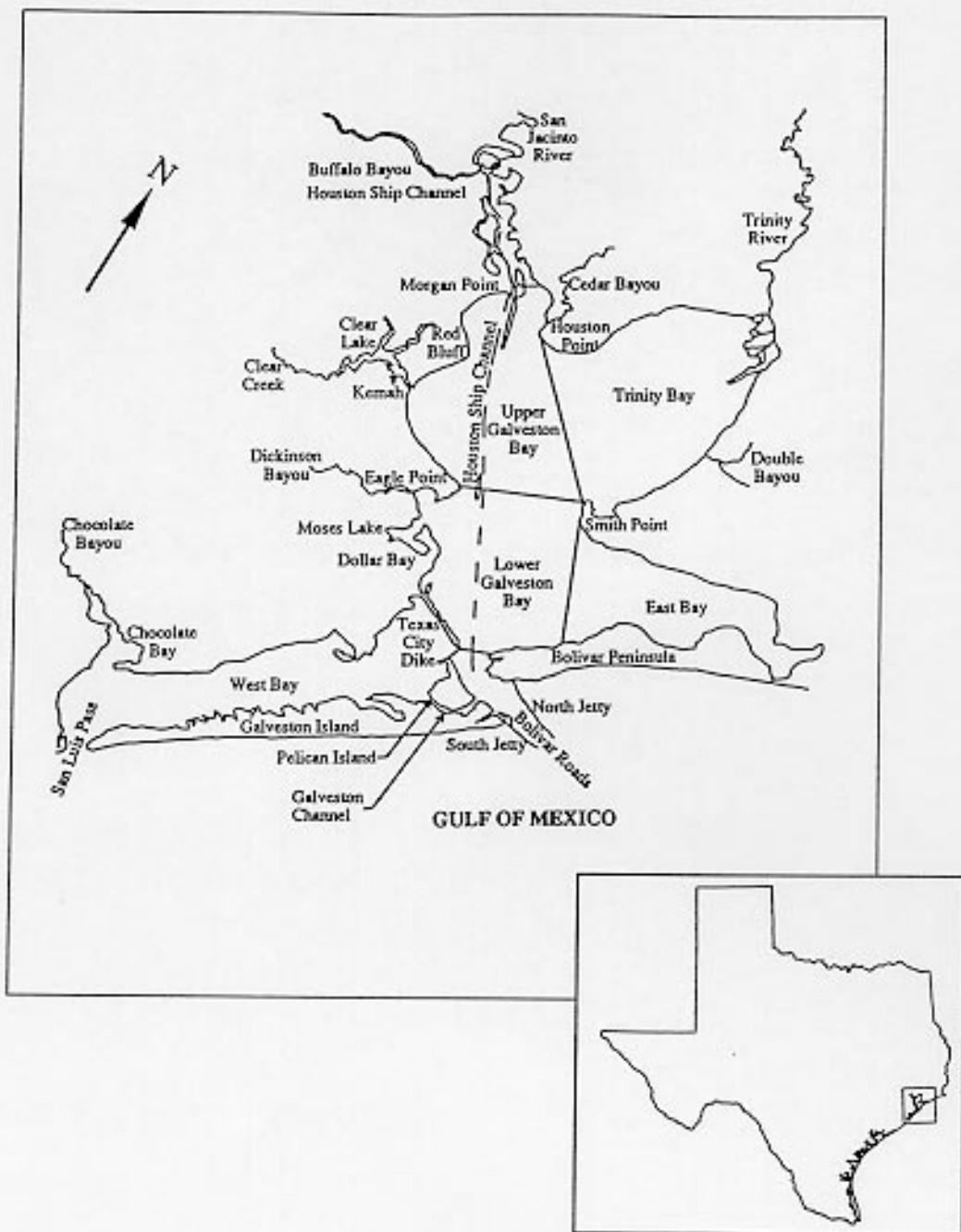
program. A relatively new programming language called Avenue allows the customization of an ArcView project to include mathematical functions, manipulation of tables and coverages, and the creation of new tables and coverages (ESRI, 1994). This new feature has made modeling and model connection through ArcView more feasible. Since this interface is more user-friendly and easier to learn, another user could very easily apply a "pre-programmed" customized project without having to relearn the programming or macro language of Arc/Info. However, general knowledge of GIS and the environmental model being used would still be necessary. The merit of this concept lies in the utilization of a GIS for watershed management in relation to investigating the benefits of engineering decisions.

To demonstrate the value of linking an environmental model to GIS, this research chose a study area which had a large amount of available data and an exhibited need for a modeling effort. In addition, the area had to display a need for future management decisions, based on environmental modeling and engineering practices. Once a GIS/model link is established, it can be enhanced to eventually be used as a management tool for engineers, city planners, and politicians, while also acting as a powerful communication tool to the general public. One area which fit these criteria is the Galveston Bay System, specifically, the Houston Ship Channel (HSC). This Bay was declared part of the National Estuary Program in 1988 by the U.S. Environmental Protection Agency (USEPA) (GBNEP, 1996). The result of this action was millions of dollars being invested into the Bay for research, data collection, and data compilation. The final product of this effort was the establishment of a comprehensive "plan" for the system, detailing the major problems associated with the Bay and outlining future management of the system (Shiple and Kiesling, 1994). These factors support the concept of using the Galveston Bay, or part of the Bay, as the study area for the connection of a water quality model to GIS. The next few sections give a brief overview of Galveston Bay, specifically the HSC; its water quality conditions, historically and presently; past water quality modeling efforts; and an outline of this project's overall objectives.

## 1.2 NEED FOR STUDY

Galveston Bay is a 1550 km<sup>2</sup> estuary located in Southeast Texas, on the Gulf of Mexico (see [Figure 1-1](#)). The estuary is home to the largest petrochemical complex in the nation (GBNEP, 1996). This profitable industry depends on the Houston Ship Channel, to provide a transportation route for ships and barges from the industrial hub of Houston to the Gulf of Mexico. The HSC, which begins at the Turning Basin, near the downstream end of the Buffalo Bayou, was first dredged around the year 1900 (Ward, 1993). After its starting point at the Turning Basin, the Channel travels in an easterly direction for approximately 25 km, where it then joins with the San Jacinto River. After the river, the HSC bends and continues in a south easterly direction, until meeting the main part of Galveston Bay at Morgan's Point. The dredged channel then travels through the Main Bay in a southern direction, where it finally reaches the Gulf of Mexico at the Bolivar Inlet, between the jetties located off of Pelican Island and the Bolivar Peninsula (see [Figure 1-1](#)). The channel averages about 10 m in depth, 1000 m in width and has a total length of approximately 85 km. The upstream 25 km of the Channel has an average discharge and velocity of about 25.5 m<sup>3</sup>/sec and 0.02 m/sec (Ward, 1993).

Galveston Bay is believed to have once provided a stable environment for a thriving ecosystem of marine plants and animals. However, since the 1950's more than 20% of the wetlands and 90% of the sea grasses have been lost (GBNEP, 1996). Point sources from the numerous petrochemical industries and non-point sources (NPS) from the urbanized area of Houston have both contributed to the poor water quality found in the Bay (GBNEP, 1996). The area of the Bay that has been most affected by these discharges is the 25 km section of the HSC between the Turning Basin and the San Jacinto River, termed the Upper Houston Ship Channel. Nearly 350 point source flows from industry eventually reach this part of the Channel. Of those 350 sources, approximately 70 discharge directly into the channel waters, while the rest are located along the tributary bayous (Visnovski, 1996). Besides the point sources, about 2600 km<sup>2</sup> of primarily urbanized land drains into this 25 km reach.



**Figure 1-1** Galveston Bay and the Houston Ship Channel (Clarke, 1993). Inset shows location of Galveston Bay on Texas coast.

Of particular concern, both historically and present day, is the dissolved oxygen deficiency that exists in the Upper HSC. It is generally accepted that a daily minimum average of 5 mg/L of dissolved oxygen (DO) is needed to support marine life (Thomann and Mueller, 1987). However, records have indicated, that DO levels in the upper HSC have historically been at or very near zero (Gloyna and Malina, 1964; Ward and Armstrong, 1992). The primary sink of DO is biochemical oxygen demand (BOD). Typically, wastewater treatment plants are the primary dischargers of this pollutant (Thomann and Mueller, 1987). Gloyna and Malina (1964) found that average BOD concentrations from sewage treatment plants which discharged into the channel ranged from 0 mg/L to 160 mg/L for monthly data in the 1930's, with the yearly average in 1933 being around 20 mg/L. The same study reported that by 1941, the average annual BOD from those same dischargers had reduced to about 8 mg/L, due to increased treatment practices. Measurements taken in the channel waters in 1951 showed the BOD daily averages at around 10 mg/L (Gloyna and Malina, 1964). Because of these dischargers and ongoing pollution problems, prior to the 1970's, the USEPA listed the HSC as one of the 10 most polluted bodies of water in the United States (USEPA, 1980).

Since the early 1970's millions of dollars have been invested in the HSC and Galveston Bay to help improve the water quality and reduce the pollutant loadings from municipal and industrial dischargers. Consequently, significant improvements in the DO concentration and BOD loadings have been seen in the Galveston Bay Area, especially in the HSC. Ward and Armstrong (1992) reported that the DO in the Upper HSC has increased by 4 mg/L since 1970. In addition, the same study reported that the BOD loading from the point sources has declined by a factor of 20, in the last two decades (Ward and Armstrong, 1992). In 1980, the USEPA singled out the HSC as the "most notable improvement" for Texas waterways in relation to its water quality (USEPA, 1980).

Although numerous improvements have been made, notable problems still exist. A recent study has indicated that the marine life in the Upper Houston Ship Channel was deficient due to lack of DO (Shipley and Kiesling, 1994). Although the point source

dischargers have improved their effluents, BOD loadings have also been attributed to the runoff originating in the urbanized areas of Houston and ultimately reaching the HSC. In the Upper HSC, both point source and NPS BOD are still a concern. Armstrong and Ward (1994) found that of all of the annual point source BOD loadings entering the entire Galveston Bay System (i.e. the five bay sections and the HSC shown in [Figure 1-1](#)), those loads that eventually reach just the Upper HSC account for a little over 50% of the total. In 1992, Newell, *et al.* listed the subwatersheds draining into the Upper HSC as the top concerns in relation to non-point source BOD loading.

Modeling efforts concerning the BOD/DO relationship and the HSC have been conducted in the past. In 1968, a study conducted by Hydroscience, Inc. developed one of the first mathematical models for the Channel to help explain the DO deficiency in the waters (Hydroscience, 1968). This early research set the foundation for a report published in 1971 by Tracor, Inc. In the Tracor report, the 1968 model was further refined to model the BOD/DO relationship in the Channel. The 1971 investigation concluded that portions of the Upper HSC experienced periods of zero DO (Espey, *et al.*, 1971). In 1984, the Texas Department of Water Resources (TDWR) conducted a full scale modeling effort on the channel, in order to evaluate waste loads entering the HSC (TDWR, 1984). In 1994, Espey, Huston, and Associates studied sanitary sewer overflow effects on the surface water quality of particular bayous of the Bay (EH&A, 1994). However, this study only looked at two particular bayous on the upstream end of the HSC and did not consider the entire system.

Although these models were accurate at the time performed, changes to the point source loadings and new knowledge of NPS loading has resulted in the need for a redevelopment of a water quality model in the Houston Ship Channel. In addition, with the continually changing area of Houston, a model which can be modified to reflect future conditions or possible management practices is a benefit when trying to project the effectiveness of a particular engineering decision, such as flow augmentation, land use changes, or effluent flow diversion. Much of the data on the HSC is available, but it does not exist in a easily attainable or reproducible form. By importing data related to the Channel,

such as point source discharge locations and loads, watershed land uses, projected runoff, and measured precipitation, into GIS, the information can be viewed and queried more easily. Once the data is formatted into GIS, the computer software can model important watershed characteristics, such as the areal distribution of NPS loads or the potential runoff over an area. These models could be standardized in GIS and used to depict the changes, if watershed characteristics were to be modified. In addition, a link established between GIS and a water quality model serves as a way of illustrating the effects of these watershed changes on the water quality of the natural system.

### **1.3 OBJECTIVE**

The objective of this research was to develop a two-way (forward and reverse) link between the GIS software, ArcView and the USEPA water quality model, WASP5. This initial link could serve as the groundwork for a decision support system within GIS, for describing and modeling possible changes in the natural system, given a modification of the current engineering practices or the environment. The test area for this link was the Upper Houston Ship Channel, which has a substantial amount of data from previous research to utilize the powerful database management system found within GIS. The area also supports a need for a method of evaluating the effects of a proposed engineering practice or watershed characteristic change on the water quality of the area. This overall objective was met by the following specific objectives:

1. Represent all data necessary to perform a BOD/DO modeling effort in GIS format. The data may take the form of tables, coverages, or grids within the GIS software, ArcView and Arc/Info.
2. Determine the non-point source loadings into the Upper Houston Ship Channel, using a method developed from the methods of Newell *et al.*, (1992) and Saunders (1996).

3. Develop a simple BOD/DO model for the Upper Houston Ship Channel, looking at steady-state conditions and considering both point and non-point source loadings while using the USEPA supported model, WASP5.
4. Establish a link between the data represented in GIS, and the water quality model, WASP5, by using the ArcView programming language, Avenue.

## 1.4 SCOPE

This project was completed in five phases: literature review, GIS data development, non-point source BOD loading determination, WASP5 modeling, and model connection to GIS. The following are specific tasks which were completed within each phase.

### *Phase I Literature Review*

This phase investigated previous modeling efforts performed in the study area, along with additional research conducted related to water quality and watershed management. Previous studies utilizing the connection of GIS with existing numerical models were also examined. The specific steps were:

1. Review literature on previous research and modeling efforts done in the Houston Ship Channel.
2. Review literature on point and non-point source loadings into the Houston Ship Channel.
3. Review literature on the uses of GIS in watershed management, water quality modeling, and linkage to numerical models.

### *Phase II GIS Data Development*

This phase was centered on the development of existing data in to a format that is representable in GIS. The majority of this step was concerned with the creation of coverages and grids to digitally and spatially represent known information, along with the generation of related data tables. The specific steps were:



1. Digitally represent the Upper Houston Ship Channel waterway, watersheds, tributaries, current water quality, and model segmentation.
2. Digitally represent the Upper Houston Ship Channel watershed, including surface terrain, watershed boundary, land use, and any other spatial data, such as areal precipitation.
3. Develop tables within GIS to represent channel characteristics and model parameters, such as depth, reaeration coefficients, and tributary flow. Link these tables to the coverages of the channel and its model segmentation.

*Phase III Non-point Source BOD Loading Determination*

Using the coverages, grids, and tables developed in Phase II, this step estimated the steady-state BOD loading from non-point sources. This phase followed a method developed by Saunders (1996), where land-use-based constituent concentrations were linked to projected runoff to obtain an areal distribution of NPS loading. The specific steps were:

1. Establish the average runoff per unit cell over the watershed area grid.
2. Determine the estimated mean concentrations (EMCs) for each land use in the land use coverage from Phase II.
3. Link the EMCs with the land use and, likewise, multiply by the areal runoff to obtain a NPS load of BOD for each unit cell
4. Route the areal loading over the watershed area to the Upper Houston Ship Channel and establish the total steady-state NPS loading to each model segment.

*Phase IV WASP5 Modeling*

This phase established the model parameters and channel characteristics needed to run WASP5. The model executed was actually EUTRO5, a WASP5 subprogram designed for dissolved oxygen and eutrophication modeling. A level one complexity for EUTRO5, involving simple Streeter-Phelps BOD/DO equations, was used. In addition, the level one complexity for WASP's subprogram TOXI5, was used to calibrate the model. The specific steps were:

1. Become familiar with WASP5 and its eutrophication sub-model, EUTRO5.
2. Establish the model segmentation and parameters to be used for the Upper Houston Ship Channel.
3. Calibrate the model with WASP5's subprogram TOXI5, using salinity as the conservative substance.
4. With the information established in Phases II and III, determine the total steady-state BOD loading (point and non-point sources) into each model segment
5. Perform steady-state modeling scenario.
6. Compare the model predictions of DO and BOD with the historical measurements.

*Phase V           GIS/WASP5 Connection*

The connection was established using ArcView's programming language, Avenue and simple formatting FORTRAN programs in order to write the necessary WASP5 input file from GIS coverages and tables. The model output was also read by FORTRAN programs and rewritten in a form that was readable by Avenue.

1. Connect the WASP5 model to GIS, by using Avenue programming and FORTRAN to produce the necessary formatted input files for the model from the ArcView coverages and tables.
2. Reformat the WASP5 output files, using FORTRAN, so that they may be imported back into GIS, through ArcView.
3. Display the model results, using Avenue programming to allow the user generate new tables, charts, and coverages.

These phases, although itemized here, were performed concurrently. For example, it would have been redundant to establish the location of the point sources into the segments by hand and then by GIS. Instead, that information was established through GIS and initially entered into the model by hand. Then, once the GIS/WASP5 connection was established, the manual input step was eliminated. The final product of this research was an established link between the WASP5 model, developed for the Upper HSC, and ArcView, using Avenue and

FORTTRAN. Although the initial link has some limitations, once established, it can be further enhanced as the information needed becomes available.

## **1.5 CONTRIBUTION TO KNOWLEDGE**

The completion of this project introduces some new concepts which are important in the area of water quality modeling, GIS, and the Houston Ship Channel. The primary contributions of this research to existing knowledge are:

- The establishment of an ArcView-based connection of models to GIS, rather than Arc/Info-based.
- The linkage of the a spatially represented land surface pollutant loading model to an in-stream water quality model.
- The assembly of a GIS database for the Houston Area, including, but not limited to, digital representation of land use, precipitation, runoff, water bodies, water quality parameters, and their related parameters.
- The development of a non-point source loading analysis for the Upper Houston Ship Channel, using GIS. Specifically, the establishment of a method for spatial distribution of runoff and flow over the watershed area, so that land-based estimated mean concentrations can be used to determine areal loads.

## **Chapter 2**

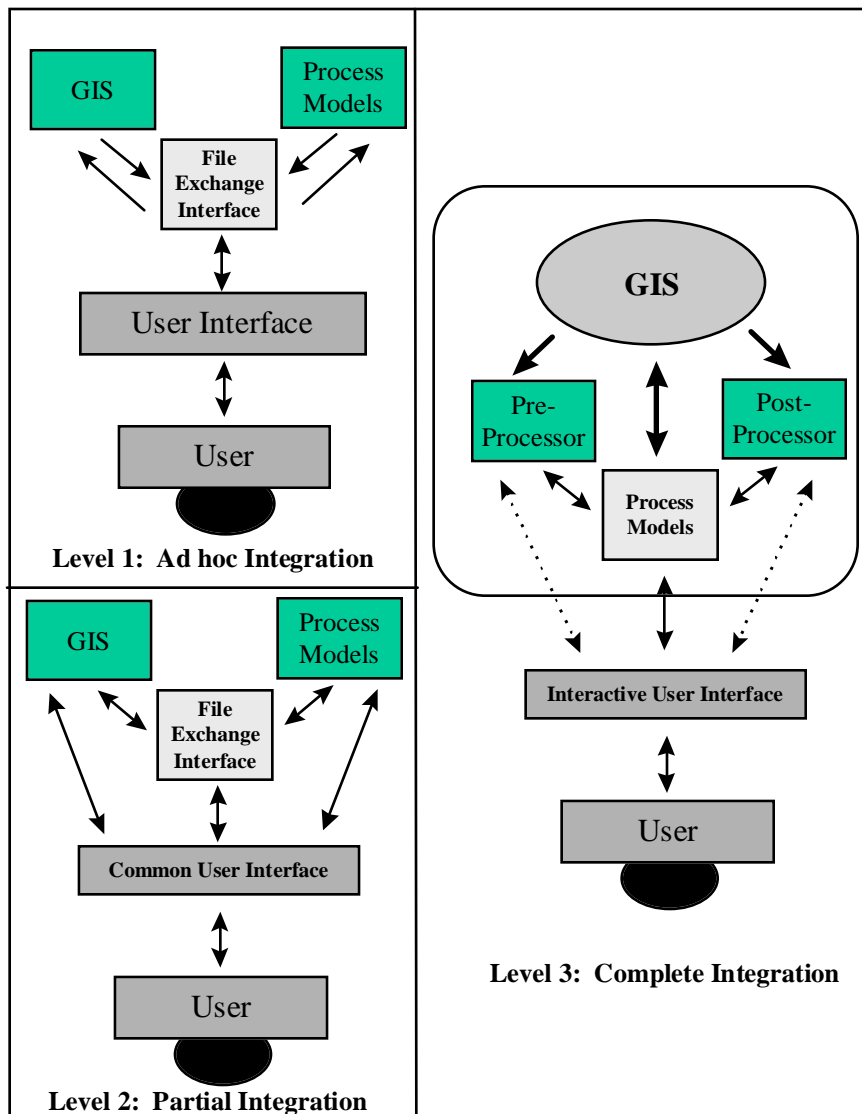
### **Literature Review**

#### **2.1 INTRODUCTION**

The purpose of this literature review is to provide the reader with a general overview of environmental modeling as it pertains to GIS. Recently, this concept has become a popular subject in many science and engineering fields. Within the past five years, many conferences have been hosted which primarily discuss research in environmental modeling and engineering practices using GIS. The first part of this chapter gives a brief description of past and current work that integrates modeling into GIS. Next, examples of existing models which have been connected to or utilize GIS in some fashion is presented. Finally, a brief overview of connections between GIS and the WASP5 water quality model is given.

#### **2.2 ENVIRONMENTAL MODELING USING GIS**

When examining previous research, attention should be given to the type of connection established between the environmental model and the GIS software. In addition, the type of GIS software used is a concern. Tim and Jolly (1994) present a good overview of three types of model interfaces possible with GIS (Figure 2-1). Their paper describes the three levels of integration as 1) ad hoc integration, 2) partial integration, and 3) complete integration. In the first level, the GIS data structure and environmental model are developed independently. The data is extracted from GIS, the model run separately, and the output analyzed at the user's discretion. The second level results in GIS playing more of an integrated role in the modeling. GIS supplies the data and then accepts the modeling results for processing and presentation. The third level consists of complete model development within the GIS software. The user has a single operating environment, where the data stored in the GIS is structured to meet the demands of the model and vice versa (Tim and Jolly, 1994). It should also be noted that there are numerous types of GIS software with which a model link can be accomplished. Most literature reviewed for this project utilized



**Figure 2-1** Levels of GIS/model integration.  
 Source: Tim and Jolly (1994)

the GIS software, Arc/Info and ArcView; although, some articles discussed below did accomplish links to other GIS software programs.

*Environmental Modeling Within GIS -- Level 3: Complete Integration*

For the past decade, the integration of GIS with environmental modeling has become an important research topic. The use of GIS for modeling provides ease and accuracy in the management and spatial representation of data. Recent projects which have conducted environmental modeling directly in the GIS have included studies in simulating hydrologic processes, water balancing a river basin, predicting chemical concentrations in rivers, and assessing non-point source loading over a watershed (Maidment, 1992; Stuart and Stocks, 1993; Olivera and Maidment, 1996; Ye, 1996; Maidment, *et al.*, 1996; Mizgalewicz, 1996; Saunders, 1996; Newell, *et al.*, 1992).

The area of hydrologic processes is one of the primary areas of study in the literature reviewed for this project. One application of hydrologic modeling in GIS incorporated a set of generic modeling tools within an unspecified GIS software to implement a semi-distributed hydrological model (Stuart and Stocks, 1993). A set of pre-established hydrologic equations, which explain surface saturation, were integrated into GIS with the macro language of the GIS software utilized. Spatial information needed for the equations, such as land slope, rainfall intensity, and soil transmissivity, was stored in the form of GIS coverages. By using a GIS tool interface, which incorporated the needed equations, along with the data coverages, the spatial distribution of the surface saturation was calculated over a study area (Stuart and Stocks, 1993).

Besides hydrologic processes related to land saturation, numerous sources were found pertaining to the concept of flow and transport modeling over the land surface. Maidment (1992) investigated concepts in hydrologic modeling with GIS, using the Arc/Info software, such as grid based watershed delineation, geographic representation of the stream network, and dynamic segmentation for hydrologic feature locations on this stream network.

This research proposed a “hybrid grid network” which consisted of the delineated watershed grid and its connected stream network created from the delineation process. The modeling of the spatially distributed hydrologic processes over the grid was combined with a model of flow and transport processes in the stream network to result in a network of linearly connected flow systems. The overall model was developed in GIS, while using basic hydrologic concepts and equations to explain the flow and transport parameters.

In an ongoing study, an application of flow process modeling uses the concepts of spatially distributed land surface parameters to calculate runoff over a watershed area (Olivera and Maidment, 1996). This research applies the concept of spatially distributed unit hydrographs over an area of cells. These unit responses, which are each independent of the surrounding responses, are convoluted along a watershed flow path to produce a total runoff response at the watershed outlet. This entire process is executed within the GIS software, Arc/Info and Arc/Info’s subprogram, Grid (Olivera and Maidment, 1996). In contrast, a slightly different approach to GIS surface water flow simulation is being conducted by Ye (1996). This model, which utilizes the newer GIS programming language, Avenue through the software, ArcView, is designed to simulate surface flow runoff based on the precipitation or soil moisture contents defined on the subwatersheds of a river basin. This research is not only investigating time varying flow processes over a watershed, but is also incorporating a groundwater flow model with the surface water flow model. Both of these models interact with the user through tool buttons in ArcView which activate Avenue programs (Ye, 1996).

Another environmental concept which has recently been integrated into GIS is water balancing over a river basin. Specifically, a water balance model has been developed for the Niger Basin in North Africa (Maidment, *et al.*, 1996). This model used a grid-based analysis to explain a relationship between flow, precipitation, evapotranspiration, water demand, and water storage. Similar water balancing efforts using GIS have been performed on Texas and the Souss Basin in Morocco (Olivera and Maidment, 1996).

Besides hydrologic processes, GIS has been used to assess pollutant loadings entering a water body and to explain the transport of chemicals in surface water. Various studies have investigated the concept of non-point source (NPS) loadings from watershed areas. Two projects in particular used GIS to develop projected areal loadings of different chemical constituents. Saunders (1996) developed a grid-based model which assessed NPS loadings of nitrogen, phosphorous, cadmium, and fecal coliforms into a small coastal bay from the 6000 km<sup>2</sup> San Antonio-Nueces Basin in South Texas. The method used a grid of land use-based estimated mean concentrations (EMCs) multiplied by spatially distributed runoff volumes to obtain an annual areal loading over the watershed (Saunders, 1996). A similar study also utilized the concept of EMCs and runoff volumes to develop an assessment of NPS loads into Galveston Bay, Texas (Newell, *et al.*, 1992). Newell, *et al.*(1992), also utilized GIS to spatially distribute runoff volumes, land use characteristics, EMCs, and final loading values. A slightly different pollutant study currently in progress is applying GIS to project chemical concentrations in the Upper Mississippi River Basin (Mizgalewicz, 1996). Using data collected in the United States Geological Survey (USGS) toxic chemical program throughout the Midwest, this GIS model is meant to explain the relationship between chemical concentrations in a stream and parameters such as chemical application, runoff, precipitation, season, and watershed characteristics. In addition, this research aims to describe chemical losses due to transport downstream using GIS as the ultimate modeling interface for these processes (Mizgalewicz, 1996).

#### *Model Connections to GIS -- Level 1: Ad hoc Integration and Level 2: Partial Integration*

As discussed in the previous section, various concepts have been developed within the GIS framework to assist in traditional environmental modeling. Of more concern for this research, though, is the establishment of a connection between an existing environmental model and the GIS software. Many research endeavors have investigated the feasibility of linking various models to GIS to assist in data management, manipulation, and output processing. Of particular interest for this project were those previous studies which concentrated on water quality and quantity model links. These projects have ranged from



incorporating an entire model into the GIS software, to concentrating on a subprogram of the model to connect to the interface.

In the area of groundwater, literature reviewed included links to the USGS model, MODFLOW, and a European simulation program entitled MICRO-FEM (Biesheuvel and Hemker, 1993. Brown, *et al.*, 1996; Rindahl, 1996). Two studies included investigations with MODFLOW. Rindahl (1996) established an "easy to use interface" through GIS to display drawdowns, stream flow, and aquifer elevations simulated from the groundwater modeling program. The research, which developed the link primarily for ease in output presentation, utilized the GIS software, ArcView 2.1 and ArcView's programming language, Avenue. The study established a polygon coverage of the modeled grid, attributed with the information resulting from a typical MODFLOW model run. The attributes were in tabular format and joined to the polygon coverage through a model "identification number" termed a "Loc-Tag". Once joined to the corresponding coverage, the model results could be spatially displayed through ArcView. Avenue scripts were also compiled which assisted the user in output display. Another study investigating a GIS link to MODFLOW centered on the use of Arc/Info to provide an efficient means of data preparation and visualization of simulation results (Brown, *et al.*, 1996). ModelGIS, an interface using FORTRAN 77 and Arc Macro Language (aml), generated model grids, model layer elevations, aquifer properties, surface water data, and model output. The options in ModelGIS were executed from a customized menu developed in Arc/Info's subprogram, ArcTools. The different choices converted coverages to model based data, assembled the data into MODFLOW input, executed the MODFLOW program, and assisted in the evaluation of the modeling results. Throughout these processes, the user interacted with the interface in the creation of the model grid and data inputs. This link also has the capability of establishing a three dimensional modeling surface (Brown, *et al.*, 1996).

The final GIS/groundwater model connection reviewed utilized the GIS software, Integrated Land and Water Information System (ILWIS) (Biesheuvel and Hemker, 1993). This study was primarily concerned with investigating the pre- and post-processing ability of

the GIS software with the European groundwater model, MICRO-FEM. The final product resulted in the establishment of a GIS/model link by allowing ILWIS to perform the following: 1) build a model network, 2) determine values for the model input, 3) combine results of the MICRO-FEM model run with other types of data, and 4) create background maps which can be used during the modeling process (Biesheuvel and Hemker, 1993).

More closely related to this current research are past studies which incorporated links to water quality models. Literature shows many different types of models and levels of connections being established, including a link for the widely used Hydrological Simulation Program (HSPF), as well as a connection with a number of selected equations, adapted from existing models (Al-Abed and Whiteley, 1995; Chen, *et al.*, 1995).

The HSPF model is used for simulation of watershed hydrology and water quality for a variety of pollutants. The model requires many parameters which describe watershed characteristics, including surface terrain, soils, land use, and vegetation. In addition, time series data for rainfall, temperature and other climatic attributes are needed to run HSPF. Al-Abed and Whiteley (1995) developed a loose connection to this watershed management model by using information stored in GIS coverages as the HSPF input. The research used the concept of look up tables and macros in Arc/Info to create spatial representations and tables of watershed characteristics. This spatial information was then entered into the HSPF model by the user (Al-Abed and Whiteley, 1995).

Two different articles relating to water quality modeling of agricultural chemicals were reviewed. One project discussed a GIS interface to four Agricultural Research Service (ARS) pollutant loading models: Agricultural Non-Point Source (AGNPS), A Basin Scale Simulation Model for Soil and Water Resources Management (SWRRBWQ), Erosion Productivity Impact Calculator (EPIC), Ground Water Loading Effects of Agricultural Management Systems (GLEAMS) (Geter, *et al.*, 1995). The goal of this research was to develop an interface which resulted in standardized and consistent input data to all of the water quality models, while providing a platform to interpret the model results through

tables, graphs, and maps. The user first enters the necessary model data in the form of attributed coverages within the GIS software, Geographic Resource Analysis System (GRASS). The total connection requires five raster based maps linked to sixteen attribute tables. This base information is then interpreted by the GRASS interface and consistent model input is determined. The link established actually writes the derived input into the formatted file necessary for the models' input, and the connection did provide a means for the user to view the model output through charts, tables, and raster maps (Geter, *et al.*, 1995).

The second agricultural chemical pollutant article (Tim and Jolly, 1994) was interested in demonstrating the concept of integrating the aforementioned water quality model, AGNPS, with an Arc/Info interface. GIS provided the means to generate and spatially organize the data needed for the non-point source modeling effort, while AGNPS was used to predict water quality related parameters such as soil erosion and sedimentation. A partial integration link was established (see [Figure 2-1](#)), by developing computer programs which provided "access points" between the GIS database, the AGNPS model, and the user. The link read the model input from raster coverages imported into Arc/Info's subprogram, Grid. Once the grid-based data were converted to a readable format by AGNPS and the model executed, the output was re-imported into Grid and displayed through ArcPlot (Tim and Jolly, 1994).

### **2.3 PREVIOUS GIS/WASP MODEL CONNECTIONS**

Research was conducted to review any connections with the WASP5 model which have been accomplished in the past. Findings indicated that one project utilized equations and concepts from WASP5 in a GIS connection, while another study connected WASP4's subprogram, EUTRO4, to the Arc/Info software (Chen, *et al.*, 1995; DePinto, *et al.*, 1993; DePinto, *et al.*, 1994).

The first study developed a model to evaluate the impact of land use and watershed management practices on the water quality of a Reservoir in Taiwan (Chen, *et al.*, 1995).

The article did not specify exactly which GIS software was utilized for the model development, but it did use equations and subprograms from the WASP5 program to model nutrient cycles. Geographic data was imported into the model from GIS and spatial attributes of some watershed and reservoir characteristics are entered through dialog boxes. Since the article discussed the use of the program in a Windows-based environment, ArcView with Avenue programming may have been used, but the accuracy of this assumption is unknown. The overall program, termed Integrated Watershed Management Model (IWMM), was run with menus. The menu choices activated object-oriented programs to execute the equations which modeled everything from reservoir hydraulics and hydrology, to water chemistry and nutrient cycles (Chen, *et al.*, 1995).

The second research project, conducted by the Great Lakes Program at the State University of New York at Buffalo and a water quality modeling firm, Limno-Tech, Inc, was the most developed connection to the WASP model found in this literature search (DePinto, *et al.*, 1993; DePinto, *et al.*, 1994). The result of this project was termed Geographically-based Watershed Analysis and Modeling System (GEO-WAMS). GEO-WAMS, which linked WASP4 (version 4 of the WASP model) to Arc/Info, performs the following functions: spatial and temporal exploratory analysis of system data; model scenario management; model input configuration; model input data editing; model input conversion to the proper file format; model output interpretation, reporting, and display; and model calibration, confirmation, and application. The interface also considered time variance in the system, along with possible three dimensional segmentation. In addition to the WASP4 connection, the system also performed a loading analysis with a spatial watershed model. All of these processes were conducted under a menu driven atmosphere created in Arc/Info and executed with Arc/Info's macro language (DePinto, *et al.*, 1993; DePinto, *et al.*, 1994). Although this connection could be termed a "partial integration" as discussed in Tim and Jolly (1994) (see [Figure 2-1](#)), the overall link from the model to Arc/Info was rather complex. The data necessary for the watershed loading model and WASP4 link included, but were not limited to, system geometry, morphometry, hydrology, soil properties, land-use, and point-source loadings, all in the form of GIS coverages, grids and tables. The research

used a portion of the Buffalo River as the study area for the prototype link and connected just WASP4's subprogram, EUTRO4, in order to model dissolved oxygen in the reach (DePinto, *et al.*, 1993; DePinto, *et al.*, 1994).

## Chapter 3

### Data Sets

#### 3.1 DATA FORMATS

For this project, numerous types of data were acquired in order to produce the model inputs and GIS coverages. Although each data set is discussed in more detail in later sections, [Table 3-1](#) shows the general data sets used in the methodology.

**Table 3-1 Digital Data used for GIS Database Development**

Data	Original Format	Source	Section(s) Discussed
Digital Line Graphs	Exported Coverage	USGS, 1993	3.2.2
Digital Elevation Models	Grid	USGS, 1996	3.2.3
Land Use/Land Cover	Exported Coverage	Newell, <i>et al.</i> , 1992 and USEPA, 1996	3.2.4
Precipitation	Grid	Daly, <i>et al.</i> , 1994	3.2.5
Flow Data	ASCII	Texas USGS, 1996	3.2.6
Point Source Dischargers	Paper, ASCII	Armstrong and Ward, 1994, and Visnovsky, 1996	4.4
Channel Segmentation Information	Paper	Espey, <i>et al.</i> , 1971	4.2
Water Quality Information	ASCII Computer File	Ward and Armstrong, 1992	5.4

In addition, [Table 3-1](#) also shows the particular section in which the data is further discussed, the original format of data, and the source from where the data was obtained. The data obtained was usually available in one of five different forms:

- Grids -- data which consisted of a spatial, uniform grid, with a measurement value applied to the center of each cell in the grid; imported into GIS using Arc/Info
- Coverages -- data obtained in an exported format (e00) and imported into GIS, using Arc/Info

- ASCII Files -- data, in a delimited format, obtained from the internet, an anonymous ftp site, or a personal acquisition from an organization
- Computer File -- data obtained in a software format (e.g., Microsoft Excel)
- Paper -- data which were obtained from a literature search or personal communication and keyed into the computer

Most of the data was imported into the computer and formatted for use in a PC or UNIX software environment. Subsequent sections provide further discussion on the manipulation of the data for use in a computer, specifically GIS, environment.

### **3.2 OBTAINING A DIGITAL DATABASE**

To utilize GIS, characteristics of the study area needed to be digitally represented in a format compatible with the GIS software, Arc/Info 7.03 and ArcView 3. These characteristics included, but were not limited to, surface terrain, streamflow, areal precipitation, land use, and area hydrography. A majority of this data was obtainable via internet. **Table 3-2** gives the internet addresses, current as of August 1996, for the data that was downloaded through the world wide web or file transfer protocol (ftp). The following sections describe how data for this digital representation were imported into GIS, using Arc/Info and ArcView. The result was a series of coverages, grids, and tables which contained information used in the digital representation of the study area. The attributes of many of the coverages, grids, and tables contained the parameters and input needed to run the water quality model. **Appendix A** contains a detailed list of all coverages, grids, and tables generated in GIS for this project.

**Table 3-2 Internet Addresses for Data Sources**

Data Source	Internet Address
Digital Line Graphs	<a href="http://sun1.cr.usgs.gov/eros-home.html">http://sun1.cr.usgs.gov/eros-home.html</a>
Digital Elevation Models	<a href="http://sun1.cr.usgs.gov/eros-home.html">http://sun1.cr.usgs.gov/eros-home.html</a>
Land Use/Land Cover Files	<a href="ftp://earth1.epa.gov">ftp://earth1.epa.gov</a>
Precipitation Data	<a href="ftp://fsl.orst.edu">ftp://fsl.orst.edu</a>
USGS Gauge Station Locations	<a href="http://txwww.cr.usgs.gov/cgi-bin/nwis1_server/">http://txwww.cr.usgs.gov/cgi-bin/nwis1_server/</a>
USGS Daily Discharge Values	<a href="http://txwww.cr.usgs.gov/cgi-bin/nwis1_server/">http://txwww.cr.usgs.gov/cgi-bin/nwis1_server/</a>

### 3.2.1 Map Projection

In order to represent the coverages and grids on a plane surface, a map projection was chosen. All of the coverages and grids for this were projected into USGS National Albers Equal Area Map Projection (USGS-Albers). USGS-Albers does not distort the area of the projected polygons. This quality is important in hydrologic modeling, since many characteristics of a watershed may be expressed per unit area. The parameters for this projection are found in [Table 3-3](#), while the projection file, `geoalb.prj` referred to in various procedures is found in [Appendix B](#). There are two datums listed in this table, World Geodetic System 1984 (WGS84) and North American Datum 1983 (NAD83). Since coverages initially existing in WGS72 cannot be projected into NAD83, WGS84 is used, instead. The errors involved with using NAD83 and WGS84 to spatially represent data in the same area are minimal.

**Table 3-3 USGS Albers Equal Area Projection Parameters**

Datum	World Geodetic System 1984 (WGS84) or North American Datum 1983 (NAD83)
Units of Length	Meters
First Standard Parallel	29° 30' 00" N
Second Standard parallel	45° 30' 00" N
Central Meridian	96° 00' 00" W
False Easting	0.0
False Northing	0.0

### 3.2.2 Digital Line Graphs

Digital Line Graphs (DLG) which are available from the US Geological Survey (USGS), are a 1:100,000 scale representation of the hydrography in the study area and surrounding watershed (USGS, 1993). The maps, which are organized by USGS 1° x 2°



quad name, show streams, rivers, creeks, canals, lakes, and shorelines (Figure 3-1). For this project, the quads: Houston, Conroe, Anahuac, and Beaumont encompassed the entire watershed area. The zipped files, titled by each quad name, were downloaded from CD ROM (USGS, 1993). The unzipped files consisted of up to eight separate 15' x 15' maps which, when joined together, constituted the hydrography for an entire 1° x 2° quad. Over 30 of these smaller coverages were joined together to form one large map of the Houston Area (Figure 3-1). A short program, written in Arc Macro Language (aml), developed by Saunders (1996) and shown in Appendix C-1, explains how to obtain the files from CD ROM, unzip the files in UNIX, import them into Arc/Info, erase the map borders, and join the entire set of 15' x 15' maps to form one large map.

### 3.2.3 Digital Elevation Models

Digital Elevation Models (DEM) are available via the internet from the USGS web site (USGS, 1996). One DEM file is a grid which covers a 1° x 1° surface area and contains an elevation value every 3" . The result is a 1201 x 1201 cell grid with 1,440,000 data points describing the surface terrain. Four 1:250,000 scale DEM grids were merged to include the Houston Ship Channel area: Houston-West, Seguin-East, Beaumont-West, Austin-East (see Figure 3-2). These grids are originally in geographic coordinates (latitude and longitudes in decimal degrees). Once imported into Arc/Info, they were projected into USGS-Albers. Procedure 3-1 annotates the process used to import these grids from downloaded form to a merged and projected grid. The final product was a digital representation of the elevation over the entire potential study area, projected to 100 m x 100 m cells. With a process developed in Arc/Info (Section 4.3), this surface terrain model was used to delineate the watershed area draining into the Upper Houston Ship Channel (Figure 3-2).

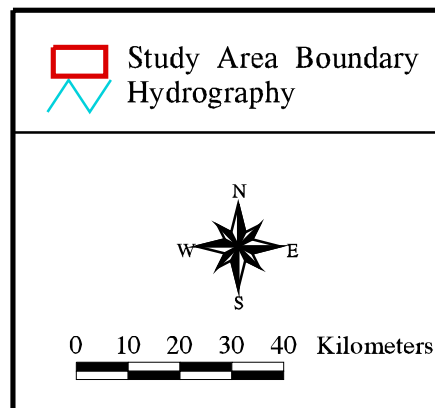
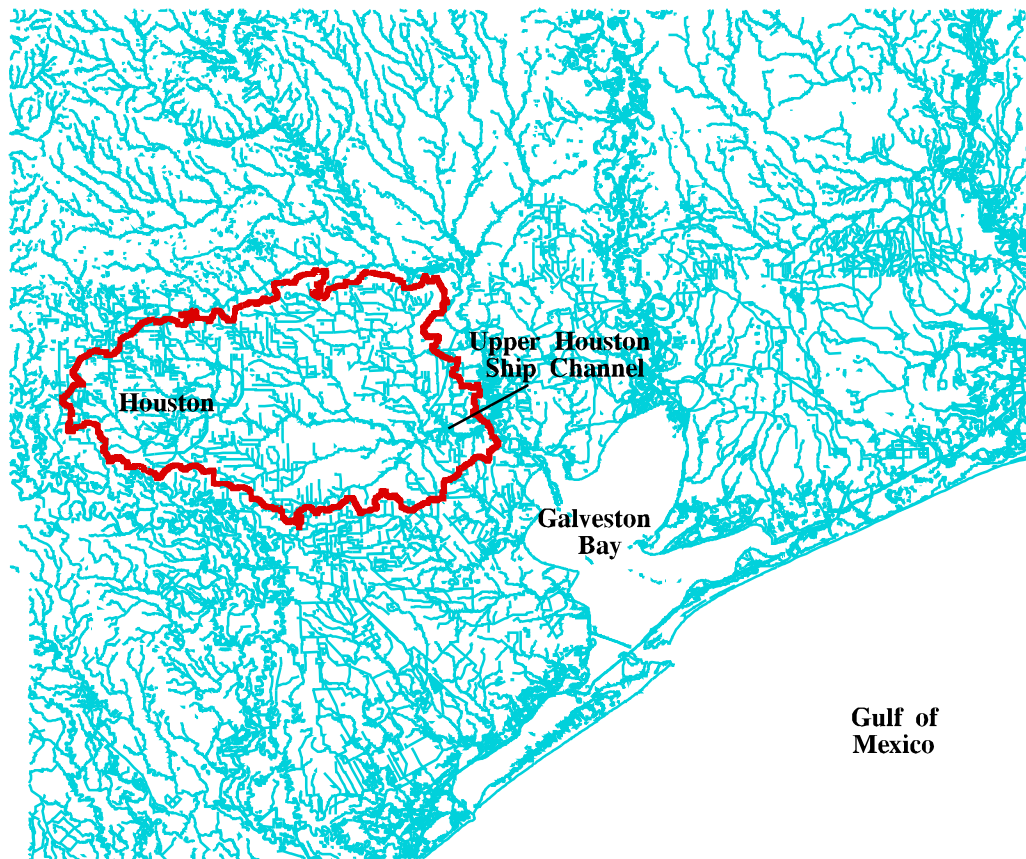


Figure 3-1 1:100,000 scale Digital Line Graph of the hydrography of the Galveston Bay area. Outline is the watershed of the study area which includes greater Houston area (USGS, 1993).

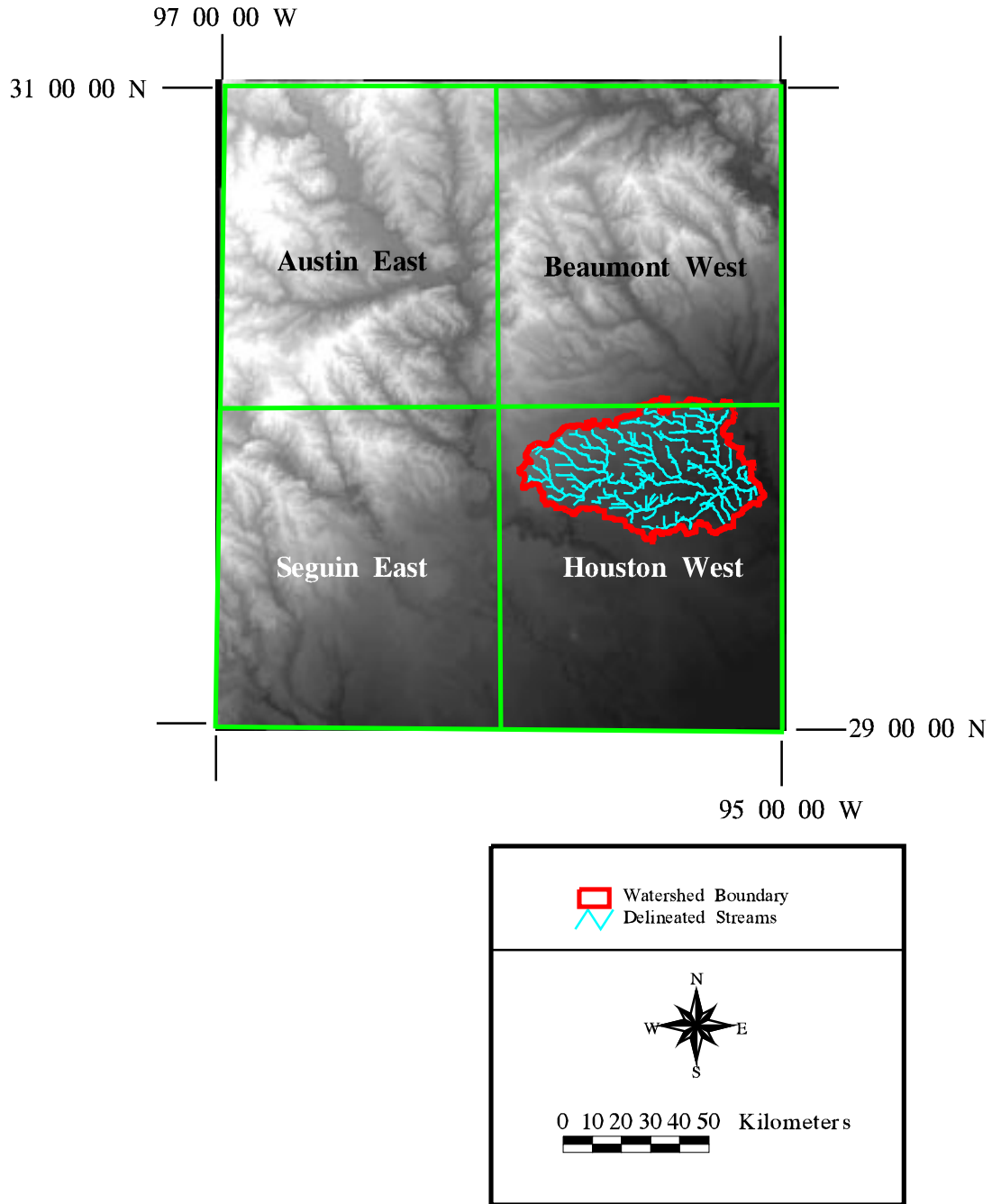


Figure 3-2 Digital Elevation Model (DEM) showing locations of USGS quads. Delineated subwatershed and streams are also shown (USGS, 1996).

**\$uncompress filename**

*uncompresses the file downloaded from USGS*

**\$dd if = <input> of = <output> cbs = 1024 conv = unblock**

*typed at c-shell prompt to put the downloaded DEM obtained from the USGS website into arc format*

**\$arc**

*invokes Arc/Info*

arc: **demlattice <input file> <output grid> usgs**

*converts the input file to a readable grid in geographic coordinates*

arc: **grid**

*invokes the Arc/Info subprogram, Grid*

grid: **<totalgrid> = merge ( <grid1>, <grid2>, .... )**

*merges all of the imported grids into one grid*

grid: **<projected grid> = project ( <totalgrid>, geoalb.prj, #, 100 )**

*projects the geographic coordinate grid into a planer map projection using a cell size of 100m; the parameters of which are located in the projection file*

**Procedure 3-1** Commands used to import USGS Digital Elevation Models into Arc/Info.

### 3.2.4 Land Use/Land Cover Files

Since land use is used in this research to determine non-point source loading to the Upper HSC, it was important to obtain an accurate and recent digital representation of the land use in the Houston area. Land use/land cover for an area is usually determined via interpretation of aerial photographs or satellite imagery pictures. The interpretation typically puts the land use into one of eight categories: urban or built-up land, rangeland, agricultural land, barren land, water, forest, wetlands, or tundra. This division system, called Anderson Classification, has two levels of organization (Anderson, *et al.*, 1976). Level I consists of the categories described above. Each of these general categories is further subdivided into more descriptive categories for Level II. For example, the “urban” category in Level I is separated into residential, commercial services, industrial, transportation /communication, industrial/commercial, mixed urban, and other urban for Level II (Table 3-4).

In 1992, Newell, *et al.* (1992) developed GIS land use coverages from 1990 LANDSAT satellite imagery data. These coverages, which incorporated the entire Galveston Bay watershed area, were obtained from Rice University in exported format via ftp (LaWare, 1996). Each USGS subwatershed, as defined in the 1992 report, constituted a separate land use coverage. Table 3-5 and Figure 3-3 show the USGS subwatersheds

**Table 3-4 Anderson Land Classification System and Corresponding Newell, *et al.* (1992)**

Classifications		
Anderson Level I Class	Anderson Level II Class	Newell, <i>et al.</i> (1992) Class
1 Urban or Built-Up Land	11 Residential	6 Residential
	12 Commercial Services	1 High Intensity Urban
	13 Industrial	
	14 Transportation/Communication	
	15 Industrial and Commercial	
	16 Mixed Urban or Built-Up land	
	17 Other Urban or Built-Up Land	
2 Agricultural	21 Cropland and Pasture	3 Agriculture
	22 Orchards, Groves, Vineyards	2 Open/Pasture
	23 Confined feeding Operations	
3 Rangeland	31 Herbaceous Rangeland	N/A
	32 Shrub and Brush Rangeland	
	33 Mixed Rangeland	
4 Forest Land	41 Deciduous Forest Land	8 Forest
	42 Evergreen Forest land	
	43 Mixed Forest Land	
5 Water	51 Streams and Canals	7 Water
	52 Lakes	
	53 Reservoirs	
	54 Bays and Estuaries	
6 Wetland	61 Forested Wetlands	5 Wetlands
	62 Nonforested Wetlands	
7 Barren Land	71 Dry Salt Flats	4 Barren
	72 Beaches	
	73 Sandy Areas	
	74 Bare Rock	
	75 Strip Mines, Quarries, Gravel Pits	
	76 Transitional Area	
	77 Mixed Barren Land	
8 Tundra	81 Shrub and Brush Tundra	N/A
	82 Herbaceous Tundra	
	83 Bare ground	
	84 Wet Tundra	
	85 Mixed Tundra	
	91 Perennial Snowfields	
	92 Glaciers	

Source: Newell, *et al.*, 1992 and Anderson, *et al.* 1976.

**Table 3-5 USGS Subwatersheds which Correspond to the Upper Houston Ship Channel Study Area**

USGS Subwatershed	Newell, <i>et al.</i> (1992) Abbreviation	Area (km <sup>2</sup> )
Addicks Reservoir	ad	347
Barker Reservoir	bk	316
Brays Bayou	br	329
Buffalo Bayou	bf	272
Greens Bayou	gr	539
Sims Bayou	sm	241
Ship Channel	sc	430
Whiteoak Bayou	wo	285
<u>Total:</u>		<u>2759</u>

Source: Newell, *et al.*, 1992

from the Newell, *et al.* (1992) report which lie in the watershed used in this project. There are some discrepancies in the digitized USGS watershed obtained from Newell, *et al.* (1992) and the delineated boundary in GIS. These differences are discussed further in [Chapter 5](#). To acquire a total coverage for the entire Upper HSC watershed, all of the subwatershed land use coverages were joined into one coverage with the *append* command so that the attributes of each coverage were preserved. [Figure 3-4](#) shows the watershed boundary and the 1990 land use coverage that resulted from the joining of the 8 subwatershed land use coverages. It should be noted that the Newell *et al.* (1992) study somewhat reclassified the traditional Level I Anderson System in the following ways:

- The Anderson Level I “urban or built-up land” classification was divided into “residential” (single family homes) and “high density urban” (all other Level I urban) subclasses.
- The Anderson Level I “agricultural land” was divided into an “agricultural” subclass (i.e., cultivated land) and an “open/pasture” subclass.

The final classification system used by the 1992 study is shown in [Table 3-4](#). Newell *et al.* (1992) considered the Level I classification sufficient, since no further subdivision enhances the non-point source loading calculations (Newell, *et al.*, 1992).

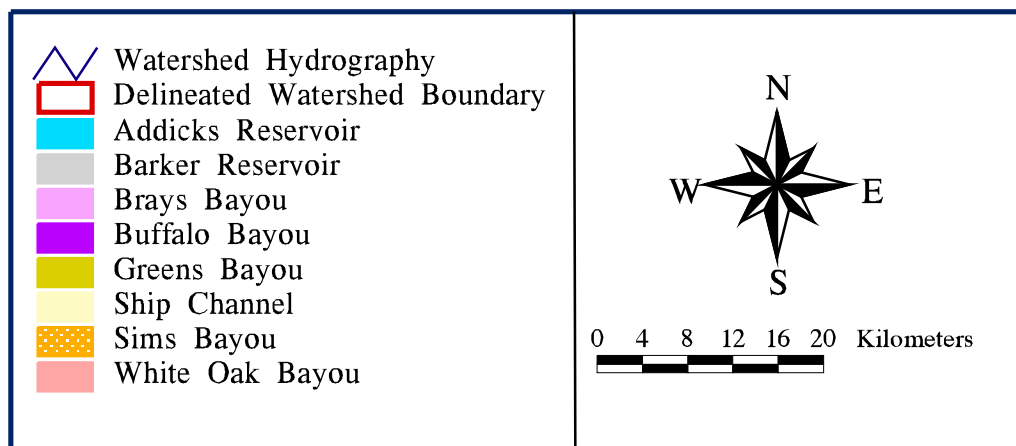
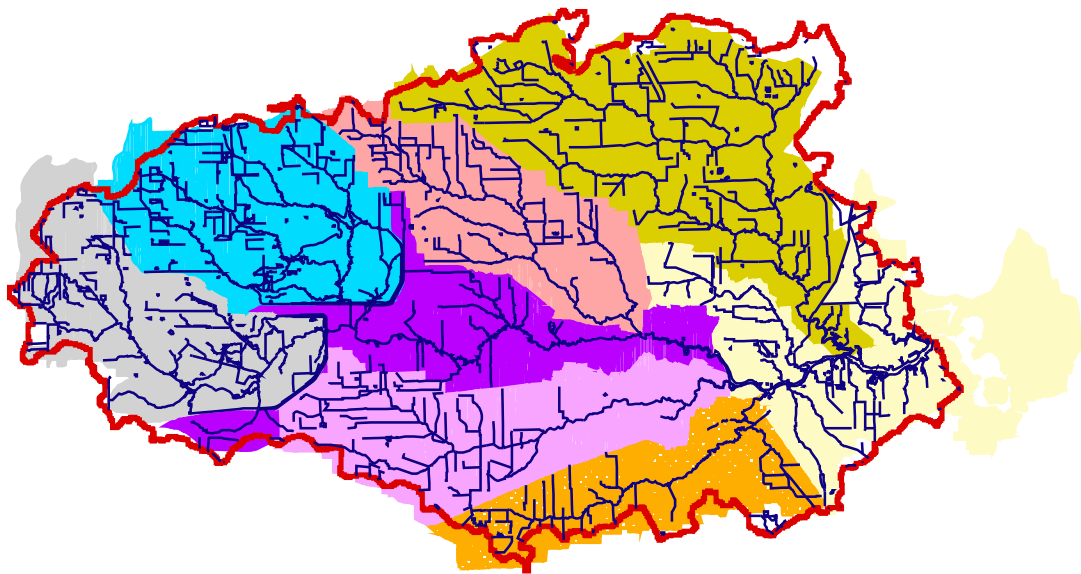


Figure 3-3 USGS subwatershed area and delineated watershed for the Upper Houston Ship Channel (Newell, et al., 1992).

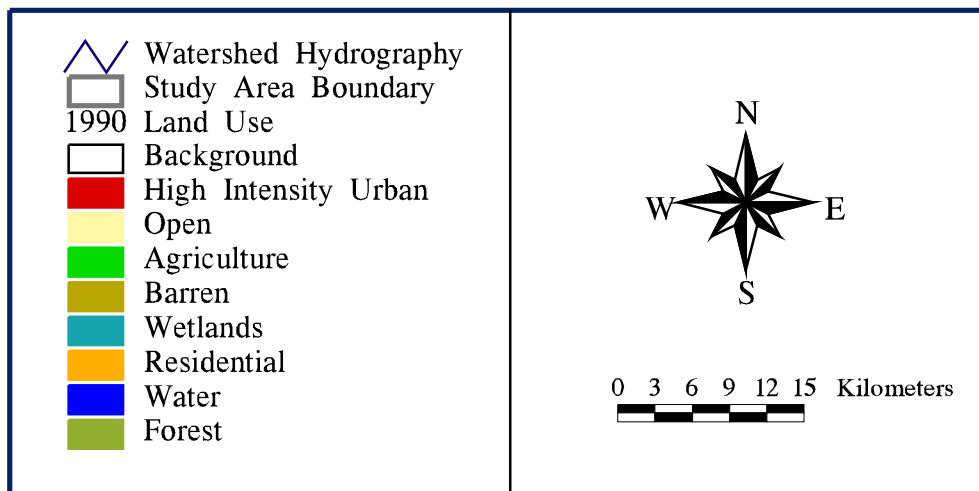
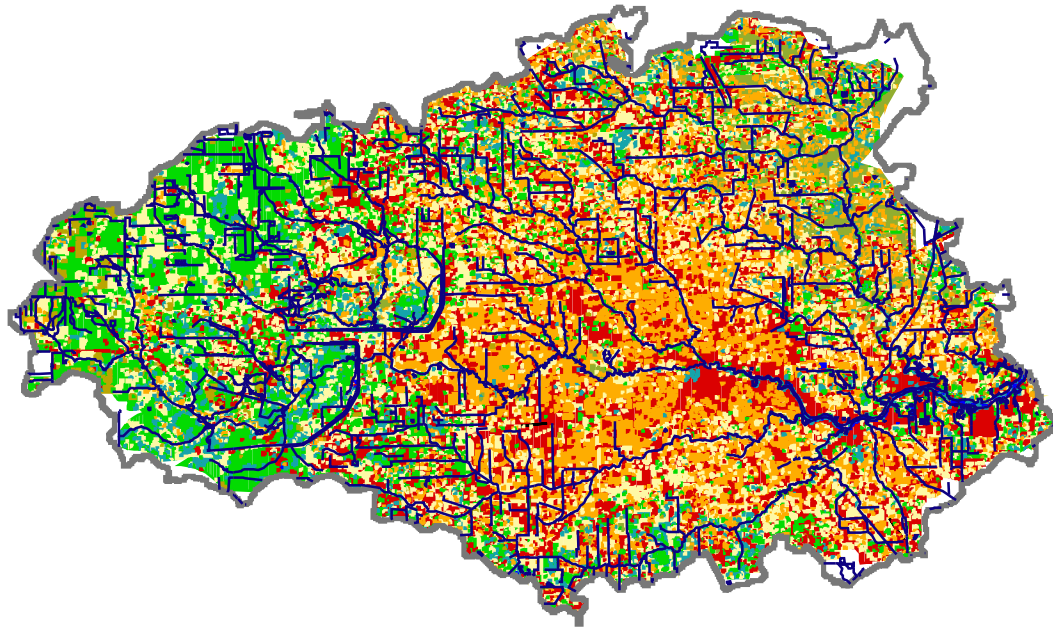


Figure 3-4 1990 land use coverage with delineated watershed boundary. Missing land use data is shown around edges of watershed boundary.



As shown in Figures 3-3 and 3-4, some outlying areas of the surface terrain (DEM) delineated watershed did not include the USGS subwatersheds. These gaps are probably due to minor errors that may have occurred in the digital delineation of the watershed or the digitization of the USGS subwatersheds by Newell, *et al.* (1992). These differences are further discussed in Section 5.2 Watershed Delineation. As a result, it was necessary to fill the gaps in the 1990 land use data, using 1980 data available from USGS. The 1:250,000 scale land use maps, organized by 1° x 2° quad name, were downloaded from the Environmental Protection Agency (USEPA) ftp site (USEPA, 1996). **Appendix C-2** provides a detailed description of the process used to determine the names of the needed files, download the files from the USEPA ftp site, and import them into Arc/Info. The quad maps were joined into one large map and the delineated watershed area was clipped from the map. The 1980 coverage was then “reclassified” so that the land use categories for the older attributes matched those categories used in the Newell, *et al.* (1992) report. The gaps missing from the 1990 data were determined by using the *erase* command in Arc/Info. The *erase* command used the 1990 land use coverage as a “cookie cutter” and deleted the information of the 1980 coverage that occupied the same area as the 1990 coverage (**Figure 3-5**). The 1990 land use coverage was then joined to the 1980 coverage which contained just the small missing areas of the delineated watershed, to obtain a total land use/land cover map for the entire watershed area (**Figure 3-6**). **Procedure 3-2** shows the process used in Arc/Info to create this coverage.

```
arc: import cover <filename.e00> <coveragename>
imports the 1980 land use coverages which are in exported format NOTE: If downloaded from USEPA
these coverages are already projected in USGS-Albers
arc: mapjoin lu_80
will create a coverage, lu_80, which consists of one map, encompassing all coverages entered by the
user, when prompted by Arc/Info
arc: dissolve lu_80 lu_80dis grid-code
arc: dissolve lu_90 lu_90dis grid-code
dissolves the coverage, lu_80 (or lu_90), by getting rid of any arcs that may exist between the adjacent
polygons with the same value in the attribute, grid-code (the land use code); the final coverage is
called lu_80dis (or lu_90dis)
arc: clip lu_80dis covtotshd lu_80shd poly
arc: clip lu_90dis covtotshd lu_90shd poly
```

**Procedure 3-2** Process used in Arc/Info to obtain a full coverage of the land use/land cover.

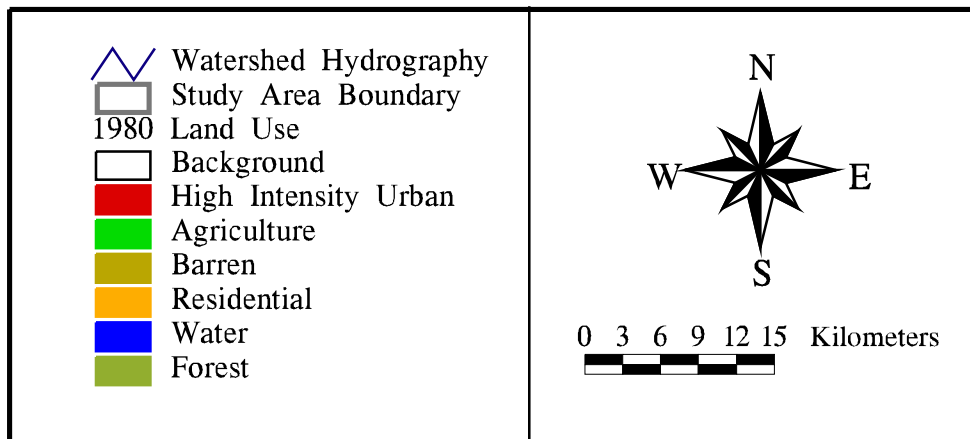
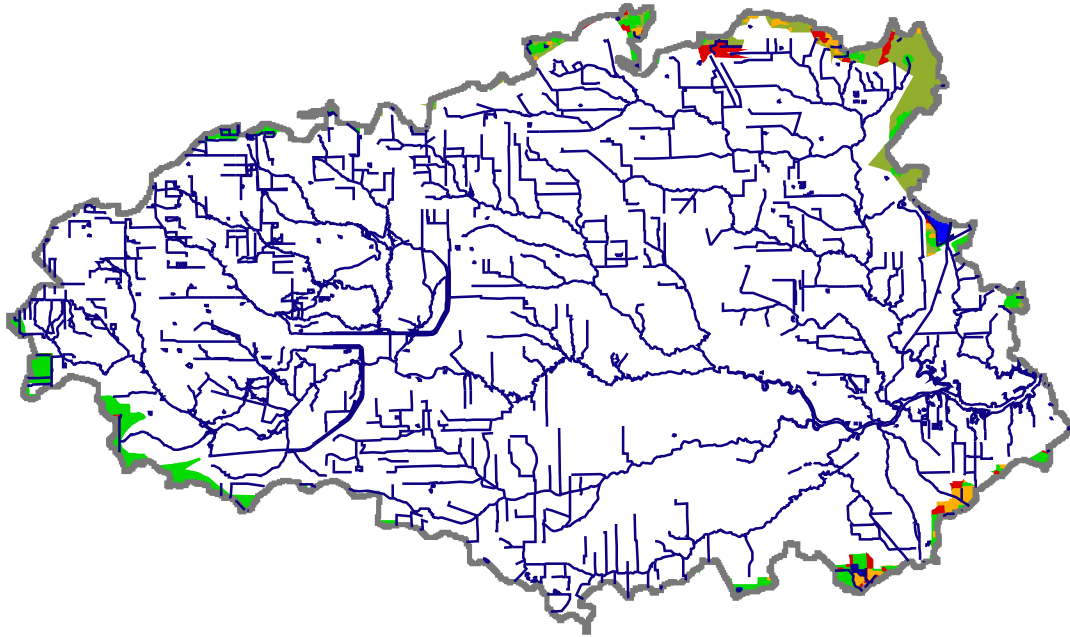


Figure 3-5 1980 land use coverage that was used to fill in the missing areas not covered by the 1990 land use data. Not that the "open" and "wetland" land uses did not exist in these areas (USGS, 1996).

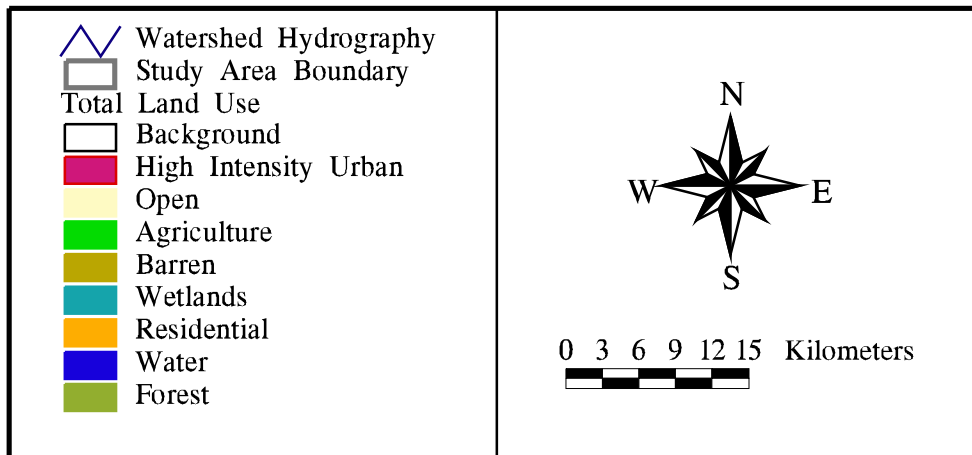
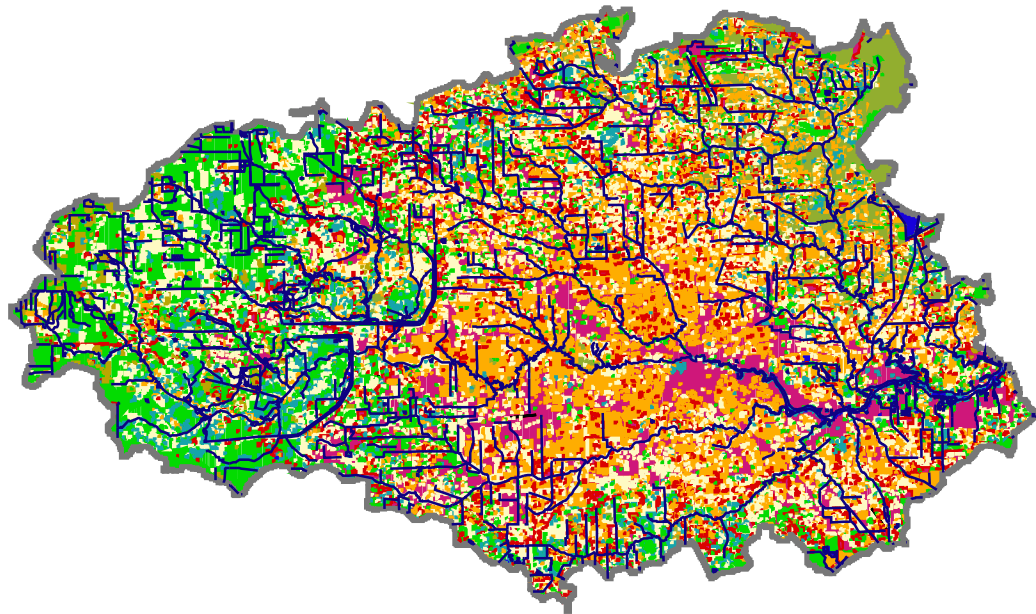


Figure 3-6 Final land use coverage, consisting of joined 1980 and 1990 land use data.

*clips the land use coverages with the watershed boundary so that only the land use within the delineated watershed are in the final coverage lu\_80shd (or lu\_90shd)*

arc: **erase lu\_80shd lu\_90shd lu\_80edge poly**

*uses the 1990 land use coverage as a cookie cutter to delete the information in the 1980 land use coverage that occupies the same area; the result is a land use coverage of just the missing areas on the edge of delineated watershed from the 1990 land use coverage*

arc: **mapjoin hsclu**

*command used to join two or more polygon coverages into one, while maintaining the attributes of each coverage in the attribute table; in this case, the coverages lu\_80edge and lu\_90shd were joined to result in hsclu*

**Procedure 3-2 (cont.)** Process used in Arc/Info to obtain a full coverage of the land use/land cover over the entire watershed area.

### 3.2.5 Precipitation Data

The Department of Agriculture, Natural Resources Conservation Service (formerly the Soil Conservation Service) has begun an effort to develop precipitation data for the entire United States. This study has produced a 2.5' x 2.5' cell grid of precipitation data over the US. There are 13 grids, consisting of precipitation values for each month (January through December) and one annual grid, averaged over the years of 1961 to 1990 (Daly, *et al.*, 1994). These grid were downloaded from the Oregon State University anonymous ftp site (see [Table 3-2](#)). From these grids, the Houston Ship Channel watershed was “clipped” from the national grids using a command called *set window* in Arc/Info’s subprogram, Grid. This command allows one to chose a smaller area from a larger grid by manually drawing a box around the area of concern. These 13 smaller grids were projected into Albers and their resolution increased to 100 m x 100 m cells, in order to match the DEM cell size. [Figure 3-7](#) illustrates an example of one of these grids. [Sections 4.3](#) and [4.5.5](#) discuss the use of these grids in the rainfall/runoff and rainfall/flow relationships development.

### 3.2.6 USGS Gauge Stations and Flow Data

To obtain a representative record of the flow for this watershed area, data from the USGS was utilized. The USGS office in Texas (Texas USGS) maintains flow gauging

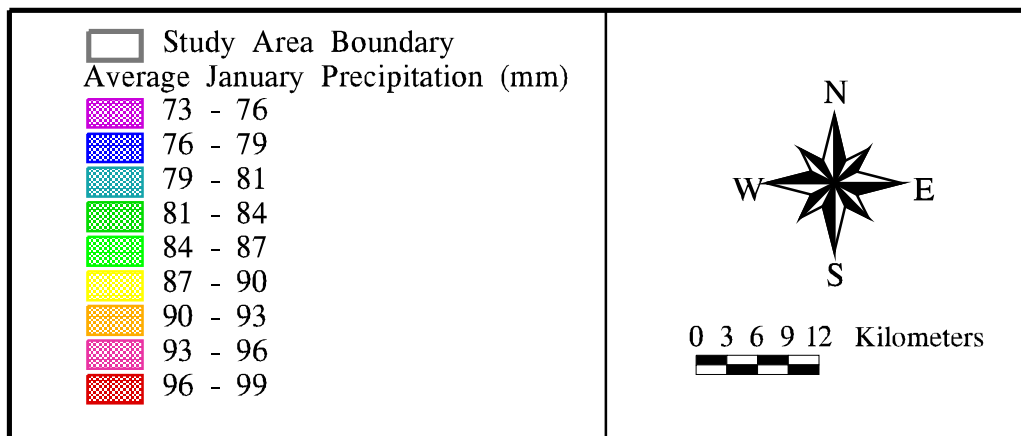
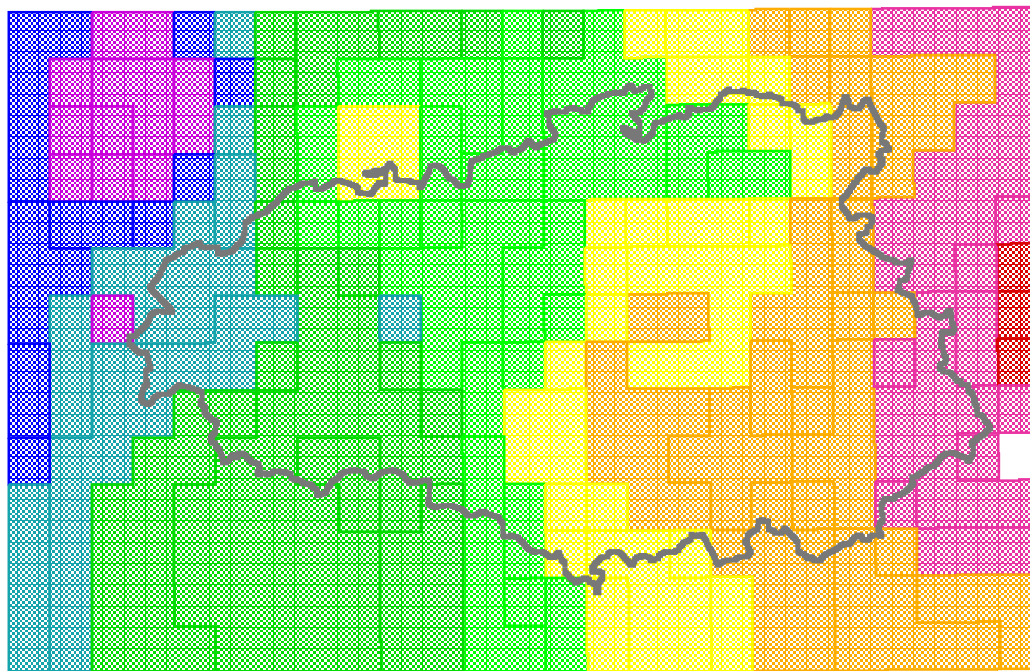


Figure 3-7 Precipitation grid for January, averaged over the years 1961-1990 (Daly, et al., 1994).

stations across the United States, some of which have continuous flow data in daily averages, as early as 1924 (Texas USGS, 1996). In the Upper HSC watershed area, there were a total of 37 possible gauging stations. Of those 37, ten were chosen for their continuous periods of record and locations (Figure 3-8). Table 3-6 shows the characteristics of each of those stations.

**Table 3-6 USGS Gauge Stations used in Methodology**

USGS Gauge Number	USGS Name	USGS Drainage Area (km <sup>2</sup> )	Location (DMS)		Period of Record
			Lat	Long	
8072730	Bear Creek near Barker Reservoir	56	29° 49' 52" N	95° 41' 13" W	07/01/77 - pr.
8073600	Buffalo Bayou at West Belt Drive	795	29° 45' 43" N	95° 33' 29" W	09/01/71 - pr.
8074500	White Oak Bayou at Houston	224	29° 46' 30" N	95° 23' 49" W	06/01/36 - pr.
8075000	Brays Bayou at Houston	246	29° 41' 49" N	95° 24' 43" W	06/01/36 - pr.
8075500	Sims Bayou at Houston	163	29° 40' 26" N	95° 17' 20" W	10/01/52 - pr.
8075730	Vince Bayou at Pasadena	21	29° 41' 38" N	95° 12' 58" W	10/01/71 - pr.
8075770	Hunting Bayou at IH 610	42	29° 47' 35" N	95° 16' 05" W	05/01/64 - pr.
8076000	Greens Bayou near Houston	178	29° 55' 05" N	95° 18' 25" W	10/01/52 - pr.
8076500	Halls Bayou at Houston	74	29° 51' 43" N	95° 20' 06" W	10/01/52 - pr.
8075900 *	Greens Bayou near Hwy 75	95	29° 34' 30" N	95° 25' 05" W	08/03/65 - pr.

\* Used for rainfall/runoff relationship verification -- not used in actual relationship development  
*Source: Texas USGS, 1996*

For each of these ten gauging stations, the flow data for 1961 to 1990, in mean daily averages, was downloaded from the website (Texas USGS, 1996). If the entire 30 years of data was unavailable, the period of record on file was obtained. This period of record was chosen to match the precipitation data's period of record (1961-1990), so an accurate rainfall to runoff relationship could be developed. In addition, a point coverage of the ten stations was developed from the location information given with each gauge (Figure 3-8). The watershed delineation process used for these stations is discussed in Section 4.3.

### 3.2.7 Tables and Coverages

Throughout the project, additional tables and coverages were needed to complete the representation of the study area in GIS. However, since the creation of these coverages and

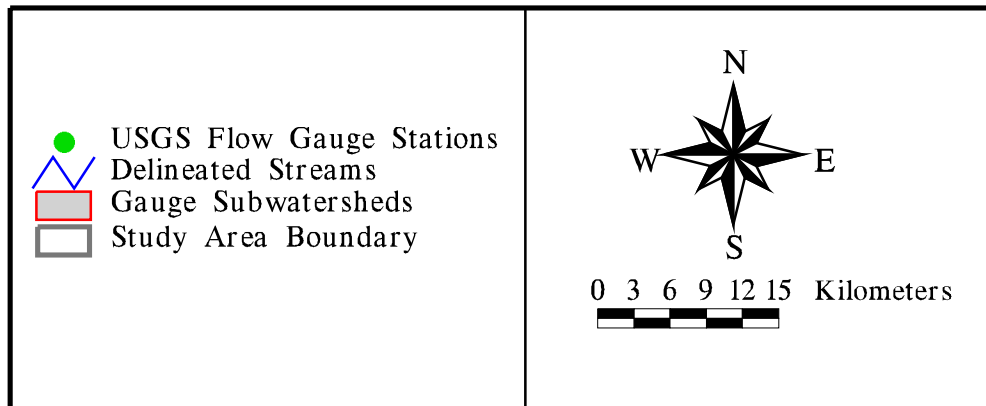
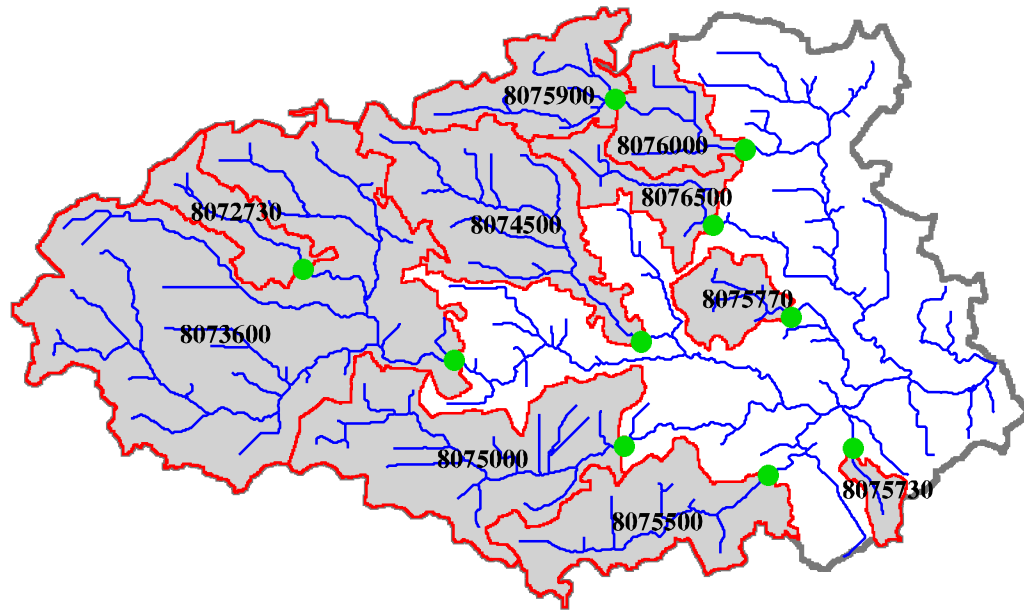


Figure 3-8 USGS flow gauging stations used in study (Texas USGS, 1996). Corresponding subwatersheds have been delineated using 3" Digital Elevation Model (DEM).

tables varied in relation to their use, their development is best discussed in detail within the methodology chapter ([Chapter 4](#)). A brief summary of the different data is given below:

- *Point Source Dischargers* -- The primary industries discharging into each water quality segment were determined from locations provided by the Texas Natural Resource Conservation Commission (TNRCC) (Visnovsky., 1996). The result was a point coverage of this information linked to another table, containing the actual discharge values for each constituent of interest.
- *Channel Segment Information* -- The segmentation utilized in a 1971 Tracor report (Espey, *et al.*, 1971) was duplicated in GIS. A table of the parameters necessary for the modeling effort was then “attached” to the segmentation coverage.
- *Water Quality Information* -- The current water quality in the channel was spatially represented in GIS by creating a polygon coverage of the water quality results for each hydrographic segment described in Ward and Armstrong (1992).



## Chapter 4

### Methodology

This chapter is divided into six main sections, with each section discussing a particular step taken in the methodology. The first five sections are dedicated to developing the data needed to perform four BOD/DO modeling runs in the Upper HSC, while the final section describes the procedure used to establish the GIS/WASP5 model connection. The first section introduces the study area by describing the Upper Houston Ship Channel and its contributing watershed. The second section provides a brief overview of water quality segmentation, previous segmentations of the HSC, and the segments used for the water quality modeling performed in this study. In addition, this section introduces the terminology of *main segment* and *boundary segment* in relation to their use in this research.

**Section 4.3** gives a procedure for the calculation of the non-point source loadings entering the Upper HSC. This procedure uses land use-based estimated mean concentrations, with spatially distributed runoff volumes to result in a areal loading of BOD to the Channel from the watershed. This section also presents a method of estimating baseflow, given a daily flow record and spatially distributing that runoff, using existing data and land use characteristics. It is important to note that the loading calculation performed in this section assumes that the non-point source load is only transported by the runoff volume. The following section (**Section 4.4**) discusses the point source loadings to the modeled reach. Only those dischargers located along the Upper HSC shoreline were considered in the point source loading determination. In contrast to the non-point source loads, the point source loads were assumed to be transported by the channel baseflow, which is discussed further in **Section 4.5**.

**Section 4.5** presents the WASP5 model development, including determination of model constants, estimation of water quality segment parameters, and execution of the model calibration and model runs. Within the section, the channel flow, which is needed for the

WASP5 model, is discussed. This flow is spatially distributed in the same way as the runoff was in [Section 4.3](#). The final flow values for each segment, along with the corresponding runoff values from [Section 4.3](#), are used to determine the baseflow in each water quality segment. This baseflow is necessary when looking at dry weather conditions, where just point source loads are entering the system (i.e., no runoff or non-point source loadings). The four modeling cases are also presented in this section: an average year case, a dry weather condition, and two cases to test model sensitivity.

Finally, [Section 4.6](#) discusses the WASP5/GIS model connection through the software ArcView, while using Avenue and FORTRAN programming. The discussion presents the menus created in ArcView to execute the Avenue scripts which read and write the model input information from tables and coverages. This section also gives a step-by-step procedure, along with an outline of the necessary tables and coverages, which is used to run the model connection. This section also contains an overview of the WASP5 input blocks as they relate to the interface, including assumptions and defaults set in the creation of the input file.

## **4.1 STUDY AREA**

The model developed incorporated the Upper Houston Ship Channel and all land draining into this section of the channel ([Figure 4-1](#)). [Figure 4-1](#) also shows the major highway systems in the Houston area for reference. The western boundary for the Upper HSC, the Turning Basin, receives the majority of its input from Buffalo Bayou, whose watershed is primarily covered by the greater metropolitan area of Houston. This water reach then travels east, receiving water and loadings from the Brays, Sims, Berry, Green, Hall, Carpenter, and Vince Bayous, and draining about 2600 km<sup>2</sup> of land. The San Jacinto Monument creates the eastern boundary of the 25 km section studied and is located just west of the confluence of the San Jacinto River. For digital representation, the channel was depicted in GIS two ways: 1) USGS Digital Line Graph (DLG); and 2) a segmented line drawn down the centerline of the channel.

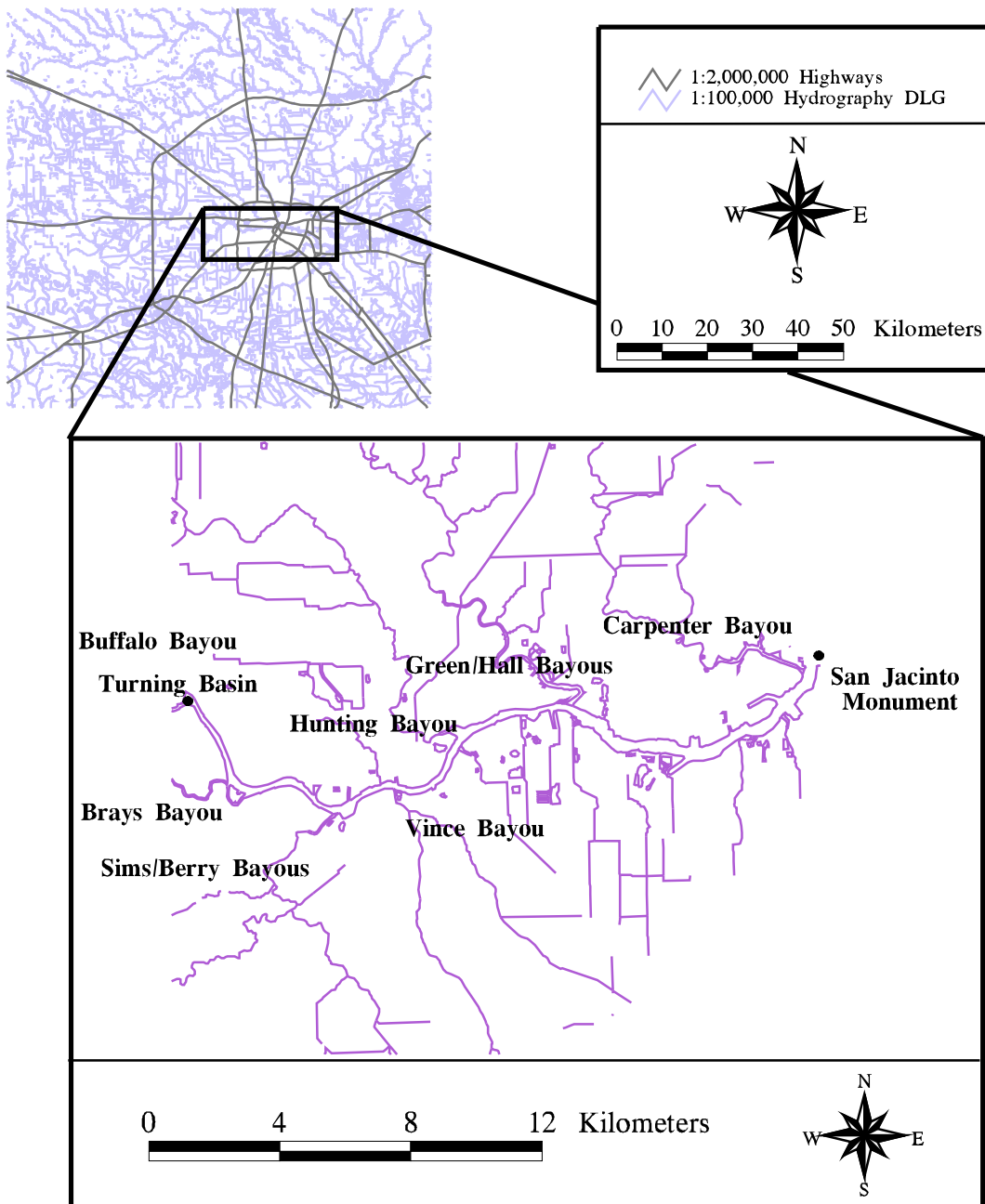


Figure 4-1 Study area, which includes the Upper Houston Ship Channel from the Turning Basin to the San Jacinto Monument and all land draining into the reach. Streams and channel are represented here by 1:100,000 DLG (USGS, 1993). Highways are also shown.

The DLG (Figure 4-1) depicts the Channel as a double-lined water reach; however, for digital representation in GIS and modeling purposes, the channel is depicted as a single flow line. Since the channel width averages only about 1000 meters at its widest point in that area, it could be modeled in just two dimensions (length and depth). Because of this detail, a single line could accurately be used to represent the channel in GIS. Therefore, a centerline was manually-drawn onto the DLG and the shoreline of the Upper HSC deleted in the Arc/Info subprogram, ArcEdit. Figure 4-2 shows the final result of this process. Section 4.2 further discusses the centerline representation of the Channel in GIS.

## 4.2 CHANNEL SEGMENTATION

In order to model the channel in WASP5, it is necessary to divide the reach into water quality segments. A segment is assumed to have uniform modeling parameters, such as depth, cross sectional area, dispersion coefficients, etc. Each segment is considered to be a completely mixed reactor. After the point and non-point loadings into each segment are determined; these loads, along with the necessary physical and chemical parameters, are read into WASP5 to produce a dissolved oxygen profile.

### 4.2.1 Previous Segmentation in the Houston Ship Channel

The Texas Water Commission (TWC -- now Texas Natural Resource Conservation Commission -- TNRCC) divided the entire Galveston Bay System into 40 segments (Ward and Armstrong, 1992). However, since the determination of these segments was controlled by regulatory reasons, homogeneous hydrography within a TWC segment could not be assumed. In addition, the reach considered for this study made up only two of the TWC segments (Figure 4-3). This resolution was not fine enough for an accurate modeling effort. Ward and Armstrong (1992) further divided the Bay and Channel into smaller, hydrographic segments (Figure 4-4). But, a full modeling effort, concerning DO had not been performed with this finer segmentation. In another earlier study (Espey et al., 1971), the entire channel was divided into 28 segments (Figure 4-5), from the Turning Basin to Morgan's Point (located at the mouth of the Channel flowing into the main bay). The 1971 effort, performed by Tracor, Inc., modeled the entire HSC for DO and BOD.

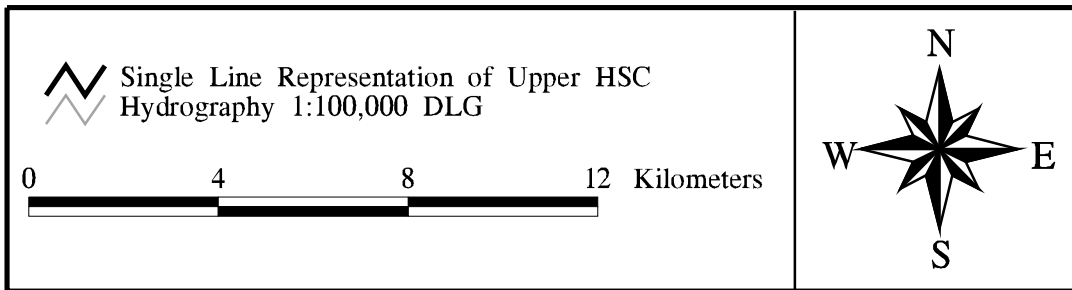
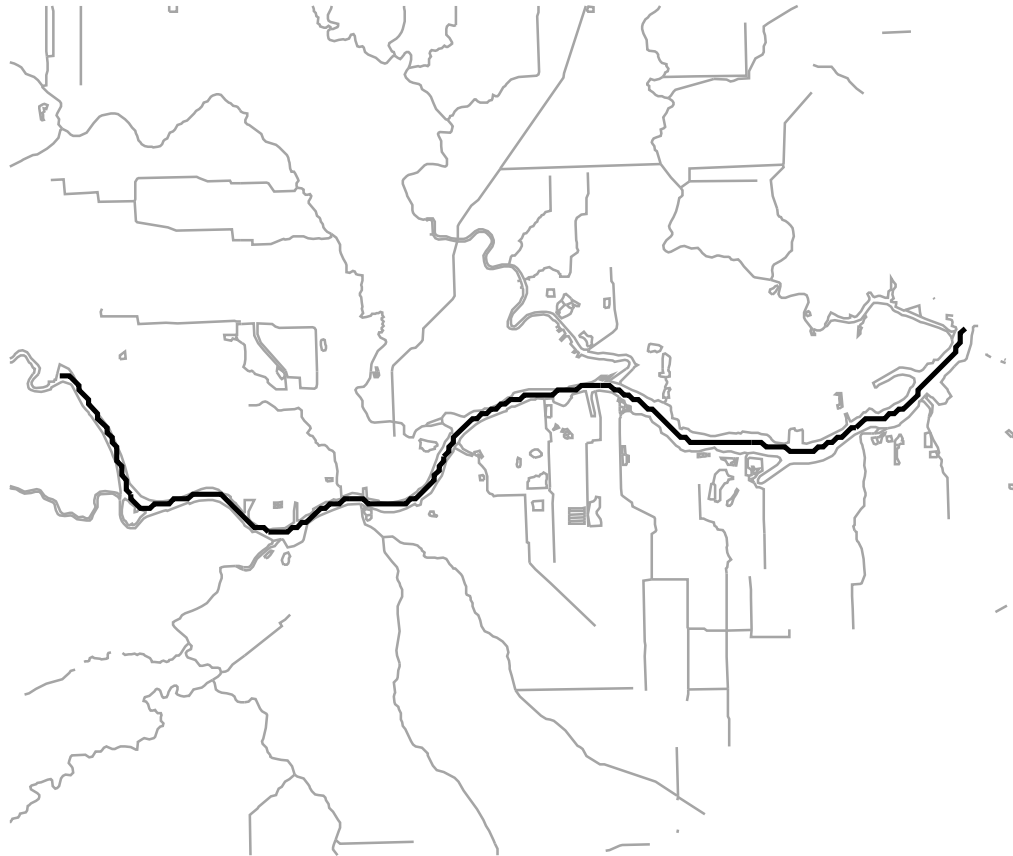


Figure 4-2 Single line representation of the Upper Houston Ship Channel used in GIS.

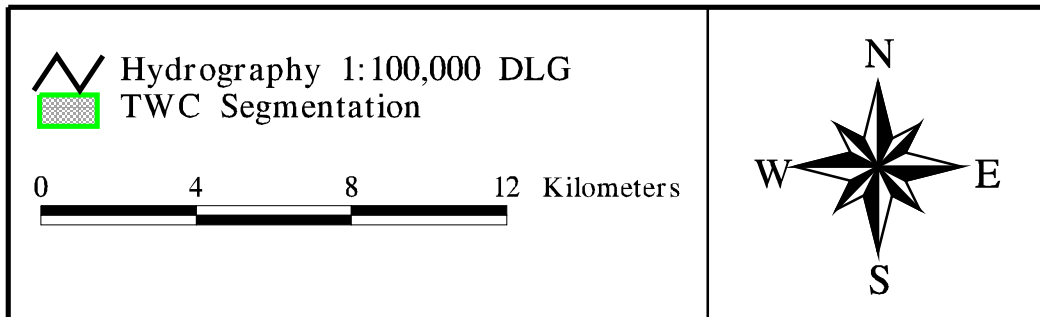
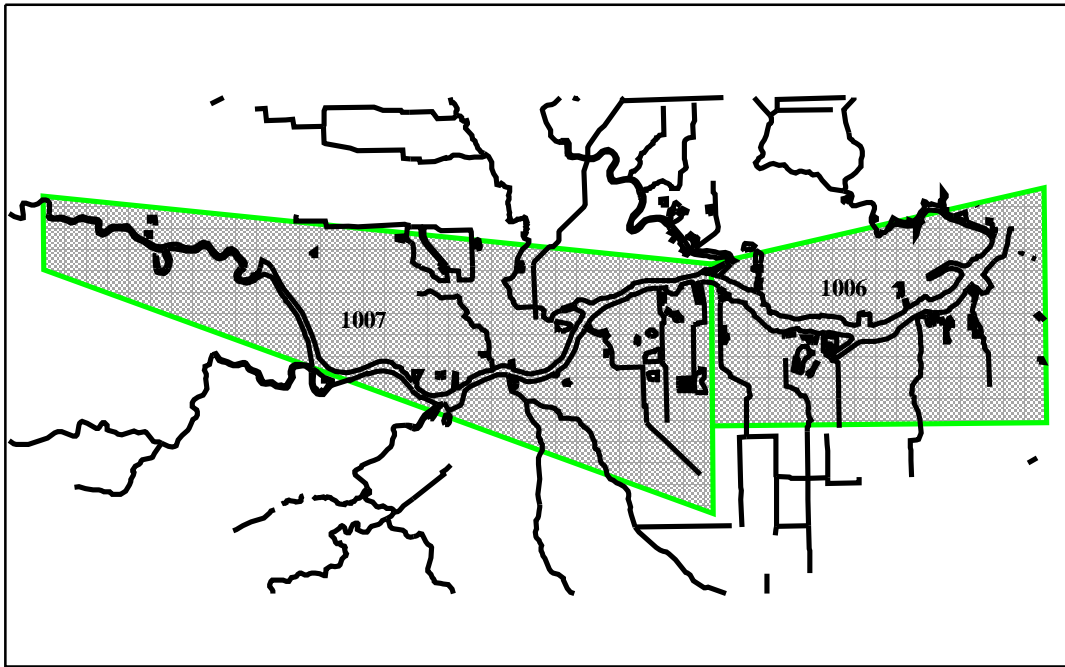


Figure 4-3 Upper Houston Ship Channel showing related TWC (now TNRCC) segments.

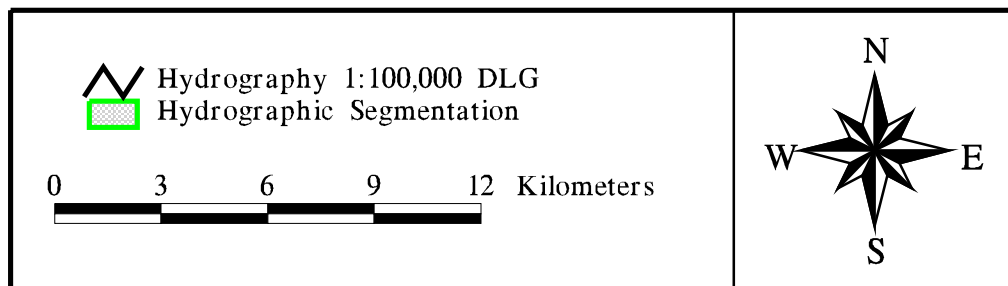
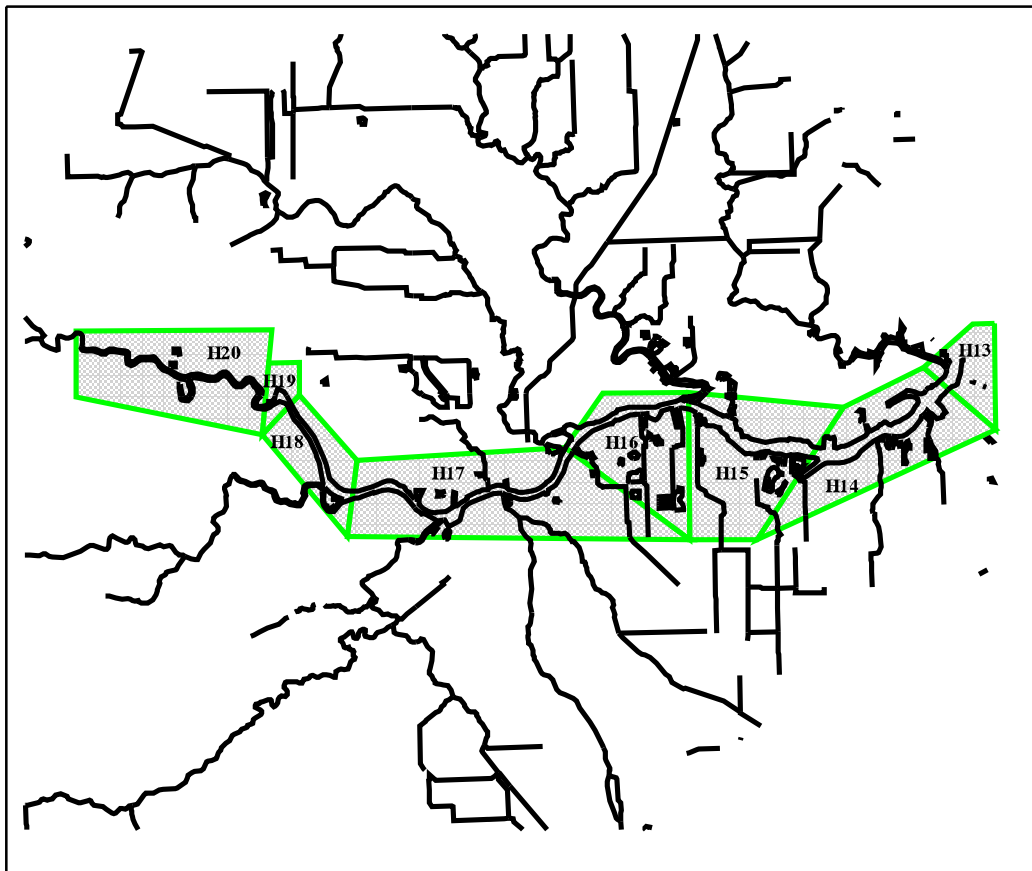


Figure 4-4 Ward and Armstrong (1992) hydrographic segmentation over the study area.

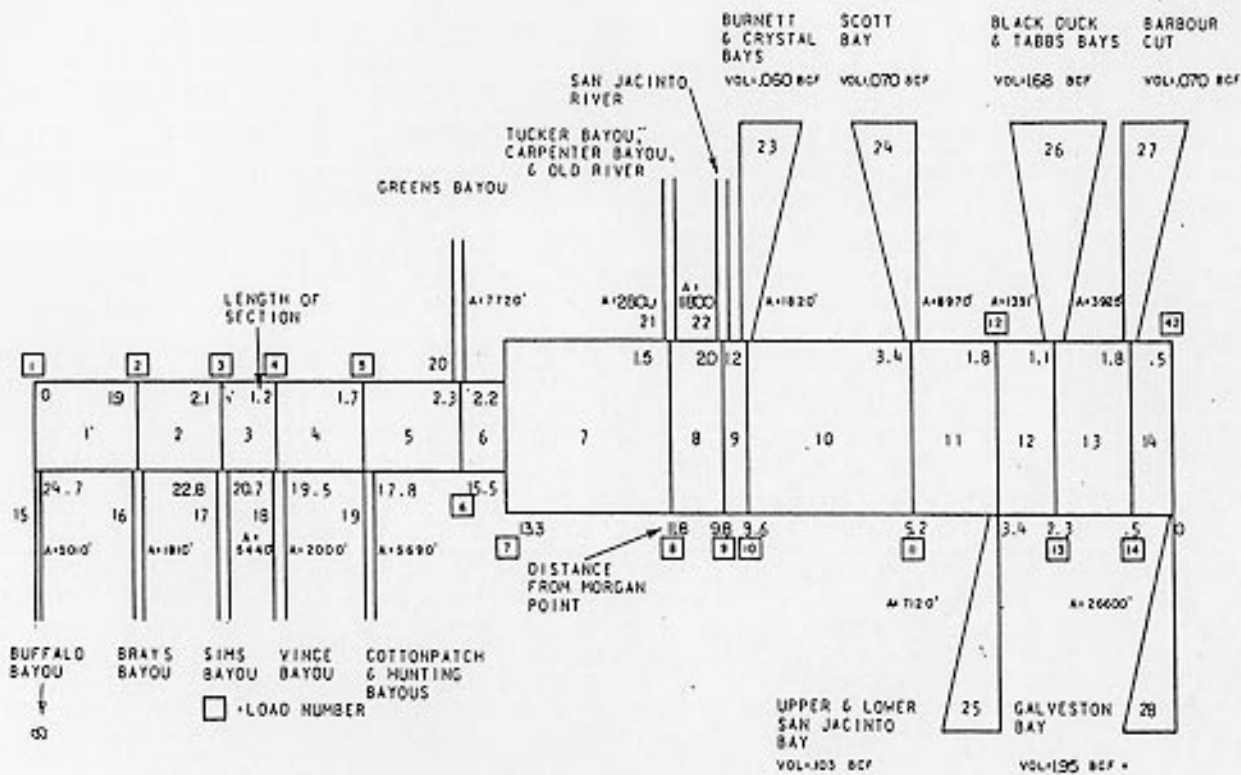


Figure 4-5 Original segmentation used by 1971 Tracor modeling effort (Espey *et al.*, 1971).



## 4.2.2 Segmentation Chosen

For this study, the Upper HSC reach was divided into eight of the hydrographic segments developed in the Tracor Inc. modeling effort of the HSC (Espey, *et al.*, 1971). By using this segmentation, the results of the modeling from this present study could be compared to the results of the 1971 study. Since this research considered just the Upper Houston Ship Channel, only the first eight segments of the 1971 report were used as the main segments for the modeling effort. Figure 4-6 shows the final segmentation used, while Table 4-1 gives some general characteristics of each main segment. In addition, some of the information provided in the 1971 report, concerning the incoming tributaries was used to develop the model boundary conditions (see Sections 4.2.3 and 4.5).

**Table 4-1 General Characteristics of the Main Segments used for the Modeling Effort**

Segment Number	Length (km)	Cross-sectional Area (m <sup>2</sup> )	Depth (m)	% Total Length
1	3.1	1625.8	9.1	12.8
2	3.4	1625.8	9.1	14.1
3	1.9	1625.8	9.1	8.1
4	2.7	1625.8	6.1	11.4
5	3.7	1625.8	9.1	15.4
6	3.5	1625.8	6.1	14.8
7	2.4	2471.3	7.9	10.1
8	3.2	2471.3	7.9	13.4
Total		24.0		

Source: Espey, *et al.*, 1971

## 4.2.3 Segment Terminology in this Research

In this report, a *main segment* is a term used in reference to the eight water quality segments described in the previous section and used to represent the 25 km of the Upper HSC modeled in this study. In addition, nine *boundary segments* are defined in this research. Model boundaries are those segments which import, export, or exchange water with the locations outside the main network. A boundary segment represents either a tributary inflow, a downstream outflow, a sediment layer, or an open water end of the model network across which dispersive mixing can occur. Within GIS, arcs were defined to represent the Buffalo,

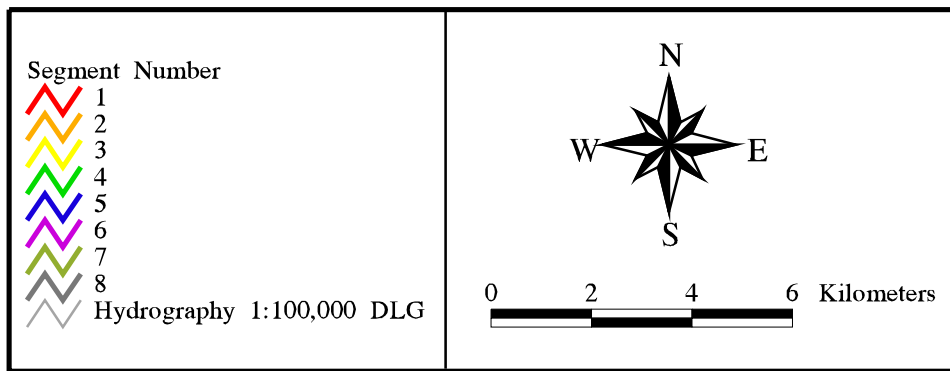
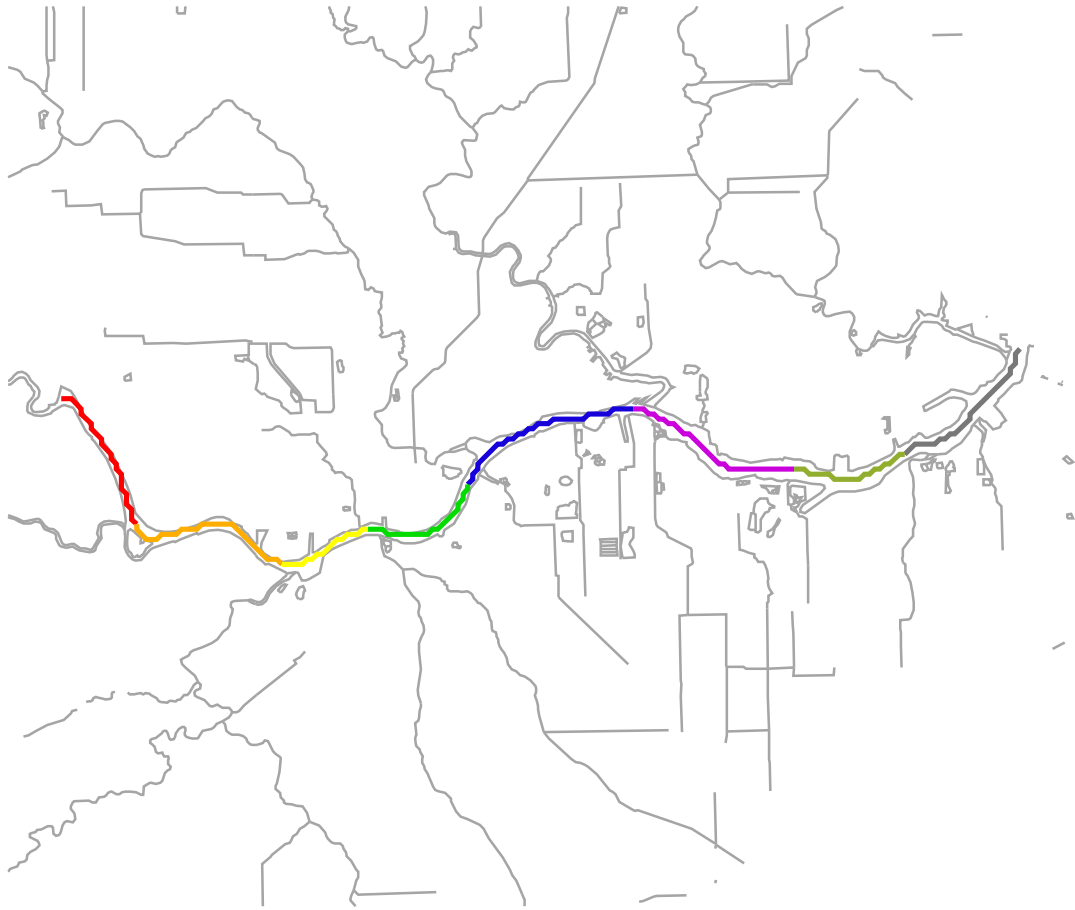


Figure 4-6 Final segmentation used as depicted in GIS (Espy et al., 1971).

Brays, Sims/Berry, Hunting, Vince, Green/Hall, and Carpenter Bayous (see [Figures 4-1](#) and [4-7](#)) to account for the tributaries entering the Upper HSC. The eighth boundary segment is the most downstream segment, which includes an input from the San Jacinto River (segment #17). The length of these arcs is arbitrary, but the lengths of the segments in the model are set at 3.2 km for all but the Buffalo Bayou, which is set at 8 km. These actual lengths, which are defined in a GIS table related to the arc coverage, are meant to depict infinite boundary conditions. Finally, the ninth boundary segment (segment #9), represents the underlying sediment layer. Further discussion on the main and boundary segments, their parameters, and their use in the model is provided in [Section 4.5](#).

#### **4.2.4 Segmentation in GIS**

Since the objective of this research was to connect WASP5 to GIS, the channel segmentation needed to be digitally represented in Arc/Info and ArcView. As mentioned earlier, the most efficient way to depict the channel was as a single line in GIS. As a result, the channel was viewed in GIS as a stream, into which numerous other streams (i.e., bayous) drained ([Figure 4-6](#)).

However, it was necessary to get GIS to recognize the channel as eight different segments, instead of one long stream. The desired result was eight arcs, each carrying their respective segment number as an attribute. To accomplish this task, a process called “flowlength” in Arc/Info’s subprogram, Grid, was executed on the flow direction grid of the DEM (see [Section 4.3](#) for further explanation of the flow direction grid). The flowlength command produced a grid in which each cell value corresponded to the distance, in meters, from that respective cell to the ultimate outlet of the grid. The flowlength values for just the single line representation of the Upper HSC were isolated with a Grid Boolean query (see [Procedure 4-1](#)). The result of this effort was a “single-lined” grid of the Upper Houston Ship Channel, with each 100m x 100m cell containing its respective flowlength value.

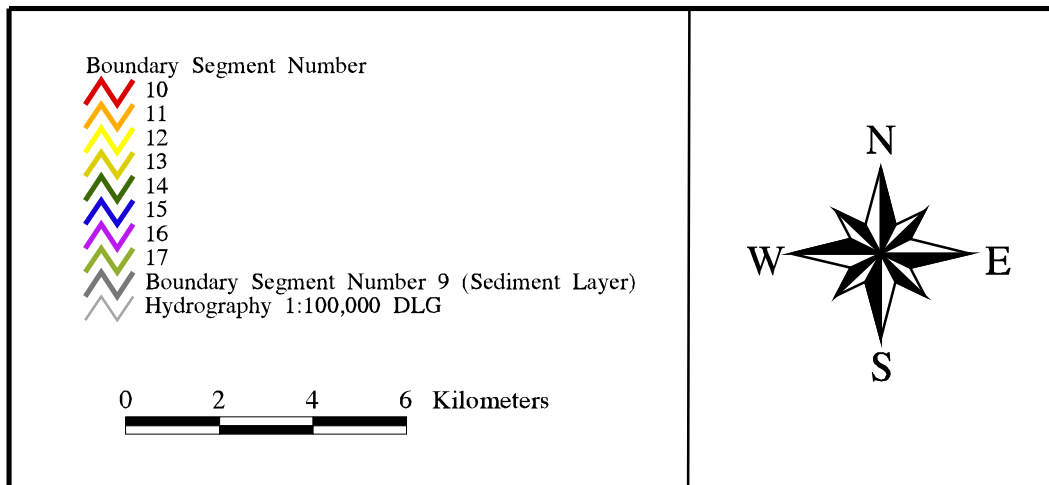
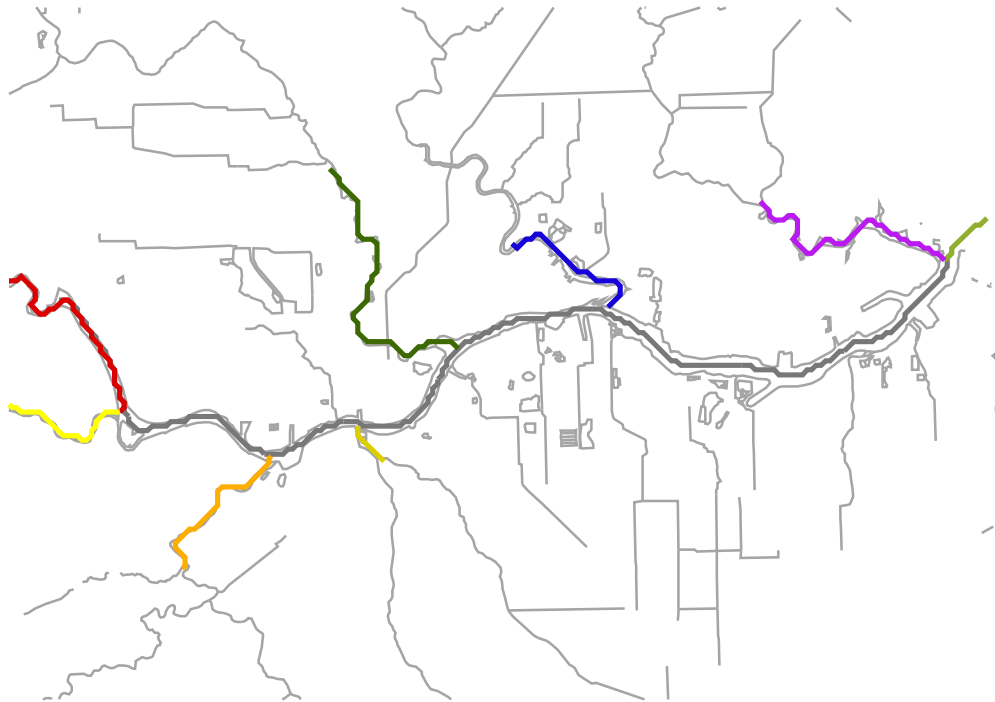


Figure 4-7 Arcs representing boundary segments for the water quality modeling.

Grid: **hscgrid = linegrid ( hscalone )**

*creates a grid, called hscgrid, of the single-line representation of the Houston Ship Channel*

Grid: **hscgrid1 = con ( hscgrid > 0, 1)**

*puts the value of "1" in each cell of hscgrid which lies on the channel*

Grid: **hscfl = flowlength ( hscfdr )**

*performs the flowlength command on the flow direction grid of the watershed; the result is hscfl, in which each cell carries the value of the length from the ultimate watershed outlet to that respective cell*

Grid: **channelfl = con ( hscgrid1 == 1, hscfl )**

*a Boolean query that produces a grid, channelfl, which holds just the flowlength values for the hscgrid*

Grid: **display 9999**

*brings up Grid's display window*

Grid: **mape hscgrid1**

Grid: **gridpaint bound**

*displays the grid of the start and end points of the modeling segmentation; this grid was produced by creating a point coverage of the two boundary points, projecting it into USGS-Albers, and gridding the coverage*

Grid: **cellvalue channelfl \***

*allows the user to query the cellvalue of the two boundary points by clicking on them within the display window*

**Procedure 4-1** Process used to determine flowlength values for the start and end points of the segmentation.

In order to determine the actual extent of the modeled reach in GIS, the locations of the start and end points for the channel segmentation were read from a 1:24,000 USGS quad sheet (Table 4-2). A point coverage of these two locations was created in Arc/Info and projected into USGS-Albers Projection. The point coverage was overlaid onto a grid of the single line representation of the Houston Ship Channel (Figure 4-8). The two grid points which corresponded to the start and end points were isolated using ArcEdit. Afterwards, their flowlength values from the isolated flowlength grid were determined with the "cellvalue" command in Grid (Procedure 4-1). The result of this entire process was the start and end flowlength values for the segmentation. By subtracting these two values, the total length of the segmented reach as represented in GIS was determined (Table 4-2). This length (25.1 km) only differed from the length given in the 1971 report (24 km) by 4.6% (Espey, *et al.*, 1971).

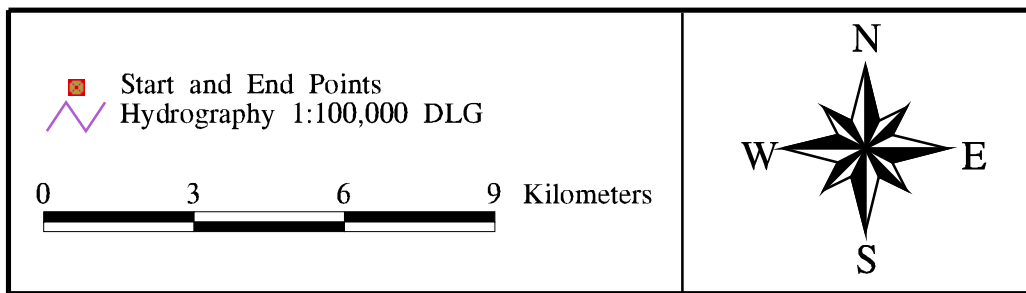
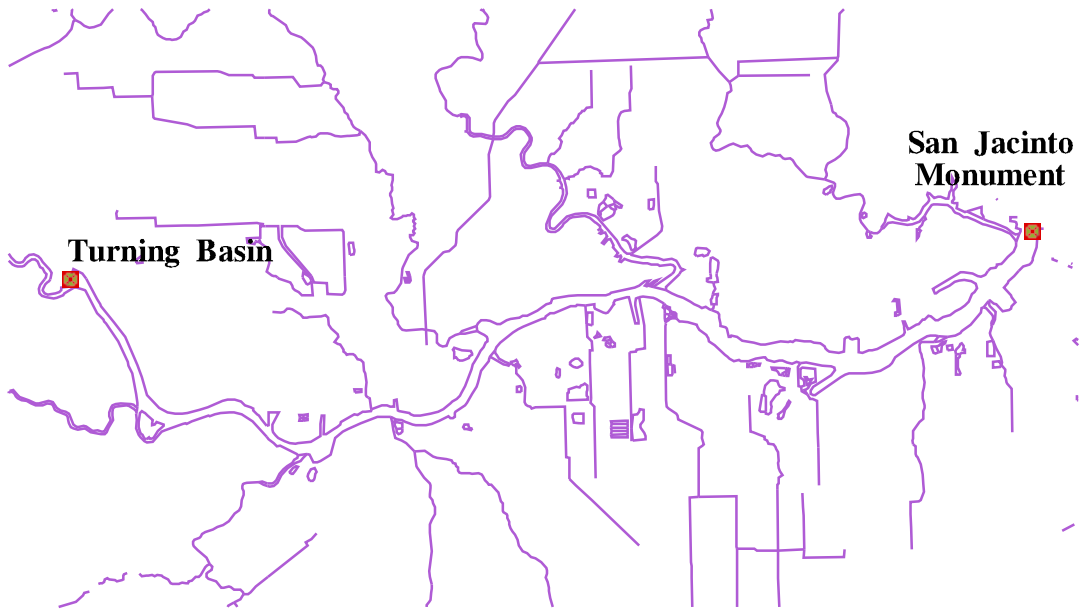


Figure 4-8 Upper Houston Ship Channel showing start and end points used for the modeled reach.

**Table 4-2 Start and End Points for Model Segmentation. Table also shows the flowlength values for each point and the resulting reach length in GIS**

<b>Point</b>	<b>Location</b> (From 1:24,000 USGS Quad Sheet)	<b>Flowlength Value</b> (m)
Turning Basin	29° 44' 58.4" N 95° 17' 25.4" W	38654.648
San Jacinto Monument	29° 45' 24.5" N 95° 5' 20.0" W	13502.429
	Difference:	25152.219
	Length of reach in GIS	25.1 km
	% Difference from Espey, <i>et al.</i> (1971):	4.6

Since the scale of the DLG may not have been on the exact same scale as the map used to determine the 1971 segmentation, proportional segmentation was used. To accomplish this task, the percent of the total reach length for each segment was calculated from the lengths given in the 1971 report. Those percentages, as shown in [Table 4-1](#), were then applied to the total stream length of the arc within GIS, as determined from the procedure above. A detailed description of this process is outlined in [Procedure 4-2](#). The final result, as illustrated in [Figure 4-6](#), was an eight-arc coverage of the segmentation, with each arc carrying an attribute corresponding to its segment number.

Grid: `seg_1 = con ( channelfl le 38654.648, 0) + con ( channelfl gt 35447.318, 1 )`  
*puts a value of "1" (1 + 0) in each cell that has a flowlength value (see [Procedure 4-1](#)) less than or equal to 36854.648 (the upper bound for segment one) and greater than 35447.318 (the lower bound); the total length of this segment is 12.8 % of the total segment length (see [Table 4-1](#))*  
 Grid: `seg_2 = con ( channelfl le 35447.318, 1) + con ( channelfl gt 31902.374, 1 )`  
 Grid: `seg_3 = con ( channelfl le 31902.374, 1) + con ( channelfl gt 29876.692, 2 )`  
 Grid: `seg_4 = con ( channelfl le 29876.692, 2) + con ( channelfl gt 27006.975, 2 )`  
 Grid: `seg_5 = con ( channelfl le 27006.975, 2) + con ( channelfl gt 23124.418, 3 )`  
 Grid: `seg_6 = con ( channelfl le 23124.418, 3) + con ( channelfl gt 19410.668, 3 )`  
 Grid: `seg_7 = con ( channelfl le 19410.668, 3) + con ( channelfl gt 16878.562, 4 )`  
 Grid: `seg_8 = con ( channelfl le 16878.562, 4) + con ( channelfl gt 13502.425, 4 )`  
*the above statements perform the same function as the first, only for each respective segment, the flowlength values change to encompass the necessary segment length and location*  
 Grid: `hsc_seg = merge ( seg_1, seg_2, seg_3, seg_4, seg_5, seg_6, seg_7, seg_8 )`  
*merges each individual grid, corresponding to each segment, into one grid*  
 Grid: `segarc = gridline ( hsc_seg, #, #, #, #, grid-code )`  
*creates an arc coverage of the grid and stores the segment number in the aat under "grid-code"*

**Procedure 4-2** Commands used to segment the single line representation of the HSC into eight arcs.

Once the segmentation was recognized in GIS, the parameters of each segment were attached to the attribute table of the eight-arc segmentation coverage. It was then possible for GIS to read the necessary input parameters for the model run. This concept is further discussed in [Section 4.6](#).

### **4.3 NON-POINT SOURCE LOADS**

#### *Introduction*

A non-point source (NPS) load is defined as any input into the HSC waters that is a result of runoff, which flowed over the land and picked up constituents from the land surface. Although the flow may have been channelized into a tributary by the time it reached the Houston Ship Channel, if the constituents originated from the land surface, as opposed to an outfall pipe, the load was considered to be a non-point source load. Determining the actual loading of constituents caused by overland flow has been a subject of numerous reports (Newell, *et al.*, 1992; Saunders, 1996). The method used in this report is similar to procedures described in Saunders (1996) and Newell, *et al.* (1992). The process utilized GIS to assist in the non-point source loading calculations. The basic concept of the method incorporated the following general equation:

$$\text{Concentration (mass/volume)} * \text{Volume of Water (volume)} = \text{Load (mass)} \quad (4-1)$$

The method developed, discussed in more detail below, was a grid-based model that calculated the non-point source load for each 100m x 100m cell of the watershed. The process used values called Estimated Mean Concentrations (EMCs), which, when associated with land use areas, provided the contribution of a given constituent to the runoff flowing over that area. Actual runoff and precipitation measurements were compiled and correlated to help spatially distribute the runoff over the entire 2600 km<sup>2</sup> area. This distributed runoff, combined with the land use based EMCs, established the NPS loadings into the Upper HSC.



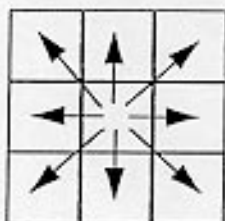
The processed specifically incorporated the following steps:

- Delineate watershed area (total and area draining into each segment)
- Spatially distribute the runoff
- Determine land use and concentration (EMC) from each cell
- Use [Equation \(4-1\)](#) to determine loading from each cell
- Determine the NPS load into each water quality segment

These steps are discussed in more detail in the following paragraphs.

### *Watershed Delineation*

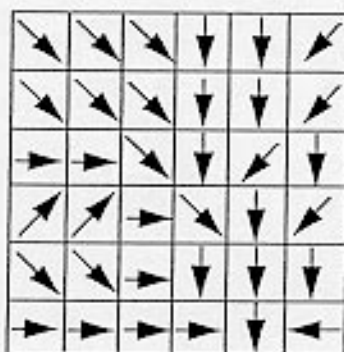
The grid-based watershed delineation has been used in other projects to produce a digital representation of all land draining into a body of water (Saunders, 1996). The concept in the watershed delineation is the use of the 3" DEM (see [Section 3.2.3](#)) to determine the direction of flow over the surface terrain. The basis of this concept is the application of the "eight direction pour point model" (Maidment, 1993). As shown in [Figure 4-9](#), the eight direction pour point model employs the theory that, if a drop of water falls onto a given cell, it can flow in eight different directions. The direction chosen is that of the steepest slope. Once the direction of the water flow is determined (termed the flow direction grid), Arc/Info's subprogram, Grid, accumulates the flow down to a given outlet (or the ultimate outlet of the grid) by counting the number of cells upstream that flow into that particular cell. A stream network is then delineated from a certain threshold value. In other words, a cell with a certain minimum number of cells draining into it was considered part of the stream network. [Procedure 4-3](#) shows a detailed description of this entire process in Grid. It is important to note that the DEM used is one that has been projected into USGS-Albers and a 100m x 100m cell size resolution. In addition, the point which represents the San Jacinto Monument was considered the ultimate outlet of the study area; therefore, a grid containing one cell, corresponding to this ultimate outlet point, was created through ArcEdit.



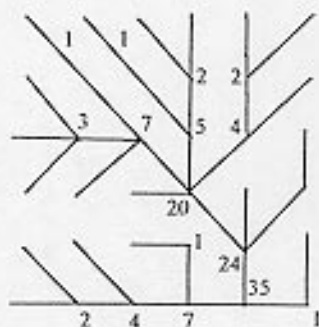
8-Direction Pour Point Model

78	72	69	71	58	49
74	67	56	49	46	50
69	53	44	37	38	48
64	58	55	22	31	24
68	61	47	21	16	19
74	53	34	12	11	12

Grid of Elevation Values



Flow Direction Grid



Flow Accumulation Grid

Figure 4-9 Eight direction pour-point model used in Grid for watershed delineation (Maidment, 1993).

Grid: **fill hscdem hscfil SINK**

*fills any "pits" or large differences in elevations between neighboring cells that may cause delineation errors. The "SINK" at the end of the statement tells Grid to look for cells which are lower than its surrounding cells.*

Grid: **hscfdr = flowdirection ( hscdem )**

*creates a flowdirection grid of the dem; each cell carries a value which indicates the direction of flow from that cell*

Grid: **hscfac = flowaccumulation ( hscfdr )**

*creates a flowaccumulation grid of the flowdirection grid; each cell carries a value which corresponds to the number of cells that drain into it*

Grid: **str\_500 = con ( hscfac > 500, 1 )**

*creates a grid of a stream network on the 500 level threshold; all cells that contain a flow-accumulation value of 500 or higher is considered part of the stream network and given a value of 1*

Grid: **totalshed = watershed ( hscfdr, outlet )**

*delineates the watershed from a given outlet point, in this case a grid containing one cell which is located nearest to the San Jacinto Monument. The outlet grid was developed by creating a point coverage of the location of the San Jacinto Monument (Table 4-2) and then creating a grid which has just one cell, corresponding to the point location through ArcEdit.*

Grid: **covstr\_500 = streamline ( str\_500 )**

Grid: **covtotsd = gridpoly ( totalshd )**

*converts the stream network grid and the watershed grid into line and polygon coverages, respectively*

**Procedure 4-3** Procedure used to delineate a watershed from a DEM for a given outlet.

The procedure above was determined using pure elevation data from the DEM. However, the final product of this attempt produced a poor digital representation of the stream network and watershed boundary (Figure 4-10). This erroneous result was mostly due to the relatively flat terrain in the area. Therefore, it was necessary to "burn in" the streams, using a cleaned 1:100,000 DLG. In this process, which is also employed in Saunders (1996), the original DLG was edited in ArcEdit to eliminate any circular arcs (i.e., lakes, reservoirs, etc.) and connect any dangling streams that are meant to be continuous. In addition, during the edit process, instream lakes and double-lined rivers or channels were replaced with representative streamlines. The streams were then gridded at the same resolution as the DEM (100m x 100m) and the elevations in the DEM, except for the cells which corresponded to the DLG stream cells were raised five meters. A new watershed could then be delineated, producing a more accurate digital representation of the ridgeline (see Procedure 4-4). As done previously, the ultimate outlet for this watershed was chosen to be a cell which was located nearest to the San Jacinto Monument (i.e. the last cell found in

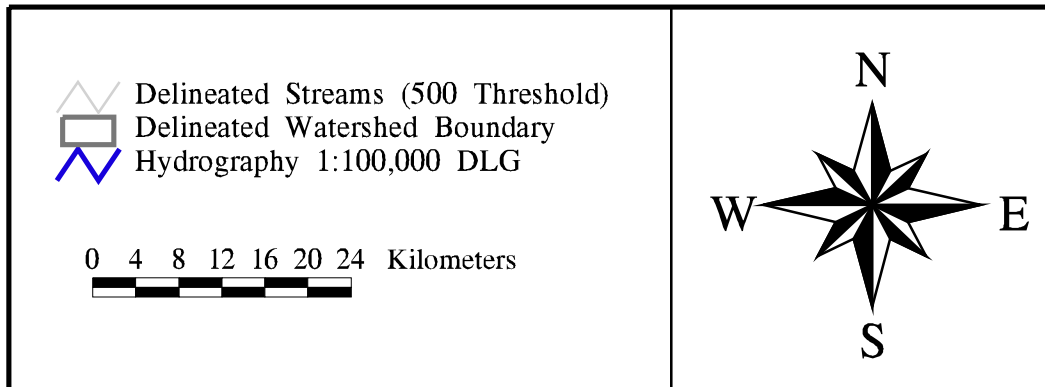
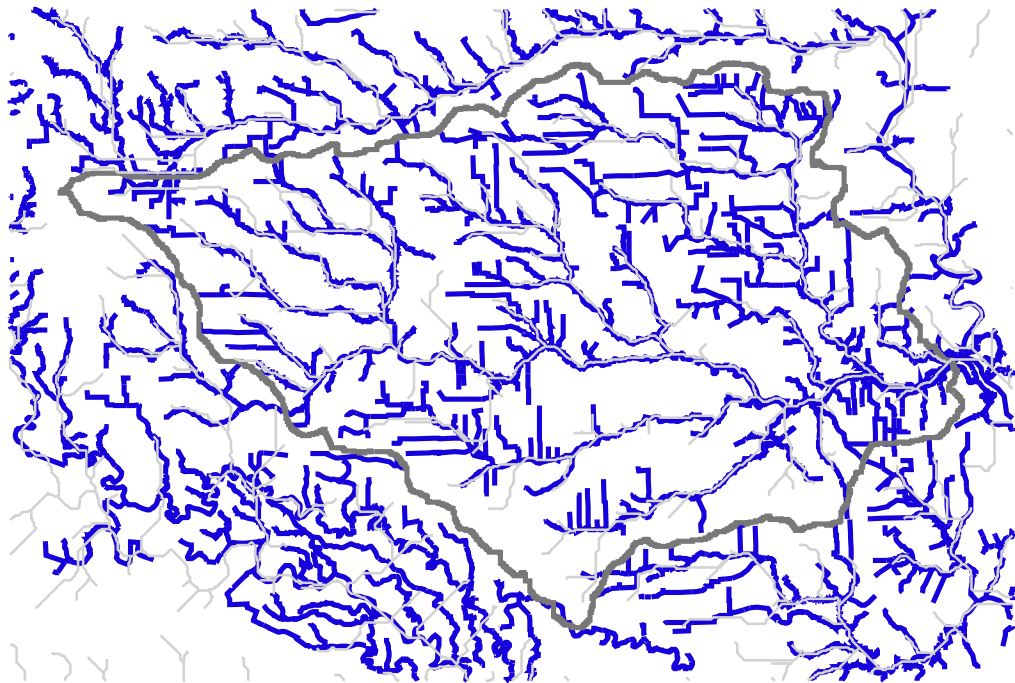


Figure 4-10 Delineated watershed and streams from original DEM. Errors exist between the DLG streams and the delineated ones; the watershed boundary is also inaccurate.

water quality segment eight). **Figure 4-11** depicts the final watershed and stream network resulting from this procedure.

Grid: **dlggrid = linegrid ( dlgedit, #, #, #, 100, 0 )**

*grids the edited DLG into 100m x 100m cells and places the value of zero in those cells not corresponding to the streams*

Grid: **hscburn = con ( dlggrid > 0, 0, hscdem + 5 )**

*increases the elevation values of the DEM grid by five meters and places the value of zero in any cell which corresponds to the stream network*

Grid: **fill hscburn hscfil SINK**

*fills any “pits” or large differences in elevations between neighboring cells that may cause delineation errors*

Grid: **hscfdr = flowdirection ( hscdem )**

*creates a flowdirection grid of the dem; each cell carries a value which indicates the direction of flow from that cell*

Grid: **hscfac = flowaccumulation ( hscfdr )**

*creates a flowaccumulation grid of the flowdirection grid; each cell carries a value which corresponds to the number of cells that drain into it*

Grid: **str\_500 = con ( hscfac > 500, 1 )**

*creates a grid of a stream network on the 500 level threshold; all cells that contain a flowaccumulation value of 500 or higher is considered part of the stream network and given a value of 1*

Grid: **totalshed = watershed ( hscfdr, outlet )**

*delineates the watershed from a given outlet point, in this case a grid containing one cell which is located nearest to the San Jacinto Monument*

Grid: **covstr\_500 = gridline ( str\_500 )**

Grid: **covtotsd = gridpoly ( totalshd )**

*converts the stream network grid and the watershed grid into line and polygon coverages, respectively*

**Procedure 4-4** Procedure for “burning in” the DLG streams and delineating the corresponding watershed.

The above procedures produced the total watershed (approximately 2600 km<sup>2</sup>). However, the area draining into the segments described in **Section 4.2** was of more importance when determining the NPS loading into each of the eight reaches. As a result, an outlet was defined at the downstream point of the segment, for each reach by locating the maximum flowaccumulation value in each zone (i.e. water quality segment) of the segmentation grid (see **Procedure 4-5**). The result was a grid of eight outlets, from which eight subwatersheds were delineated. **Figure 4-12** illustrates the final coverage of the areas draining into each segment, while **Table 4-3** gives the delineated areas for each subwatershed. **Procedure 4-5** describe the commands used in Grid to produce this result.

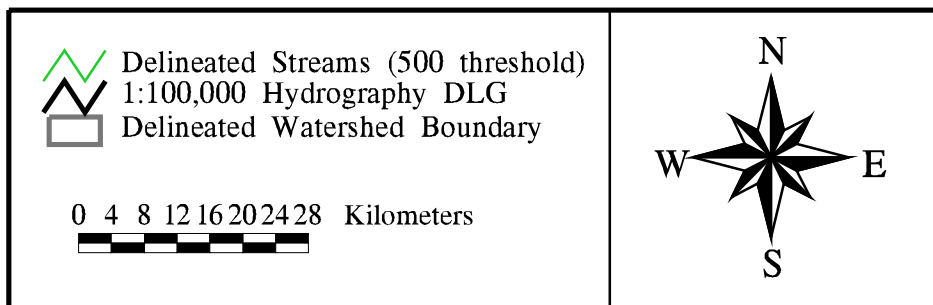
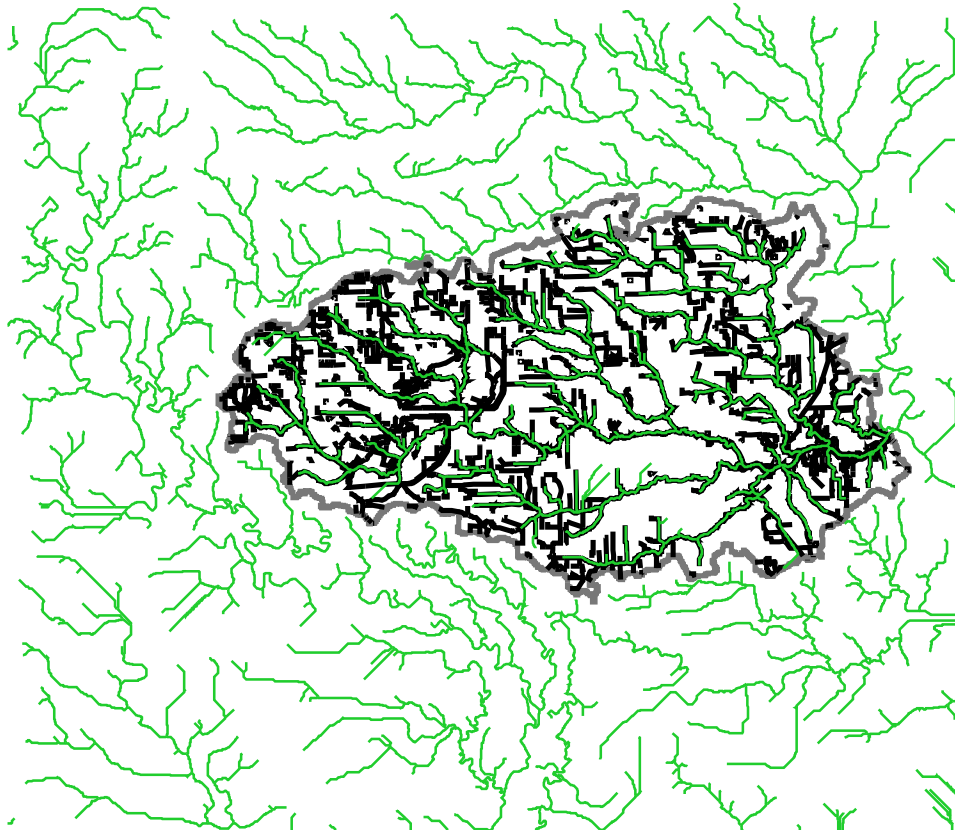


Figure 4-11 Comparison of streams delineated from "burned" DEM grid and 1:100,000 DLG of hydrography. Delineated streams correspond well to the DLG.

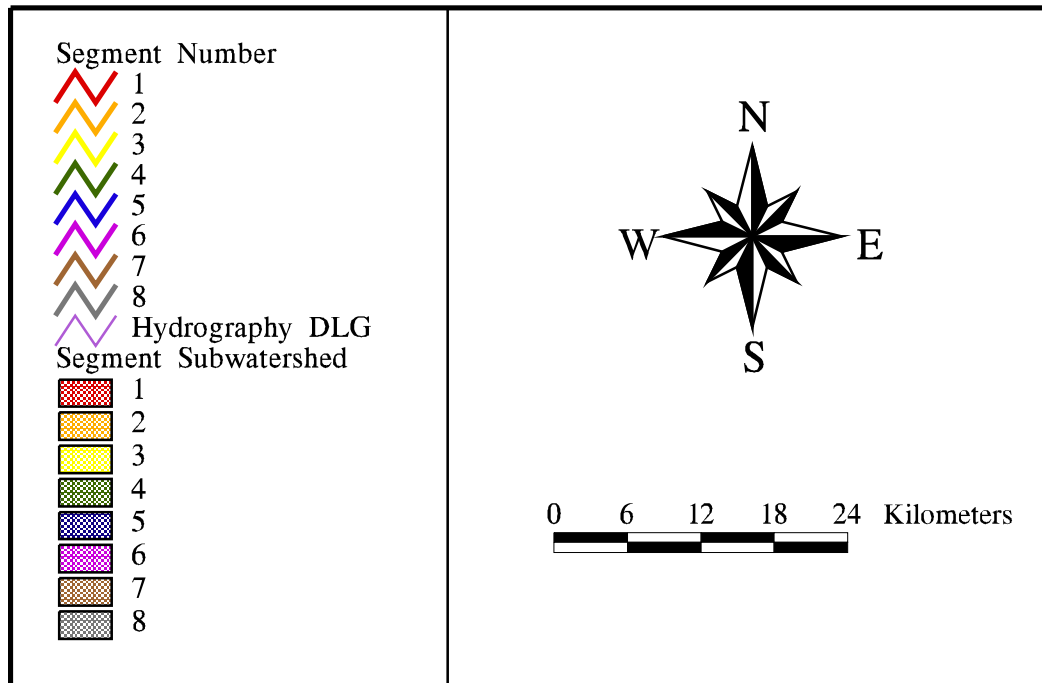
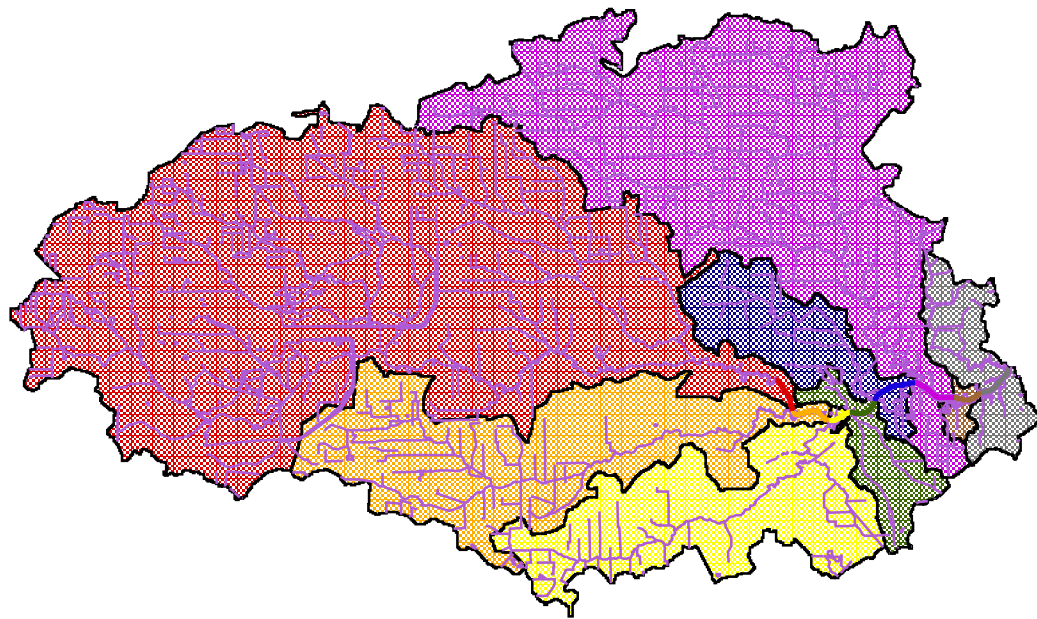


Figure 4-12 Segmentation with corresponding watersheds.

**Table 4-3 Delineated Areas of Segment Subwatersheds**

Segment Number	Area (km <sup>2</sup> )
1	1143.98
2	330.59
3	241.43
4	58.21
5	111.73
6	635.76
7	10.04
8	74.69
Total	2606.43

Grid: **acc\_seg = zonalmax ( hsc\_seg, hscfac )**

*locates the maximum flowaccumulation value in each zone; in this case hsc\_seg is a grid consisting of eight zones -- one for each segment*

Grid: **out\_seg = con ( acc\_seg == hscfac, hsc\_seg )**

*places the values of the segment number in the cell which corresponds to the maximum flowaccumulation value; the result of this is the outlet grid*

Grid: **seg\_shd = watershed ( hscfdr, out\_seg )**

*delineates the watersheds for the eight given outlets*

**Procedure 4-5** Commands used to develop segment subwatersheds.

#### *Spatial Distribution of Runoff*

After the watershed was delineated, it was necessary to obtain an average runoff volume generated from each cell. As mentioned in [Section 3.2.6](#), 37 USGS streamflow gauges were located in the study area ([Figure 4-13](#)). Of these 37 gauges, nine were chosen for their long period of records to determine a spatial distribution of runoff for the entire watershed ([Table 3-6](#)). For each of the nine gauges, the watershed areas were delineated from the flowdirection grid by selecting an outlet point at every gauge location. This selection was performed by choosing the cell, through ArcEdit, which was located on the delineated stream network and was nearest to the point representing the actual gauge location. [Table 4-4](#) shows the watershed areas determined from this process and a comparison of these delineated areas to the areas given by USGS (see [Figure 3-8](#)). Most areas delineated by Arc/Info fall within about 10% of the USGS area. Differences are



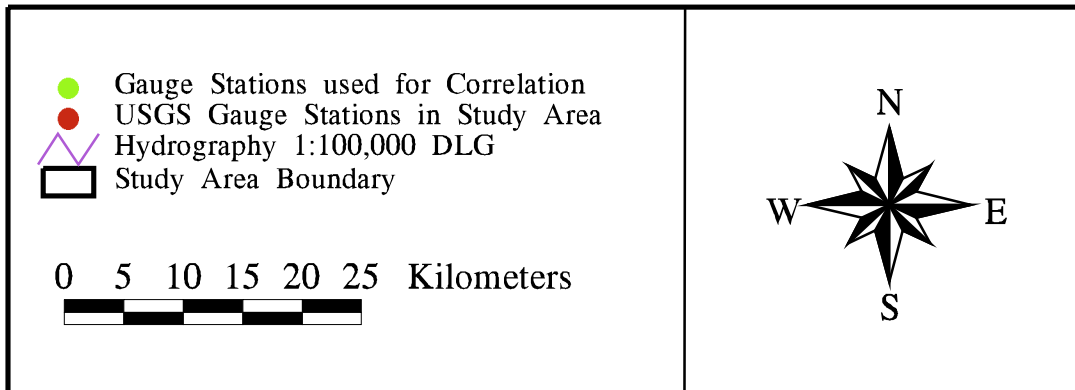
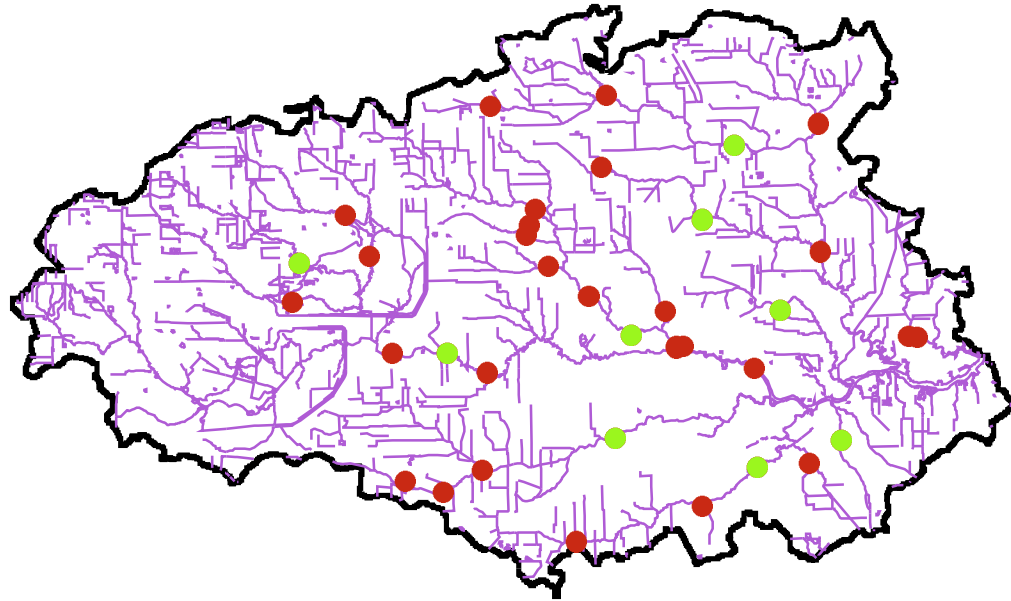


Figure 4-13 Study area showing USGS streamflow gauges.

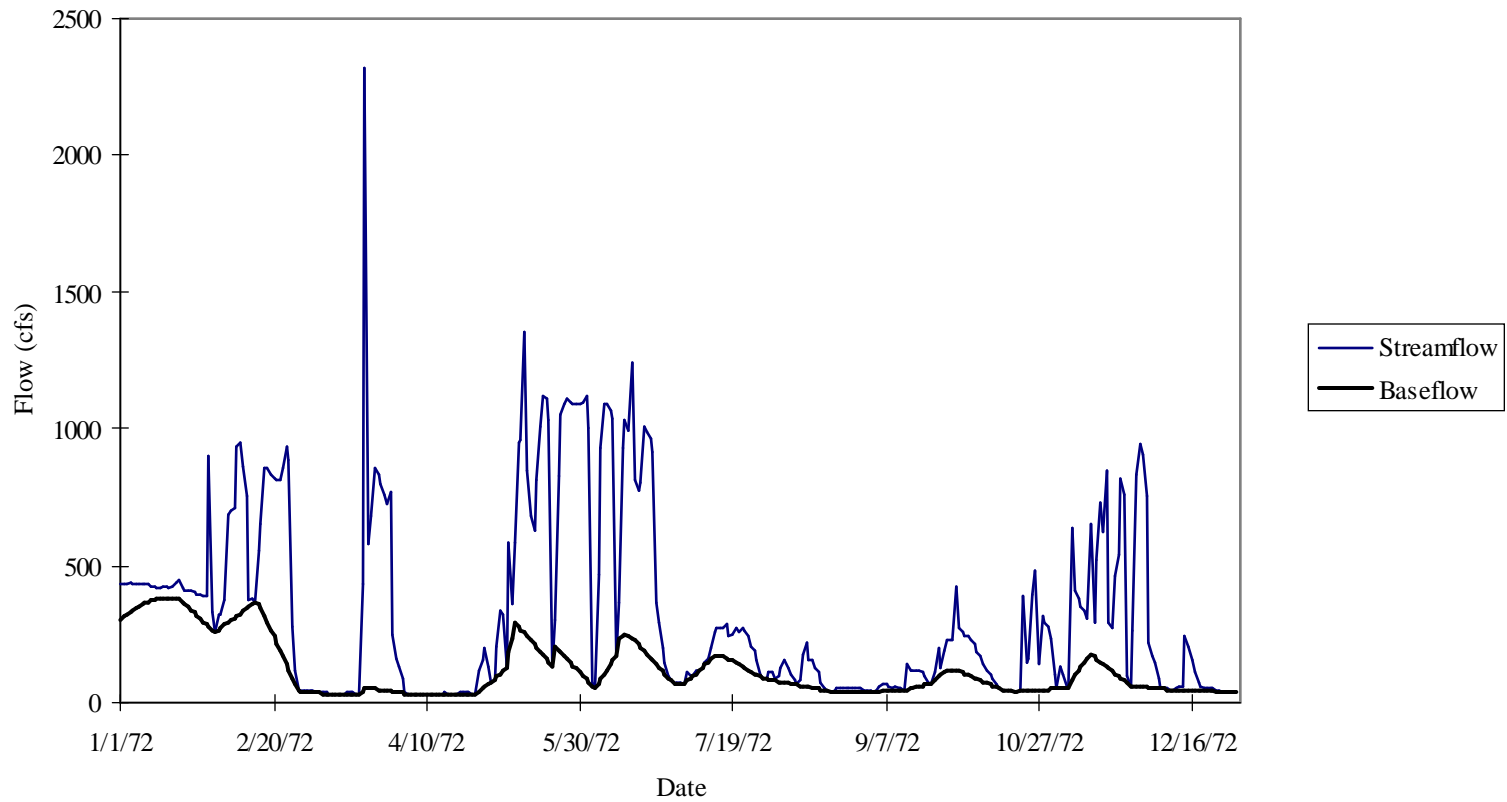
probably due to small inaccuracies in the DEM or possible errors in the given USGS areas. Since these nine stations only encompassed about 50% of the entire watershed area (see [Figure 3-8](#)), it was necessary to “redistribute” this flow data over the total watershed to obtain a runoff value from every cell in the study area.

**Table 4-4 USGS Watershed Areas Compared to Delineated Watershed Areas**

USGS Station	USGS Drainage Area (mi <sup>2</sup> )	Delineated Drainage Area (mi <sup>2</sup> )	%Error
8072730	21.5	25.78	19.92
8073600	307.0	273.18	-11.02
8074500	86.3	79.16	-8.28
8075000	94.9	97.94	3.20
8075500	63.0	61.12	-2.98
8075730	8.3	7.82	-5.33
8075770	16.1	20.06	24.61
8076000	68.7	69.04	0.50
8076500	28.7	27.82	3.08

*Source: Texas USGS, 1996*

Flow records formatted as average daily values are available for every gauge from the Texas USGS via internet (Texas USGS, 1996). Since the data available are in total streamflow, but the EMCs are meant to be applied to surface runoff, baseflow is subtracted out of the flow using a FORTRAN program (Olivera, 1996). This program, shown in [Appendix C-3](#), uses a concept of minimum slopes to estimate the daily baseflow. For the very first day in the period (Day 1), the streamflow value is assumed to be the baseflow. The program then calculates the slopes between the baseflow value to the streamflow values of the next 12 days (Days 2 through 13). The minimum slope found is assumed to be the baseflow slope; the baseflow for Day 2 is calculated using this minimum slope. The program then moves to Day 2 and calculates the slopes between the determined baseflow at Day 2 and the streamflow values of Days 3 through 14; hence the entire process is repeated. This method continues until an estimated baseflow value is obtained for each day in the period of record. [Figure 4-14](#) shows an example of one year of streamflow data, with the estimated baseflow. Since the program assumes that the value on the first day of the record is the



**Figure 4-14** Flow record for gauge 8073600, shows the estimated baseflow along with the measured streamflow (Texas USGS, 1996).

baseflow, streamflow values for a few days preceding and following the desired record are included in the baseflow estimation. In this way, any errors involved with this assumption are avoided with the ability to disregard the first and last few baseflow values.

The percentage of the total flow which accounts for the baseflow varied from station to station, with the average being 22%. Stations 8072730 and 8075730, which had small drainage areas, both had a baseflow/total flow percentage of 7%. In contrast, the two larger drainage areas from stations 8073600 and 8075000 resulted in 36% of the total flow being composed of baseflow. The other five stations all had baseflow/total flow ratios of about 21 to 30 %. The calculated baseflow is discussed later in this chapter in relation to the water quality modeling parameters (see [Section 4.5.5](#)).

When possible, the average daily flow data was downloaded for the 30 year period of 1961 - 1990 so that an accurate comparison could be performed with the precipitation data for the same period. As shown in [Table 3-4](#), various periods of record existed for each gauge. Once the baseflow was subtracted from the flow, any station containing an incomplete record between 1961 and 1990 was adjusted to fit the studied period of record by using the following equation:

$$(R_x)_{1961-1990} = \left[ \frac{(R_y)_{\text{available}}}{(R_y)_{1961-1990}} \right] (R_x)_{\text{available}} \quad (4-2)$$

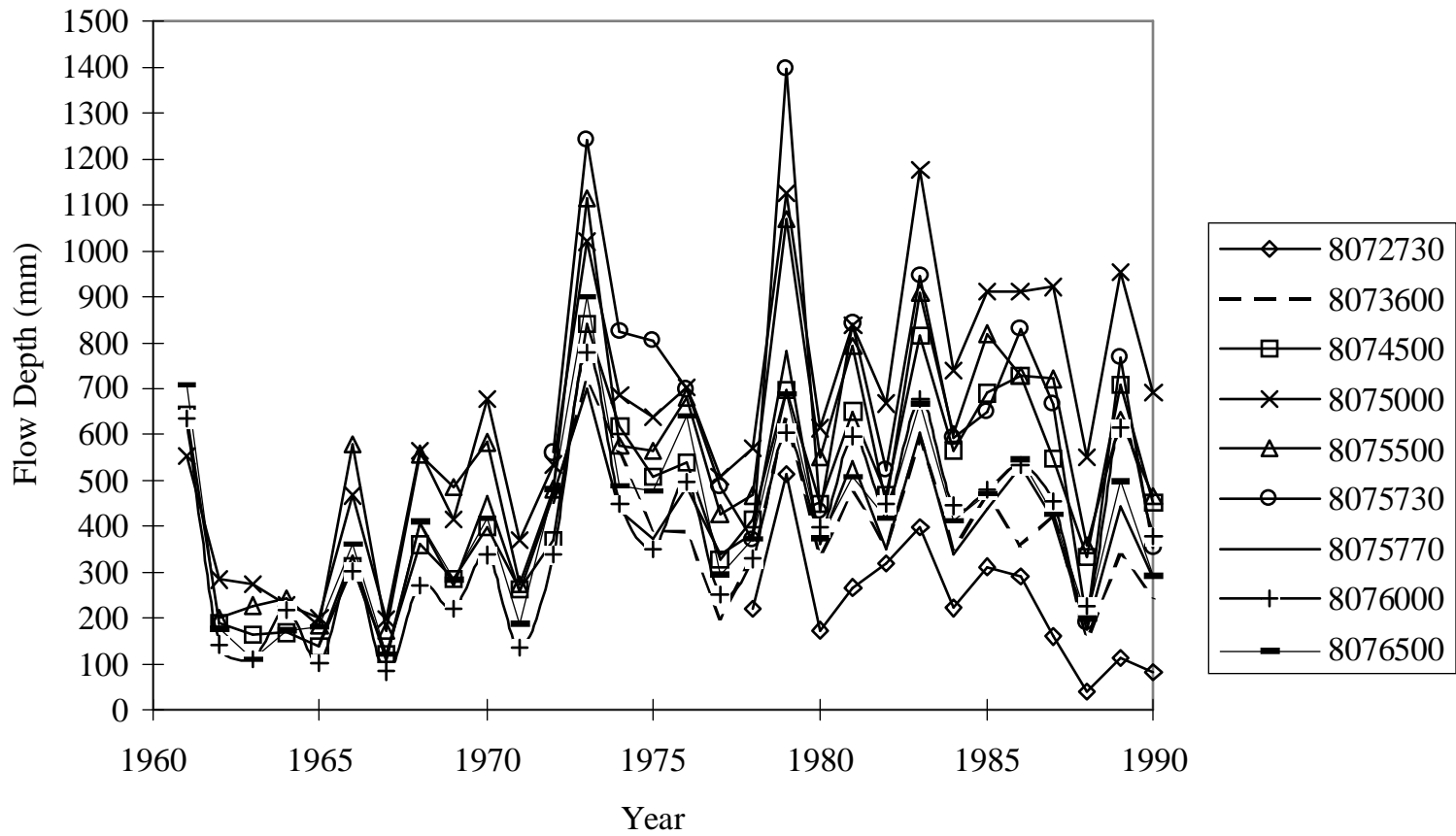
where:

- $(R_x)_{1961-1990}$  = average yearly runoff depth for a given gauge, x, adjusted to represent the entire period, 1961 to 1990 (mm/yr)
- $(R_y)_{\text{available}}$  = average yearly runoff depth of four gauged stations with complete records, averaged over the record available for gauge x (mm/yr)
- $(R_y)_{1961-1990}$  = average yearly runoff depth of for gauged stations with complete records, averaged over the record, 1961 - 1990 (mm/yr)
- $(R_x)_{\text{available}}$  = average yearly runoff depth for partially gauged station, averaged over the record available (mm/yr).

Gauges 8073600, 8074500, 8076000, 8076500, which combined, cover about 44% of the total watershed area, all have 30 years of data available. Although Table 3-6 shows that station 8075500 had a full period of record, some errors existed in the earlier data, resulting in an incomplete record for 1961 (Figure 4-15). To apply Equation (4-2), it is necessary to assume that the response of the entire watershed is similar to the response shown for those four stations. Figure 4-15 indicates that the overall response of the watershed to rainfall events is relatively consistent for gauge to gauge. Therefore, the use of Equation 4-2 is accurate. In addition, since the runoff data was calculated as average daily data, a macro was written in Excel which added the data to obtain yearly data. The final annual flow was divided by the delineated station subwatershed area to obtain depth of runoff per year. Figure 4-15 illustrates the annual flow depths for each station and how they varied over time. In addition, the final adjusted runoff values for each station are shown in Table 4-5. The relative runoff coefficient shown in Table 4-5 is discussed in the following paragraphs, while further discussion on the use of the baseflow in this study is found in Section 4.5.5, in relation to the water quality modeling parameters.

**Table 4-5 Streamflow Gauges with Runoff, Precipitation, and Relative Runoff Coefficient**

Gage	Average Runoff (mm)	Adjusted Avg Runoff (mm)	Precipitation (mm)	Runoff/Precipitation	Relative Runoff Coefficient
8072730	219.84	190.89	1111.92	0.17	0.28
8073600	248.68	211.95	1235.21	0.17	0.39
8074500	361.16	361.16	1187.18	0.30	0.56
8075000	404.85	404.85	1180.70	0.34	0.44
8075500	380.55	389.18	1220.16	0.32	0.49
8075730	643.50	548.46	1262.08	0.43	0.59
8075770	314.14	301.93	1223.32	0.25	0.43
8076000	304.32	304.32	1182.56	0.26	0.41
8076500	317.92	317.92	1199.45	0.27	0.52
8075900	241.01	226.56	1180.73	0.43	0.19



**Figure 4-15** Streamflow records for nine gauges used in rainfall/runoff correlation. Graph shows the similar response of the watershed to precipitation.

Once the runoff was determined, the annual precipitation data had to be considered. By using the annual precipitation grid described in [Section 3.2.5](#), an average annual depth was found by performing a weighted flow accumulation on the flow direction grid with the precipitation grid (see [Procedure 4-6](#)). The flow accumulation value at each station was then determined by using the “cellvalue” command. This value, which was actually in units of depth x total number of cells upstream from the given cell, was divided by the number of cells in the station subwatershed to obtain an average precipitation depth. The results of this process are shown in [Table 4-5](#). The average runoff depth was divided by the average precipitation depth at each gauge to obtain an estimate for the average yearly percentage of precipitation which becomes runoff.

grid: **pannalb = project ( p\_ann, geoalb.prj, #, 100 )**

*projects the annual precipitation grid over the study area from geographic coordinates into USGS-Albers with a 100 m x 100 m cell size*

grid: **pannfac = flowaccumulation ( hscfdr, pannalb )**

*performs a weighted flow accumulation on the flow direction grid by adding up the precipitation cells that flow into a given cell; the final grid contains cells with the total amount of rainfall multiplied by the number of cells flowing into a given cell*

grid: **pannvalues = con ( gage > 0, pannfac )**

*puts the value of the flow accumulation grid cell in to corresponding cell that represents the USGS gauge location*

grid: **mape gage**

grid: **gridpaint gage**

grid: **cellvalue pannvalues \***

*allows one to query the gauge grid and obtain weighted flow accumulation values at each station*

**Procedure 4-6** Procedure used to determine the flow accumulation values of precipitation at each gauge station (i.e. subwatershed outlet).

The highly urbanized quality of the watershed provides support to a correlation between this runoff/precipitation ratio and land use. To do this correlation, a value was assigned to each land use cell to characterize the amount of runoff that cell would produce. In classical urban hydrology, it is common to use runoff coefficients to help characterize the amount of runoff produced from a given storm event for a given area. In a similar way, runoff coefficients can provide a relative measure of the urbanization of an area, by assigning high values to paved areas and low values to open, grassy land. This latter concept was employed to get a “relative measure of urbanization” for the Upper HSC watershed. Values

of runoff coefficients vary depending on the source (Chow, *et al.*, 1988; Browne, 1990; Pilgrim and Cordery, 1993). The coefficients found in **Table 4-6** were chosen from the researched literature and assigned to each land use (Browne, 1990). The coverage of land use was then gridded and the value of the runoff coefficient was retained as the measurement in each 100m x 100m cell. In a manner similar to the steps followed to obtain average precipitation depth (**Procedure 4-6**), the average “runoff coefficient” over each station subwatershed was determined (see **Table 4-5**). It is important to note that this value gives only a relative measure of the urbanization for the watershed; it can not be used in an absolute manner. A small coefficient value indicates less urbanization over a watershed area, as compared to an area with a higher coefficient value.

**Table 4-6 Runoff Coefficients Used for Relative Measure of Watershed Urbanization \***

Land Use	Runoff Coefficient
Urban <sup>+</sup>	0.89
Open	0.22
Agriculture	0.24
Barren	0.22 (estimated)
Wetlands	0.8
Residential <sup>++</sup>	0.34
Water	1
Forest	0.15

*Source: Browne, 1990 (Table 7.6)*

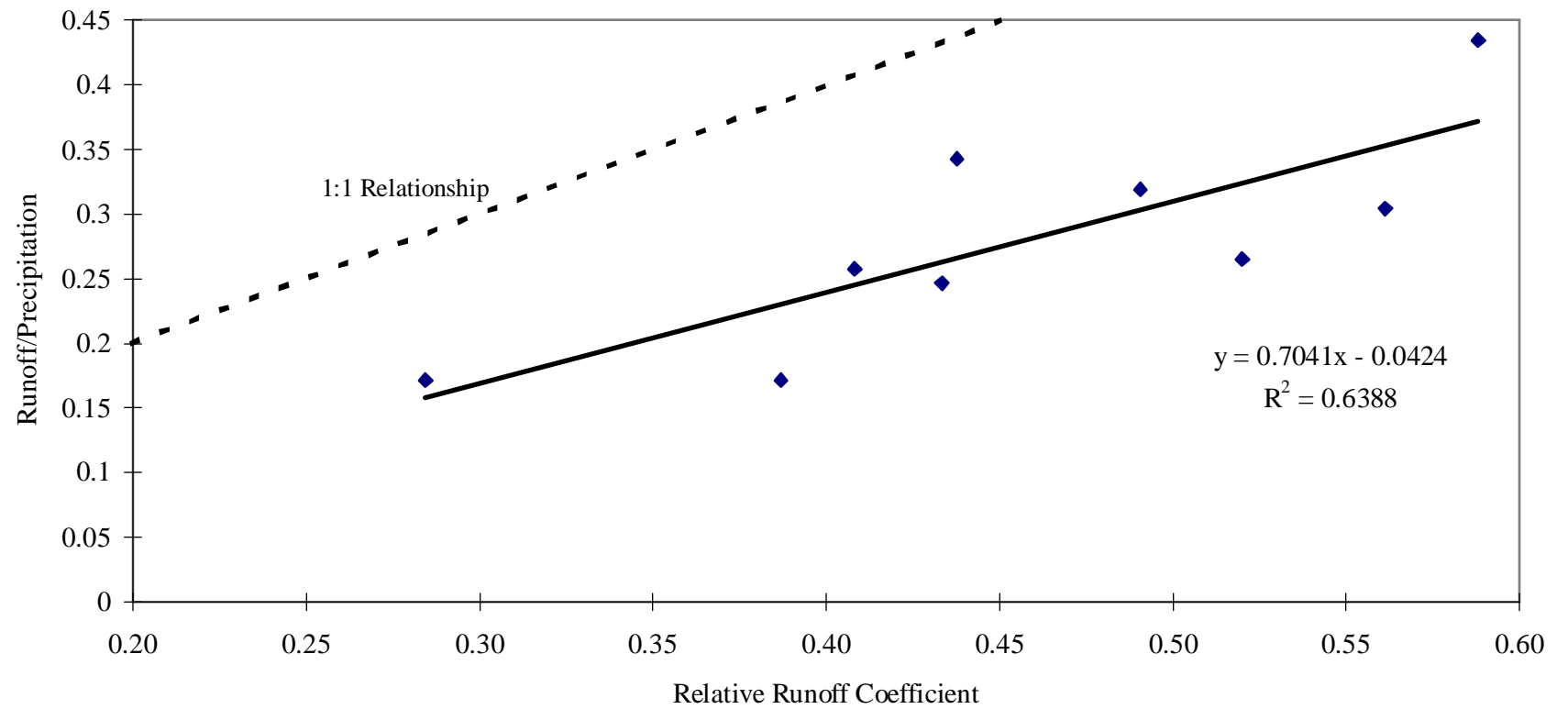
\* Assumptions: Soil Type D, Design Storm 25 + yrs, Flat Slopes

+ Assumed Commercial

++ Assumed 1/2 acre lots

The runoff/precipitation ratios were plotted against these relative runoff coefficients to obtain a relationship between percentage of precipitation which becomes runoff and extent of urbanization over the land surface. **Figure 4-16** displays the final correlation graph with a linear regression best fit to the points. The 1:1 line on the graph illustrates the relationship which exists if the relative runoff coefficient represented the percentage of precipitation which eventually becomes runoff. The actual values which resulted from the procedure described in the previous paragraphs, are about 60% of the values on the 1:1 line. This





**Figure 4-16** Relation between ratio of mean annual surface runoff divided by mean annual precipitation to runoff coefficients for the same area, based on land use and standardized runoff coefficient table. 1:1 line illustrates the relationship that would exist if the runoff coefficient was an absolute measure of the percentage of precipitation which becomes runoff.

equation was then used to redistribute the runoff over the entire watershed area. This process was accomplished by taking the runoff coefficient grid and using it as “input” to the equation to produce a runoff/precipitation grid (Procedure 4-7). The ratio grid, which contained a value of runoff/precipitation for every 100m x 100m cell, was multiplied by the precipitation grid discussed in Section 3.2.5. The final result was a grid of estimated annual runoff (Figure 4-17) in dimensions of depth.

grid: **hsccoeff = polygrid ( hscclu, runoff\_coeff, #, #, 100 )**

*grids the land use coverage to a 100 m x 100 m cell size and retains the runoff coefficient attached to the particular land use, as the value in each cell*

grid: **r\_pann = hsccoeff \* 0.704 - 0.0424**

*uses the correlation shown in Figure 4-16 to calculate a runoff/precipitation value for every cell*

grid: **r\_ann = r\_pann \* pannalb**

*creates a grid of runoff by multiplying the calculated runoff/precipitation grid by the measured precipitation grid*

**Procedure 4-7** Procedure used to distribute average annual runoff over the entire watershed area.

#### *Estimated Mean Concentrations*

An event mean concentration is the average concentration of water quality constituents over the course of a storm event from a defined drainage area with a given land use. Since this study examines steady-state responses, instead of just one particular storm event, a more accurate name for this factor is Estimated Mean Concentration (EMCs). Numerous studies have been undertaken to determine accurate EMCs for various areas (Newell, *et al.*, 1992). Research has shown that most EMCs are site-specific; therefore, it is best to use values that have been determined for either a particular area of study, or for an area with similar land usage. Newell, *et al.*, (1992) performed an extensive investigation to obtain accurate Estimated Mean Concentrations values for the Houston area. Most of the EMCs determined from this 1992 study were derived from the analysis of point and non-point source water quality data for the Houston area and previous water quality reports dealing with NPS loading. Although the modeling effort for this current study required only BOD, Table 4-7 shows some other typical values used in the Newell, *et al.*, (1992) study. In addition, Figure 4-18 shows the distribution of the BOD EMC values over the watershed area.

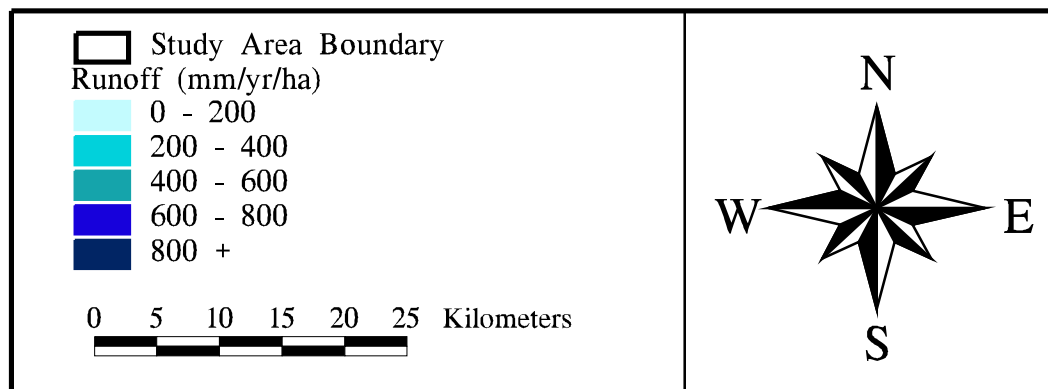
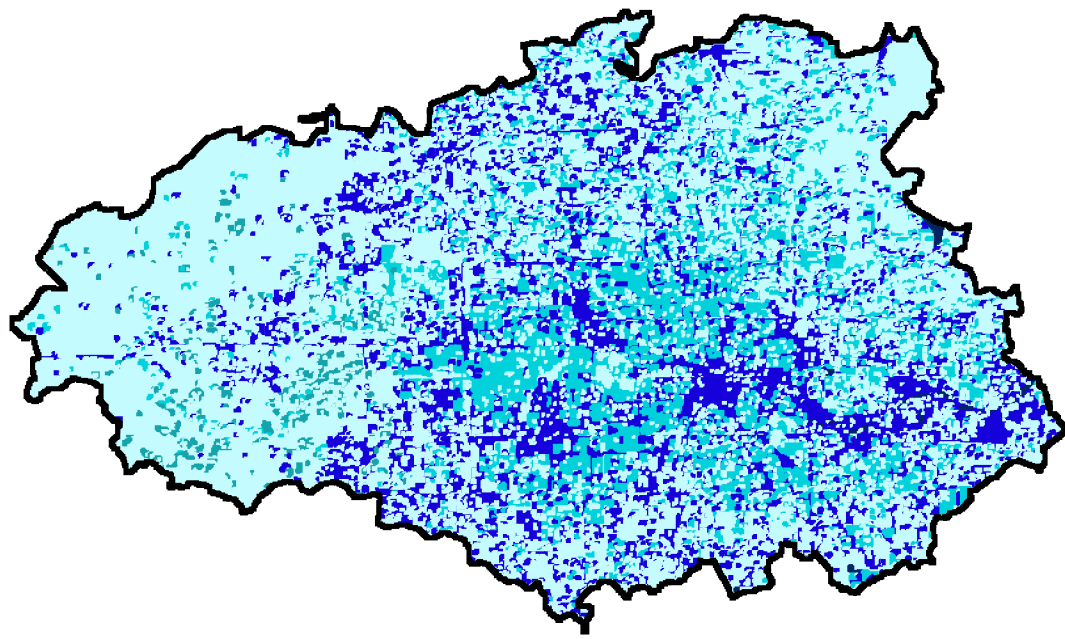


Figure 4-17 Final runoff distribution over watershed area.

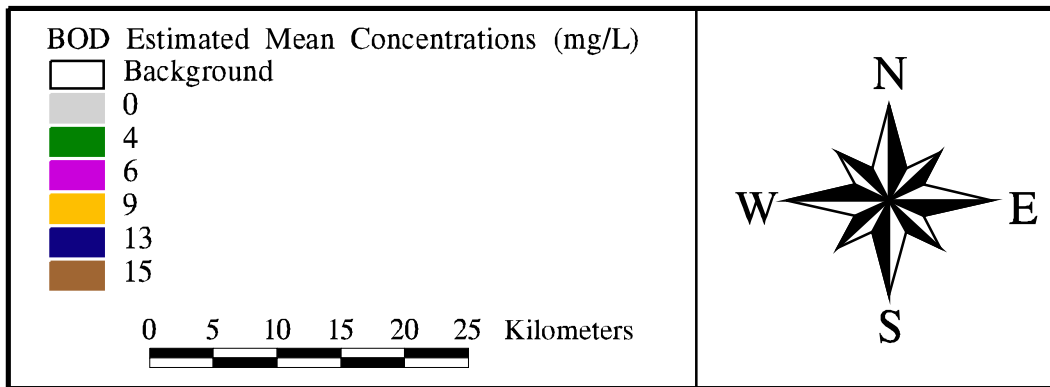
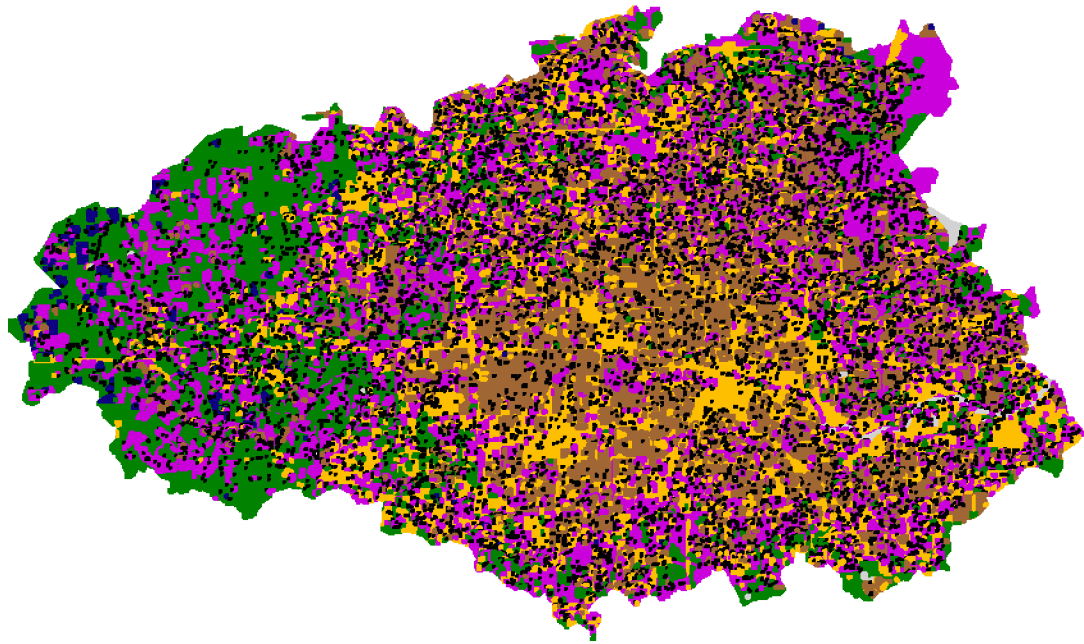


Figure 4-18 Estimated Mean Concentrations for BOD distributed over the watershed area.

**Table 4-7 Estimated Mean Concentration Values Used for Non-Point Source Loading**

Land Use Category	Total Suspended Solids (mg/L)	Total Nitrogen (mg/L)	Total Phosphorus (mg/L)	Biochemical Oxygen Demand (mg/L)
High Density Urban	166	2.10	0.37	9
Residential	100	3.41	0.79	15
Agricultural	201	1.56	0.36	4
Open/Pasture	70	1.51	0.12	6
Forest	39	0.83	0.06	6
Wetlands	3	0.83	0.06	6
Water		0.00	0.00	0
Barren	2200	5.20	0.59	13

Source: Newell, et al., 1992

#### *Final Loading Calculations*

Equation (4-1) required a runoff volume multiplied by a constituent concentration to obtain a final NPS load. As discussed at the beginning of this chapter, the baseflow, which is presented in Section 4.5.5, carries only the point source loadings, while the runoff transports the non-point source loadings. With the runoff distribution determined above and the BOD concentration from the EMC values, the BOD loading due to non-point sources could be calculated. This procedure was accomplished by multiplying the EMC grid with the runoff grid and correcting for unit conversions (Procedure 4-8). The final result was a grid containing the BOD loading in kg/yr, for each cell. This grid was converted into a coverage and shown in Figure 4-19.

As mentioned earlier, the NPS loading into each water quality segment was the value of interest for this project. These inputs were determined by running a weighted flow accumulation of the BOD loading grid and obtaining the flow accumulated value at each segment “outlet” (Procedure 4-8). Since these values were accumulated, they had to be subtracted, successively. For example, the actual loading to segment two is the flow accumulation value at the outlet to two, minus the flow accumulated value at segment one’s outlet. The results of this process are shown in Table 4-8.

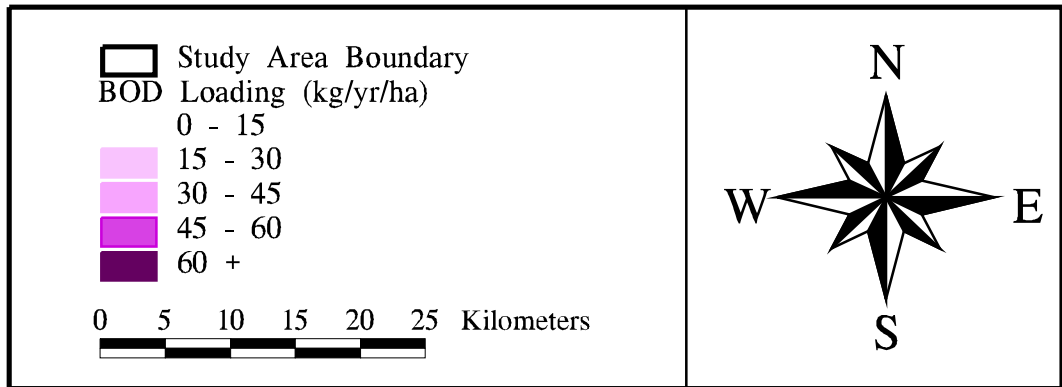
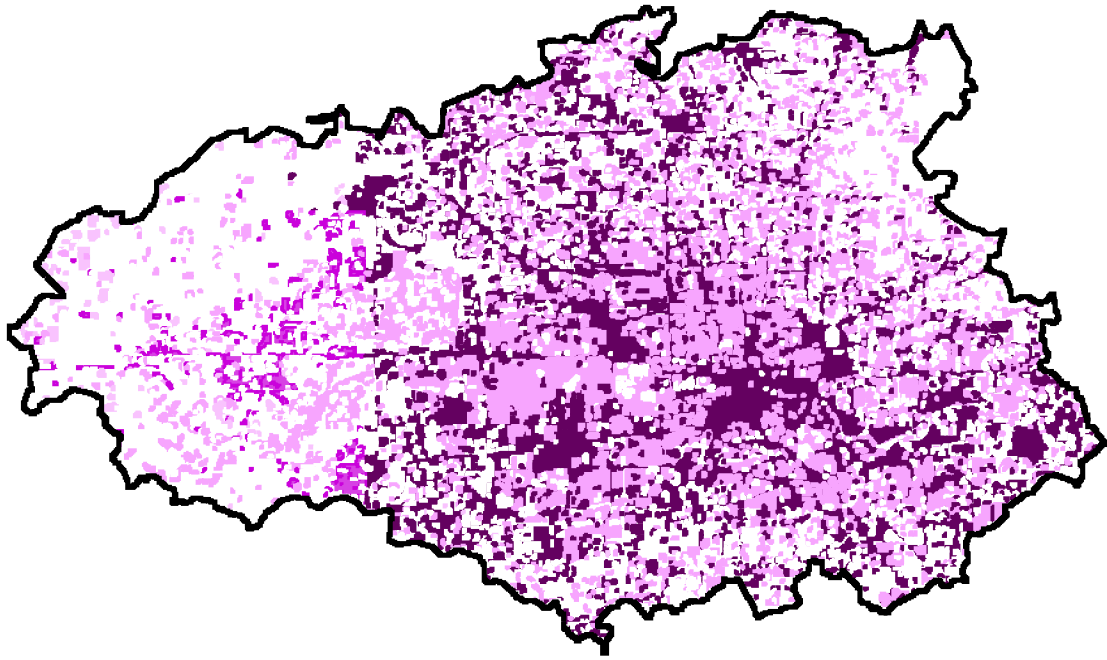


Figure 4-19 Final biochemical oxygen demand (BOD) non-point source loading distribution.

grid: **emcbodgr** = polygrid ( **hsclu**, **emc\_bod**, # , #, **100** )  
*grids the land use coverage into 100m x 100m cells and retains the biochemical oxygen demand EMC as the value in each cell*

grid: **bodann** = ( **emcbodgr** \* **r\_ann** ) / **100**  
*multiplies the emc grid (mg/L) by the runoff grid (mm/yr) and corrects for units to obtain the BOD loading in kg/yr*

grid: **bodfac** = flowaccumulation ( **hscfdr**, **bodann** )  
*performs a weighted flow accumulation on the flow direction grid with the BOD loading grid*

grid: **bodseg** = con ( **out\_seg** > **0**, **bodfac** )  
*puts the flow accumulation value for each segment outlet into a grid called bodseg*

grid: **bodsegin** = int ( **bodseg** )  
*truncates the bodseg grid to have just integer values so that it can be combined with the outlet grid*

grid: **bod\_out** = combine ( **out\_seg**, **bodsegin** )  
*combines the segment outlet grid and the BOD loading grid to obtain a value attribute table of the segment number with the corresponding accumulated load value*

**Procedure 4-8** Steps taken to establish BOD NPS loading over watershed area and into each segment.

**Table 4-8 Non-Point Source BOD Loading into Each Segment**

Segment Number	Flow Accumulation Value (kg/yr)	Incremental Loading (kg/yr)
1	2,922,613	2,922,613
2	4,267,839	1,345,226
3	5,027,275	759,436
4	5,263,575	236,300
5	5,760,916	497,341
6	7,376,897	1,615,981
7	7,419,253	42,356
8	7,671,837	252,584

#### 4.4 POINT SOURCE LOADS

A point source, for this study, is defined as any permitted municipality or industry which discharges directly into the Upper Houston Ship Channel, through an outfall pipe. The point source loadings are assumed to be carried by the baseflow in the system. In addition, point sources do not include dischargers along the bayous, since their inputs are taken into consideration as boundary concentrations in the tributary baseflow within the water quality model. Although a 1994 point source report exists which specifies each

discharger in the Houston area, the report only indicates the TWC segment to which the industry discharged (Armstrong and Ward, 1994). As mentioned in [Section 4.2](#), the TWC segments were on a much larger resolution than the segmentation used for the modeling. Since it was necessary to know the how much point loading was going into each water quality modeling segment, this spatial resolution is unacceptable.

The locations for each point source were obtained from TNRCC (Visnovski, 1996). Of the approximately 1800 point sources in Houston for which the locations were known, about 70 discharge directly into the Upper Houston Ship Channel shoreline. However, only half of these point dischargers have reported BOD measurements from the 1994 report. [Figure 4-20](#) shows just the dischargers along the Upper HSC and those which have BOD measurement attached to them. Once a point coverage of the 1800 dischargers was created in Arc/Info, the 70 shoreline dischargers were isolated, a new point coverage was created, and the available measurement data was joined to the point attribute table (pat) through ArcView. [Table 4-9](#) gives a summary of the total amount of BOD entering into each segment from the point sources. These numbers, although accurate for the reported dischargers, are not representative of the system. In reality, there are about 35 other point dischargers for which the BOD loading is not known. In addition, the locations of Combined Sewer Overflows (CSOs) are not known. CSOs are typically a large source of BOD during heavy rainfall periods (EH&A, 1994). The shortcomings of the point source data are discussed in more detail in [Chapter 5](#).

**Table 4-9 Summary of BOD Discharged into Each Segment, from the Available Point Source Data**

Segment	Number of Point Sources	BOD Loading (kg/day)
1	1	2.62
2	6	19.44
3	1	0.00
4	2	1470.06
5	7	56.39
6	6	234.31
7	1	24.79
8	8	226.71



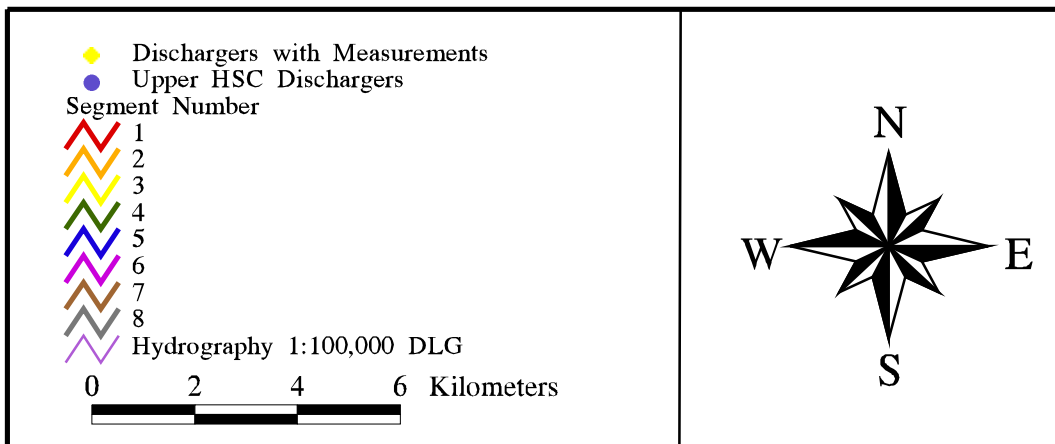
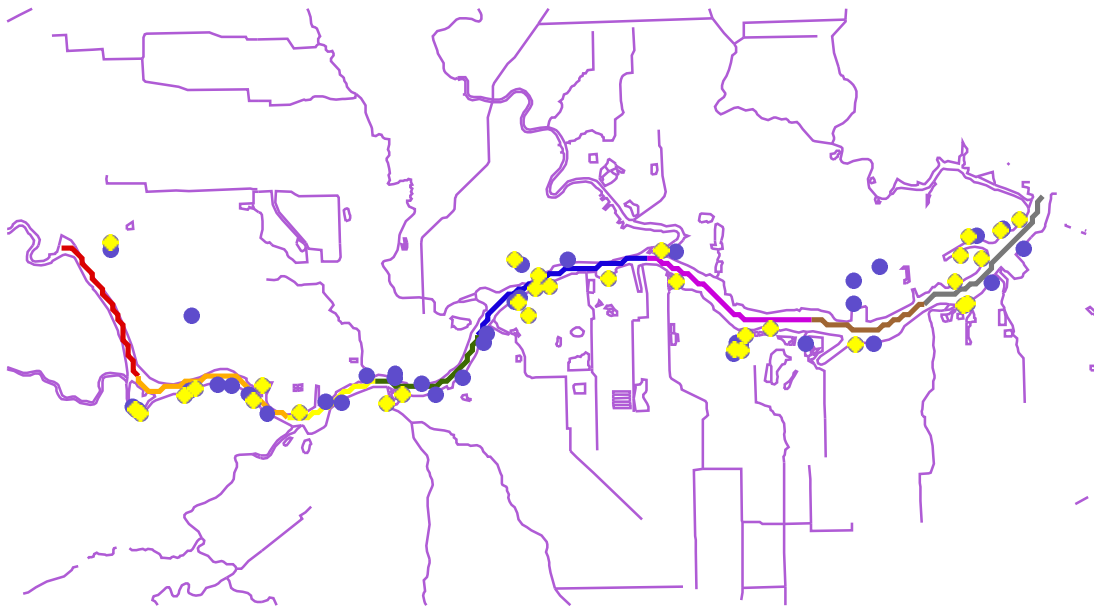


Figure 4-20 Point source dischargers along the Upper Houston Ship Channel. Lighter points are those dischargers for which BOD measurements are available.

## 4.5 WASP5 MODEL DEVELOPMENT

### 4.5.1 Introduction

The Water Quality Analysis Simulation Program (WASP5) is distributed by the USEPA at the Center for Exposure Assessment Modeling (CEAM) in Athens, Georgia. The program consists of a main program, WASP5, and three subprograms: EUTRO5, TOXI5, and DYNHYD5. EUTRO5 is used to model BOD/DO and eutrophication, TOXI5 is used for toxic chemicals and model calibration, and DYNHYD5 for system hydrodynamics. The current version of WASP5, version 5.10, is available as shareware from the USEPA Homepage on the World Wide Web ([ftp://ftp.epa.gov/epa\\_ceam/wwwhtml/wasp.htm](ftp://ftp.epa.gov/epa_ceam/wwwhtml/wasp.htm)). Although the current version has a user interface entitled WISP (WASP Interactive Support Program), the ArcView connection discussed in this project does not utilize this interface. Due to present memory constraints, it is not possible to run WISP, while running the Windows environment necessary for ArcView. As a result, only the EUTRO5 and TOXI5 executables (with their related error and message files) are needed for the ArcView connection.

The WASP5 model, which helps interpret water quality responses of a natural system given man-made pollution or natural events, is written and compiled in Lahey FORTRAN. The program is capable of one, two, or three dimensional modeling; steady-state or time-varying conditions; and model parameter customization. Although the hydrodynamics of a system can be modeled with a separate program or with DYNHYD5, simple hydrodynamics can be simulated directly in EUTRO5 or TOXI5. The basic principle of both the hydrodynamics and the water quality program is the conservation of mass. In other words, the water volumes and water quality constituents are tracked over time and space using a series of mass balancing equations, which are solved with a basin finite differencing method (Ambrose, *et al.*, 1993).

This project concentrated primarily on the BOD/DO model, EUTRO5. Within EUTRO5, there are eight state variables and six levels of complexity. The eight state

variables are ammonia nitrogen, nitrate nitrogen, inorganic phosphorus, phytoplankton carbon, carbonaceous BOD (CBOD), dissolved oxygen (DO), organic nitrogen, and organic phosphorus. As the level of complexity increases, the number of state variables modeled also increases. The complexity levels in EUTRO5 are:

1. Simple Streeter-Phelps BOD/DO with sediment oxygen demand (SOD)
2. Modified Streeter-Phelps with nitrogenous BOD
3. Linear DO balance with nitrification
4. Simple eutrophication
5. Intermediate eutrophication
6. Intermediate eutrophication with benthos

For this study, the first level of complexity was investigated. Level one only considers the state variables, BOD and DO, and incorporates SOD in the mass balances. The equations used in the constituent mass balance for Level one are shown below. Equation 4-3 explains the change in BOD concentration over time, while Equation 4-4 defines the change in DO concentration over time (Ambrose, *et al.*, 1993).

$$\frac{\partial C_5}{\partial t} = -k_D \Theta_D^{(T-20)} \left( \frac{C_6}{K_{BOD} + C_5} \right) C_5 - \frac{v_{s3}(1-f_{DS})}{D} C_5 \quad (4-3)$$

$$\frac{\partial C_6}{\partial t} = k_2 (C_s - C_6) - k_D \Theta_D^{(T-20)} \left( \frac{C_6}{K_{BOD} + C_5} \right) C_5 - \frac{SOD}{D} \Theta_S^{(T-20)} \quad (4-4)$$

where;

- $C_5$  = concentration of carbonaceous biochemical oxygen demand (mg/L)  
(interpreted as total BOD for level one),
- $C_6$  = concentration of dissolved oxygen (mg/L),
- $k_D$  = deoxygenation rate @ 20 °C (/day),
- $\Theta_D$  = deoxygenation temperature coefficient (--),

T	=	temperature ( °C),
K <sub>BOD</sub>	=	half saturation constant for oxygen limitation (mg O <sub>2</sub> /L),
V <sub>s3</sub>	=	organic matter settling velocity (m/day),
f <sub>DS</sub>	=	fraction of dissolved CBOD,
D	=	depth of the overlying water column (m),
k <sub>2</sub>	=	reaeration rate ( /day),
C <sub>s</sub>	=	dissolved oxygen saturation (mg/L),
SOD	=	sediment oxygen demand @ 20 °C (g/m <sup>2</sup> -day),
Θ <sub>s</sub>	=	temperature coefficient (--).

As discussed earlier, EUTRO5 uses a finite differencing method to solve the above equations to explain the change in concentration over time. For spatial distribution, both advective and dispersive flows affect the concentration. Equation 4-5 describes the change in mass in a given segment due to dispersive exchange.

$$\frac{\partial M_{ik}}{\partial t} = \frac{E_{ij}(t)A_{ij}}{L_{cij}}(C_{ik} - C_{jk}) \quad (4-5)$$

where

M <sub>ik</sub>	=	mass of chemical "k" in segment I (g),
C <sub>ik</sub> , C <sub>jk</sub>	=	concentration of chemical "k" in segments "i" and "j" (mg/L),
E <sub>ij</sub> (t)	=	dispersion coefficient time function for exchange "ij" (m <sup>2</sup> /day),
A <sub>ij</sub>	=	interfacial area shared by segments "i" and "j" (m <sup>2</sup> ), and
L <sub>cij</sub>	=	characteristic mixing length between segments "i" and "j" (m).

#### 4.5.2 Model Constants

For EUTRO5, level one complexity, only two constants had to be set in the model: the reaeration rate and the deoxygenation rate. The reaeration rate (k<sub>2</sub> in Eqn. 4-4) helps determine the rate of gas transfer of oxygen from the overlying atmosphere into the surface

water. The deoxygenation rate ( $k_d$  in Eqns. 4-3 and 4-4) explains the rate of oxygenation of the BOD in the water column.

### *Reaeration Rate*

The three main sources of oxygen to water are DO from incoming streams, gas transfer, and photosynthesis from marine plants. Typically, the primary source of oxygen for most natural systems is gas transfer by mixing induced from wind and high flow conditions (Thomann and Mueller, 1987). Within EUTRO5, there are three options on the model's approach to reaeration rate. These options include:

1. A single reaeration constant can be specified, with an internal temperature coefficient of 1.028.
2. Spatially varying reaeration constants can be input and varied through time.
3. EUTRO5 calculates a reaeration rate from water velocity, depth, wind velocity (default set to 0.6 m/sec), water temperature, and air temperature (defaulted to 15°C).

Historically, defining reaeration rates in the Upper HSC has proven to be difficult (Bales and Holley, 1992; Holley, 1996). Since relatively low flow conditions exist in the channel, Option 3 would not produce representative results for the system. In addition, past studies have shown that the reaeration rates are extremely dependent on the mechanical mixing resulting from heavy ship traffic in the area. This mechanical mixing and the lack of hydrodynamic mixing from low flow conditions make it difficult to measure reaeration rates for different areas in the channel (Bales and Holley, 1992). As a result, a spatial variation of the reaeration rate would be hard to establish, since accurate measurements are unlikely to exist. For these reasons, a constant reaeration rate (Option 1) was set initially set at 0.1 /day. This value corresponds to the same number established by Espey, *et al.* (1971) for the Upper HSC.

### *Deoxygenation Rate*

In typical BOD/DO analysis, the total rate of BOD removal is considered. This overall loss rate, termed  $k_r$ , considers the effects of settling and oxidation on BOD. However, in EUTRO5, the effects of settling are taken into account with the second term of Equation (4-3). Consequently, it is only necessary to set the rate at which BOD employs oxygen to stabilize the pollutant material present,  $k_d$ . Since the estimation of  $k_d$  cannot easily be determined from laboratory incubation tests, many studies have attempted to link physical channel characteristics to the deoxygenation rate (Thomann and Mueller, 1987). In addition, as the level of treatment in wastewater treatment plants increases, the BOD which reaches the receiving waters represents the less easily oxidizable portion of the pollutant. Typical values of  $k_d$  range from 0.1 to 0.5 /day for bodies of water deeper than five feet (Thomann and Mueller, 1987). Since the treatment of wastewater discharging to the HSC waters has improved in past years, it would be conservative to choose a deoxygenation rate at the lower end of the scale. In the earlier modeling efforts of the Upper HSC, two different values for  $k_d$  were employed. Espey, *et al.* (1971) and Hydroscience (1968) used spatially and time constant numbers of 0.10 /day and 0.15 /day, respectively. For this research, a value of 0.10 /day for the entire 25 km modeled reach was assumed.

### **4.5.3 Main Segment Characteristics**

Table 4-10 summarizes the physical characteristics of the main segmentation for the Upper HSC. The table also shows additional model parameters needed for EUTRO5, such as sediment oxygen demand, water temperature, and salinity. The physical attributes of the Channel were obtained from the 1971 modeling study (Espey, *et al.*, 1971). A majority of the other parameters were extracted from the 1992 water quality study performed for the Galveston Bay National Estuary Program (GBNEP) (Ward and Armstrong, 1992).

The segment length was originally obtained from Espey, *et al.* (1971). However, due to scaling differences and the method in which the segmentation was imported into GIS (Section 4.2), the actual lengths used in this research varied slightly from those given in the 1971 report (see Table 4-1). The segment depth and area, however, are the same as those

**Table 4-10 Physical Characteristics and Model Parameters for Upper HSC Main Segmentation**

Seg #	Length (m)	Area (m <sup>2</sup> )	Depth (m)	Exchange Coefficient (m <sup>2</sup> /sec)	Temp. (°C)	Salinity (ppt)	SOD (g/m <sup>2</sup> /d)	Θ <sub>s</sub>	Initial DO (mg/L)	Initial BOD (mg/L)	Hydraulic Coefficients*			
											a	b	c	d
1	3238.5	1625.8	9.14	704.5	28.0	5.82	1.5	1.068	1.36	7.18	0.004	0.4	1.2	0.6
2	3538.5	1625.8	9.14	704.5	23.8	7.87	1.5	1.068	1.81	5.04	0.004	0.4	1.2	0.6
3	1990.0	1625.8	9.14	704.5	23.8	7.87	1.5	1.068	1.81	5.04	0.004	0.4	1.2	0.6
4	2631.4	1625.8	6.10	704.5	23.8	7.87	1.5	1.068	1.81	5.04	0.004	0.4	1.2	0.6
5	4038.5	1625.8	9.14	704.5	26.4	9.96	1.5	1.068	0.68	6.25	0.004	0.4	1.2	0.6
6	3697.1	1625.8	6.10	704.5	24.2	9.45	1.5	1.068	2.25	3.53	0.004	0.4	1.2	0.6
7	2490.0	2471.2	7.92	704.5	24.0	9.85	1.5	1.068	1.64	5.06	0.004	0.4	1.2	0.6
8	3404.2	2471.2	7.92	704.5	24.0	9.85	1.5	1.068	1.64	5.06	0.004	0.4	1.2	0.6

Sources: Espey, et al., 1971 and Ward and Armstrong, 1992

\* a,b,c, and d are empirical coefficients as per Equations 4-6 and 4-7

given in the earlier report (Espey, *et al.*, 1971). The exchange coefficients were also obtained from the Tracor modeling effort. In this 1971 study, it was concluded that the dispersion (or exchange) coefficient varied with the magnitude of the net advective flow in the Houston Ship Channel. The study plotted dispersion coefficient versus net flow on a log-log plot and obtained a linear relationship (Figure 4-21). Given the flow values for the Upper HSC, the resulting dispersion coefficient only varies from about 15 to 25 mi<sup>2</sup>/day. Since this range is rather small, the 1971 study assumed an average flow for all segments upstream up the San Jacinto River and chose one exchange coefficient of 809 m<sup>2</sup>/sec (27 mi<sup>2</sup>/day) for the entire 25 km reach. However, the constant value chosen in 1971 seemed high given the graph and average flow. The actual data for this graph was not given in the report and the scale of the chart was relatively large. As a result, it was difficult to reproduce this function accurately. As a result, the same graph was used to obtain a new dispersion coefficient which seemed more representative. The average flow for the eight segments was recalculated at about 900 ft<sup>3</sup>/sec, resulting in an exchange coefficient of 704.5 m<sup>2</sup>/sec (24.5 mi<sup>2</sup>/day).

The temperature, salinity, initial (in time and space) DO, and initial BOD measurements were all obtained from average measurements provided in Ward and Armstrong (1992). Five of the hydrographic segments discussed in Section 4.2 (see Figure 4-4) encompassed the eight main segments of this present modeling effort. Table 4-11 shows the correspondence of the main segmentation to this hydrographic segmentation.

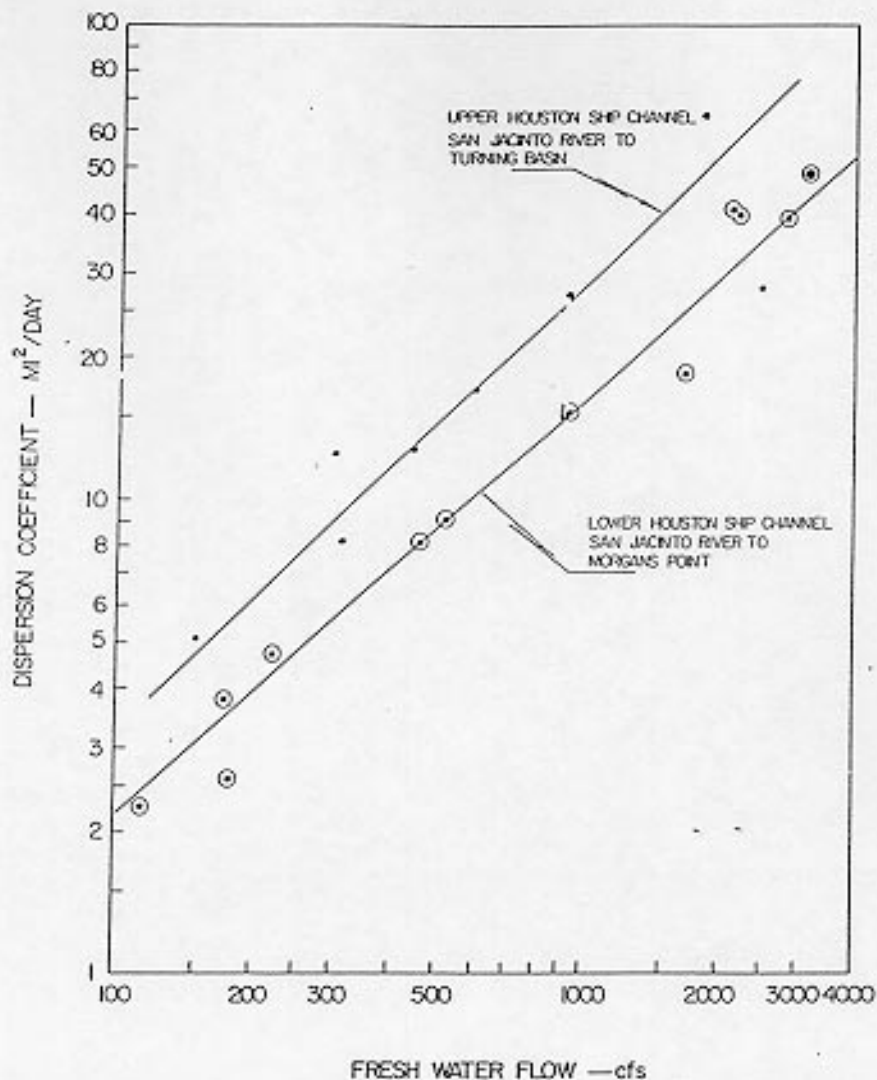
**Table 4-11 Hydrographic Segments Corresponding to Main Segmentation**

Hydrographic Segments	Corresponding Main Segment (s)
H12	17 *
H14	7,8
H15	6
H16	5
H17	2,3,4
H18	1
H20	10 *

*Source: Ward and Armstrong, 1992*

\* Boundary Segments





**Figure 4-21** Original graph from Espey, *et al.* (1971) showing dispersion (i.e. exchange) coefficient versus net freshwater flow. For the Upper HSC, an average flow value of 900 cfs was assumed and an overall exchange coefficient of 24.5 mi<sup>2</sup>/day was chosen.

The surface layer of the sediment layer directly under the water column usually undergoes aerobic decomposition and, in the process, removes oxygen from the overlying water. This effect is usually measured in sediment oxygen demand (SOD). Since actual measurements of SOD for the Upper HSC could not be located, a constant value of 1.5 g/m<sup>2</sup>/day was assumed for the entire reach. This number corresponds to the approximate average of the range for estuarine mud given in Thomann and Mueller (1987). In addition, the SOD  $\Theta$  used to correct for temperatures varying from 20°C was set at the typical value of 1.068 (Thomann and Mueller, 1987).

Finally, the hydraulic coefficients are those coefficients and exponents related to the following equations:

$$V = aQ^b \quad (4-6)$$

$$D = cQ^d \quad (4-7)$$

where:

- V = channel velocity (m/sec),
- Q = channel flow (m<sup>3</sup>/sec),
- D = channel depth (m), and
- a,b,c,and d = empirical coefficients or exponents.

For rectangular channels, values of 0.4 and 0.6 can be assumed for b and d, respectively (Ambrose, *et al.*, 1993). Although the Upper HSC is not exactly rectangular, it is similar enough to assume these values without considerable error. Then, since the flow, depth, and velocity (flow/area) are known for each segment, average values of a and c are calculated to be 0.004 and 1.2, respectively. Since EUTRO5 only uses these numbers for reaeration and volatilization calculations and not transport functions, these coefficients do not effect the present model established for the Channel (Ambrose, *et al.*, 1993).

#### 4.5.4 Boundary Segment Characteristics

Model boundaries consist of those segments that import, export, or exchange water with locations outside the main network. A boundary segment is either a tributary inflow, a downstream outflow, or an open water end of the model network across which dispersive mixing can occur. In the Upper HSC, each main segment was assigned a boundary condition to incorporate the main tributaries flowing into the Channel. [Table 4-12](#) gives the characteristics and necessary model parameters for these boundary conditions established for the Upper HSC, while [Figure 4-22](#) shows the conceptual segmentation for the system. Also in [Table 4-12](#) is the name of each bayou assigned to the boundary condition.

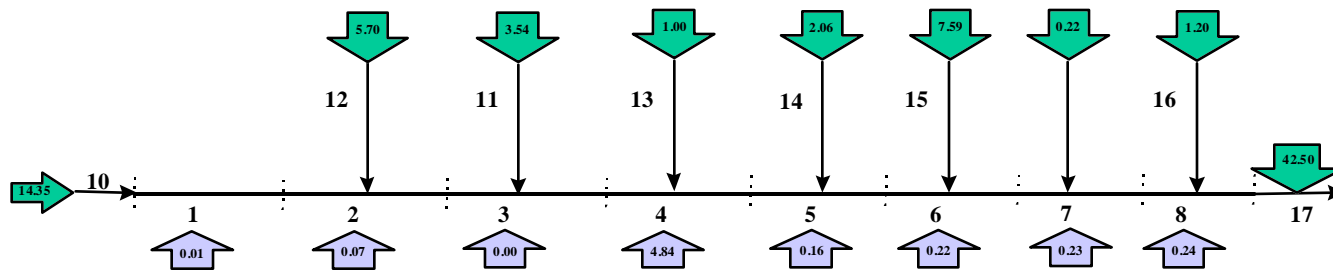
The exchange coefficients, cross-sectional areas and depths were all obtained from Espey, *et al.* (1971). [Figure 4-21](#) was again utilized to obtain the exchange coefficient for segment 17, while a small dispersion coefficient of  $119 \text{ m}^2/\text{sec}$  ( $4 \text{ mi}^2/\text{day}$ ) was assumed for all tributaries. All boundaries, excluding the Buffalo Bayou (segment 10), had lengths set at 3.2 km (2 miles) to emulate a somewhat "infinite" condition. The Buffalo Bayou was set at 8 km (5 miles).

For the boundary BOD concentration in these segments, a number of sources were consulted. For segments 10 and 17, Ward and Armstrong (1992) provided average BOD measurements from the hydrographic segmentation established in that report (see [Table 4-11](#)). For the remaining boundary segments, two sources were compared and values for BOD were assumed from the measurements taken for these studies (Armstrong and Ward, 1994; TDWR, 1984). In relation to DO values, a conservative value of 5 mg/L was assumed for all boundaries; except segments 10 and 17, where averages from Ward and Armstrong (1992) were employed. The value of 5 mg/L is the widely accepted minimum needed to maintain marine life (Thomann and Mueller, 1987). Some DO measurements, which ranged from 6 - 7 mg/L, exist for some tributaries (EH&A, 1994). However, since these measurements were taken only during storm events, they were probably not representative of the baseflow conditions because high flow conditions during storms usually result in higher DO

**Table 4-12 Physical Characteristics and Model Parameters for Upper HSC Boundary Segmentation**

Seg #	Length (m)	Area (m <sup>2</sup> )	Depth (m)	Exchange Coefficient (m <sup>2</sup> /sec)	Temp. (°C)	Salinity (ppt)	SOD (g/m <sup>2</sup> /d)	Θ <sub>s</sub>	Initial DO (mg/L)	Initial BOD (mg/L)	Down-stream Segment	Name
9	2500.3	0.10	0.10	0.001	20.0	0.00	1.5	1.068	0.00	0.00	0	Sediment Layer
10	8046.9	465.4	9.14	119.9	23.6	1.82	1.5	1.068	3.03	8.14	1	Buffalo Bayou
11	3218.7	505.4	4.88	119.9	20.0	0.20	1.5	1.068	5.00	8.40	3	Sims/Berry Bayous
12	3218.7	168.2	3.05	119.9	20.0	0.20	1.5	1.068	5.00	6.90	2	Brays Bayou
13	3218.7	185.8	1.83	119.9	20.0	0.20	1.5	1.068	5.00	8.00	4	Vince Bayou
14	3218.7	528.6	3.35	119.9	20.0	0.20	1.5	1.068	5.00	8.60	5	Hunting Bayou
15	3218.7	717.2	10.06	119.9	20.0	0.20	1.5	1.068	5.00	5.90	6	Greens/Hall Bayou
16	3218.7	260.1	3.96	119.9	20.0	0.20	1.5	1.068	5.00	6.00	8	Carpenters Bayou
17	3218.7	2471.2	4.88	119.9	25.3	10.90	1.5	1.068	3.64	7.42	0	Dwnstr. Boundary

Sources: Espey, et al., 1971 and Ward and Armstrong, 1992.



**Figure 4-22** Schematic of final segmentation used in Upper HSC model. Numbers next to lines indicate segment numbers. Outline also shows total flows entering each segment (in m<sup>3</sup>/sec). Downward pointing arrows are tributary flows, while the upward arrows are point source inflows from Armstrong and Ward (1994). Figure is not to scale.

concentrations. Finally, the SOD for each boundary segment and its corresponding  $\Theta$  were set at the same values as for the main segmentation (Thomann and Mueller, 1987).

Temperature and salinity values for most boundary segments were set at 20 °C and 0.2 parts per thousand (ppt), respectively. The only exceptions were the concentrations and temperatures in segments 10 and 17. Since these two segments were located along the main channel, the average measurements determined in Ward and Armstrong (1992) could be used. Since the remaining boundaries were relatively freshwater inflows, 0.2 ppt was a reasonable assumption for salinity.

One final boundary segment that needs to be established is a benthic sediment layer (segment #9). This layer, which acts as a sink for particulate BOD due to settling, is established along the entire length of the main network (segments 1 through 8). The depth of this segment was set at 10 cm to represent the active layer of the sediment. In addition, a vertical exchange rate of  $10^{-4}$  m<sup>2</sup>/sec was established between the water column and sediment pore water to simulate a possible sink or source of DO.

#### **4.5.5 Flow and Baseflow**

As mentioned earlier, each main segment has an established amount of steady state total flow and baseflow. The total flow, which consists of runoff plus baseflow, is used to represent average year conditions. For this case the runoff is assumed to carry the non-point source loadings, while the baseflow carries both point source loadings. In contrast, the dry weather conditions, only considers the baseflow as flow into the system. This condition is meant to represent a worst case scenario, where no runoff enters the channel. For this situation, only the point source loadings are input to the model, since there is no runoff to carry any non-point source pollution.

## *Flow*

To determine the total flow, a rainfall/flow/urbanization relationship was defined. This relationship is very similar to the rainfall/runoff equation developed in [Section 4.3](#). Following the same reasoning used to distribute the runoff over the watershed area, the flow (i.e. baseflow not subtracted) was also distributed over the land surface. [Figure 4-23](#) illustrates the final relationship between percentage of precipitation which become flow versus relative urbanization of a given USGS subwatershed. The 1:1 line on the graph illustrates the relationship which exists if the relative runoff coefficient represented the percentage of precipitation which eventually becomes flow. The actual values that result in the equation are about 80 % of the values on the 1:1 line. Using the steps outlined in [Procedure 4-7](#) and substituting the new equation from [Figure 4-23](#), grids of flow/precipitation and flow over the entire watershed area were calculated. Once this flow grid was determined, a weighted flow accumulation was performed on the flow direction grid (see [Procedure 4-9](#)). The accumulated flow values at each segment outlet were found and converted from mm/yr/ha to m<sup>3</sup>/sec. These final numbers are shown in [Table 4-13](#), along with the flow measurement for the San Jacinto River entering segment 17. Although some of this flow enters the segment by way of diffuse runoff, for modeling purposes, it was all assumed to enter the segment at the boundary segment ([Figure 4-22](#)).

grid: **ffac = flowaccumulation ( hscfdr, flcalc)**

*performs a weighted flow accumulation on the flow direction grid, weighted with the grid of flow*

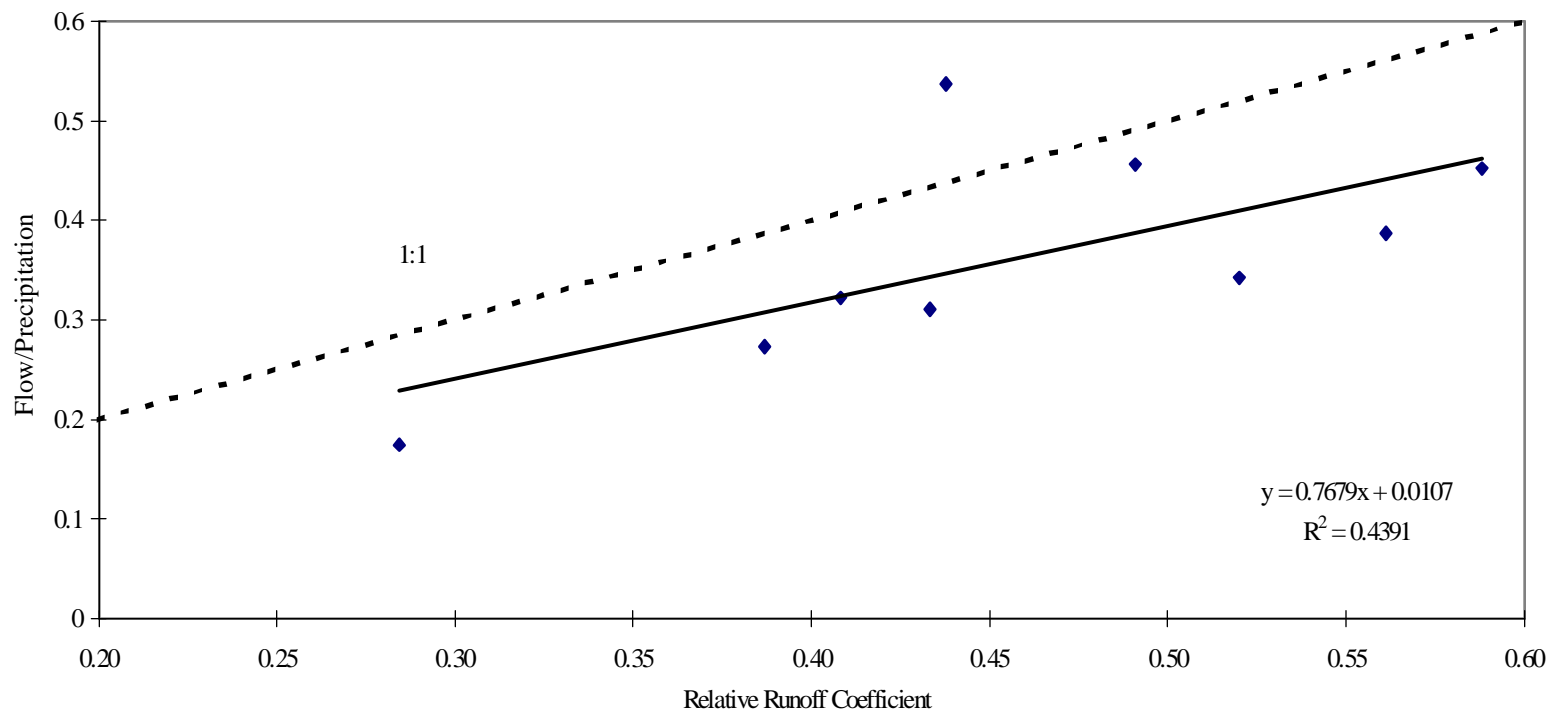
grid: **ffacint = int( ffac )**

*truncates the weighted flow accumulation values so they can be combined with the outlet grid*

grid: **flout = combine ( out\_seg, ffacint )**

*combines the segment outlet grid and the flow accumulation grid to obtain a value attribute table of the segment number with the corresponding accumulated flow value*

**Procedure 4-9** Steps taken to determine total flow (mm/yr/ha) into each main segment. Flow grid was established using method outline in [Procedure 4-7](#) and relationship shown in [Figure 4-23](#)



**Figure 4-23** Relation between ratio of mean annual surface flow divided by mean annual precipitation to runoff coefficients for the same area, based on land use and standardized runoff coefficient table. 1:1 line illustrates the relationship that would exist if the runoff coefficient was an absolute measure of the percentage of precipitation which becomes flow.



**Table 4-13 Final Flow, Runoff and Baseflow Values for Main Segment**

Segment	Incremental Total Flow (m <sup>3</sup> /sec)	Point Source Flow* (m <sup>3</sup> /sec)	Incremental Runoff (m <sup>3</sup> /sec)	Incremental Baseflow (m <sup>3</sup> /sec)	Baseflow: % of Total Flow
1	14.35	0.01	10.96	3.39	24
2	5.70	0.07	4.57	1.13	20
3	3.54	0.00	2.75	0.79	22
4	1.00	4.84**	0.79	0.21	21
5	2.06	0.16	1.66	0.40	19
6	7.59	0.22	5.68	1.91	25
7	0.22	0.23	0.18	0.04	18
8	1.20	0.24	0.93	0.27	23
17	42.5***	n/a	n/a	n/a	n/a

\* Source: Ward and Armstrong (1992)

\*\* 3.05 cms is power plant outflow

\*\*\* From Espey, et al., 1971 San Jacinto River flow

### Baseflow

The final steady state baseflow for the water quality model was calculated by subtracting the runoff determined in Section 4-3 from the flow determined above. The accumulated runoff into each segment was established in the same way the flow to each segment was calculated in Procedure 4-9, but the flow accumulation was weighted with the runoff grid instead of the flow grid. The total flow, runoff, and final baseflow to each segment is shown in Table 4-13. The last column of Table 4-13 shows the percentage of total flow which is composed of baseflow. These values are relatively consistent from segment to segment, ranging from 18% for segment 7 to 25% for segment 6, with the average being 22%. This average compares well to the average percentage discussed in Section 4.3 for the calculated baseflow at each USGS gauge station (also 22%). As mentioned earlier, the baseflow is meant to transport the point source loadings and represent dry weather conditions when input to the model without the runoff.

### Point Source Flows

Also shown in Table 4-13 are the point source flows into each segment (Armstrong and Ward, 1994). As mentioned in Section 4.4, these flows only account for about half of the point source dischargers along the channel. Therefore, the entire system cannot be represented accurately until more data is obtained on the other dischargers. In addition, the part of the point source flow entering segment 4 (3.05 m<sup>3</sup>/sec), although very large, is a

power plant. Since power plants recycle a majority of their intake, this measurement is ignored. For the final input to the model, only the steady state flow conditions from the tributaries were considered. The point source flows were ignored since they represented only about 2.5% of the total flow shown above. Finally, since there was no major tributary entering segment 7 and the calculated flow was minimal, this flow value was ignored in the final input to the model.

#### 4.5.6 Constituent Loading

As discussed in Sections 4.3 and 4.4, the NPS and point source BOD loading was determined for each segment. These steady state values are summarized in Table 4-14. The point source data did not have flows "attached" to them. But, since these flows are relatively small, the error introduced by their omission is minimal. The non-point source, however, were obtained using the steady state runoff results discussed in Section 4.3. This aspect does affect the model input and is discussed further in Section 4.5.8.

**Table 4-14 Final Steady State Loading Values for BOD**

Segment Number	NPS BOD Loading (kg/day)	Point Source BOD Loading* (kg/day)
1	8007.16	3.26
2	3685.55	24.16
3	2080.65	0.00
4	647.40	1827.28
5	1362.58	70.09
6	4427.35	291.25
7	116.04	30.81
8	692.01	281.80
<b>Total</b>	<b>21,018.74</b>	<b>2,528.65</b>

\* Source: Ward and Armstrong (1992)

#### 4.5.7 Model Calibration

The final model was calibrated to ensure that it accurately represented the Upper HSC. Salinity was chosen as the chemical to calibrate WASP5 because it is considered a conservative material and it is an excellent water mass tracer. The objective of the calibration was to produce results similar to those reported in Ward and Armstrong (1992).

To accomplish this objective, boundary segments 10 and 17 were set to their long-term average values of 1.82 and 10.9 ppt, respectively (Ward and Armstrong, 1992), while all other main segments were set to 0 ppt and boundary segments to 0.2 ppt. WASP5's subprogram, TOXI5, was then run at a level one complexity until quasi-steady state was reached in the Upper HSC. The input file for this calibration is shown in [Appendix D-1](#) and the results are discussed in [Chapter 5](#).

#### 4.5.8 Model Runs

The input file for the BOD/DO model run representing average year conditions is in [Appendix D-2](#). Since this study was centered on connecting the water quality model to GIS, time constraints resulted in fewer model runs than originally desired. However, the model was successfully executed with loadings, flows, and parameters developed in this section, representing long term, steady state conditions. In addition, the sensitivity of the model to the constants ( $k_2$  and  $k_d$ ) was also investigated. Further research could provide a method for studying the sensitivity of the model to the segmentation, to changes of the water quality due to land use changes or engineering practices; and to time-varying inputs; all using GIS as the interface to implement these changes.

An overview of the four cases investigated in this research is shown in [Table 4-15](#). For cases 1, 3, and 4, the point source and NPS loadings along with the steady state flows were applied to the Upper HSC. Case 2 was established to model "dry weather" conditions, resulting in just steady state baseflow ([Table 4-13](#)) being used with the point source loadings.

**Table 4-15 Cases Applied for BOD/DO Model in the Upper Houston Ship Channel**

Case	Flow Conditions	NPS Loads?	Point Source Loads?	$k_2$ (/day)	$k_d$ (/day)
I	Average Year	Y	Y	0.1	0.1
II	Dry Year	N	Y	0.1	0.1
III	Average Year	Y	Y	0.5	0.1
IV	Average Year	Y	Y	0.1	0.3

Special attention was given to the boundary concentration since WASP5 accounts for loadings into the main reach from boundaries. The model applies the following equation to determine the loading entering a downstream segment from an upstream boundary:

$$V_i S_{bik} = Q_{0i}(t) C_{bik} \quad (4-8)$$

where:

- $V_i$  = volume of segment i ( $m^3$ ),
- $S_{bik}$  = boundary loading rate response of chemical "k" in segment, "i" ( $g/m^3$ -day),
- $Q_{0i}$  = upstream inflow into boundary segment, "i" ( $m^3/day$ ), and
- $C_{bik}$  = concentration in boundary segment, "i" ( $mg/L$ ).

The way the model input is set, this boundary concentration is applied to the total flow; however, the loading from the runoff has already been considered with the non-point source calculation. As a result, if this calculation does occur with the total flow, it could be viewed as a somewhat "double load" to the system. As a result, the boundary concentrations were adjusted so that the loading calculated in Equation. 4-8 was equal to that of the boundary concentration multiplied by just the steady state baseflow. To accomplish this adjustment, the following equations were employed:

$$V_i S_{bik} = Q_{bf}(t) C'_{bik} \quad (4-9)$$

$$C'_{bik} = \frac{V_i S_{bik}}{Q_{tot}} \quad (4-10)$$

where:

- $Q_{bf}$  = steady state baseflow upstream of segment "i" ( $m^3/day$ ),
  - $C'_{bik}$  = adjusted concentration for boundary segment "i" ( $mg/L$ ),
  - $Q_{tot}$  = total flow upstream of segment "i", and
- other variables are previously defined.

## 4.6 GIS/WASP5 CONNECTION

### 4.6.1 Introduction

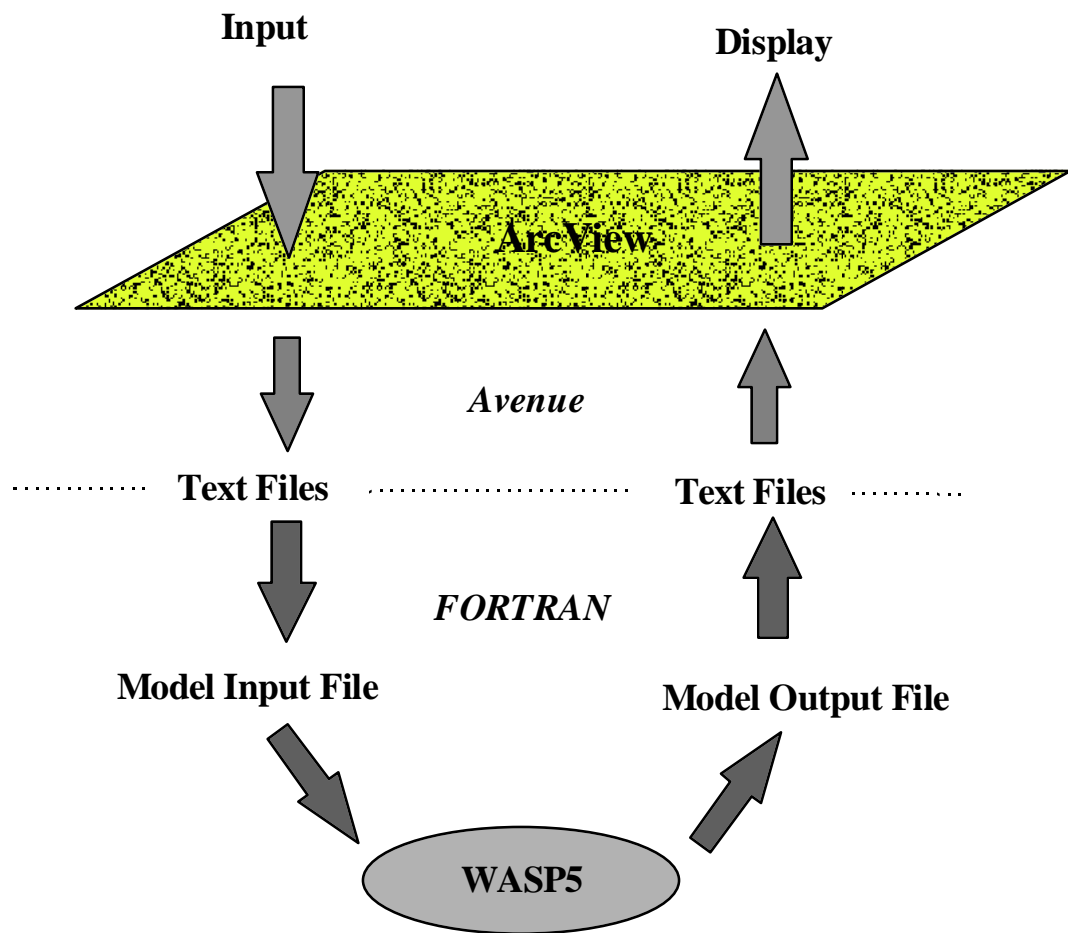
The concept behind this GIS/model connection, shown in [Figure 4-24](#), allows programs within GIS to produce the necessary information for the exterior model's input files. The GIS software then executes formatting programs in order to obtain a properly spaced input file and executes the model. Finally, the model output is processed, imported back into GIS, and viewed as charts, tables, and coverages.

Specifically for this research, the model input is read by the ArcView programming language Avenue (see [Section 4.6.2](#)) from tables and coverages within a customized project. The Avenue programs, consisting of compiled scripts, are executed through the ArcView interface by way of menu choices. This information is written into 13 "free form" text files ([Appendix E](#)): one for each WASP5 input block and three character files holding filenames, and the model description. These text files become the input for a FORTRAN program which formats all of the data into WASP5's space sensitive input file. After WASP5 is executed, the model's output file is processed by another FORTRAN program to read the necessary results. These results are printed to a new text file in the form of an array and imported back into ArcView, using Avenue. After importing, the output is viewed in the form of tables, charts, and newly created coverages.

### 4.6.2 Programming Involved

#### *Avenue*

Avenue is an object-oriented programming language which is used to create customized ArcView projects. The language works by creating "objects" of project documents, such as views, coverages, and tables and processing, reading, or manipulating information from these objects to create additional objects within the project. As with any typical programming language, loops, input statements, and print statements are possible using Avenue. Once an Avenue script is written and compiled, it is "attached" to a menu



**Figure 4-24** Concept behind GIS/WASP5 model connection. Avenue, the programming language in ArcView, creates text files of the necessary model information. FORTRAN formatting programs then read this information and write the WASP5 input file. After the model is run, the output file is read by another FORTRAN program and imported back into ArcView, via Avenue.

selection or tool bar button and executed numerous times. It is important to remember that the scripts in Avenue are based on a particular customized project. For this reason, to re-execute a script, the original objects, such as a given table or view, must always be present in the project.

For this research, 33 Avenue scripts (see [Appendix F](#)) were written to create the model connection shown in [Figure 4-24](#). An overview of each of these scripts is outlined in [Table 4-16](#). These scripts wrote the free form text files for each model input block, executed the FORTRAN programs from the ArcView interface, and created charts and coverages to present the output. All of these scripts, except one, are invoked through menu choices in the customized project ([Figure 4-25](#)).

#### *FORTRAN*

Also included in the connection are four FORTRAN programs used to format the input and read the output (see Appendices C-4 and C-5). These programs, also outlined in [Table 4-16](#), were written and compiled in Microsoft FORTRAN. Since the final connection performed BOD/DO modeling, along with model calibration, different programs were needed to deal with both TOXI5 (calibration model) and EUTRO5 (BOD/DO model) input and output files. Consequently, separate input formatting programs and output reading programs were written for each model, resulting in four compiled FORTRAN codes.

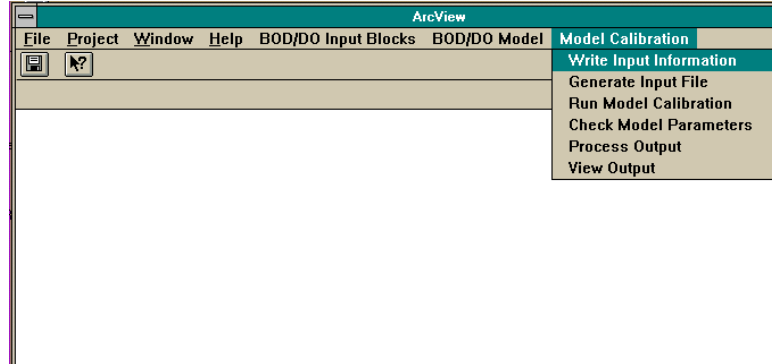
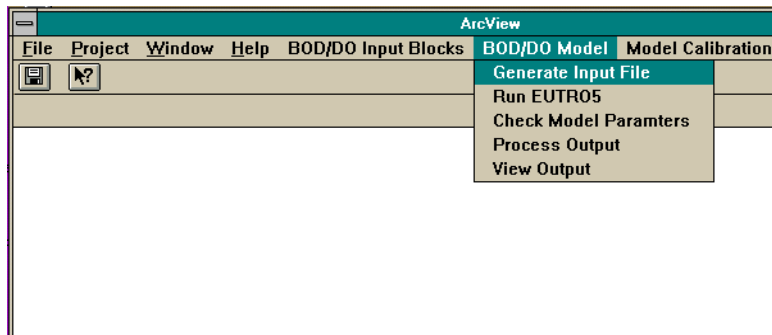
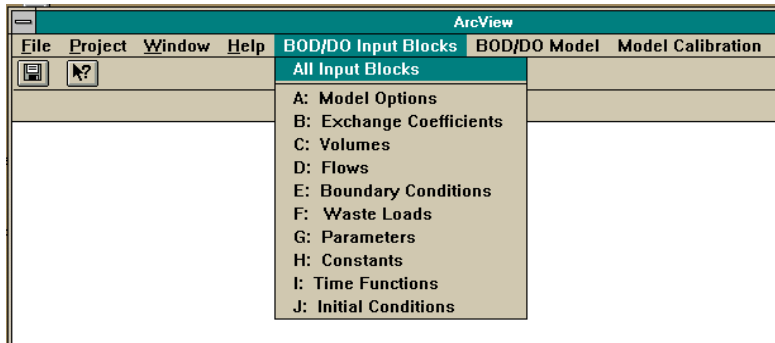
#### **4.6.3 Menu Overview**

There are three primary choices on the menu bar related to the model connection ([Figure 4-25](#)): *BOD/DO Input Blocks*, *BOD/DO Model*, and *Calibration Model*. Under the first choice, the user may write the necessary information for a EUTRO5, level one complexity, steady-state model run. There are options to either create the all 13 of the text files with the input file information (*All Input Blocks*), or to change just one input block and its related text file by selecting it off of the menu. The second choice, *BOD/DO Model*, is executed after all of the input block information has been written to text files. The first step under this option is *Generate Input File*. By invoking this command, the FORTRAN

**Table 4-16 Scripts and Programs Created for ArcView/WASP5 Connection**

Program or Script Name	Type	Function
all	Avenue	Runs all scripts to create texts file for EUTRO5 input file
bttm	Avenue	Controls the "bug" icon on the "Segmentation" view to plot a chart
calgen.exe	FORTRAN	Formats the 13 text files into a TOXI5 calibration input file
calinputa	Avenue	Writes the text files for Input Block A in the model calibration
calinputall	Avenue	Runs scripts which create text files for TOXI5 input file
calout.exe	FORTRAN	Processes the "tdf" file to produce an array of salinity at each segment vs. time
cal_parchk	Avenue	Checks the parameters on the output file to process for the calibration model
eutrorun	Avenue	Runs the EUTRO5 model from ArcView
frame1 - frame4	Avenue	Produces a new theme on the "Segmentation" to observe the change in concentration over time; each script is time delayed
gen_file_eutro	Avenue	Executes outgen.exe from ArcView
gen_file_toxi	Avenue	Executes calgen.exe from ArcView
help	Avenue	Executes the help file for the ArcView/WASP5 connection
inputa	Avenue	Writes the text files for Input Block A for the BOD/DO model input file
inputb	Avenue	Writes the text files for Input Block B for the BOD/DO or calibration model input file
inputc	Avenue	Writes the text files for Input Block C for the BOD/DO or calibration model input file
inputcale	Avenue	Writes the text files for Input Block E for the calibration input file
inputcalf	Avenue	Writes the text files for Input Block F for the calibration input file
inputcalg	Avenue	Writes the text files for Input Block G for the calibration input file
inputcalh	Avenue	Writes the text files for Input Block H for the calibration input file
inputcalj	Avenue	Writes the text files for Input Block J for the calibration input file
inputd	Avenue	Writes the text files for Input Block D for the BOD/DO or calibration input file
inpute	Avenue	Writes the text files for Input Block E for the BOD/DO input file
inputf	Avenue	Writes the text files for Input Block F for the BOD/DO input file
inputg	Avenue	Writes the text files for Input Block G for the BOD/DO input file
inputh	Avenue	Writes the text files for Input Block H for the BOD/DO input file
inputi	Avenue	Writes the text files for Input Block I for the BOD/DO or calibration input file
inputj	Avenue	Writes the text files for Input Block J for the BOD/DO input file
mod_parchk	Avenue	Checks the parameters on the output file to process for the BOD/DO model
modout.exe	FORTRAN	Processes the "edf" file to produce arrays of DO and BOD at each segment vs. time
outgen.exe	FORTRAN	Formats the 13 calibration text files into a EUTRO5 input file
run_calout	Avenue	Executes calout.exe and creates a dbf table for salinity at each segment over time
run_modout	Avenue	Executes modout.exe and creates dbf tables for DO and BOD at each segment over time
toxirun	Avenue	Executes TOXI5 from ArcView
vwout	Avenue	Controls the output presentation processes -- all five options





**Figure 4-25** Customized menus created to run Avenue scripts. *BOD/DO Input Blocks* creates the text files needed for the EUTRO5 model run. *BOD/DO Model* formats this information and runs the EUTRO5 model. *Model Calibration* assists the user in running TOXI5 to calibrate the model with salinity.

program `outgen.for` ([Appendix C-4](#)) is invoked and the properly formatted WASP5 input file is created from the 13 text files ([Appendix E](#)). Once complete, EUTRO5 may be executed with the second option under this menu choice, *Run EUTRO5*. After the model run, the output is processed and viewed by choosing the last three menu options: *Check Model Parameters*, *Process Output*, and *View Output*, in succession (see [Section 4.6.8](#)). The last choice on the menu bar is *Model Calibration*. The options located under this choice are similar to those options found under *BOD/DO Model*. However, with the model calibration, all of the input blocks are always created with the first menu option: *Write Input Information*. In other words, the ability to change just one input block of the entire input file is not possible; primarily because the calibration input file is typically much simpler to create than the BOD/DO model input file. In addition, WASP5's subprogram, TOXI5 is executed to perform the model calibration, using salinity as the conservative tracer (*Run Model Calibration*). The options related to the output processing and viewing are identical to the output commands found under *BOD/DO Model*.

[Table 4-17](#) presents a more detailed description of the processes which occur during the *BOD/DO Input Blocks* menu choices. Specifically, [Table 4-17](#) gives the menu option, the tables read during its execution, and the text files produced for each option. For a detailed description of the tables in the connection, refer to [Appendix A](#) and the following section.

#### **4.6.4 Tables Needed for Connection**

Presently, eight tables within ArcView are needed to run the model connection. These various tables, outlined below, are read by Avenue as objects and the information written to text files. It is extremely important that all tables listed here are included and opened in the project, before the model input file is created. It is also important that all tables are named (either by alias or real name) as they are written below. In addition, since the Avenue scripts contain unit conversions which are performed on values within the tables, the field names, field units, and field order for each table have to adhere to those outlined in

**Table 4-17 Detailed Description of BOD/DO Input Blocks Menu Options**

Menu Option	Tables Read from Project	Fields Read from Table	Text File(s) Produced	Tables Produced
A: Model Options	Boundary Segments	grid-code	a.txt	none
	Main Segment Parameters	grid-code	title.txt	
	Main Segmentation	grid-code	inptnme.txt	
B: Exchange Coefficients	Boundary Segments	act_length	b.txt	none
		dwnstr_seg		
		ex_coeff		
		grid-code		
		perpin		
		type		
		upstr_seg		
	width			
	Main Segment Parameters	dwnstr_seg		
		ex_coeff		
grid-code				
C: Volumes	Boundary Segments	act_length	c.txt	none
		bttm_seg		
		grid-code		
		type		
		x_area		
	Main Segment Parameters	bttm_seg		
		grid-code		
		length		
		type		
		x_area		
D: Flows	Boundary Segments	dwnstr_seg	d.txt	flow.dbf
		grid-code		
		type		
		upstr_seg		
	Flow Accumulation Values	flow accumulation		
		grid-code		
	Main Segment Parameters	dwnstr_seg		
		grid-code		
		upstr_seg		
	Runoff Accumulation Values	grid-code		
rfac				
E: Boundary Conditions	Boundary Segments	grid-code	e.txt	none
		type		
		int_do		
		int_bod		

**Table 4-17 (cont) Detailed Description of BOD/DO Input Blocks Menu Options**

Menu Option	Tables Read from Project	Fields Read from Table	Text File(s) Produced	Tables Produced
F: Waste Loads	BOD Loading Values Point Source BOD	accumulated bod loading segment bod	f.txt	load.dbf
G: Parameters	Boundary Segments  Main Segment Parameters	grid-code sal temp sod grid-code sal temp sod	g.txt	none
H: Constants	none	none	h.txt	none
I: Time Functions	none	none	i.txt	none
J: Initial Conditions	Boundary Segments  Main Segment Parameters	grid-code int_do int_bod grid-code int_do int_bod	j.txt	none

**Appendix A.** Below is an alphabetical list of the tables and a small description of each; the specific attributes for each table and its corresponding units are found in **Appendix A.**

#### *BOD Loading Values*

A table (value attribute table -- INFO format) of the BOD flow accumulation values for each "outlet" (i.e., its most downstream point) of the main water segments (see **Procedure 4-8**). These values are usually obtained by running a flow accumulation over the watershed area, weighted by a grid of BOD load, in Arc/Info's subprogram, Grid. This weighted flow accumulation is then "combined" with a grid of the outlet points to obtain this table (see **Figure 4-26**). This table accounts for the NPS loading from the watershed land surface.

#### *Boundary Segments*

A dBase file (dbf) table which contains all boundary segments (water and sediment), and their corresponding parameters. This file can be created in directly in dBase, or in Arc/Info and exported out of ArcView to a dBase format. If the table is not a dBase format,

BOD Loading Values	
<i>Grid-code</i>	<i>Accumulated BOD Loading</i>
1	2922613
2	4267839
3	5027275
4	5263575
5	5760916
6	7376897
7	7419253
8	7671837

Flow Accumulation Values	
<i>Flow Accumulation</i>	<i>Grid-code</i>
45257496	1
63230692	2
74402944	3
77553856	4
84044216	5
107994720	6
108688720	7
112447960	8

Runoff Accumulation Values	
<i>Accumulated Runoff Value</i>	<i>Grid-Code</i>
34567252	1
48963920	2
57637824	3
60140312	4
65362556	5
83266360	6
83835368	7
86783136	8

**Figure 4-26** Flow accumulation tables needed in model connection. All of these tables are value attribute tables (vat's) which were formed by performing a combine with the respective weighted flow accumulation grid and a grid of the outlet points to each segment. The order of the fields is not important in these tables; however the accumulation units are important (see [Appendix A](#)).

it is not possible to edit and change parameters in the table. The numbering of the boundary segments starts with the next number after the last main segment. A portion of this table is shown in [Figure 4-27](#).

#### *Flow Accumulation Values*

A table (a value attribute table -- INFO format) of the flow accumulation values for each "outlet" (i.e., its most downstream point) of the main water segments ([Procedure 4-9](#)). These values are usually obtained by running a flow accumulation over the watershed area, weighted by a grid of flow depth, in Arc/Info's subprogram, Grid. This weighted flow accumulation is then "combined" with a grid of the outlets to obtain this table ([Figure 4-26](#)).

#### *Main Segment Parameters*

A dbf file which contains the attributes of the main segments for the water quality model. This file can be created directly in dBase, or in Arc/Info and exported out of ArcView to a dBase format. If the table is not a dBase format, it is not possible to edit and change parameters in the table. This table is joined, during the generation of Input Block A, to the "Main Segmentation" table below. A portion of this joined table is shown in [Figure 4-27](#).

#### *Main Segmentation*

The arc attribute table (aat) of the main segmentation coverage. The numbering of the main segments starts with "1" and continues, in order, until the last main segment is numbered.

#### *Point Source BOD*

A dbf file which contains the annual BOD loading into each segment from point sources ([Figure 4-28](#)). This file can be created directly in dBase, or in Arc/Info and exported out of ArcView to a dBase format. If the table is not a dBase format, it is not possible to edit and change parameters in the table..

Boundary Segments													
Grid-Code	$E_x$ coeff	$X_c$ area	Upstr_seg	Dwnstr_seg	Bltn_seg	Int_bod	Int_do	Sod	Temp	Sal	Act_length	Depth	Type
9	0.0010	0.10	0	0	0	0.00	0.00	1.50	20.0	0.00	2502.80	0.10	3
10	119.9000	465.40	0	1	0	1.92	3.03	1.50	23.6	1.82	8046.90	9.14	1
11	119.9000	505.40	0	3	0	1.87	5.00	1.50	20.0	0.20	3218.70	4.88	1
12	119.9000	168.20	0	2	0	1.37	5.00	1.50	20.0	0.20	3218.70	3.05	1
13	119.9000	185.80	0	4	0	1.68	5.00	1.50	20.0	0.20	3218.70	1.83	1
14	119.9000	528.60	0	5	0	1.67	5.00	1.50	20.0	0.20	3218.70	3.35	1
15	119.9000	717.20	0	6	0	1.48	5.00	1.50	20.0	0.20	3218.70	10.06	1
16	119.9000	260.10	0	8	0	1.35	5.00	1.50	20.0	0.20	3218.70	3.96	1
17	959.3000	2471.20	8	0	0	7.42	3.63	1.50	25.3	10.90	3218.70	4.88	1

Main Segment Parameters														
Grid-code	$E_x$ coeff	$X_c$ area	Upstr_seg	Dwnstr_seg	Bltn_seg	Int_bod	Int_do	Sod	Temp	Sal	Width	Depth	Type	
1	704.50	1625.80	10	2	9	7.18	1.36	1.50	28.00	5.82	1	9.14	1	
2	704.50	1625.80	1	3	9	5.04	1.81	1.50	23.80	7.87	1	9.14	1	
3	704.50	1625.80	2	4	9	5.04	1.81	1.50	23.80	7.87	1	9.14	1	
4	704.50	1625.80	3	5	9	5.04	1.81	1.50	23.80	7.87	1	6.10	1	
5	704.50	1625.80	4	6	9	6.25	0.68	1.50	26.40	9.96	1	9.14	1	
6	704.50	1625.80	5	7	9	3.53	2.25	1.50	24.20	9.45	1	6.10	1	
7	704.50	2471.20	6	8	9	5.06	1.64	1.50	24.00	9.85	1	7.92	1	
8	704.50	2471.20	7	17	9	5.06	1.64	1.50	24.00	9.85	1	7.92	1	

**Figure 4-27** *Boundary Segments* and *Main Segment Parameters* .tables as they appear in ArcView. These tables hold the segment parameters and data necessary for the WASP5 model. The last column of the *Boundary Segments* table is not shown due to space constraints.

Point Source BOD	
<i>Grid-code</i>	<i>Load</i>
1	2.62
2	19.44
3	0.00
4	1470.06
5	56.39
6	234.31
7	24.79
8	226.71

**Figure 4-28** *BOD Point Source Loading* table used for model connection. The point source loads are in units of  $10^3$  lbs/yr.



### *Runoff Accumulation Values*

A table (a value attribute table -- INFO format) of the runoff accumulation values for each "outlet" (i.e. its most downstream point) of the main water segments. These values are usually obtained by running a flow accumulation over the watershed area, weighted by a grid of runoff depth, in Arc/Info's subprogram, Grid. This weighted flow accumulation is then "combined" with a grid of the outlet points to obtain this table (see [Figure 4-26](#)).

### *Water Boundary Segmentation*

The aat of the water boundary segment reaches. It is important not to join this table with the "Boundary Segments" table, since the boundary segments table may have segments which are not Type 1 (i.e., water column). If joined, segments shown on the Boundary Segments table which are not represented in the coverage as water reaches are "lost". The numbering of the boundary segments starts with the next number after the last main segment.

## **4.6.5 Coverages Needed for Connection**

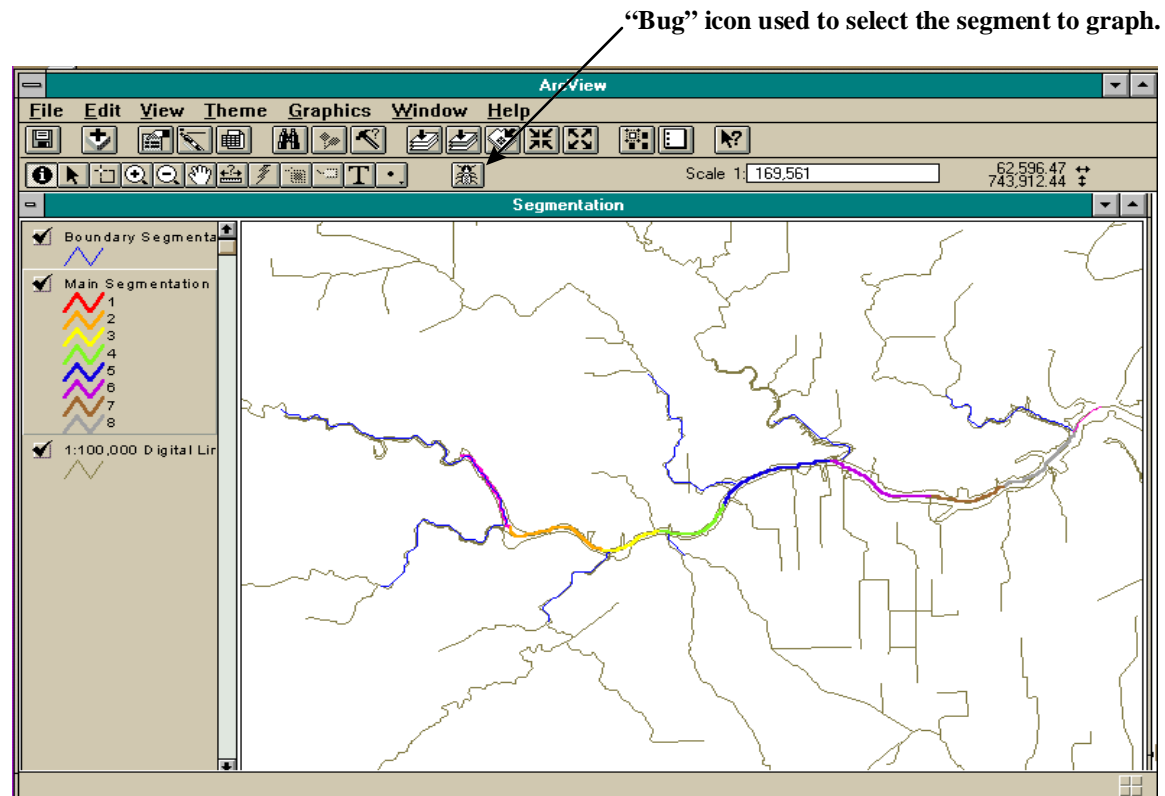
Although other coverages may be added for informational or display purposes, there are only two arc coverages needed in ArcView to run this model connection. Both are located within the same view, "Segmentation" (see [Figure 4-29](#)). The necessary coverages are described below:

### *Boundary Segmentation*

An arc coverage of the water boundary segments. The aat attached to this theme is named "Water Boundary Segmentation".

### *Main Segmentation*

An arc coverage of the main segmentation being modeled by WASP5. The aat to this theme is named "Main Segmentation". This coverage is named "segarc" within Arc/Info and aliased as "Main Segmentation" on the view.



**Figure 4-29** ArcView view, entitled "Segmentation". This view holds the two necessary themes (see view legend) to run the connection: "Main Segmentation" and "Boundary Segmentation." The "bug" icon, discussed in [Section 4.6.8](#) is shown on the lower tool bar.

## 4.6.6 Input File Creation and Modification

### *Input File Initial Creation*

The way this connection works is by reading necessary tabular information and querying the user for needed model options, during the input file generation. WASP5 has ten input blocks (A through J) and a "free form" text file is created by Avenue for each block (Appendix E). In addition, three text files containing the model run description, DYNHYD5 file name, and the input filename are created. Once the input file generation is performed, these text files (see Table 4-17) are in the working directory as a.txt, b.txt, etc. It is then possible to change just one input block (i.e., just the model constants -- Input Block H), while the rest of the parameters stay as they were originally generated.

Avenue executes all of the FORTRAN programs and writes all of the text files to what is termed the "working directory." The working directory, which also contains the WASP5 model executables and FORTRAN formatting programs, is established in running the script for Input Block A. Input Block A is always run before generating a new input file and running the model.

Creating the first input file for the system incorporates the following steps:

1. Checking that all the executables needed for the connection, EUTRO5, and TOXI5 and their related files (see Table 4-16) are in one directory (i.e. the working directory).
2. Referencing the correct coverage in the *vwout* script for the *Main Segmentation* coverage (see Section 4.6.10).
3. Having all of the necessary tables and view open. They can be minimized to icon views, but they must be open (Figure 4-30).
4. Having the "Project" window active so that the model connection menu items are shown on the main menu bar. They are: *BOD/DO Input Blocks*, *BOD/DO Model*, and *Model Calibration*.
5. Choosing *BOD/DO Input Blocks: All Input Blocks* (Figure 4-30) and allowing the project to run. The entire process takes about 2-3 minutes. The scripts query

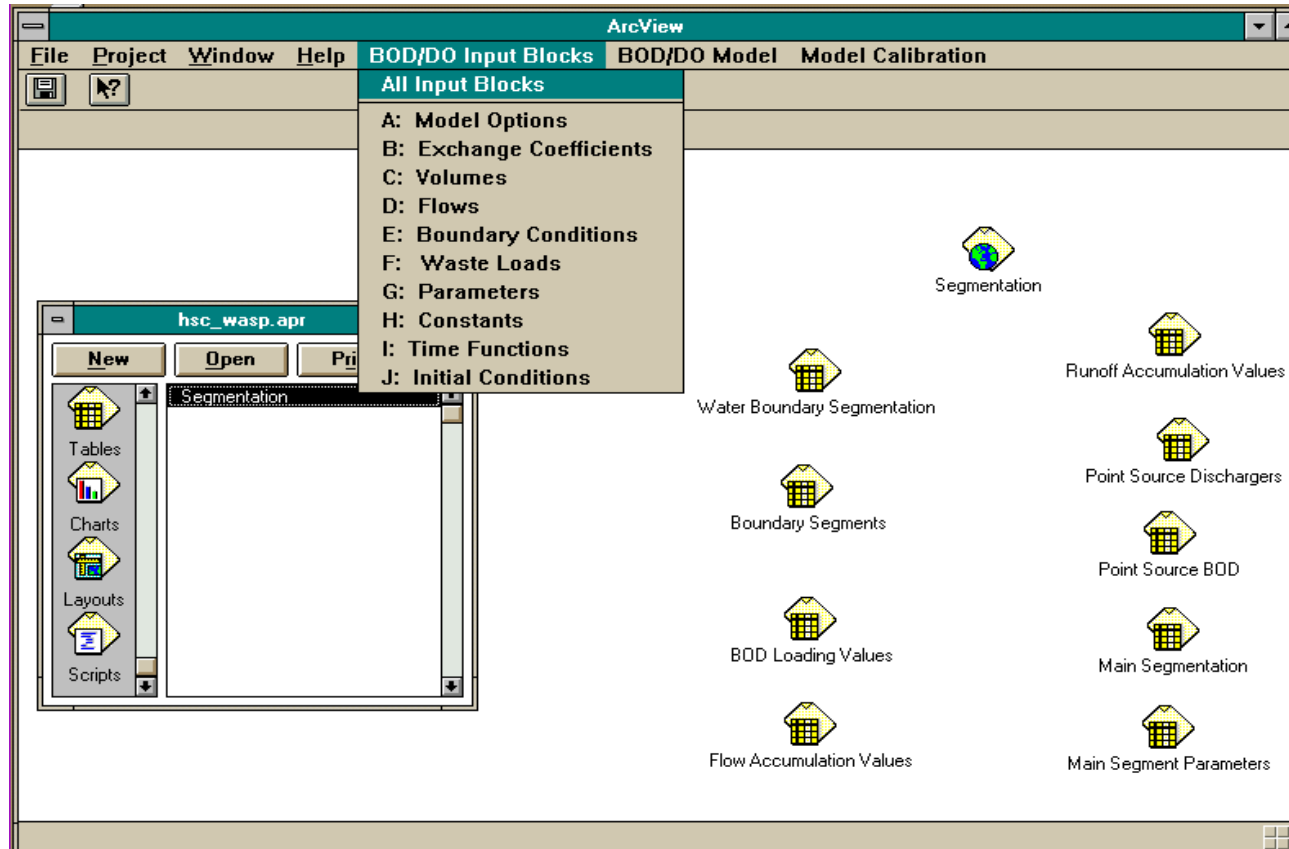
the user for some information relating to certain input blocks (Figure 4-31), while other blocks do not require any user input. The project prints a message box to the screen (Figure 4-32) each time it is done writing a particular input block.

6. Once all ten blocks are written, selecting *BOD/DO Model: Generate Input File* executes the FORTRAN program which formats the 13 text files into the WASP5 input file.

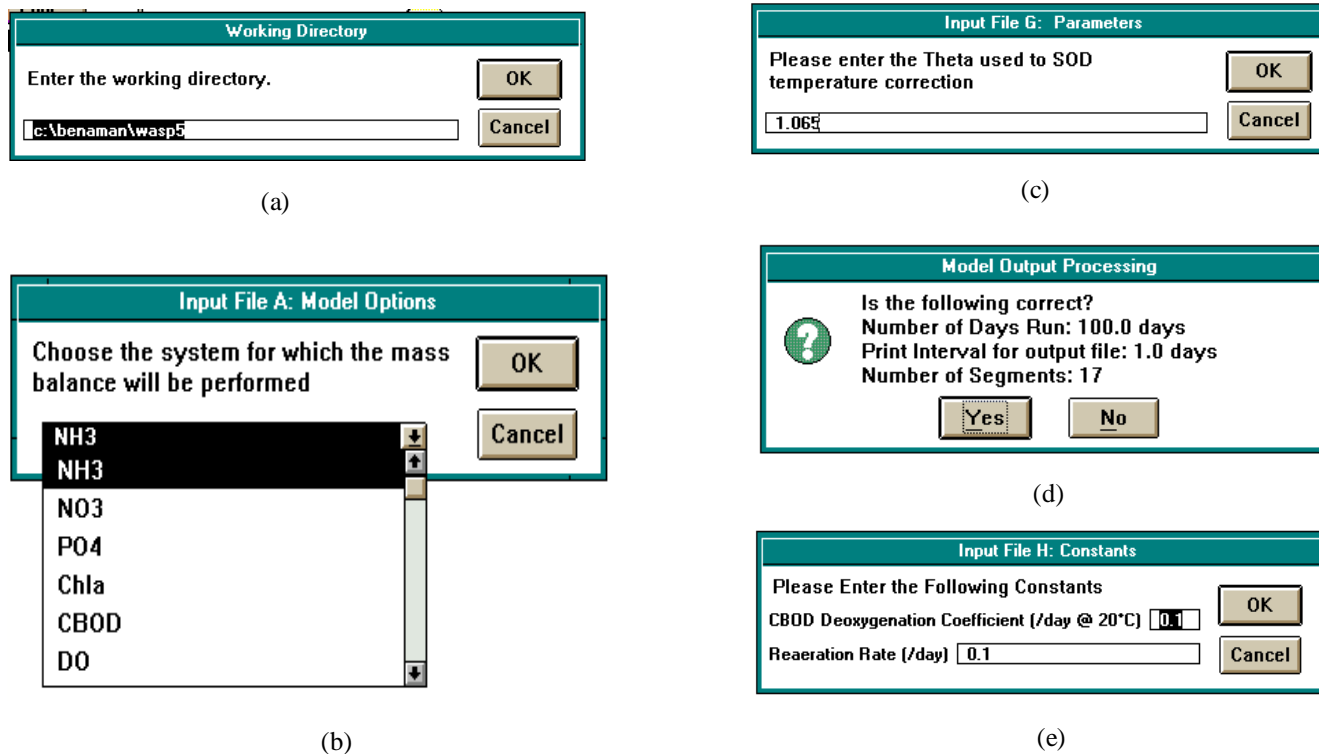
### *Input Block Modification*

Once the input file and related text files have initially been established, it is possible to change just one block of the main input file. This ability is an advantage to either correct a possible mistake or to observe the changes that may occur in the results, if a constant or a parameter changes (i.e., to investigate model sensitivity). Before accomplishing this task, the entire input file process explained above has been executed previously. It is not necessary to have run this process during the active project session; if it was run in an earlier session the free form text files created by Avenue (a.txt, b.txt, etc. -- see Table 4-17) are still be present in the working directory. It is then only necessary to rerun those blocks in which related information has been changed. Some important considerations before modifying tables and coverages are:

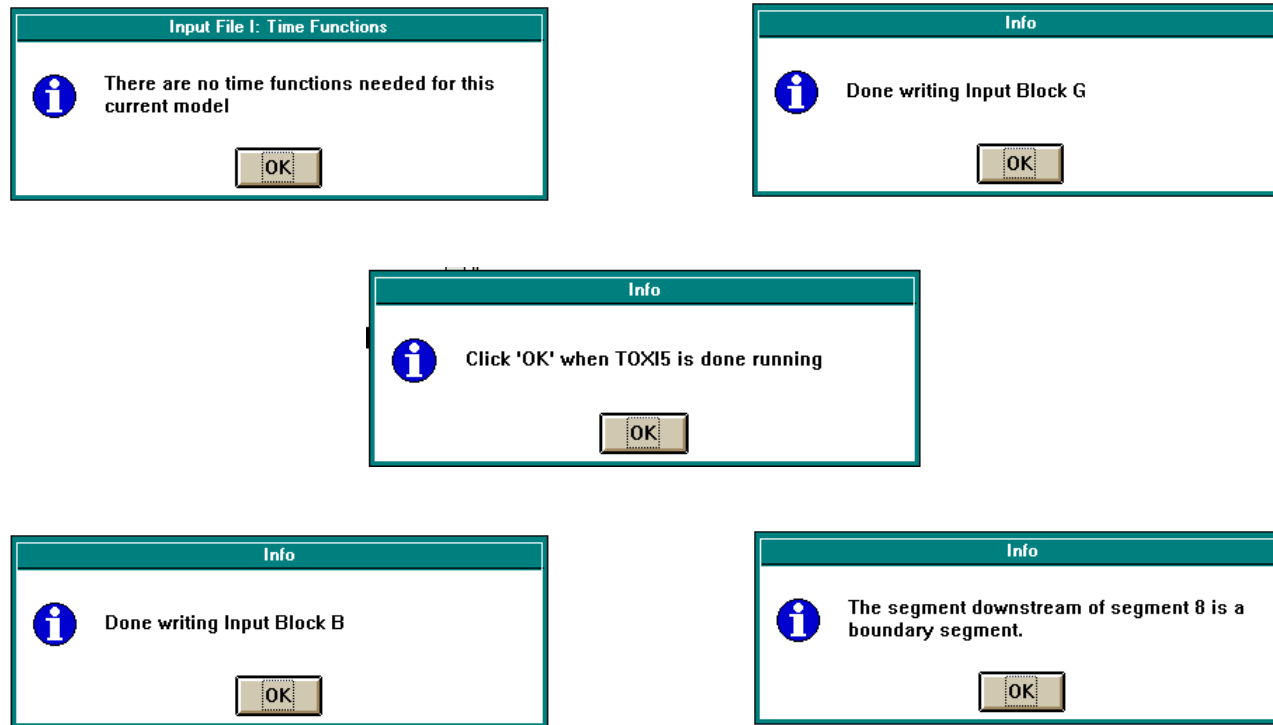
- Input Block A is always executed at the beginning of a session. This input block establishes the working directory, which is linked to the other subroutines that create Input Blocks B through J.
- The text files which are created and used by the formatting FORTRAN program are always named a.txt, b.txt, c.txt, etc. In order to change information in an input file but preserve the original input file created, the new input file must be named differently during the Input Block A generation. However, changing parameters on a project table results in the old information being overwritten. To avoid this problem, an entirely new table with the new information is created in Arc/Info or dBase, imported into ArcView, and aliased with the correct name.



**Figure 4-30** ArcView setup for model connection. This figure shows necessary views and tables opened, but minimized. The main project window (hsc\_wasp.apr) is active and the customized menus are shown. The menu choice is for "All Input Blocks". This choice writes all 13 text files necessary for the input file to WASP5.



**Figure 4-31** Examples of query boxes created by Avenue and shown during input file creation through ArcView. (a) asks the user to input the working directory during Input Block A; (b) displays a list of the state variables in EUTRO5 for the user to choose; (c) shows an input of a parameter for Input Block H; (d) queries the user during output processing; and (e) allows the user to enter the two constants needed for level one EUTRO5 BOD/DO modelling.



**Figure 4-32** Examples of various message boxes created by Avenue. Message boxes are used to inform the user a process is done, give the user information about an input block, or asks the user to click "OK" when a DOS process is finished so ArcView regains "control" of the project.

- If parameters are changed in the tables, all input blocks which are affected by those characteristics that have been modified are rerun. For example, if cross sectional areas are changed, it is necessary to rerun both Input Block B: Exchanges, and Input Block C: Volumes, since the cross sectional area is used in both of these blocks.

Most of the input block creation process reads information from the tables outlined in [Section 4.6.4](#). To change information in a within a table in ArcView, the table must be in a dbf format. Since ArcView only allows one to modify dbf tables, if an INFO file needs to be changed, it is first exported out of ArcView as a dbf file, reopened into the project, and renamed to the correct name. Once the table is a dbf file, it is changed by following the steps below:

1. To edit the table, make it active and choose *Table: Start Editing* on the menu bar (see [Figure 4-33](#)). Then, choose the editing icon from the tool bar and change the values. When complete, select *Table: Stop Editing* from the menu.
2. Set the project window active and select *BOD/DO Model* from the menu bar. A menu appears, displaying all ten input blocks ([Figure 4-30](#)). Choose the input block to recreate and Avenue regenerates the necessary text files.
3. Rerun Input Block A in order to rename the WASP5 input file and preserve the previous input file (if desired).
4. When complete, choose *BOD/DO Model: Generate Input File*, to recreate the input file with the new information.

#### *Dry Weather Conditions*

For the BOD/DO model input file, the user has the option to simulate "dry weather" conditions. During the generation of Input Blocks D and F, Avenue prompts the user to simulate average or dry weather flows. If "dry weather conditions" is chosen, the input file will have just baseflow conditions and point source loads only. It is important that the answer to the query at each input block is consistent; otherwise, an inaccurate input file which either discludes runoff flows or non-point source loads is created.



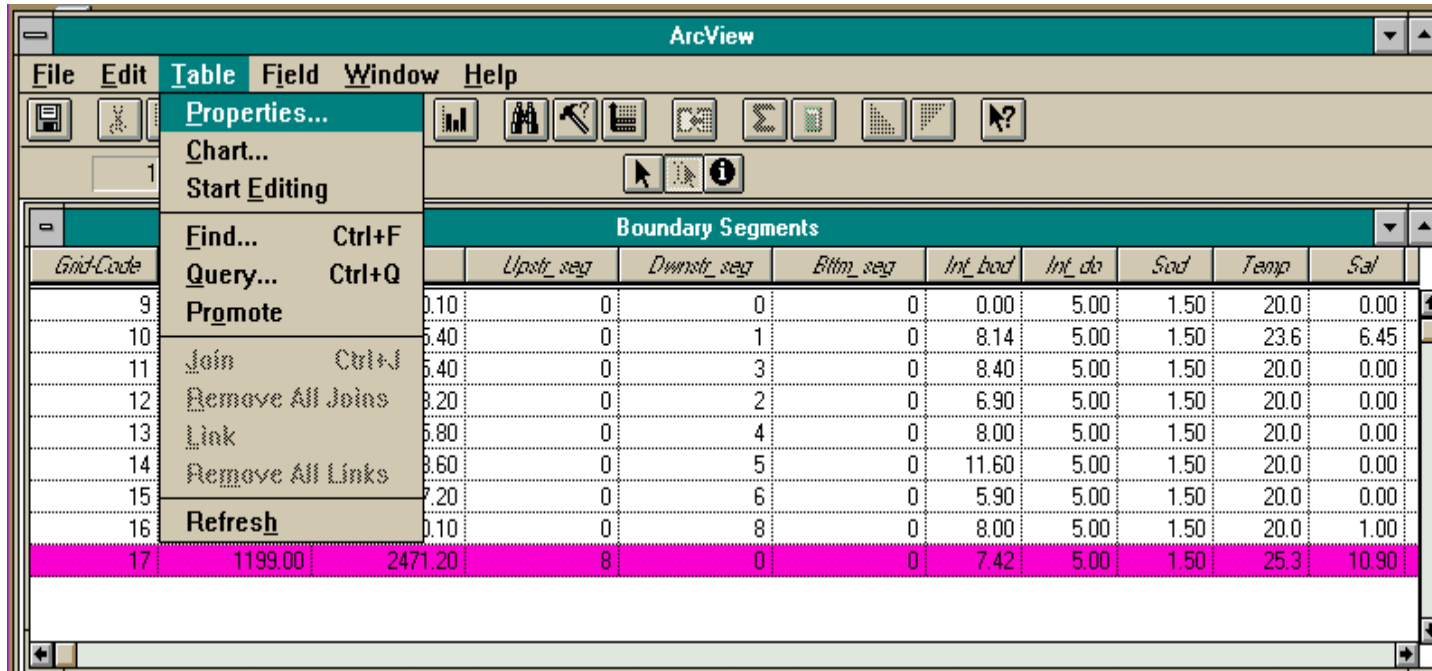


 Table edit icon located on lower tool bar

**Figure 4-33** ArcView menu showing option to *Start Editing*. Once chosen, the table edit icon (above) is highlighted and the selected table (in this case, "Boundary Segments") is changed. Only dBase tables can be edited.

### *Calibration Input File*

Before running the BOD/DO model in EUTRO5, the system being modeled must first be calibrated. The user can perform this calibration without the help of the ArcView connection, or utilize the ArcView to assist in the model calibration. For calibration, the TOXI5 is used to model a conservative substance in the system. Typically, at least two boundary conditions are set to a known concentration value of the substance; the rest of the segments are set to initial conditions of 0. The model is run and once equilibrium in the system is reached, the calculated values of the conservative substance are compared to known values.

For the calibration in this connection, the method employed is a steady state model using TOXI5, level one complexity and salinity as the conservative substance. When the menu choice *Write Input Information* under *Model Calibration* is chosen (see [Figure 4-25](#)), Avenue reads the data from the tables. In addition, the model constants and the desired segment numbers set as the boundary conditions are input by the user. As in the regular input file generation, the Avenue program creates 13 text files. A number of these files are identical to the blocks needed for a BOD/DO model run; specifically, input blocks B,C,D, and I. For this reason, the same free form text files (b.txt,c.txt,d.txt, and i.txt) are written to the working directory. For all other input blocks, the free form text files have the prefix "cal" (i.e. cala.txt).

### *An Overview of Each Input Block*

As mentioned earlier, WASP5 requires 10 input blocks for a completely formatted input file. Most of the information for each one of these input blocks is read by Avenue, using the corresponding choices on the menu bar (see [Figure 4-25](#)). However, some defaults to each input block were set within the input block creation in order to make writing the formatted input file more efficient and easier. The following is a short description of each information read from ArcView for each input block and the defaults that were set in its generation.

### Input Block A: Model Identification and Simulation Control

This input block contains basic simulation information and model preferences. The script reads how many segments are present and queries the user to choose model preferences such as length of model run, preferred time step, and print intervals. This script also asks for a model title, which is printed on the first line of the input file, and the input file name (\*.inp). This file name uses the standard DOS 8.3 file naming convention and always has the extension ".inp".

The following defaults are set in the generation of this input block:

- Presently, the connection is set to handle just EUTRO5 for Simple Street-Phelps modeling and level one complexity TOXI5 for calibration.
- Backward differencing is always used.
- A transport file is always generated.
- The first six segments' solutions are those which are displayed on the screen, during a model run.
- The same maximum time step is used throughout the model run.
- The same print interval is used throughout the model run.

### Input Block B: Exchange Coefficients

This input block describes the exchange coefficients for surface water (pore water exchanges are not set in the connection). The script reads the lengths of the segments to calculate the characteristic length between segments. It also determines if the boundary segments are perpendicular to the main segmentation and compensates for that attribute in the characteristic length. Finally, it reads the exchange coefficients and cross-sectional areas for two neighboring segments and chooses the smaller area and the exchange coefficient for printing to the text file. It also established the exchange between the benthic sediment and the water column to emulate a possible sink or source for DO.

The following defaults are set:

- Exchange coefficients are steady-state.
- Exchange coefficients are written to the text file in  $\text{m}^2/\text{sec}$  and lengths are in m.

### Input Block C: Volumes

This input block describes the segment volumes for the system. It also sets the hydraulic geometry parameters for calculating segment depth and velocity. These geometry parameters are used to calculate reaeration (if necessary) or volatilization from the segments; they are not used in transport calculations (Ambrose, *et al.*, 1993). For this connection, a constant (in time and space) reaeration coefficient has been assumed; therefore, these geometry parameters are not used to calculate reaeration in this particular set up. The script reads the cross-sectional areas ( $\text{m}^2$ ) and multiplies it by that segment's length (m) to obtain the volume. The script also asks the user to choose the following: 1) water column volume option, 2) benthic volume option, and 3) benthic time step.

The following defaults are set:

- Volumes are written to the text file in  $\text{m}^3$ .
- Geometry parameters do not spatially vary.
- Only the first benthic volume option is possible.

### Input Block D: Flows

This input block provides the advective transport flows that are used in the model. Presently, only flows for WASP5 Flow Field 1 (water column) are used in the connection. First, the script asks the user to choose a flow option. These options are described below (Ambrose, *et al.*, 1993):

1. Field one flows are specified directly by the user. Individual flows at each segment interface are summed by the model, and the net flow is applied across the interface.
2. Field one flows are specified directly by the user. Individual flows at each segment interface are applied directly by the model.

3. Flows are read from a formatted file created by DYNHYD5 or another hydrodynamic model. If this option is chosen, the user is asked to enter the file name of the text file which hold this information.

The table, "flow.dbf" (Figure 4-34) is created during this input block determination, by doing a units conversion on the "Flow Accumulation Values" and the Runoff Accumulation Values tables (Figure 4-26) (mm/yr/ha to m<sup>3</sup>/sec). This table gives the total cumulative flow, incremental flow, runoff, and baseflow into each main segment in m<sup>3</sup>/sec. The baseflow is found by subtracting the incremental runoff from the incremental flow.

The script in ArcView assumes that each boundary segment given has an associated flow and each main segment has just one flow input. The script tracks each flow input from its upper-most boundary to the most downstream segment (see Figure 4-22). To do this process, each boundary segment and its corresponding upstream and downstream segments are read from the "Boundary Segments" table. The main segment, which is downstream of a given boundary, is then found in the "Main Segmentation" table, and its corresponding downstream segment is obtained. The script continues to look for the successive downstream segments, until the most downstream main segment is reached. This flow route is printed to the text file and, at the end of the flow path, the actual flow from the original boundary segment is printed. If dry weather conditions are chosen, Avenue replaces the total flow in this text file with just the baseflow.

The following defaults are set:

- All flows are in m<sup>3</sup>/sec. The percentages of the total flow are written to the text file, along with the total flow value.
- The flow is steady state.
- The number of flow fields is set to 1: water column only (no pore water flows).

load.dbf			
<i>grid-code</i>	<i>bod_ps</i>	<i>bod_nps</i>	<i>do_nps</i>
1	3.26	8007.16	4735.24
2	24.16	3685.55	1972.15
3	0.00	2080.65	1188.21
4	1827.28	647.40	342.81
5	70.09	1362.58	715.38
6	291.25	4427.35	2452.58
7	30.81	116.04	77.95
8	281.80	692.01	403.80

flow.dbf				
<i>grid-code</i>	<i>cumm_flow</i>	<i>int_flow</i>	<i>runoff</i>	<i>baseflow</i>
8	35.66	1.20	0.93	0.27
5	26.65	2.06	1.66	0.40
7	34.46	0.22	0.18	0.04
6	34.24	7.59	5.68	1.91
4	24.59	1.00	0.79	0.21
1	14.35	14.35	10.96	3.39
3	23.59	3.54	2.75	0.79
2	20.05	5.70	4.57	1.13

**Figure 4-34** The two dbf files which are created during the BOD/DO input block execution. Flow.dbf gives the cumulative flow, incremental flow, runoff, and baseflow for each segment in m<sup>3</sup>/sec, while load.dbf shows the point source BOD loading and NPS BOD and DO loading into each segment in kg/day.

### Input Block E: Boundaries

This input block describes the boundary segments and their concentrations. Model boundaries consist of those segments that import, export, or exchange water with locations outside the main network. A boundary is either a tributary inflow, a downstream outflow, or an open water end of the model network across which dispersive mixing can occur. The boundary concentrations are read from the "Boundary Segments" table.

The following defaults are set:

- Since the EUTRO5 model is set just for the Simple Streeter-Phelps Model, only BOD and DO are considered in the boundary concentrations.
- For the calibration model, those boundary segments chosen by the user are set to their respective salinity values, while all other boundaries are set to 0.
- All concentrations are read and written in units of mg/L.
- The boundary conditions are steady state.

### Input Block F: Waste Loads

This Input Block writes the BOD and DO loads into each main network segment. The script does two things:

1. Reads the point source BOD loads into each segment (in  $10^3$  lbs/yr) from the "Point Source BOD" table and converts the value to kg/day.
2. Calculates the non-point source BOD from the "BOD Loading Values" table (Figure 4-26) in kg/day by taking each load value and subtracting the previous segment's load value to get the incremental loading to each segment.

Since the model is set for steady state, the non-point BOD loads are added to the point source BOD loads to get a total load to the segment. This input block also generates a table called "load.dbf" (Figure 4-34), which gives the non-point source loads in kg/day for BOD and DO. This table also gives the total point source BOD loading in kg/day for each main segment. If dry weather conditions is selected during this input block creation, only the

point source loads are written to the input file, since there is no runoff to carry the NPS loads.

The following defaults are set:

- All loads are steady state.
- Only BOD and DO are considered.

#### Input Block G: Parameters

This input block reads the necessary parameters for the level one EUTRO5 or level one TOXI5 models. As the complexity level increases, the number of parameters needed also increases. Presently, no parameters are needed for the level one TOXI5 model and only four parameters are required for EUTRO5 level one:

1. Temperature, read from the segment tables in °C; temperature is used to correct for deviations from the standard (20°C) and DO saturation.
2. Sediment oxygen demand (SOD), read from the tables in g/m<sup>2</sup>/day.
3. SOD theta correction, input by user; used to correct SOD for temperatures deviating from 20°C.
4. Salinity, read from the tables; used to calculate DO saturation.

The following defaults are set:

- Temperature does not vary in time.
- SOD theta does not vary in time or space.

#### Input Block H: Constants

This Input Block queries the user for the necessary constants needed to run the Simple Streeter-Phelps or the calibration models. The definition of the constants varies, depending upon the structure and kinetics of the systems comprising each model. For the present BOD/DO model, only two constants are needed:

1. CBOD deoxygenation rate at 20°C, per day.
2. Reaeration rate constant at 20°C for entire water body, per day.



For the calibration model, three constants are necessary:

1. Molecular weight of the substance being modeled in g/mol.
2. Constant partition coefficient for sorption to solids in L/kg.
3. Water column biodegradation.

The following default is set:

- The constants do not vary in time or space.

#### Input Block I: Time Functions

If the model were non-steady state for any parameter, this input block would use time functions to vary the specific parameter. Presently, none of the parameters are set to vary in time, so the default for this block is set to 0.

#### Input Block J: Initial Conditions

This input block describes the initial conditions for each system in the model. For the EUTRO5 model, only DO and BOD are considered in this script. The initial conditions are read from the segment tables. The TOXI5 model only needs to set the boundary segments that are chosen by the user to their respective salinities, while all other initial conditions (i.e., salinities) are equal to zero.

The following defaults are set:

- The dissolved fraction of BOD is set at 0.5.
- The dissolved fraction of DO is always 1.0.
- The dissolved fraction for salinity is 1.0.
- The maximum value for all systems in EUTRO5 is 1.0e8.
- The maximum value for salinity is 35,000 mg/L (sea water).
- Solids Field 3 transports BOD in its particulate form.
- Solids Field 5 transports DO.
- Solids Field 3 transports NaCl.

- All densities are set to 1.0 (EUTRO does not use those values).
- All initial conditions are in mg/L.

#### 4.6.7 Running the Model from ArcView

As mentioned before, the input file which is created has some limitations. This initial connection, although quite simplified for WASP5 capabilities, can be enhanced to eliminate some of the preset defaults presently in the code. However, at this time, this model connection has the following limitations:

- Only steady state input files are run.
- The connection is best set up for a river system, or a tidally influenced river system
- The Simple Streeter-Phelps Model is the complexity level -- the connection considers just Biochemical Oxygen Demand (BOD) and Dissolved Oxygen (DO)
- Only the water column flow is considered in this connection. The connection is not presently set up for two-layer water systems or for sediment transport.

Once an input file is created in ArcView, a model run can be executed by simply choosing *BOD/DO Model: Run EUTRO5*. Be sure that the input file has been generated before performing this step. When the model is executed, a DOS window appears and the EUTRO5 interface is shown. When a list of input files is displayed, the input file to run is executed. Do not press the "OK" button on the message box (Figure 4-32) until the model has completed executing. Pressing "OK" informs ArcView that the model is completely done running and Avenue can now exit from DOS and return to the ArcView interface. In addition, the EUTRO5 or TOXI5 models do not have to be executed through ArcView. It is possible to get ArcView to write several input files for these models and then run the model separately, through DOS, for each file. The final output can then be processed and viewed through ArcView.

#### 4.6.8 Output File Processing

There are three main steps that need to be executed in succession in order to process and view the output from a model run:

1. Checking the model parameters
2. Processing the output file
3. Viewing the output

All of these steps can be executed from either the *BOD/DO Model* menu or the *Calibration Model* menu. Be sure to choose the correct menu, corresponding to the model that is being processed (see [Figure 4-25](#)).

##### *Step 1: Checking the model parameters*

This item just reinitializes the model options and file information for ArcView to process the output. With this option, it is possible to create a number of output files with numerous model runs and then process them, one at a time, without having to go back and rerun the model. The correct output file name for the desired model run is input during this step. If an output file from the calibration model is being processed, the output file is named the same as its assigned input file name, with a ".tdf" extension. The BOD/DO model follows the same convention, with an ".edf" extension. This step is always executed before processing and viewing an output file.

##### *Step 2: Processing the output file*

This option executes a FORTRAN program which writes a text file with either salinity, or BOD and DO measurements, for every segment at each time step. The text file is then imported into ArcView as a dBase table. Avenue prompts the user to name each table, as it is processed. If a number of output files are being processed at once, their related tables are given descriptive names so that they can be differentiated in the project.

##### *Step 3: Viewing the output*

The first step that the user must do, after executing this command is to choose a table to which Avenue links the newly created charts and coverages. Although the table choice

box displays a list of all available tables (Figure 4-35), only those that were created by the above process step can be viewed using this menu option. If a table is chosen that does not have the proper format, ArcView exits out of the view output script. The table selected has to correspond to the parameters set in Step 1, above. If it does not, the user must go back and reset the parameters correctly. This need is because the output viewing steps (such as coverage creation) use some of these parameters in order to execute various Avenue script commands.

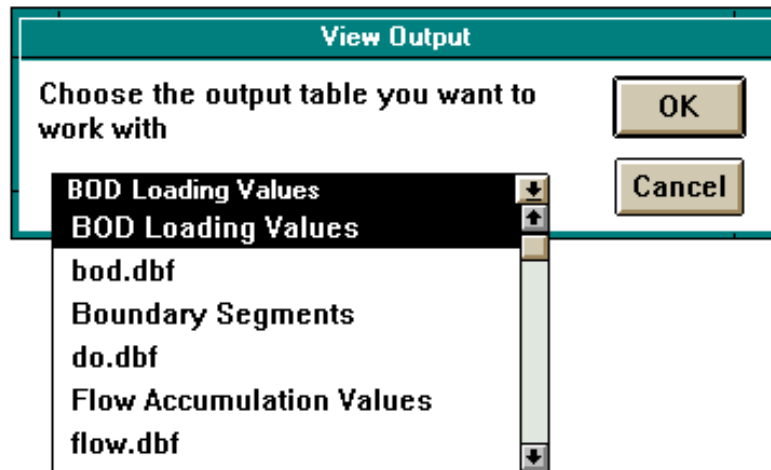
Within this step, there are five options. They are as follows:

1. View the table
2. Create a chart of concentration vs. time for a chosen segment
3. Create a chart of concentration vs. segment number for a chosen time
4. Create a coverage of the concentration at the last time step in the table
5. Create a "movie" of four coverages which display concentration at four chosen times

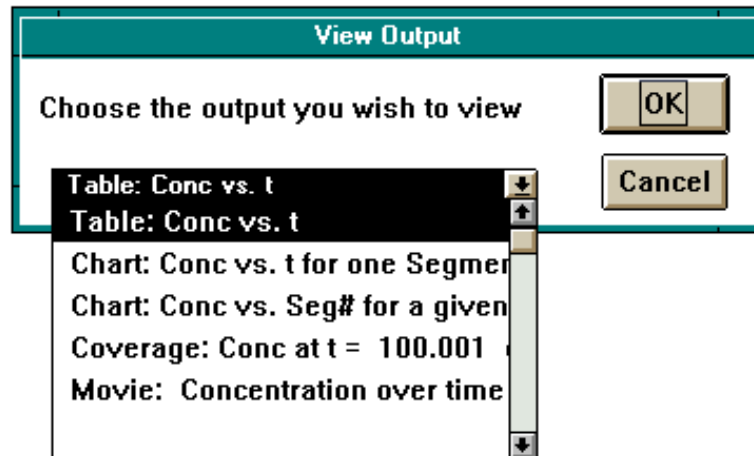
All charts which are created are always "linked" to a given table. If changes are made to that table or to the selections within the table, the chart changes accordingly. For example, if a chart of concentration vs. segment at time = 10 days has been created and another chart of concentration vs. segment at time = 19 days is created and linked to the same table, the first chart changes to reflect the new time. To avoid this problem, the user can create or add multiple copies of the same table to the project and link a chart to each table. This process is accomplished by either adding the dbf file numerous times and renaming the table so that it is more descriptive or processing the same output a number of times and changing the table name when prompted for a name. Then, only one chart is linked to each table and the charts remains independent of one another.

#### Option 1: Viewing the Table (Figure 4-36)

By choosing this option, the selected table opens and become the active view. The user can then view the output or create charts manually from this table, if desired.



(a)



(b)

**Figure 4-35** Query boxes: (a) shows the list of tables to choose from which new charts and themes are created and (b) displays the output options available to the user. Only those tables with a "Time" field are able to be processed.

salinity.dbf								
Time	1	2	3	4	5	6	7	8
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0
0.5010	0.5850	0.3900	0.3730	0.4530	0.7520	1.5100	2.4800	3
1.0010	1.0100	0.8650	0.8810	0.9840	1.2600	1.8300	2.4900	2
1.5000	1.3900	1.2600	1.2700	1.3500	1.5800	2.0500	2.5900	2
2.0000	1.6900	1.5800	1.5800	1.6600	1.8500	2.2500	2.7300	3
2.5000	1.9500	1.8400	1.8500	1.9200	2.0900	2.4500	2.9000	3
3.0010	2.1800	2.0800	2.0900	2.1500	2.3100	2.6500	3.0700	3
3.5010	2.3800	2.2900	2.3000	2.3600	2.5200	2.8400	3.2400	3
4.0010	2.5600	2.4800	2.4900	2.5500	2.7100	3.0100	3.4100	3
4.5010	2.7300	2.6500	2.6700	2.7300	2.8800	3.1900	3.5800	3
5.0010	2.8900	2.8200	2.8300	2.9000	3.0500	3.3500	3.7300	4
5.5010	3.0400	2.9700	2.9900	3.0600	3.2100	3.5000	3.8800	4
6.0010	3.1800	3.1200	3.1400	3.2100	3.3600	3.6500	4.0200	4
6.5010	3.3200	3.2500	3.2800	3.3500	3.5000	3.7800	4.1600	4
7.0010	3.4500	3.3900	3.4100	3.4800	3.6300	3.9200	4.2900	4
7.5010	3.5700	3.5100	3.5400	3.6100	3.7600	4.0400	4.4100	4
8.0010	3.6800	3.6300	3.6600	3.7300	3.8800	4.1600	4.5200	4
8.5010	3.7900	3.7400	3.7700	3.8500	4.0000	4.2700	4.6300	4
9.0000	3.9000	3.8500	3.8800	3.9500	4.1100	4.3800	4.7400	5
9.5000	4.0000	3.9500	3.9800	4.0600	4.2100	4.4800	4.8300	5
10.0000	4.0900	4.0500	4.0800	4.1600	4.3100	4.5800	4.9300	5
10.5010	4.1900	4.1400	4.1800	4.2500	4.4000	4.6700	5.0200	5
11.0010	4.2700	4.2300	4.2600	4.3400	4.4900	4.7500	5.1000	5

**Figure 4-36** Table (dbf format) which is created during the output file processing of the WASP5 output. This table shows values of salinity (in ppt) and is opened when the first choice for the output presentation is chosen.

#### Option 2: Concentration vs. Time (Figure 4-37)

This option opens the view, "Segmentation" (Figure 4-29) and prompts the user to activate the "bug" icon and choose a segment. Once the "bug" icon located on the far right bottom tool bar is active, the proper theme (either Main Segmentation or Boundary Segmentation) is highlighted. The chart is created by clicking on the segment to graph. The user is asked a few chart options, including color and the name of the y-axis. Afterwards, a chart, showing concentration vs time ( for all time steps) appears.

#### Option 3: Concentration vs. Segment Number (Figure 4-38)

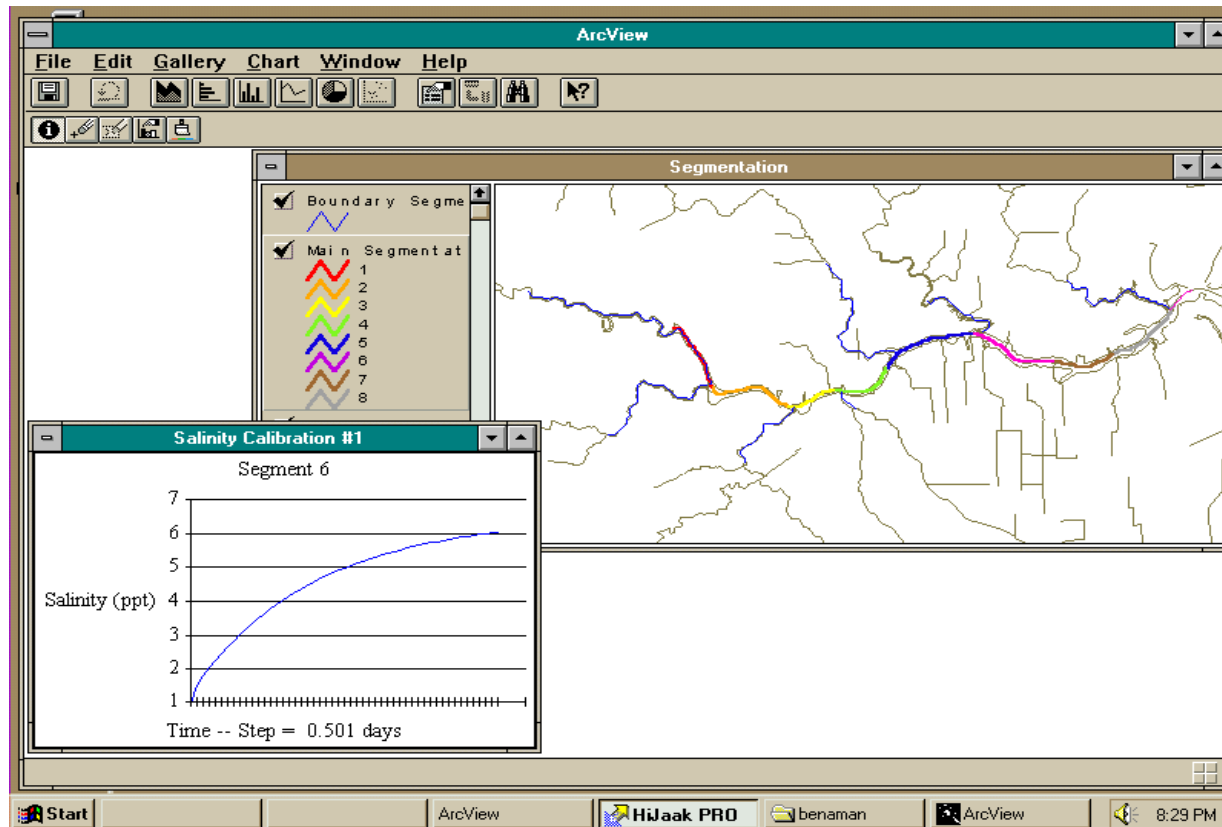
This option prompts the user for the time at which to display the concentration values. A bar chart showing the concentration for each segment (main and boundary) is then created. If a different time is desired, the new time is selected on the linked table and the chart changes to reflect the new selection.

#### Option 4: Coverage at Final Time (Figure 4-39)

This option creates an ArcView coverage of the concentration at the final time in the model run. The script automatically brings the coverage up on the view, "Segmentation", and shows the concentration values for that time, in a ramped arc coverage (from blue to red).

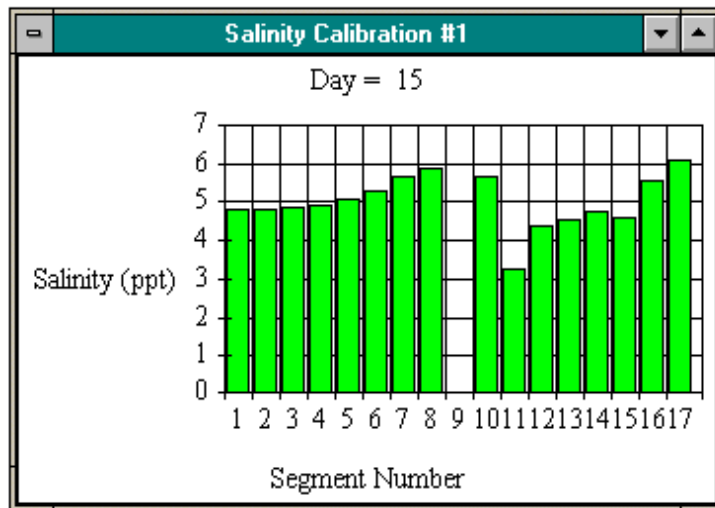
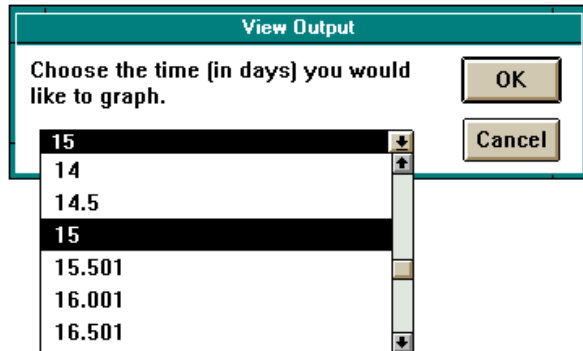
#### Option 5: Movie of Four Chosen Times (Figure 4-40)

This option allows the user to choose four given times from all possible times in a chosen table. The script then opens the view, "Segmentation", and creates a coverage, at a chosen time delay. The script determines the minimum and maximum values within the selected times and ramp the coverage from gray to blue in eight intervals. These intervals stays constant for all four times, so that changes in concentrations can be viewed consistently. After the script is complete, the user has four new coverages of concentration at each time step.

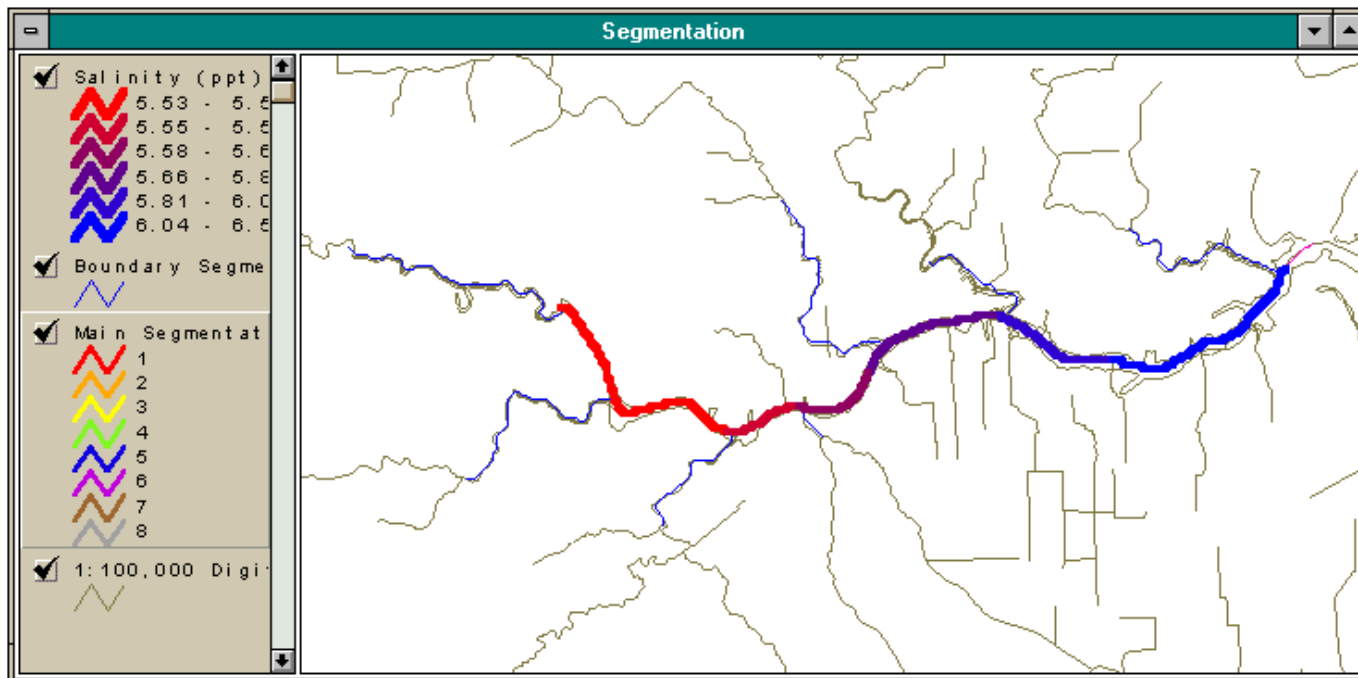


**Figure 4-37** Result of second choice in output file presentation options: chart of concentration vs. time for a given segment and chosen table. The chart is created by choosing the bug icon from the view toolbar (see [Figure 4-26](#)) and choosing the segment off of the view to chart.

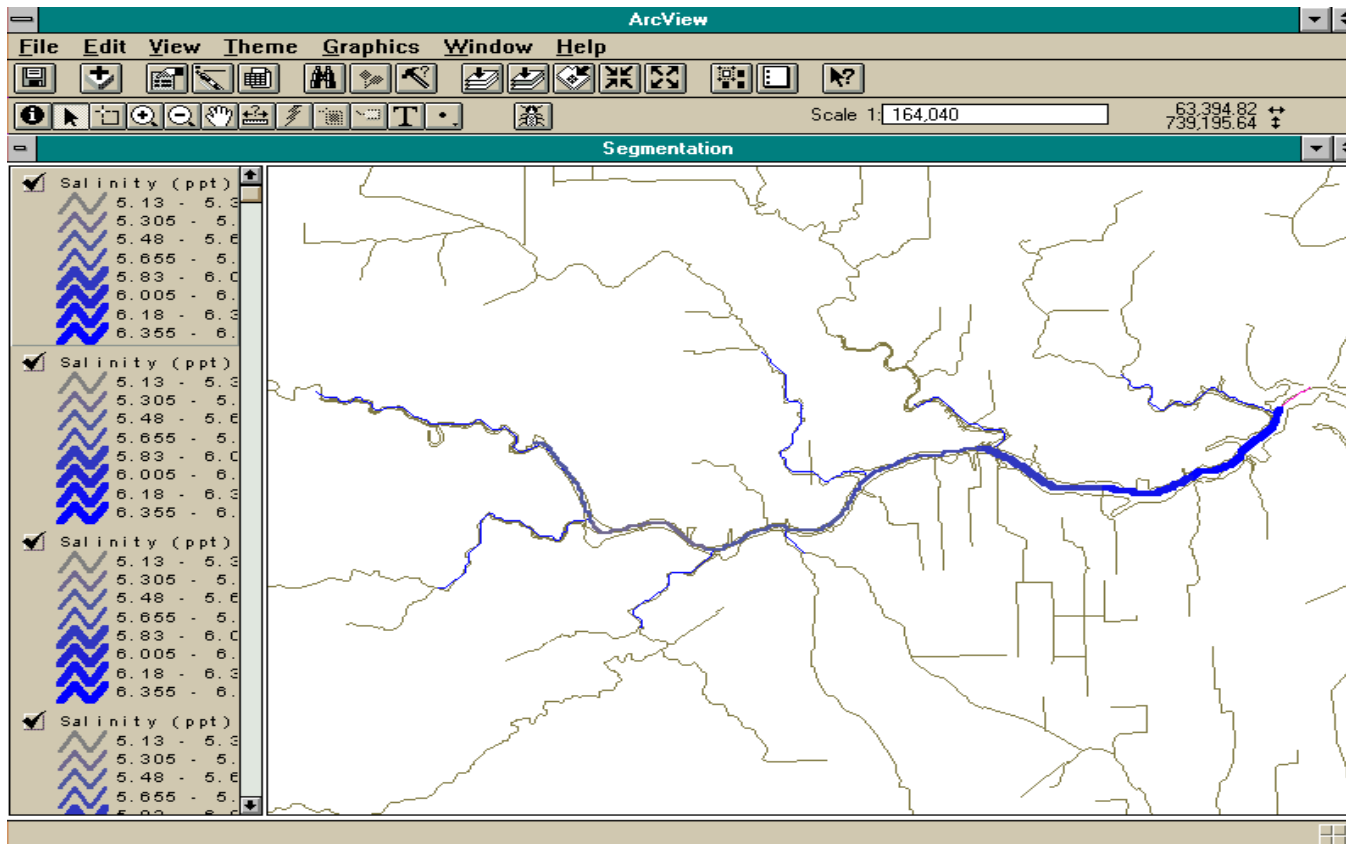




**Figure 4-38** Query box and chart for third output presentation option: concentration vs. segment number at a chosen time. The query box allows the user to pick the time at which to chart the concentrations. Segment #9 is the benthic sediment layer; no salinity is modeled in that segment.



**Figure 4-39** New theme created by fourth output presentation option. This new theme shows concentration at the last time in the model run. The arcs are ramped from blue to red.



**Figure 4-40** Themes created by the fifth output viewing option. The themes, all of concentration at different times selected by the user, are shown on the screen at a set time delay. When complete, four new themes (as shown here) are present on the view. The interval for the ramped arcs is determined by finding the maximum and minimum concentration values of all times chosen and dividing by eight.

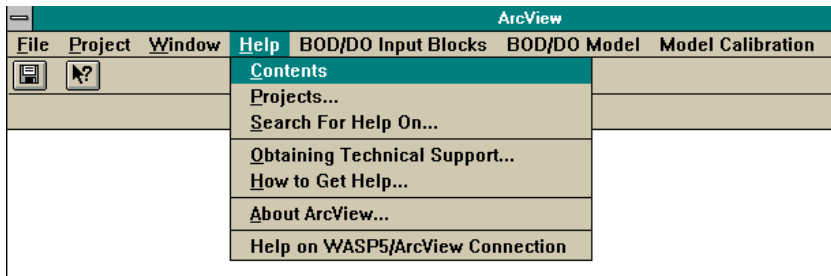
#### 4.6.9 Help File Creation

Located within this customized project is a "help" document to assist a user in the use of this model connection. To activate this help document, the option *Help on WASP5/ArcView Connection* under *Help* on the menu bar is chosen (Figure 4-41). This file is a hypertext document, compiled in rich text format and executed under Windows's help file (winhelp.exe). This help document, shown in Appendix G, gives information on the model, the connection, the input blocks, troubleshooting, limitations, and references. In addition, it outlines how to create an initial input file, to change an input block, to run the model, and to perform a model calibration.

#### 4.6.10 Setting up the WASP5/ArcView Connection

The ArcView/WASP5 model connection and a demo which shows the Houston Ship Channel study discussed in this report can be set up on any computer which has ArcView 2.1 or higher installed on the machine. In order to set up the demonstration on a personal computer, the following steps are taken:

- Install WASP5 onto the computer. WASP5 is available from the USEPA Homepage ([ftp://ftp.epa.gov/epa\\_ceam/wwwhtml/wasp.htm](ftp://ftp.epa.gov/epa_ceam/wwwhtml/wasp.htm)).
- Download the demonstration files, in zipped format, from the University of Texas, Center for Research in Water Resources Homepage for the ArcView/WASP5 connection demo (<http://www.ce.utexas.edu/prof/maidment/GISHydro/>) and unzip them into the directory that the WASP5 executables are located.
- Open the project `hsc_wasp.apr` in ArcView version 2.1 or higher. ArcView may initially ask the user to location of some coverages and tables. All necessary coverages and tables should be located with in the directory in which the downloaded file was unzipped. Most tables are in dbf format, while the coverages are in a folder entitled *cover*.



**Figure 4-41** Windows help document run by winhelp.exe and created to assist a user in the execution of the WASP5/ArcView connection.

- Open the script *vwout* and locate the two references to the arc coverage *segarc*. Both of the lines in the script read:

```
segSrcName = SrcName.Make("c:\wasp\cover\segarc arc")
```

This line informs Avenue on where to locate the main segmentation coverage. Be sure that the drive and directory name in this line is correct. Also, if this script is being recompiled for a new system, this line should reference the correct location and name for the main segmentation coverage in the new system that has been developed.

- If the *vwout* script was changed, in any way, recompile it, by clicking on the checkmark icon on the bottom toolbar of the ArcView script tools.
- Model input file creation, model runs, and output viewing can then be performed as described previously in this section.

## Chapter 5

### Results and Discussion

#### 5.1 RAINFALL/RUNOFF AND RAINFALL/FLOW CORRELATION

Newell, et al. (1992) stated that the development of a rainfall/runoff relationship in the Houston area was difficult due to the watershed's high amount of urbanization. Instead, their 1992 study produced design hydrographs from flow and precipitation data. In the present research, however, flow and precipitation data was used directly to result in the graphs shown in [Figures 4-16 and 4-23](#). These figures show relationships for runoff and flow, with precipitation and land use. Although both equations calculated have a rather low  $R^2$  values, the results obtained from these correlations compare relatively well to the measured values. Since these relationships were developed from data over 30 years, the equations are meant to represent long-term steady state conditions. In addition, the "relative runoff coefficient" introduced in this process only represents a relative measure of the urbanization of a given watershed and is not being used in an absolute sense.

To verify the calculated runoff values, flow data from a station (USGS Gauge 8075900), which was not used in the development of the runoff/precipitation/land use equation, was downloaded from Texas USGS (Texas USGS, 1996). This station, located on Greens Bayou, is further described in [Table 3-6](#). After the flow data from this station was downloaded, the baseflow was subtracted from the total flow, using the same FORTRAN program discussed in [Section 4.3](#). The average runoff depth for the available period of record was then calculated. To compare this measured runoff value with its corresponding calculated runoff, [Figures 5-1 and 5-2](#) were created. [Figure 5-1](#) shows the calculated runoff depth routed down the delineated streams, and the USGS gauge stations attributed with their measured average runoff depths. The runoff in the streams represents the average depth of runoff from any given point over all land upstream of that point. As a result, a direct comparison of the colors shown in [Figure 5-1](#), between the point coverage and the arc coverage, can be made. [Figure 5-2](#) displays a graph of

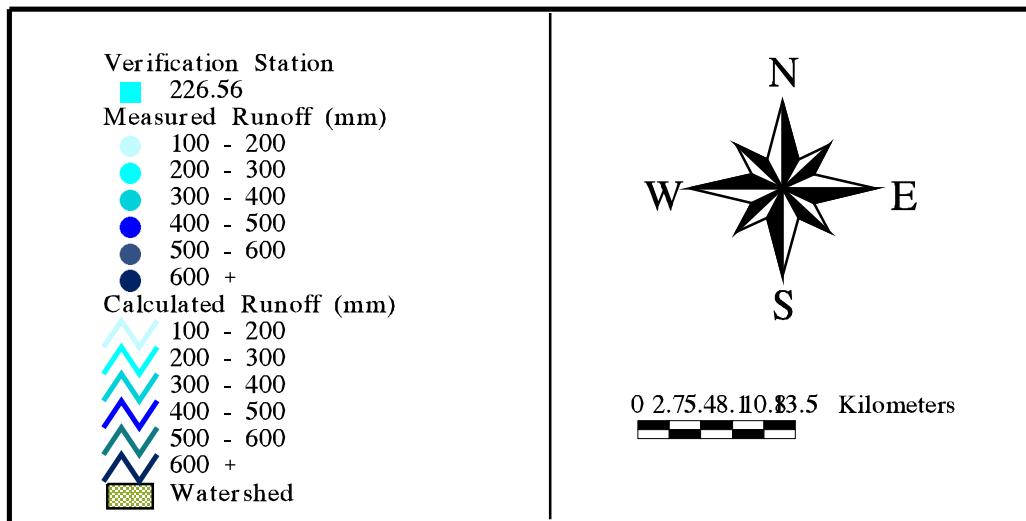
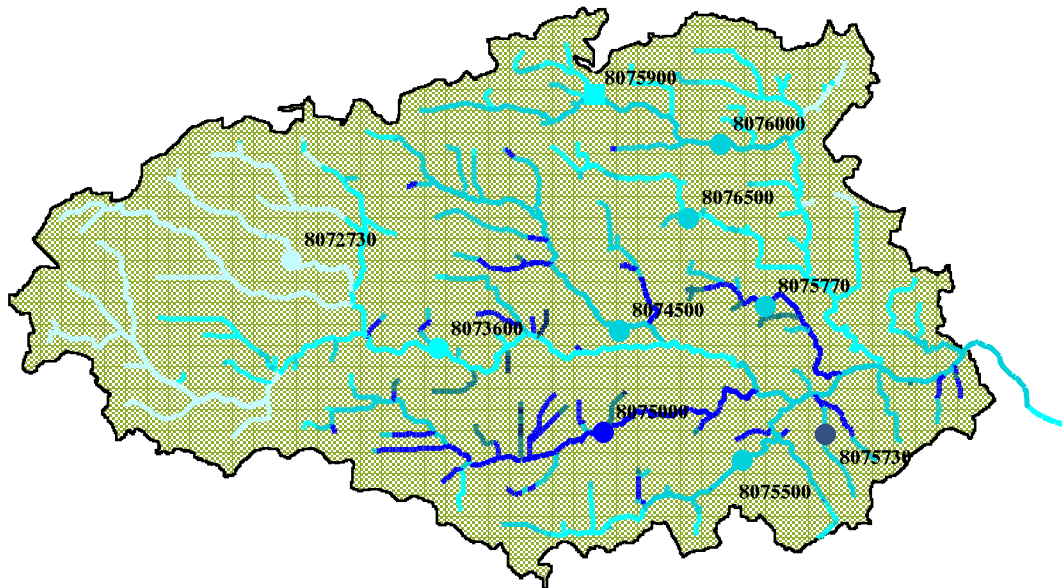
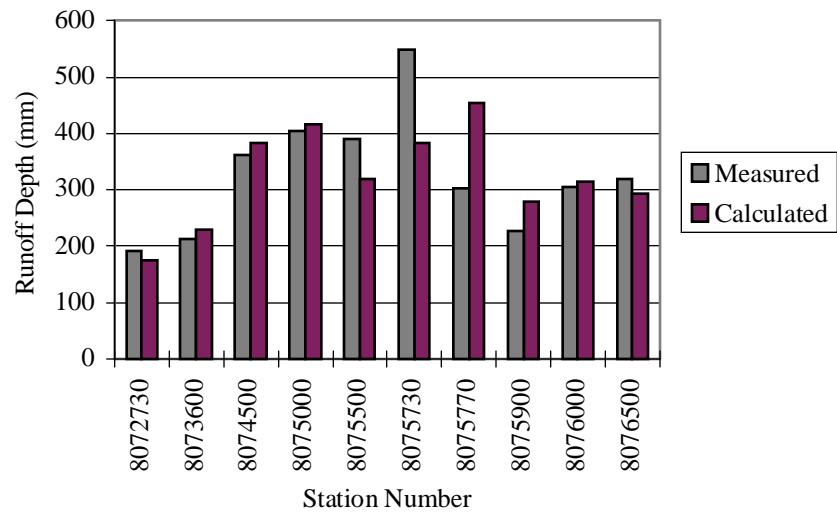


Figure 5-1 Comparison of calculated runoff depths with measured runoff depths at each USGS gauge station. The streams can be interpreted as average runoff depth from all land draining into a given point on the stream. This figure also shows station 8075900 which was not used in the rainfall/runoff/urbanization relationship, but is shown to verify the equation's results.





**Figure 5-2** Direct comparison of calculated runoff depths at same location as measured depths for each USGS gauge. Most calculated values compare reasonably well with the measured depths. Stations with smaller watershed areas seem to have more of discrepancy.

the same stations, comparing the measured values to the calculated runoff depths for each station.

As shown in [Figures 5-1](#) and [5-2](#), the measured runoff depths compare reasonably well to the calculated depths represented by the streams. Those stations that had larger drainage areas, such as 8073600 and 8075000, seemed to result in more accurate results. The station which was not used in the correlation development (8075900) also indicates that the calculated value for that area is representative of the actual measurement. Whether the calculated values over or under estimated the actual measured values is not consistent from station to station. Errors may have been introduced by the assumption that the watershed had a uniform response to the rainfall. In addition, one station in particular seemed to be an outlying data point on [Figures 4-16](#) and [4-23](#) (8075730) This station had the smallest drainage area (8.3 mi<sup>2</sup>) and was the closest station to the coast. Since the watershed area is small and the station's precipitation is possibly influenced by the coastal environment, inaccuracies may have occurred when it was included in the regional precipitation/runoff relationship.

For the flow correlation, [Table 5-1](#) gives some selected flow measurements from various sources, compared to the calculated flow values at approximately the same location. These calculated flow values correspond to the main segment flows determined for the WASP5 model. Although the calculated numbers primarily represent the main tributaries entering into given model segment on the Houston Ship Channel, they also encompass any diffuse runoff or smaller tributaries entering into that main segment. In spite of the fact that the calculated flow values are meant to represent long-term averages, some are similar to the measurements from past years. While some of this measured flow data is decades old, this coherence is not surprising, as a large percentage of Houston has been urbanized for more than 20 years. Because of this history, the flow patterns in the metropolitan area have probably not changed significantly. In particular, the calculated flows for the Buffalo, Brays, and Carpenters Bayous correspond well to the historical data. On the other hand, the Sims/Berry, Hunting, and Greens/Hall Bayous have considerably higher calculated flows

than the measured flows. Besides errors the introduced by using one equation to represent the flow over the entire area, some other possible explanations for these differences include:

- The main segment downstream of the respective bayou may receive additional flow from diffuse runoff or smaller tributaries, resulting in a higher total calculated flow for that segment.
- Urbanization or recent land use changes may have caused the flow to increase from earlier measured values.
- The average measured flows presented in **Table 5-1** do not encompass the same years used in the correlation development (Armstrong and Ward, 1994 and Espey, *et al.*, 1971).

**Table 5-1 Comparison of Calculated to Measured Flow for Various Tributaries at Different Years**

Bayou Name	Calculated Flow (m <sup>3</sup> /sec)	Downstream Main Segment	1971 Flow* (m <sup>3</sup> /sec)	1994 Flow** (m <sup>3</sup> /sec)
Buffalo	14.23	1	15.83	12.41
Brays	5.70	2	4.67	6.55
Sims/Berry	3.54	3	1.25	3.31
Vince	1.00	4	2.38	
Hunting	2.06	5	0.99	0.70
Greens/Hall	7.59	6	3.37	2.67
Carpenters	1.20	8	0.99	

Sources: \* Espey, *et al.*, 1971

\*\* Armstrong and Ward, 1994

## 5.2 WATERSHED DELINEATION

In the past, watershed delineation was performed using contour maps and possibly, ground surveying techniques. Once the boundaries were drawn on maps, they could be digitized into GIS as polygons. This was the method used for the development of the watershed in Newell, *et al.*, (1992). However, for this study, the watershed delineation method used in Saunders (1996) was employed. This process, which uses a grid-based terrain analysis to determine the direction of water flow, has proved to accurate in digitally representing a given drainage basin (Maidment, 1993 and Saunders, 1996). The final watershed boundary used in this study is compared to the USGS digitized boundary in **Figure**

3-3. The USGS watersheds shown in the figure are digitized basins for each major bayou draining into the Houston Ship Channel (Newell, *et al.*, 1992). It is difficult to perform a direct quantitative comparison on the total area encompassed by the eight USGS watersheds and the grid delineated boundary, since the USGS watersheds also include an small area which lies downstream of the final outlet set for this current project. Nevertheless, the accuracy of the digital representation of the terrain can be investigated by looking at the USGS gauge station watershed areas in comparison to those delineated with Arc/Info. Table 4-4 presents the USGS watersheds as compared to the delineated areas. For most gauges, the error in the delineated basins with the given USGS area is less than 10%. Two of those higher than 10% (Stations 8072730 and 8075770) have relatively small areas, which makes it difficult to accurately represent them. It is unclear why station 8073600 has such a large difference, except that errors may have also occurred in the area given by USGS.

The primary source of these errors lies in the highly developed characteristic of the watershed. This urban development makes it difficult to accurately represent man-made flow paths such as canals and reservoirs. In addition, the flat coastal regions have proved to be problematic when trying to digitally delineate stream networks (Saunders, 1996). Although most of these errors were avoided by “burning” in a DLG of the stream network (see Section 4.3), inaccuracies still exist. In addition, complete confidence in the USGS areas should not be taken. These areas were read from contour lines on a USGS quad sheet and cannot be considered 100% accurate.

### 5.3 NON-POINT AND POINT SOURCE LOADINGS

#### *Non-point Source Loading*

Non-point source loading has recently become a primary area of concern related to water quality of rivers, lakes, and bays. Significant amounts of harmful materials accumulated from the land surface have been found to be carried by stormwater runoff which eventually enters the water bodies. Of particular concern in urbanized areas is BOD, oil and grease, and heavy metals. For this present modeling effort, only BOD loading was a concern

(Figure 4-19). As expected, the areas which produce higher runoffs and have more urbanization, result in higher NPS loading. The values presented in Figure 4-19 represent steady state yearly averages, but, since rainfall does not actually occur in a steady state fashion, it may be more insightful to calculate the loading from just one storm event on a time varying basis. In addition, the investigation of time varying EMCs within a storm event could also be beneficial.

For this study, though, steady state values were acceptable since the model connection is initially being developed for steady state conditions. A similar non-point source loading assessment was performed on the entire Galveston Bay system in 1992 (Newell, *et al.*, 1992). Table 5-2 presents the results from the 1992 study as compared to this present research (see Figure 3-3 for USGS subwatershed locations). Since slight differences existed in the watershed areas for both studies, care should be taken when directly comparing the two average areal loading values. In particular, a downstream portion of the Ship Channel watershed from the Newell, *et al.* (1992) study was not included in this research since it was downstream of the ultimate outlet point established in this study. For this reason, a direct comparison of those BOD loading values is not completely accurate.

**Table 5-2 Comparison of BOD Non-Point Source Loading Results by USGS Watershed**

Watershed	Newell, <i>et al.</i> (1992) Loadings (kg/yr/hect)	Calculated Loadings from this Study (kg/yr/hect)
Addicks Reservoir	20.2	16.1
Barker Reservoir	19.0	13.7
Brays Bayou	51.7	40.9
Buffalo Bayou	51.5	40.5
Greens Bayou	38.8	26.3
Ship Channel	46.5	40.0
Sims Bayou	41.5	30.3
Whiteoak Bayou	52.7	38.7
<b>Total</b>	<b>312.9</b>	<b>246.5</b>

As shown in Table 5-2, the values from both studies compare relatively well. Since both method used the same EMCs (Table 4-7) and land use coverage (Figure 3-6), the differences are probably due to the runoff volumes used in each study. As mentioned in Section 5-1, Newell, *et al.* (1992) developed design storms, using a model and previous

runoff data, while this research established a rainfall/runoff/urbanization curve. [Table 5-2](#) indicates that the runoff volumes used in this study are less than those used for an "Average Year" in the Newell, *et al.* (1992) report. However, since considerably different methods were used to obtain runoff volumes, disagreements between the two results are expected. On the whole, though, the differences in the trend in the loading from watershed to watershed is the same for both studies.

#### *Point Source Loadings*

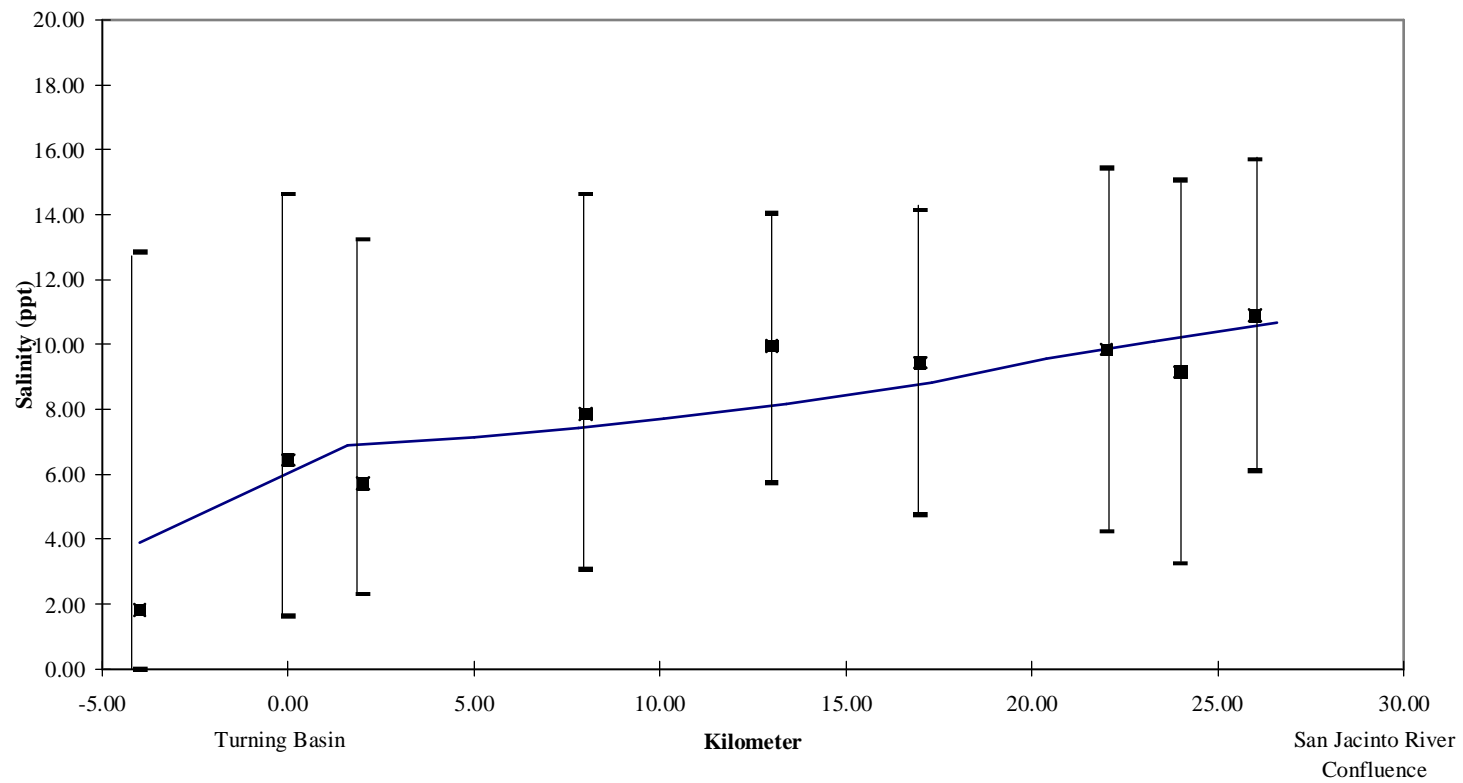
The final point source loadings from municipalities and industry along the Upper HSC are shown in [Table 4-9](#). As discussed in [Section 4.4](#), these loadings are not truly representative of the entire system, since only half of those dischargers located along the 25 km stretch had reported BOD measurements. Although these loadings were obtained from self-reporting data or estimated with standard concentrations multiplied by flow data, these values are probably more accurate than those that could be obtained directly from the industry's discharge permit. Since dischargers sometimes gain permits for effluent concentrations and flows they do not always reach, the self-reported data is more representative of the actual flow. Because more data from the missing point source dischargers could not be obtained, the loadings given in [Table 4-9](#) were used in the model. A more refined modeling effort could consider the point source effluents entering the tributaries (segments 10 through 16), instead of assigning a concentration value to the tributary's upstream boundary, as was done in this research.

Direct comparison of these point source loadings to other reports proved to be difficult. The major stumbling block was that earlier waste loading estimations for the Channel grouped all the permitted dischargers and estimated runoff loads into one category (Espey, *et al.*, 1971 and TDWR, 1984). However, for this modeling effort, those industries not directly discharging into the Upper HSC were considered in the tributary boundary conditions. In addition, the segmentation in the earlier studies used to report total loadings did not match the segmentation used here.

Also worth mentioning is the Combined Sewer Overflows (CSOs) located in the city of Houston. In recent years, these old sewer systems have become a concern as a potential source of BOD and other harmful pollutants, during runoff events (EH&A, 1994). But, since the city has over 80 overflow locations, time constraints did not permit the consideration of these inputs into the model. Further research could be conducted to develop special boundary conditions or loadings which account for the CSOs during heavy rainfall conditions.

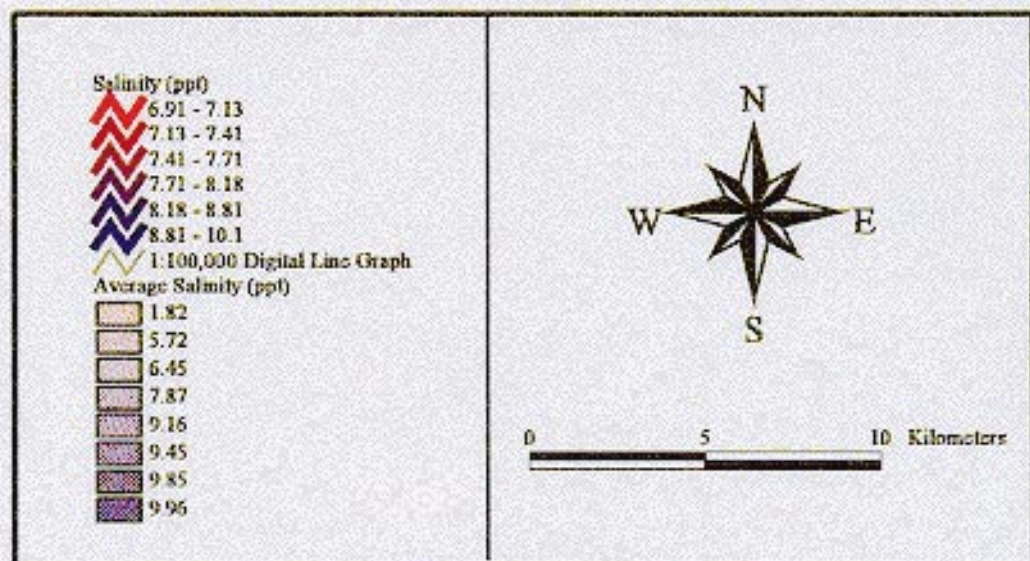
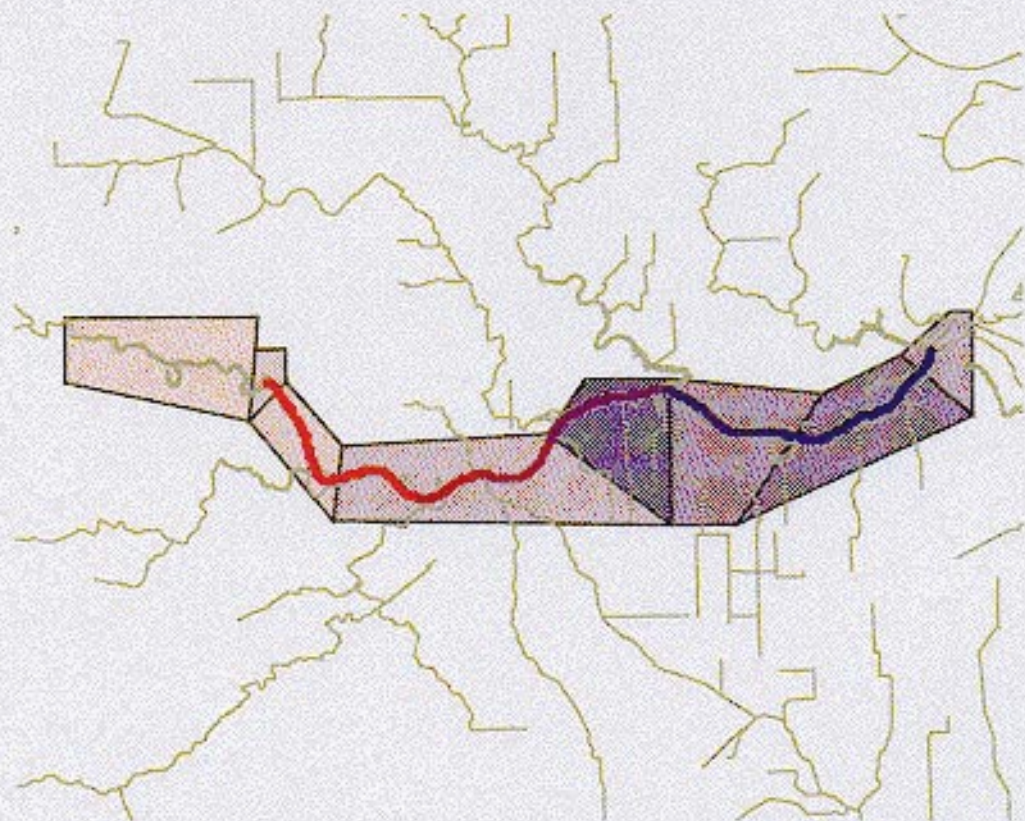
#### **5.4 MODEL CALIBRATION**

Salinity, which is considered an excellent conservative tracer, was used to calibrate the calculated flows, dispersive coefficients, and other parameters used in the Houston Ship Channel model. Two segments were set at their long-term salinity values, while the other segments were input at zero or minimal salinities. The model was executed to quasi-steady state and the calculated concentrations compared to measured values. The results of this model calibration are shown in [Figures 5-3](#) and [5-4](#). [Figure 5-3](#) compares the model results (smooth line) to the long-term averages from Ward and Armstrong (1992). The approximate midpoint of each hydrographic segment shown in [Figure 4-4](#), was used as the kilometer marker for the measurement. The turning basin was assumed to be the starting point and distances were positive, downstream from this datum. In addition, the same relative comparison is shown in [Figure 5-4](#), with the use of ArcView. The hydrographic segments are shaded to their corresponding concentrations, and the segments are also color-ramped in a similar fashion. Both figures show relatively good agreement of the model with the averages, with the downstream values being slightly better. Differences are probably most due to the fact that these point measurements are actually averages taken over as many as 50 or 60 years. As a result, these averages may not be truly representative of the system for flows and exchange coefficients from data for 1960 to 1990. To more accurately compare measurements to the model results, data from just those years should be averaged and used.



**Figure 5-3** Model calibration results. Smooth line is model concentrations, while the points are averaged measurements from Ward and Armstrong (1992). The averages also have their corresponding ranges shown.





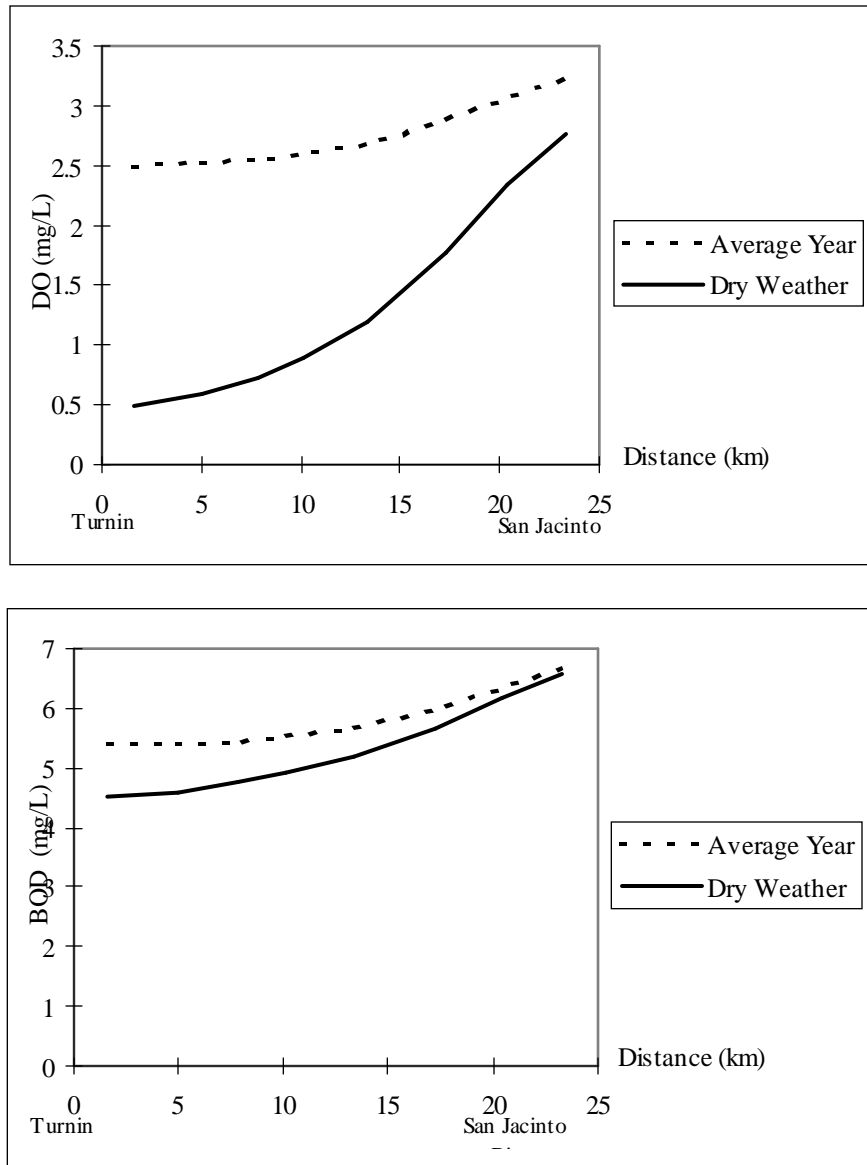
**Figure 5-4** GIS representation of model calibration results. Polygons correspond to segmentation and average measurements from Ward and Armstrong (1992).

## 5.5 BOD/DO MODEL RESULTS

Previous modeling efforts on the Houston Ship Channel resulted in varying DO profiles over distance. In 1971, the modeled DO in the Upper HSC was near or equal to zero for all segments. The 1971 study also projected the DO in the Upper Channel equal to 1.8 mg/L if secondary treatment of wastewater was to be implemented and about 3.4 mg/L if advanced treatment was used (Espey, *et al.*, 1971). Since their modeling project was performed before secondary or advanced treatment processes were actually implemented, it is more accurate to investigate a study performed after the 1970's. In 1984, the TDWR predicted the DO in the Upper HSC would vary from just under 1 mg/L at the Turning Basin to 4.5 mg/L at the San Jacinto Monument (TDWR, 1984). However, it is unclear what reaeration coefficient was used in the 1984 model.

Figure 5-5 illustrates the BOD/DO results for the initial cases modeled in this research (Case I and II: Table 4-15). In this model run,  $k_2$  and  $k_d$  were both set to 0.1 /day. For the average year results (runoff and NPS loads included) the DO varied from 2.5 mg/L to about 3.3 mg/L, while the BOD only had a 1 mg/L gradient from 5.5 mg/L to 6.5 mg/L. For the dry weather conditions (no runoff or NPS loads) the DO in the upper end of the modeled reach changed considerably to 0.5 mg/L, but then improved to 2.5 mg/L by the time the downstream segment was reached. The BOD in the dry weather scenario also decreased, but to a much lesser degree than the DO. The modeled BOD started at 4.5 mg/L in the Turning Basin to about 6.5 mg/L at the downstream end. Figures 5-6 and 5-7 also show a GIS representation of the DO and BOD modeling results.

The differences between the average year and dry weather conditions reflect a change in concentration due to loading and flow modifications. Since the dry weather flows did not include NPS loads or runoff, this case represents baseflow only with point source loads, or a worst case scenario for flow conditions. Because of this low flow condition, the DO concentrations decline, since the Upper HSC does not receive as much DO from the tributaries. The BOD concentrations also decline, although not as sharply as the DO, since



**Figure 5-5** BOD and DO modeling results for average year and dry weather conditions. Graphs show a decrease in both DO and BOD concentrations from the average year (Case I) to the dry weather conditions (Case II) .





(a) Dry Weather



(b) Average Year

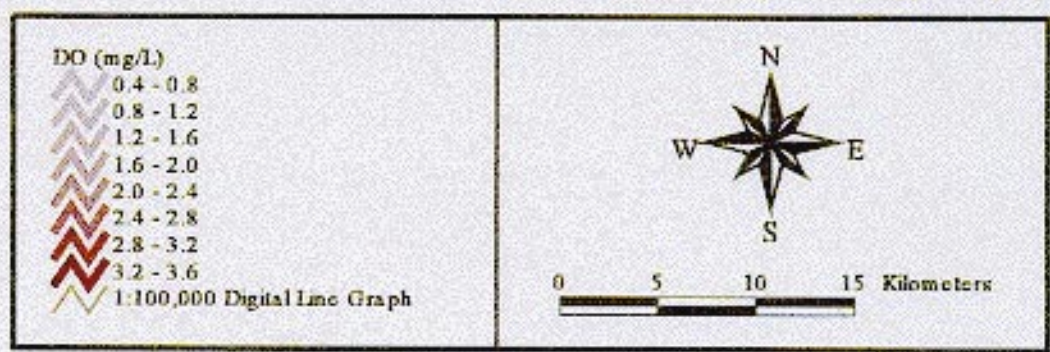
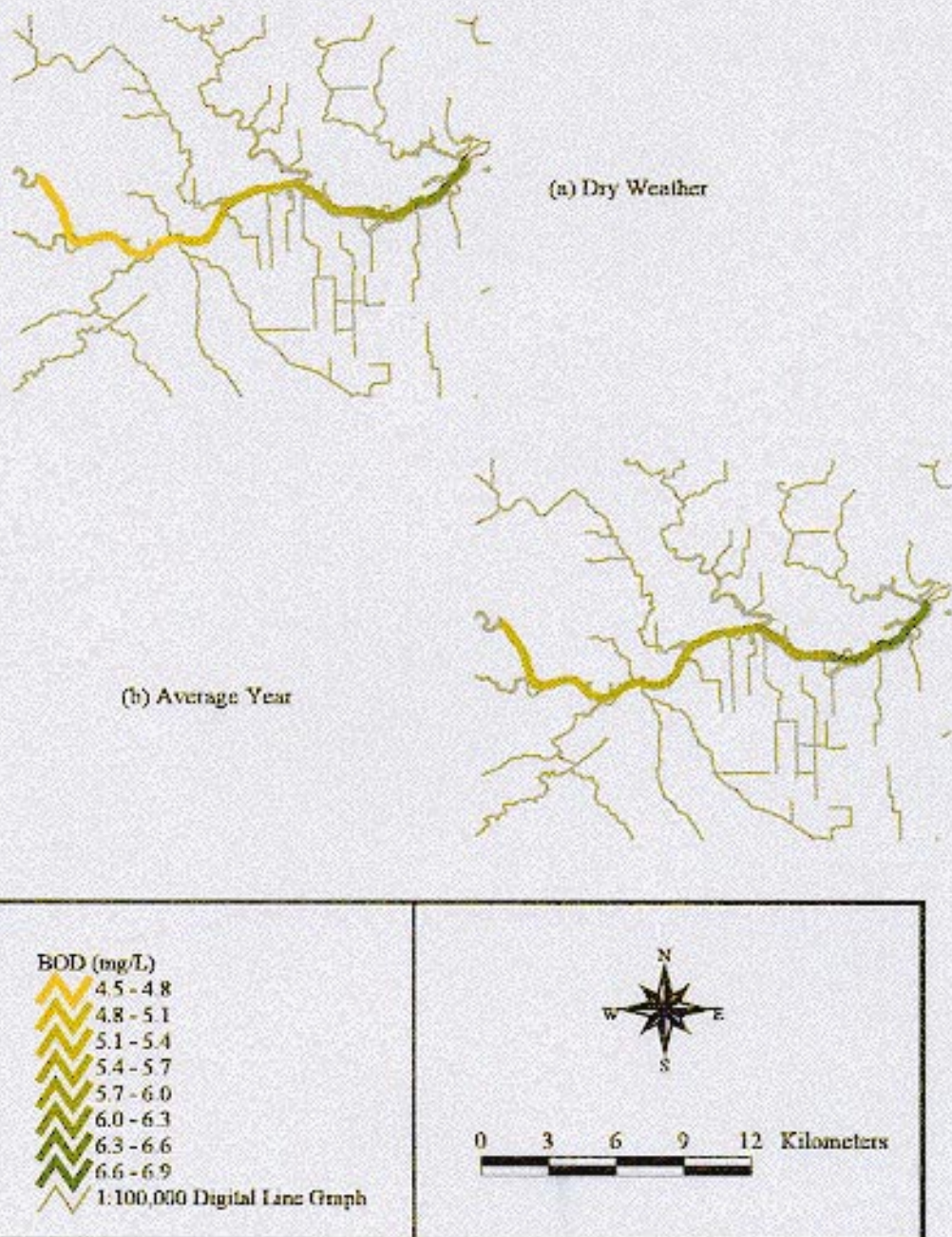


Figure 5-6 GIS representation of DO modeling results for the Upper HSC.





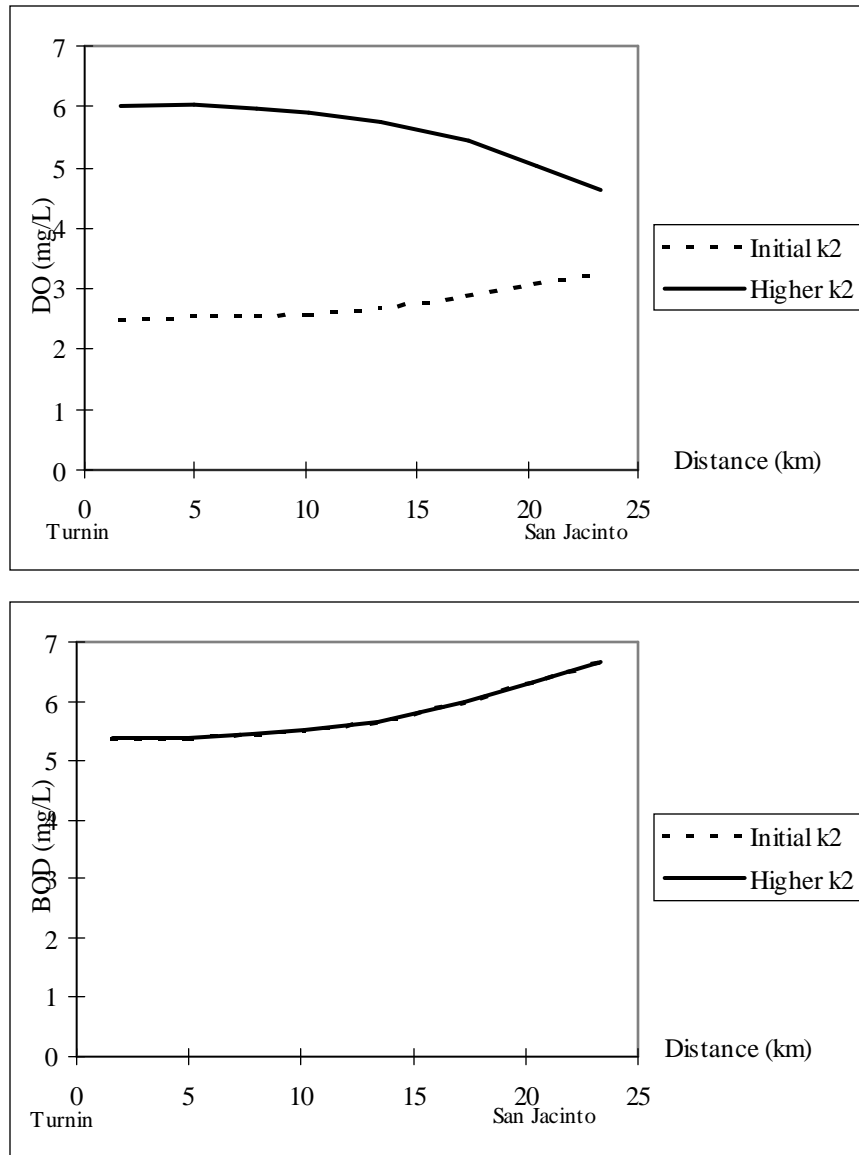
**Figure 5-7** GIS representation of BOD modeling results for the Upper HSC.

the segments are still receiving the point source effluents from the industries, but not the NPS loads. For a comparison of the daily loading to each segment from point and non-point sources, refer to [Table 4-14](#). For both cases, the upstream result in segment 8 is somewhat controlled by the boundary condition set at segment 17 (see [Figures 4-6](#) and [4-7](#)). This boundary condition (3.64 mg/L DO and 7.42 mg/L BOD) accounts for the large inflow of the San Jacinto River entering that segment. To avoid this forced boundary, the entire channel could be modeled, from the Turning Basin to Morgan's Point (the confluence of the channel with main Galveston Bay -- see [Figure 1-1](#)). As a result, Galveston Bay could be set as the downstream boundary condition instead of a portion of the channel, resulting in a more representative model of the entire system.

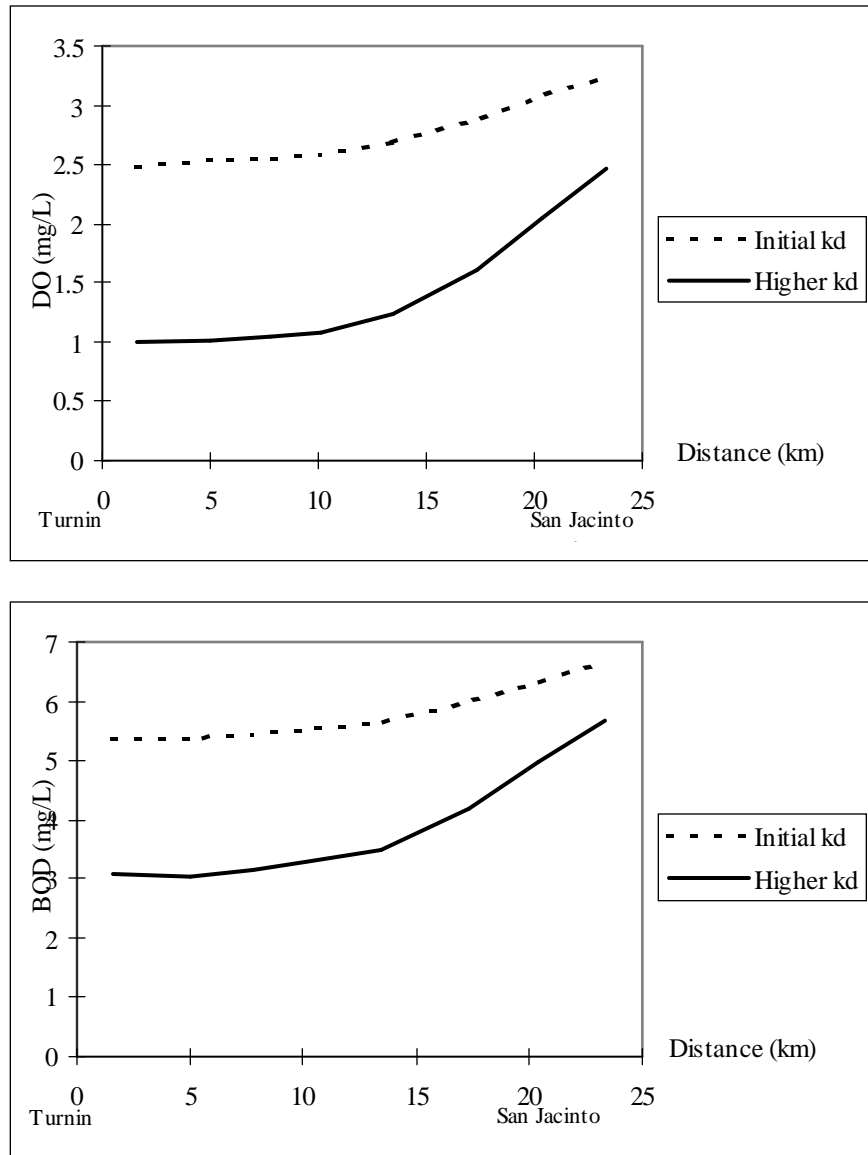
In comparison to the 1984 modeling effort, the overall averages of DO in the Upper HSC are about the same (around 2 to 2.5 mg/L). The current model shows a higher concentration in Turning Basin than the 1984 model. This result may be accurate, since the loadings to the Houston Ship Channel are said to have declined in the past decade (Shipley and Kiesling, 1994). The downstream result for the 1984 model, though, is higher than the result from this study's model (TDWR, 1984). This difference may be due to the boundary condition mentioned in the previous paragraph. In addition, it is unclear exactly what reaeration coefficient was used in the 1984 study for this 25 km reach. As a result, the differences in the two models' results could be attributed to a change in the reaeration coefficient.

## **5.6 MODEL SENSITIVITY**

To investigate the model's sensitivity to the constants used in the input, two additional cases were run on the average year conditions (see [Table 4-15](#)). The cases either increased reaeration coefficient,  $k_2$  or deoxygenation rate,  $k_d$ , while keeping all other parameters, loadings, and flows the same. [Figures 5-8](#) and [5-9](#) show the results of this effort. [Figure 5-8](#), which illustrates Cases I and III from [Table 4-15](#), indicates that that DO concentration is dependent on the value of  $k_2$ . The results show an increase of anywhere



**Figure 5-8** Comparison of Case I and Case III for BOD/DO model. Both used average year conditions, Case I:  $k_2 = 0.1$  /day, Case III:  $k_2 = 0.5$  /day. The BOD concentrations in both Cases are the same.



**Figure 5-9** Comparison of Case I and Case IV for BOD/DO model. Both used average year conditions, Case I:  $k_d = 0.1$  /day, Case IV:  $k_d = 0.3$  /day.



from 1.5 to 3.5 mg/L of DO over the modeled water body. The concentration of BOD, though, stays exactly the same for both cases. **Figure 5-9**, which represents a comparison between Cases I and IV, shows decreases in both BOD and DO concentration for an increased  $k_d$ .

The increase in DO due to an increase in  $k_2$  (Case III: **Figure 5-8**) is expected; a higher reaeration coefficient indicates that oxygen is being introduced into the water at a faster rate. The influence of this change is shown especially in the upstream portion of the HSC. However, the modeled concentrations seem more dependent on the aforementioned boundary concentration in segment 17 for the downstream segments. The fact that the BOD did not change for a higher in  $k_2$  indicates that its concentration is dependent on the loadings to the system and the value of  $k_d$ , rather than the concentration on DO in the water. However, it should be noted that more sophisticated model of the system would result in the BOD concentration depending on the DO concentration. For an increased  $k_d$  (Case IV: **Figure 5-9**) the changes in DO concentration from the initial case (Case I) are not as pronounced as for Case III. The dissolved oxygen follows the same profile in both Cases I and IV, but the overall difference only varies from about 1 to 1.5 mg/L. This decrease in DO is understandable, since the BOD is being oxidized at a faster rate. The overall result is that more BOD is used, causing lower DO concentrations. Likewise, the higher  $k_d$  also reduces the final BOD concentration. Similar modeling results were observed in the TDWR model when investigating sensitivity. A 50% change in the reaeration coefficient resulted in a significant difference in the final DO concentrations. However, the same percentage change to the oxidation coefficient did not effect the modeled DO much (TDWR, 1984).

## **5.7 WASP5/GIS CONNECTION**

### *Introduction*

The overall connection illustrated in **Figure 4-24** is accomplished in GIS through scripts written in Avenue, the programming language for ArcView. The connection allows the user to perform a number of modeling scenarios, including model calibration, average

year BOD/DO modeling, and dry weather BOD/DO modeling. At the present time, the connection set up for a Simple Streeter-Phelps model through EUTRO5 or a model calibration using level one TOXI5. In addition, this initial link only considers steady state processes.

For the connection to work, all of the tables and coverages described in [Section 4.6](#) must be opened in the ArcView project. In addition, the field names for each table have to correspond to the names described in [Appendix A](#). Presently, the GIS/WASP5 connection can be termed a "loose link", since a FORTRAN program actually formats the data exported from ArcView into the input file for WASP5 and another FORTRAN program reads the WASP5 output file to write a readable text file for ArcView. Even so, the connection can be considered quite unique in that it utilizes Avenue and provides a link into and out of the GIS software, ArcView. Most other examples of connections to models with GIS have been using Arc/Info, which is command-line based and primarily utilized in coverage creation. ArcView, on the other hand, is a user friendly Windows-based program which lends itself to customized menus, multiple display of coverages, and easy output presentation.

It is important to note that a basic knowledge of WASP5, specifically EUTRO5, and a general understanding of ArcView is necessary to use this connection to its fullest extent. The help file provided with the connection ([Section 4.6.9](#) and [Appendix G](#)) is meant to assist the user with the general concept and processes in the GIS/WASP5 link. It is not, however, intended as a stand alone document for ArcView and WASP5 methodologies or concepts. When creating this connection, it was assumed that the user understands the parameters needed for a WASP5 input file and also knows how to produce this input in the necessary GIS format. Information on WASP5 can be obtained from the USEPA Homepage (<http://www.epa.gov/>) and the WASP5 User's Manual (Ambrose *et al.*, 1993). In addition, some knowledge of ArcView is an advantage when the final output presentation stage is reached. Although the interface does create new charts and themes, the final formatting of these objects has to be performed by the user ([Section 5.8](#)).

## *Scripts*

As mentioned in [Section 4.6](#), there are 33 Avenue scripts which run the GIS/WASP5 connection. The choices on the customized menus shown in [Figure 4-25](#) control each of these scripts. In most instances, the order in which the menu items are invoked is not important, excluding the following cases:

1. *Input Block A: Model Options* is run before executing any other scripts. This input block sets the working directory which tells Avenue where to find the FORTRAN programs and model executables. This input block also sets some initial model parameters for the rest of the input block generation and allows the user to name the WASP5 input file. Note that for model calibration, Input Block A is automatically run if "Write Input Information" is chosen ([Figure 4-25](#)).
2. *Check Model Parameters* is always run before processing and viewing output data from WASP5. This script double checks the print options set for the output file and retrieves the name of the output file to process.

A beneficial characteristic of the menu structure is the ability to perform the steps that produce model output at different times. Multiple input files can be generated one day, the model runs performed later, and then any output file processed at any time. Any table resulting from the output processing can be used to create new themes and charts in the *View Output* command during any project session.

The first customized menu *BOD/DO Input Blocks* has the option of creating all 10 WASP5 input blocks at once, or executing just one script that generates the related text file for one particular block. This feature is an advantage if only one or two model characteristics need to be changed after the initial creation of an input file. For example, after an input file for a given model with the initial conditions is made, the user can create input files for the same system while changing just the model constants or boundary conditions.

### *Input Message Boxes*

The interface used to input and choose information is the Avenue input message box. These message boxes (see [Figure 4-32](#)) allow the user to choose from a list of available variables, or input required parameters by keyboard. One limitation to these message boxes is size; the size of the box is preset in Avenue. As a result, it is sometimes difficult to reduce the amount of text in a string so that all the information is displayed in the choices. This problem, shown in [Figure 4-35](#) (b) is only resolved when the string is chosen, causing the rest of the text to appear in the top selection box.

### *Limitations*

One limitation to the interface is its extreme sensitivity to the table, coverage, and field naming conventions. If a table or coverage outlined in [Sections 4.6.4](#) and [4.6.5](#) is not included, named differently, or its fields named incorrectly, the scripts are not able to read the table information to print to the text file. This error occurs because each script searches for the tables by its name or alias. In addition, the units outlined in [Appendix A](#) for the table fields have to be adhered to, since the scripts contain unit conversions to write the input information for WASP5 in the proper dimensions.

Limitations are also present in the output processing, table/chart relationships, and multiple new theme creation. The maximum length of the output file to be processed is dependent on the amount of available memory of the computer running the connection. For EUTRO5, 18 lines of text are written for each segment at every time step. In the same way, 7 lines of text are written for each segment at each time step in the TOXI5 output file. If an "out of memory" or "stack overflow" error occurs when trying to process the output data file, the number of times steps written to the output file can be reduced to decrease the number of lines in the WASP5 output files. Concerning the table/chart limitation, a chart is always related to its data table. If one or a number of records are chosen in a table, the chart will display all selected records. Once the table selection changes, the chart modifies its display, accordingly. This restriction becomes a problem when two records from the same chart want to be viewed separately. To get around this relational limitation, two identical tables are

added to the same project. A different chart is then linked to each table. Finally, the ArcView constraint related to the new theme creation has to do with joining new fields to a table. When a new theme is created by the output viewing step, the aat of the "main segmentation" coverage is actually joined with the values of concentration at a chosen time step. This field then becomes the attributed field for the theme legend. However, although multiple copies of this theme can be added to a view, only one aat truly exists for this coverage. Because of this concept, once a project is exited, ArcView only "remembers" the field for the last theme created and the concentration values from that time step control all "main segmentation" themes attributed with concentration. One way to avoid this problem is to manually create a dbf table with a field for each concentration value to be displayed. This table is joined to the "main segmentation" table and multiple themes are attributed with the desired field, instead of using the same field with different concentration values depending on the theme.

The final limitation to the connection is related to the speed of the model run through ArcView. Since WASP5 is a DOS-based model, ArcView must execute a DOS prompt to execute the model. However, when this process occurs, the graphics entailed in displaying the DOS prompt and running the model cause the program to run at a slower speed than if the model were run directly through DOS (Plover, 1996). In addition, when ArcView runs either EUTRO5 or TOXI5, ArcView is suspended and control is passed to the DOS prompt. Any further ArcView scripts are delayed with a message box display (see [Figure 4-32](#)). When the model is completely finished running, control is returned to ArcView by pressing "OK" in the message box.

#### *Comparison to Previous Connections*

As mentioned in Chapter 2, the majority of the previous connections of GIS to existing models have used the software programs, Arc/Info and GRASS. This research, on the other hand, utilized the ArcView software with the object-oriented programming language, Avenue. Specifically in relation to the WASP model, the only other connection that was found in the literature used Arc/Info and its corresponding macro language to

establish a link between EUTRO4 and GIS (DePinto, *et al.*, 1993 and DePinto, *et al.*, 1994). Although both this current connection and the EUTRO4 connection, called GEO-WAMS, would be considered "partial integrations" as described by Tim and Jolly (1994) (see [Figure 2-1](#)), the Arc/Info interface is more advanced than this current ArcView link. The connection for this present research is still in its initial stages and there is much room for modeling advancements in the interface. However, it seems that the current trend in GIS modeling incorporates ArcView and Avenue more frequently, since ArcView is more visual and user-friendly than Arc/Info. As a result, the possibility exists to further develop this connection to the level that GEO-WAMS has attained.

#### *Future Possibilities*

Now that the initial connection to the WASP5 model is established, the possibility for further refinement and advancements in the link is completely possible. As newer versions of ArcView are released, updates to Avenue may make it easier to format the data directly from Avenue to a WASP5 input file, instead of using FORTRAN to format the files. Another possibility to investigate would be to further utilize GIS's topological concepts by having the scripts determine direction of flow and neighboring segments and possibly develop the option for more complex modeling (i.e., overlying water segments, etc.). In the future, the connection can be updated to run all levels of complexity of WASP5, along with time varying parameters. A final goal could be to develop the pollutant loading model within ArcView so that impacts on land use or engineering practices to the receiving waters could be observed directly through ArcView, without having to perform the NPS loading analysis in Arc/Info.

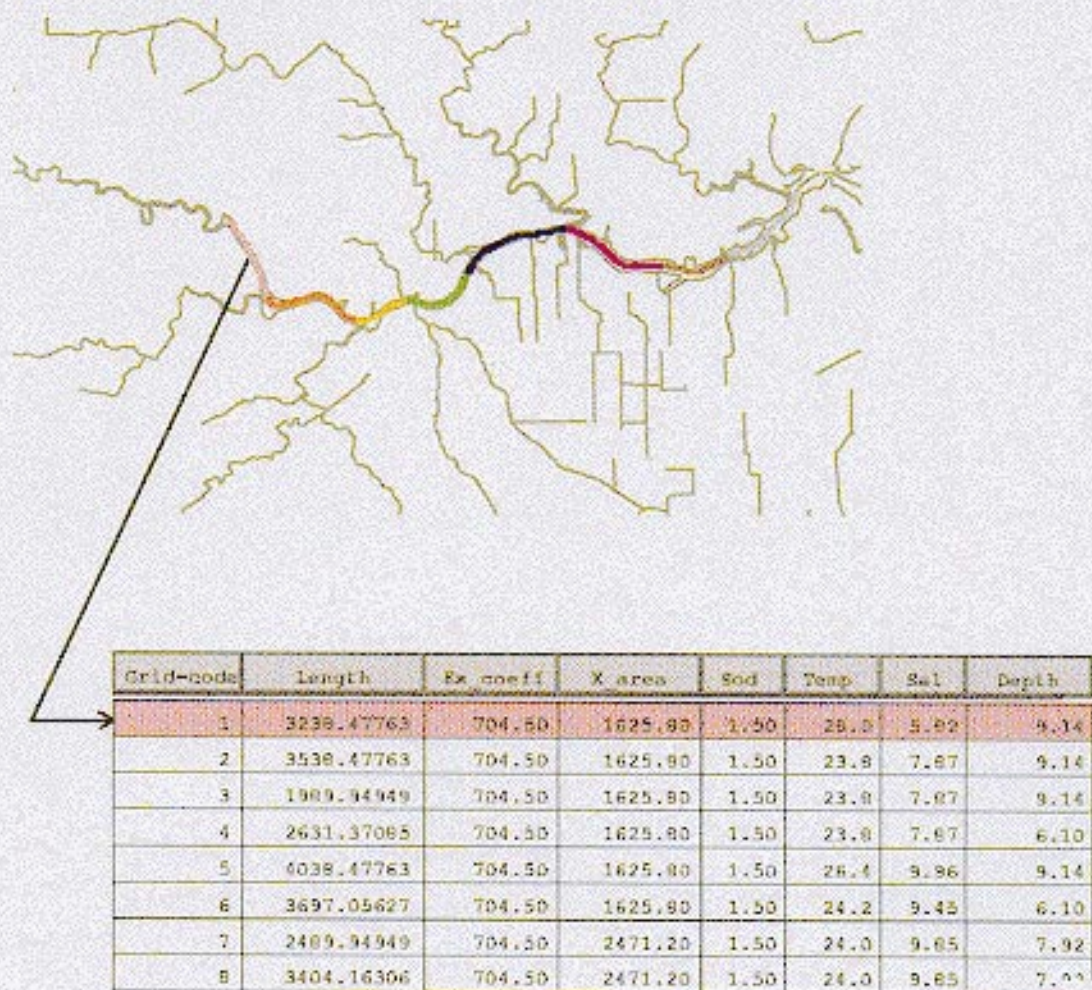
## **5.8 OUTPUT PRESENTATION**

One of the powers of ArcView lies in its output presentation capabilities. Once a general knowledge of the software is gained, the possibilities for communicating information to engineers, city planners, or the general public are limitless. With the output data processed, different tables and charts can be related, along with various new themes, to display the model results. [Figures 5-10](#) through [5-13](#) illustrate just a few examples of the

output formats possible through ArcView. **Figure 5-10** shows the relational capabilities of GIS through a one-to-one relate. The table shown on the bottom of the page is related to the segmentation coverage in that each arc in the coverage has one corresponding record in the table. This ability is an advantage if the user wants to see the given information about one segment, quickly and easily. By highlighting the segment on the coverage, the corresponding record highlights in the table; the opposite occurs if the table record is first highlighted. In addition, multiple tables can be linked together to allow even more information to be displayed. For example, if concentration measurements for the segments at different times are known, a table containing this information can be made in ArcView and a one-to-many relational link established between the present table and this new table. The user can then view all measurements taken in a particular segment by just clicking on that segment.

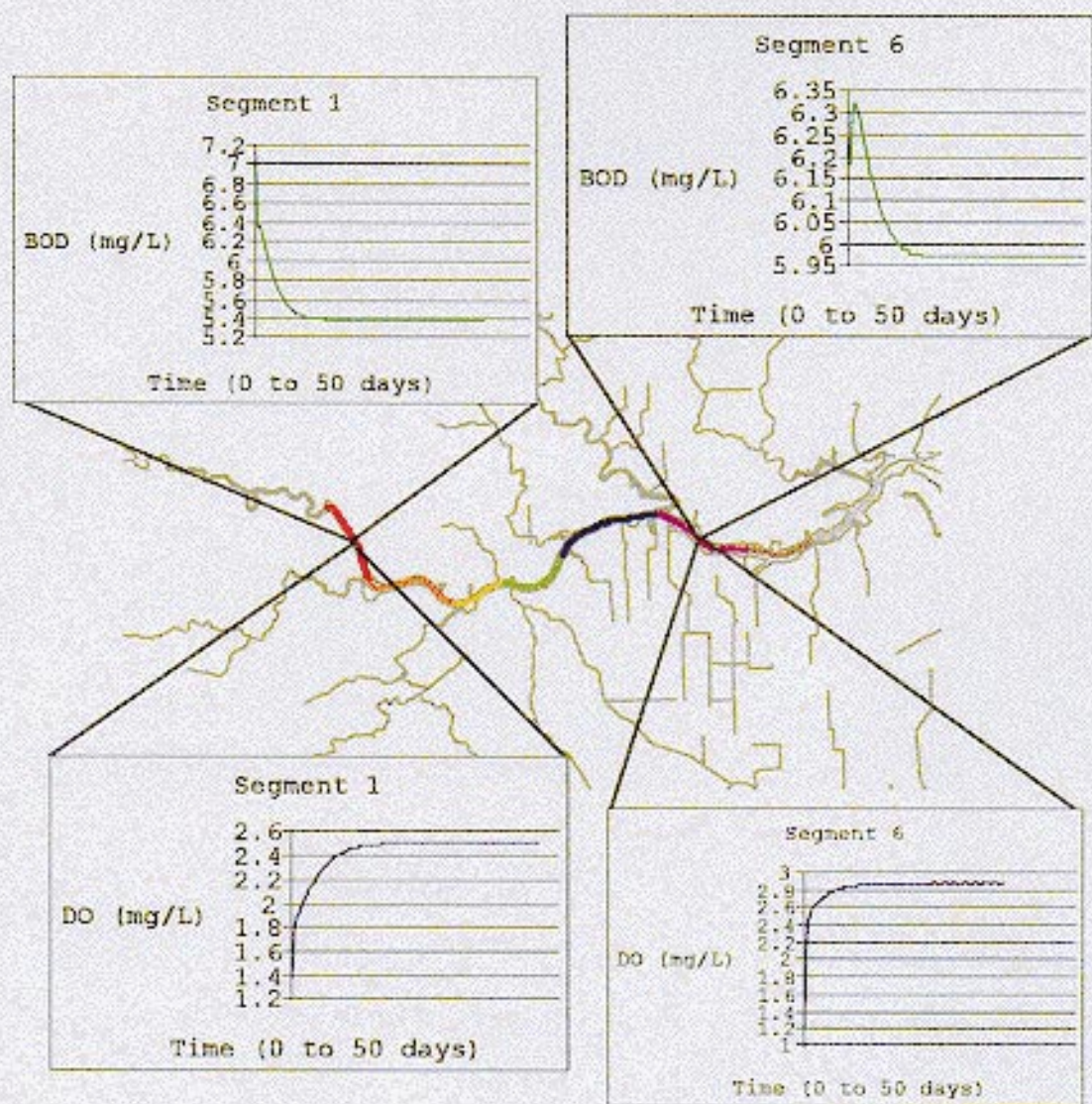
**Figure 5-11** illustrates the charting capabilities of ArcView for output presentation. This figure shows BOD and DO concentrations versus time for two segments in the main reach. By formatting the charts and themes in an ArcView layout, the user can depict the different concentration changes in different segments over time, with one picture. Unfortunately, the chart formatting options in ArcView are limited which makes it difficult to obtain consistent graphs in **Figure 5-11**. **Figure 5-12** depicts the output from a "movie"; one of the output viewing options for the GIS/WASP5 link. With this figure, one can see the change in concentration of DO over time. Although, for steady state model runs, these views only illustrate the model reaching quasi-steady state, this option lends itself well to output viewing if time-variant modeling is attempted. Finally, **Figure 5-13** is meant to show the multiple display ability of ArcView. The number of objects which can be displayed on layout is only limited by style and space. In **Figure 5-12**, a table, chart, and view, all portraying dissolved oxygen results, are placed on the layout. The chart allows the user to see the final concentration in all model segments, while the table and view permit representation of the concentration in just the main modeled reach.





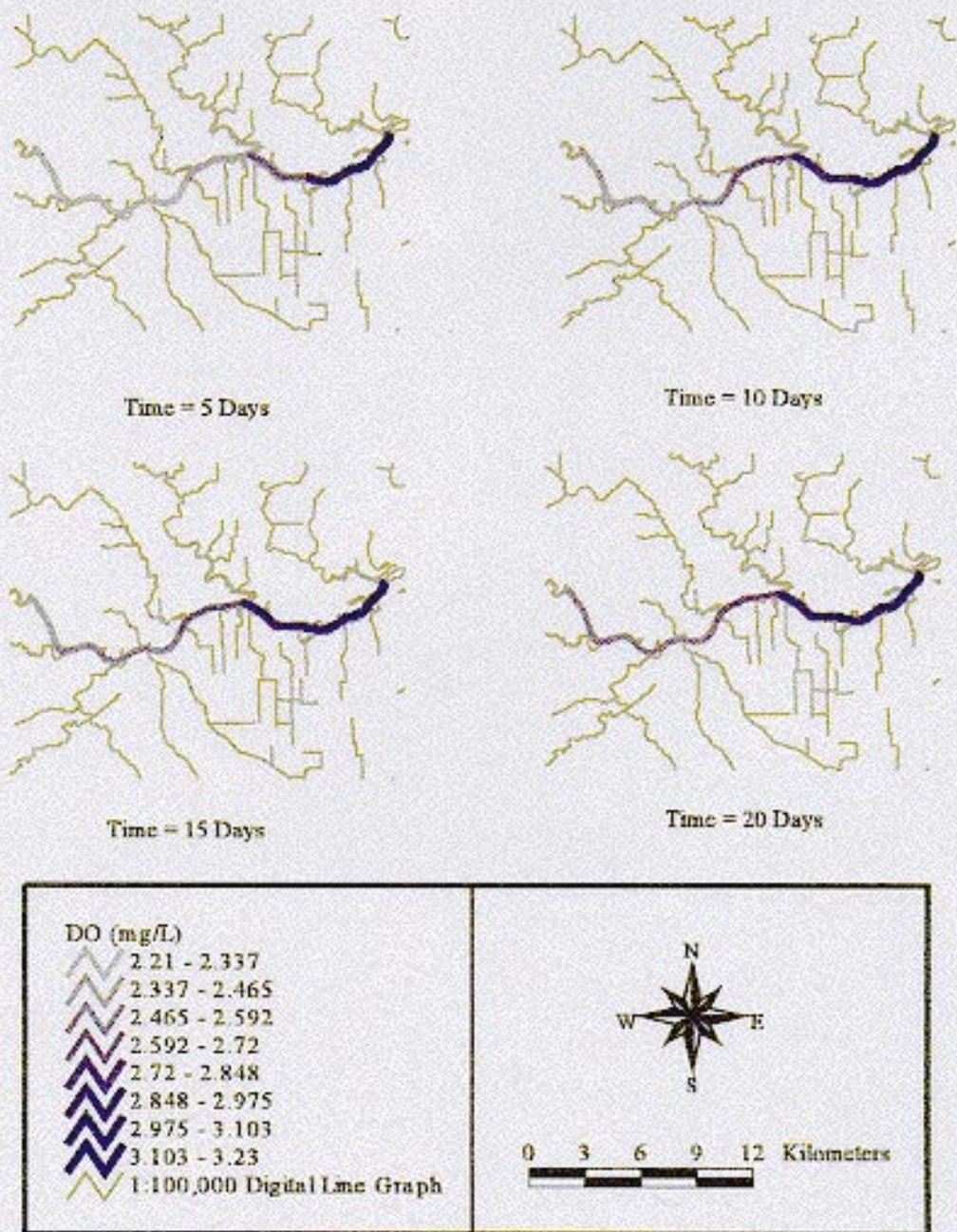
**Figure 5-10** One-to-one relation which exists between the main segmentation theme and its corresponding parameters table. Since these two objects are joined together through the coverage's arc attribute table, the related record on the table will highlight when the segment is chosen, or vice versa (pink highlight).





**Figure 5-11** Layout showing the charting capabilities of ArcView. Graphs for DO and BOD versus time are displayed for two different segments. Presently, the charts illustrate the model run reaching quasi-steady state. If time-variant inputs were used, the charts could display the change in concentration in response to a loading or flow change.





**Figure 5-12** Four views created from "movie" display option in the output presentation process from the GIS/WASP5 connection. These views presently show the model reaching steady state, but can also show the response of the system to a pulse input, if time-variant parameters are established in the connection.

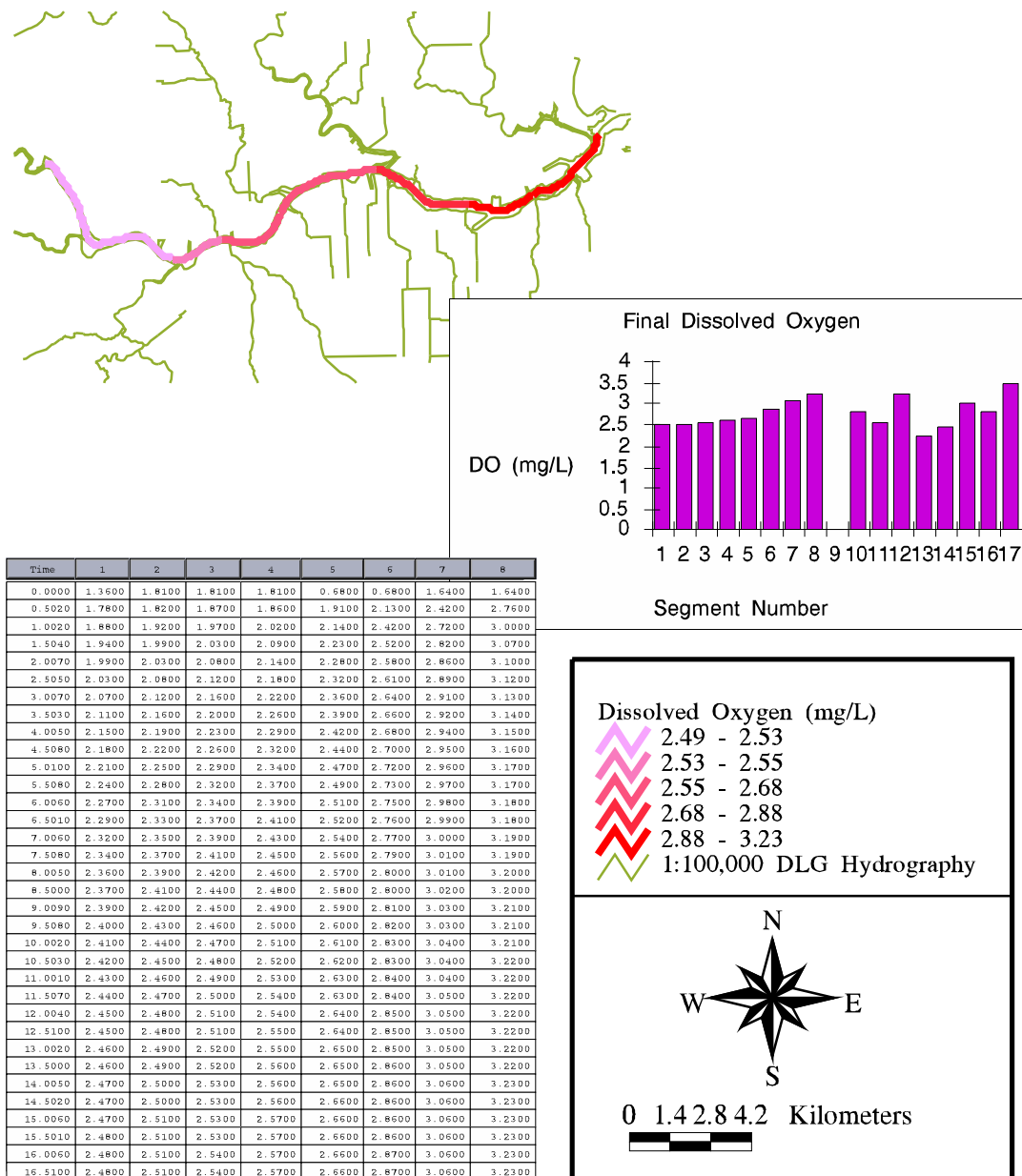


Figure 5-13 Layout illustrating the multiple display capabilities of ArcView. A view of the segmentation with the final DO concentration gradient is shown, along with the table from where the final data was obtained. The chart shows the final concentration of DO in every model segment, including the boundary ones.

## Chapter 6

### Conclusions and Recommendations

In this research project, the GIS software, ArcView was connected to the USEPA water quality model, WASP5. The study area utilized as a test case for this linkage was the Upper Houston Ship Channel, located in Galveston Bay, Texas. The connection successfully reads the necessary model input for the model and writes it to text files, executes a FORTRAN program which formats the data into the WASP5 input file, runs the WASP5 model, processes the output, and allows the user to view the output in the form of charts, GIS coverages, and tables. This initial connection establishes a primary link, which, with further development, can be used as a watershed management tool to investigate the impact of engineering practices on the water quality of a given system.

Specific conclusions, limitations, and recommendations for the five phases outlined in [Chapter 1](#) are enumerated below.

*Phase I            Literature Review*

- Previous model connections and environmental modeling integrations were found in the literature. However, the majority of these modeling efforts were performed using the GIS software, Arc/Info.

*Phase II           GIS Data Development*

- The Upper Houston Ship Channel and its surrounding watershed characteristics were digitally represented through GIS using coverages and tables.
- In particular, the watershed boundary was successfully established using an Arc/Info, Grid-based watershed delineation technique developed from previous research efforts. In addition, the watershed areas draining into each water quality segment were delineated.

- The land use, precipitation, streams, water quality data, channel segmentation, and point source loading data were also represented through GIS.
- The accuracy of these coverages and tables is dependent on the source and scale from which the original data was obtained.

*Phase III Non-point Source BOD Loading Determination*

- A rainfall/runoff/urbanization relationship was established for the Upper Houston Ship Channel watershed, using available flow, precipitation, and land use data, to spatially distribute the steady state runoff over the entire watershed area. This relationship proved to be reasonably accurate for long-term steady state results.
- The total steady state areal BOD loading across the watershed was determined by multiplying the spatially distributed runoff and a land use-based estimated mean concentration (EMC), determined specifically for the Houston area. This pollution assessment procedure was performed with a method developed in Arc/Info.
- The total annual steady state BOD loading into each water quality segment was calculated by "routing" all of the areal loads from a segment's subwatershed to its final outlet point.
- The NPS loadings established in this study are a good representation of average steady-state values; however, more representative results would be possible if both time-varying loading and EMC values were investigated.
- Future work in this area should include the possibility of developing this loading analysis through ArcView to extend the current GIS/WASP5 connection.

*Phase IV WASP5 Modeling*

- A water quality model was established in the Upper Houston Ship Channel and calibrated using salinity, with WASP5's subprogram, TOXI5. Once calibrated, EUTRO5 was employed to model BOD and DO in the Channel.

- For the EUTRO5 model, the total point source and NPS loading of BOD to each model segment was determined from previous calculations and available data. However, the point source loading used was incomplete, since it represented only about half of dischargers along the channel. In addition, each tributary was modeled as a boundary, with a concentration and flow to account for the loadings.
- Four model runs were performed to simulate steady state BOD/DO processes with a simple Streeter-Phelps level of complexity. The runs included an average year, a dry weather condition, and two cases to investigate model sensitivity to the system constants.
- The flow conditions were established using a relationship similar to the rainfall/runoff/urbanization correlation from Phase III. Once the flow and runoff (from Phase III) was known, the baseflow was calculated by subtracting the runoff from the total flow.
- For the average year modeling case, total flow, along with point and NPS loadings were input to the system. In contrast, the dry weather condition, which was meant to represent a worst case scenario in relation to flow, only considered baseflow with the point source discharges.
- The DO results were found to be extremely sensitive to the choice of the reaeration coefficient,  $k_2$ , but not as dependent on the value of the deoxygenation coefficient,  $k_d$ . The BOD results were dependent only on the  $k_d$ .
- The DO and BOD concentrations resulting in the channel also depended on the value chosen for the downstream boundary condition.
- Future modeling would be more accurate if it were to consider the entire Houston Ship Channel, from the Turning Basin to Morgan's Point. In this way, the influence of the downstream boundary condition may be reduced.
- More research into the point source loadings and the boundary concentrations should be performed to more accurately represent the system.
- Time varying inputs, flows and parameters should also be investigated to explain the NPS loading and overall system more accurately.



*Phase V GIS/WASP5 Connection*

- The WASP5 model was connected to GIS, using FORTRAN programming and the ArcView programming language, Avenue.
- The current Arc/Info connection to WASP4, GEO-WAMS (DePinto, *et al.*, 1993 and DePinto, *et al.*, 1994) is more developed than this ArcView interface; however, with more research, this link has the potential to reach, and possibly exceed, the level GEO-WAMS has attained.
- This connection includes the ability to create WASP5 input files, to execute WASP5, and to view the output from the model run, all through an ArcView interface. The results of the model can be viewed through ArcView as either new charts, tables, or themes.
- The interface allows the user to choose three types of modeling cases: model calibration with salinity, average year BOD/DO conditions, or dry weather conditions.
- Presently, this connection is set only for steady state conditions and the BOD/DO modeling is established only for level one complexity in WASP5 (simple Streeter-Phelps equation).
- Limitations to the connection included the slowness of a model run through ArcView, the inability to relate more than one chart to a table, the constraint on number of time steps the output program could process, and the restriction on the number of new themes that are saved to a project.
- The constraint on the naming of tables and coverages should be studied in relation to the Avenue language.
- New versions of ArcView would support investigation into the establishment of a more direct link to WASP5 as the text formatting capabilities of Avenue become more refined.
- More development of the connection should alleviate the dependency of the scripts on tables and attempt to utilize GIS's relational capabilities within the coverage topology beyond what is established in this study.

Future work should also include the consideration of linking higher levels of complexity in the WASP5 model, which would entail the incorporation of more state variables, parameters, and constants. In addition, the modeling of time-varying conditions within the connection should be studied.

- The final ultimate goal of future research should be to establish a representative model of the loading analysis and overall system within GIS so that, with a link to a water quality model, the overall connection could be used as a powerful watershed management tool.



**Appendix A**  
**Data Dictionary**

### Coverages \*

Coverage	Feature	Class	Attribute	Value	Description
alb5900	USGS gauge station 8075900	Point	usgs_stat	varies	USGS gauge station number
			location	varies	Physical location as described by USGS
			drain_area	varies	USGS drainage area (mi <sup>2</sup> )
			per_of_rec	varies	Available per. of record for flow data
			msrd_run	varies	Avg. msrd runoff depth (1961-1990)
bcseg	Tributaries entering the Upper Houston Ship Channel	Arc	none		
bodgr30	Visualization of areal BOD loading over the Upper HSC watershed (only shows polygons greater than 30000 m <sup>2</sup> -- cannot be used for modeling purposes)	Poly.	grid-code	varies	Areal BOD loading (kg/yr/ha)
cvemc30	Visualization of EMC distribution over the Upper HSC watershed (only shows polygons greater than 30000 m <sup>2</sup> -- cannot be used for modeling purposes)	Poly.	grid-code	0	EMC for water
				4	EMC for agriculture
				6	EMC for open, wetlands, and water
				9	EMC for high density urban
				13	EMC for barren
15	EMC for residential				
cvnoedit	Watershed area delineated without DLG "burn"	Arc	none		
cvpsmsrd	38 point source dischargers along the Upper HSC shoreline which have BOD measurements from Armstrong and Ward (1994)	Point	tnrcc_per_num	varies	TNRCC water quality permit number
			x-coord	varies	x location, in albers units
			y-coord	varies	y location in albers units
cvsegshd	Eight subwatersheds for each water quality segment in the Upper HSC	Poly.	grid-code	1 to 8	Corresponding water quality segment which drains this area
cvshd590	Subwatershed for USGS gauge 8075900	Poly.	grid-code	8075900	USGS gauge which drains this area
dlgedit	Final, edited digital line graph used for digital elevation model "stream burn"	Arc	none		
dlgnohsc	Digital line graph without the HSC defined as a center line	Arc	none		
dlgshd	DLG coverage encompassed by the Upper HSC watershed	Arc	none		

Coverage	Feature	Class	Attribute	Value	Description
gagesd10	All 10 subwatersheds of the USGS gauge stations used in this study	Poly.	grid-code	varies	USGS gauge which drains the corresponding area
gageshd	Each subwatershed for each of the nine USGS gages used in the rainfall/flow and rainfall/runoff correlations	Poly.	grid-code	varies	USGS gauge which drains the corresponding area
gbaydlg	Original 1:100,000 digital line graph of the Galveston Bay area	Arc	none		
gbyhyseg	Geographic projection of the Galveston Bay hydrographic segmentation as given in the GBNEP Report-22 (1992)	Poly.	segment-id	varies	Segment # Ward and Armstrong (1992)
			x-coord	varies	Longitude of label point in decimal deg.
			y-coord	varies	Latitude of label point in decimal deg.
Gbhysegpr	Alber projection of gbyhyseg		segment-id	Same as for gbyhyseg	
			x-coord	varies	x location, in albers units
			y-coord	varies	y location in albers units
hscalone	Arc of the centerline used for the HSC definition	Arc	none		
hsclu	Final landuse coverage used for EMC and Runoff Coefficient values. Combines, primarily 1990 data w/ missing areas of ws filled in by 1980 data	Poly.	grid-code	0	Background
				1	Urban
				2	Open
				3	Agriculture
				4	Barren
				5	Wetlands
				6	Residential
				7	Water
				8	Forest
				curve_num	
				97	CN for grid-code = 1
				80	CN for grid-code = 2
			81	CN for grid-code = 3	
89	CN for grid-code = 4				
67	CN for grid-code = 5				
87	CN for grid-code = 6				
100	CN for grid-code = 7				
77	CN for grid-code = 8				

Coverage	Feature	Class	Attribute	Value	Description
			runoff_coeff	0.89	c for grid-code = 1
				0.22	c for grid-code = 2
				0.24	c for grid-code = 3
				0.22	c for grid-code = 4
				0.8	c for grid-code = 5
				0.34	c for grid-code = 6
				1	c for grid-code = 7
				0.15	c for grid-code = 8
hsc_pscv	68 point source dischargers which lie directly on the Upper HSC shoreline	Point	tnrcc_per_num	varies	TNRCC water quality permit number
			x-coord	varies	x location, in albers units
			y-coord	varies	y location in albers units
lu_80dis	Dissolved (no boundary lines) 1980 land use with Anderson Classification	Poly.	lucode	11	Residential
				12	Commerical Services
				13	Industrial
				14	Transportation, Communications
				15	Industrial and Commercial
				16	Mixed Urban or Built-Up Land
				17	Other Urban or Built-Up Land
				21	Cropland and Pasture
				22	Orchards, Groves, Vineyards, Nurseries
				23	Confined Feeding Operations
				31	Herbaceous Rangeland
				32	Shrub and Brush Rangeland
				33	Mixed Rangeland
				41	Deciduous Forest Land
				42	Evergreen Forest Land
				43	Mixed Forest Land
				51	Streams and Canals
				52	Lakes
				53	Reservoirs
				54	Bays and Estuaries

Coverage	Feature	Class	Attribute	Value	Description
				61	Forested Wetlands
				62	Nonforested Wetlands
				71	Dry Salt Flats
				72	Beaches
				73	Sandy Areas other than Beaches
				74	Bare Exposed Rock
				75	Strip Mines, Quarries, and Gravel Pits
				76	Transitional Area
				77	Mixed Barren Land
			grid-code	Same as grid-code for hsclu	
lu_80ed	1980 land use of the areas within the hsc watershed which were missing the 1990 data	Poly.	lucode	Same as lucode for lu_80dis	
			grid-code	Same as grid-code for hsclu	
lu_90dis	1990 land use coverage of watershed area as defined by Newell et al. (1992)	Poly.	grid-code	Same as grid-code for hsclu	
lu_90shd	1990 land use coverage of watershed area defined by delineation (mapjoined w/ lu_80ed to get hsclu)	Poly.	grid-code	Same as grid-code for hsclu	
p1cov	Coverage of January rainfall depth grid	Poly.	grid-code	varies	Avg. precip. depth for Jan, 1961-1990
panncov	Coverage of annual rainfall depth grid	Poly.	grid-code	varies	Avg. annual precip. depth, 1961-1990
quadpr	Coverage of four USGS quads encompassing the study area	Poly.		none	
ranncv30	Visualization of runoff distribution over the Upper HSC watershed (only shows polygons greater than 30000 m <sup>2</sup> -- can not be used for modeling purposes)	Poly.	grid-code	varies	Areal calculated runoff depth (mm/yr/ha) calculated using rainfall/runoff/urbanization correlation
rstrev	Calculated runoff depth from correlation, routed down the streams	Arc	grid-code	varies	Calculated runoff depth using rainfall/runoff/urbanization eqn. (mm/yr/ha)
segarc	Eight water quality segments	Arc	grid-code	1 to 8	Water quality segment # for model
stat10	All 10 USGS gauge stations used in this study	Point	none		

Coverage	Feature	Class	Attribute	Value	Description
stat_alb	71 USGS gauging stations within Harris, Waller, and Fort Bend Counties	Point	usgs_stat	varies	USGS gauge #
			x-coord	varies	x location, in albers units
			y-coord	varies	y location in albers units
stat_hsc	37 USGS gauging stations in HSC watershed		usgs_stat	varies	USGS gauge #
			x-coord	varies	longitude in decimal degrees
			y-coord	varies	latitude in decimal degrees
stendalb	Start and end points of segmentation for the Upper HSC (Turning Basin and San Jacinto Monument) as read from 1:24,000 USGS quad sheet	Point	x-coord	varies	x location, in albers units
			y-coord	varies	y location in albers units
strnoedit	Stream coverage, delineated without DLG "burn" in	Arc	none		
strshd	Delineated streams (with burn) in the Upper HSC watershed	Arc	none		
str_500	Final coverage of the stream delineation on the 500 cell threshold	Arc	none		
studstat	Coverage of the nine USGS gauges used for this study	Point	usgs_stat	varies	USGS gauge station number
			location	varies	Physical location as described by USGS
			drain_area	varies	USGS drainage area (mi <sup>2</sup> )
			per_of_rec	varies	Available per. of record for flow data
			msrd_run	varies	Avg. msrd runoff depth (1961-1990)
tnrcc_ps	1839 point source dischargers within the Galveston Bay area (obtained from TNRCC)	Point	tnrcc_per_num	varies	TNRCC water quality permit number
			x-coord	varies	x location, in albers units
			y-coord	varies	y location in albers units
totshd	entire watershed area (delineated with burn), with San Jacinto Monument defined as the outlet	Poly.	none		
twcpr	TWC segmentation over the Upper HSC	Poly.	twc_id	1006	TNRCC water quality segment number
				1007	TNRCC water quality segment number

\* NOTES:

The attributes are those in the pat or aat which are additional to the normal topology for those classes. And, all coverages are in USGS-Albers, unless otherwise specified

### Grids \*

Grid	Feature	Type	Z Units
acc_seg	Flow accumulation values along the HSC centerline	Floating Point	# of cells
ad	Grid of digitized USGS watershed for the Addicks Reservoir	Integer	n/a
bf	Grid of digitized USGS watershed for the Buffalo Bayou	Integer	n/a
bk	Grid of digitized USGS watershed for the Barker Reservoir	Integer	n/a
bodad	Areal BOD loading determined by this study for the Addicks Reservoir watershed	Floating Point	kg/yr/ha
bodann	Areal BOD loading over the entire Upper HSC watershed	Floating Point	kg/yr/ha
bodbf	Areal BOD loading determined by this study for Buffalo Bayou watershed	Floating Point	kg/yr/ha
bodbk	Areal BOD loading determined by this study for the Barker Reservoir watershed	Floating Point	kg/yr/ha
bodbr	Areal BOD loading determined by this study for the Brays Bayou watershed	Floating Point	kg/yr/ha
bodfac	Flow accumulation, weighted by BOD loading, over the Upper HSC watershed	Floating Point	kg/yr/ha
bodgr	Areal BOD loading determined by this study for the Greens Bayou watershed	Floating Point	kg/yr/ha
bodsc	Areal BOD loading determined by this study for the Ship Channel watershed	Floating Point	kg/yr/ha
bodsm	Areal BOD loading determined by this study for the Sims Bayou watershed	Floating Point	kg/yr/ha
bodwo	Areal BOD loading determined by this study for the Whiteoak Bayou watershed	Floating Point	kg/yr/ha
bod_out	BOD loading into each water quality segment	Integer	kg/yr/ha
bound1	Start and end points of the HSC study area	Integer	n/a
br	Grid of digitized USGS watershed for the Brays Bayou	Integer	n/a
cchint	Relative runoff coefficient value for station 8075900	Integer	unitless
chnlfl	Flowlength values, just along the HSC study area	Floating Point	m
coeffac	Weighted flow accumulation of relative runoff coefficient over Upper HSC watershed	Floating Point	unitless
demfull	Geographic projection of 1:250,000 digital elevation model of four merged USGS DEMs: houston_west, seguin_east, beaumont_west, and austin_east	Floating Point	m
dlggrid	Grid of edited digital line graph used for HSC burn	Integer	n/a
emcbodgr	Grid of BOD EMC values for Upper HSC watershed	Floating Point	mg/L
emcbodin	Grid of BOD EMC values for Upper HSC watershed	Integer	mg/L
fcalc	Flow depth calculated from rainfall/flow/urbanization relation over Upper HSC ws.	Floating Point	mm/yr/ha
flfac	Flow accumulation, weighted by calculated flow, over the Upper HSC watershed	Floating Point	mm/yr/ha
flfacint	Flow accumulation, weighted by calculated flow, over the Upper HSC watershed	Integer	mm/yr/ha
flout	Total cumulative flow values at each water quality segment outlet	Floating Point	mm/yr/ha

<b>Grid</b>	<b>Feature</b>	<b>Type</b>	<b>Z Units</b>
f_p	Flow/Precipitation as calculated by the rainfall/flow/urbanization equation	Floating Point	unitless
gage	Grid of USGS gauge stations, moved onto 500 threshold streams	Integer	n/a
gage5900	Grid of gauge 8075900, moved onto 500 threshold streams	Integer	n/a
gage_shd	Subwatersheds for each USGS gauge station	Integer	n/a
gr	Grid of digitized USGS watershed for the Greens Bayou	Integer	n/a
grtotshd	Total watershed area with San Jacinto Monument defined as the outlet point, delineated from "burned" DEM	Integer	n/a
hscburn	DEM with the edited digital line graph "burned in"	Floating Point	m
hsccoef	Grid of relative runoff coefficients for Upper HSC watershed area	Floating Point	unitless
hscdem	Original DEM before stream "burn", Alber projection of demfull	Floating Point	m
hscdmalb	Albers projection of DEM, encompassing just beyond the watershed area	Floating Point	m
hscfac	Flow accumulation over the Upper HSC watershed, from hscfdr with stream "burn"	Floating Point	# of cells
hscfdr	Flow direction grid of the HSC watershed	Floating Point	n/a
hscfil	Filled DEM for the HSC watershed	Floating Point	m
hscfl	Flowlength of the HSC watershed	Floating Point	m
hscgrid1	Grid of just the centerline HSC, with all cells set to a value of 1	Integer	n/a
hsc_seg	HSC segmentation for study area	Integer	n/a
outlet	Grid of single outlet point of study area: San Jacinto Monument	Integer	n/a
out_seg	Outlet points defined for each channel segment	Integer	n/a
p1alb - p12alb	Average monthly precipitation grids (1961-1990) for each month	Integer	mm/month/ha
pannalb	Average annual precipitaion grid (1961-1990)	Integer	mm/yr/ha
pannchk	Cumulative annual precipitation value for gauge 8075900	Integer	mm/yr/ha
pannvinn	Cumulative annual precipitation value for nine USGS gauges used in relationship	Integer	mm/yr/ha
rcalc	Calculated runoff from rainfall/runoff/precipitation equation over HSC watershed	Floating Point	mm/yr/ha
rfac	Flow accumulation, weighted by calculated runoff, over the Upper HSC watershed	Floating Point	mm/yr/ha
rstr	Average runoff depth over an area upstream of a given point on the delineated streams	Integer	mm/yr
r_annint	Calculated runoff from rainfall/runoff/precipitation equation over HSC watershed	Integer	mm/yr/ha
r_out	Total cumulative runoff values at each water quality segment outlet	Integer	mm/yr/ha
sc	Grid of digitized USGS watershed for the Ship Channel	Integer	n/a
seg_shd	Subwatersheds for each segment in the HSC study area	Integer	n/a



<b>Grid</b>	<b>Feature</b>	<b>Type</b>	<b>Z Units</b>
shd5900	Delineated watershed for gauge 8075900	Integer	n/a
sm	Grid of digitized USGS watershed for the Greens Bayou	Integer	n/a
str2000	Delineated streams, from burned DEM, at 2000 threshold	Integer	n/a
str500	Delineated streams, from burned DEM, at 500 threshold	Integer	n/a
strnobrn	Delineated streams from original DEM (w/o burn), 500 threshold	Integer	n/a
totshd5	Delineated watershed, from original DEM (no stream burn)	Integer	n/a
wo	Grid of digitized USGS watershed for the Whiteoak Bayou	Integer	n/a

\* NOTES:

All grids are 100mx100m cells, in USGS-Albers, unless otherwise specified

#### Shape Files

<b>Shape File</b>	<b>Feature</b>	<b>Class</b>	<b>Coverage</b>
hydroseg.shp	Hydrographic segmentation over the Upper HSC (Ward and Armstrong, 1992)	Poly.	n/a
bc_seg.shp	Boundary water quality modeling segments for Upper HSC	Arc	bcseg
ps_msrd.shp	68 point source dischargers along the Upper HSC	Point	hsc_pscv

### Tables

Table	Feature	Alias*	Attribute	Description	Units
bnd_seg.dbf	dBase File which contains boundary segment modeling parameters	Boundary	grid-code	Boundary segment #	n/a
		Segments	ex_coeff	Exchange coefficient	m <sup>2</sup> /sec
			x_area	Cross-sectional area	m <sup>2</sup>
			upstr_seg	Upstream segment	n/a
			dwnstr_seg	Downstream segment	n/a
			btm_seg	Segment below this segment	n/a
			int_bod	BOD boundary concentration	mg/L
			int_do	DO boundary concentration	mg/L
			sod	Sediment oxygen demand	g/m <sup>2</sup> -day
			temp	Temperature	° C
			sal	Salinity	ppt
			act_length	Length of boundary segment	m
			depth	Depth of segment	m
			type	Type of segment (WASP5 standards)	n/a
			perpin	Y or N to indicate orientation to main channel	n/a
	name	Descriptive name of the boundary segment	n/a		
bc_seg.dbf	dBase File attached to bc_seg shape file	n/a	No additional attributes		
bod_out.vat	vat of BOD flow accumulation value for each segment outlet	BOD	value	Arc assigned cell value	n/a
		Loading	count	# of cells with this value	n/a
		Values	grid-code	Main segment #	n/a
			bodfac	accumulated bod loading	kg/yr/ha
bod_ps.dbf	dBase File of total point source loading into each segment	Point Source	segment	Main segment #	
		BOD	BOD	BOD loading	10 <sup>3</sup> lbs/yr
bodavyr.dbf	dBase File of modeling results of BOD for average year conditions	n/a	Time	Time	days
			1 through 17	BOD concentration for the respective segment #	mg/L
boddry.dbf	dBase File of modeling results of BOD for dry weather conditions	n/a	Time	Time	days
			1 through 17	BOD concentration for the respective segment #	mg/L
bodhgk2.dbf	dBase File of modeling results of BOD for average year conditions, high k <sub>2</sub>	n/a	Time	Time	days
			1 through 17	BOD concentration for the respective segment #	mg/L
bodhgkd.dbf	dBase File of modeling results of BOD for average year conditions, high k <sub>d</sub>	n/a	Time	Time	days
			1 through 17	BOD concentration for the respective segment #	mg/L
doavyr.dbf	dBase File of modeling results of DO	n/a	Time	Time	days

Table	Feature	Alias*	Attribute	Description	Units
	for average year conditions		1 through 17	DO concentration for the respective segment #	mg/L
dodry.dbf	dBase File of modeling results of DO for dry year conditions	n/a	Time	Time	days
			1 through 17	DO concentration for the respective segment #	mg/L
dohgk2.dbf	dBase File of modeling results of DO for average year conditions, high k <sub>2</sub>	n/a	Time	Time	days
			1 through 17	DO concentration for the respective segment #	mg/L
dohgkd.dbf	dBase File of modeling results of DO for average year conditions, high k <sub>4</sub>	n/a	Time	Time	days
			1 through 17	DO concentration for the respective segment #	mg/L
flout.vat	vat of flow, flow accumulation value for each segment outlet	Flow Accumulation Values	value count grid-code flfac	Arc assigned cell value # of cells with this value Main segment # accumulated flow depth	n/a n/a n/a mm/yr/ha
flow.dbf	dBase File created during Input Block D execution in ArcView connection	n/a	grid-code cumm_flow int_flow runoff baseflow	Main segment # Cumulative calculated flow Incremental calculated flow Incremental calculated runoff Incremental calculated baseflow	n/a m <sup>3</sup> /sec m <sup>3</sup> /sec m <sup>3</sup> /sec m <sup>3</sup> /sec
gbayps.dat	INFO file of dischargers into Galveston Bay	n/a	Table not available for publication		
hsc_dis.dbf	dBase File of dischargers into Galveston Bay	n/a	Table not available for publication		
hsc_res.dbf	dBase File summarizing modeling results for Upper HSC	n/a	grid-code do_aveyr bod_aveyr do_drywtr bod_drywtr salinity	Main segment # Modeled DO for average year conditions Modeled BOD for average year conditions Modeled DO for dry weather conditions Modeled BOD for dry weather conditions Modeled salinity	n/a mg/L mg/L mg/L mg/L ppt
hsc_sal.dbf	dBase File of modeling results of calibration	n/a	Time 1 through 17	Time Salinity concentration for the respective segment #	days ppt
hyd_seg.dbf	dBase File summarizing measured values for hydrographic segmentation (Ward and Armstrong, 1992)		segment_id DO BOD salinity	Hydrographic segment # Average measured DO Average measured BOD Average measured salinity	n/a mg/L mg/L ppt
hydroseg.dbf	dBase File attached to hydroseg shape file	n/a	No additional attributes		
load.dbf	dBase File created during Input Block F execution in ArcView connection	n/a	grid-code bod_ps	Main segment # Total point source loading into main segment	n/a kg/day

Table	Feature	Alias*	Attribute	Description	Units
main_seg.dbf	dBase File which contains main segment modeling parameters	Main Segment Parameters	grid-code	Main segment #	n/a
			ex_coeff	Exchange coefficient	m <sup>2</sup> /sec
		x_area	Cross-sectional area	m <sup>2</sup>	
		upstr_seg	Upstream segment	n/a	
		dwnstr_seg	Downstream segment	n/a	
		btm_seg	Segment below this segment	n/a	
		int_bod	BOD initial concentration	mg/L	
		int_do	DO initial concentration	mg/L	
		sod	Sediment oxygen demand	g/m <sup>2</sup> -day	
		temp	Temperature	° C	
		sal	Salinity	ppt	
		width	Width of main segment	m	
		depth	Depth of segment	m	
type	Type of segment (WASP5 standards)	n/a			
ps_msrd.dbf	dBase File of the dischargers along the HSC with measurements -- attached to shape file	Same as uphscps.dbf	Table not available for publication		
r_out.vat	vat of runoff flow accumulation value for each segment outlet	Runoff Accumulation Values	value	Arc assigned cell value	n/a
			count	# of cells with this value	n/a
			grid-code	Main segment #	n/a
			rfac	accumulated runoff depth	mm/yr/ha
uphscps.dbf	dBase File of the dischargers along the HSC with measurments	Same as ps_msrd.dbf	Table not available for publication		

\* Alias refers to name needed for ArcView/WASP5 connection

## Appendix B

### Projection File: geoalb.prj

```
input
projection geographic
units dd
parameters
output
projection albers
units meters
spheroid grs80
datum NAD83
parameters
29 30 00
45 30 00
-96 00 00
23 00 00
0.0
0.0
end
```

**Appendix C**  
**Programs, Macros, And Procedures**

## Appendix C-1 Procedure and Macro for Importing Digital Line Graphs from CD ROM to ArcInfo

Source: Saunders, 1996

```

/* An ARC AML FOR PREPARING DLG DATA FOR REGIONAL ANALYSIS
/* AML NAME: dlghydro.aml (run from the "Arc" prompt)
/* FUNCTION: Prepares selected DLG data for analysis with respect to a
/*           particular hydrologic or political region.
/*
/* INPUTS:
/*   -all compressed ("zipped") DLG files corresponding to the region of
/*     interest. These zipped files are downloaded from the USGS EROS
/*     Data Center at http://sun1.cr.usgs.gov/eros-home.html.
/*     Alternatively the DLG files can be accessed from US Geodata
/*     1:100,000-Scale DLG Data Compact Disc (USGS, 1993).
/*   -a projection file that will allow for conversion from utm map
/*     coordinates to whatever projection is desired.
/*   -a polygon coverage delineating the boundary of the hydrologic or
/*     political region of interest.
/*
/* AUTHOR:
/*   William Saunders, Graduate Student, University of Texas at Austin,
/*   Environmental and Water Resources Division, Department of Civil
/*   Engineering, April 1996
/*
/* MODIFIED BY:
/*   Jennifer Benaman, Graduate Student, University of Texas at Austin,
/*   Environmental and Water Resources Division, Department of Civil
/*   Engineering, April 1996
/*
/* *****
/* BEGIN AML EXECUTION
/*
/* Assuming that 4 zipped DLG hydro files have been downloaded (in this case
/* from CD-ROM using the following commands:
/*
/* cp /cdrom/100k_dlg/beaumont/bm4hydro.zip ./
/* cp /cdrom/100k_dlg/conroe/bm3hydro.zip ./
/* cp /cdrom/100k_dlg/anahuac/ho2hydro.zip ./
/* cp /cdrom/100k_dlg/houston/holhydro.zip ./
/*
/* This first set of commands are the only one that the user of the AML must
/* change. Store the number of zipped DLG files into the variable dlgnum.
/* Then, for each zipped DLG file, define sequential variables called dlg# as
/* the first 3 characters of each of the zipped files.
/*
&sv dlgnum = 4
&sv dlg1 = ho1
&sv dlg2 = ho2
&sv dlg3 = bm3
&sv dlg4 = bm4

```

```

&do &while %count2% le 8
  &do &while [exists %filename%0%count2% -cover]
    %filename%0%count2%
    &sv count2 = %count2% + 1
  &end
&sv count2 = %count2% + 1
&end
&end
~
Y
Y
build bigmap line
/*
/* Once "bigmap" has been created, each of the coverages that were merged to/*
build it are no longer necessary. This part of the AML kills off all of
/* the intermediate level coverages used to append "bigmap".
/*
&sv count = 1
&do &while %count% le %dlgnum%
  &sv filename = [value dig%count%]
  &sv count = %count% + 1
  &sv count2 = 1
  &do &while %count2% le 8
    &do &while [exists %filename%0%count2% -cover]
      kill %filename%0%count2% all
      &sv count2 = %count2% + 1
    &end
  &sv count2 = %count2% + 1
&end
&end
&end
/*
/* The "bigmap" coverage is then reprojected to the desired map projection
/* and coordinates. The example below shows projection from Universal
/* Transverse Mercator coordinates (the projection of the files as they exist
/* on CDROM or from the Internet site) to the Texas State Mapping System.
/* Accordingly, this projection file must be located in the path specified
/* in the project statement. For this statement, the utmtsms.prj file is
/* located in the same directory as the "bigmap" coverage.
/*
project cover bigmap hydrocov utmtsms.prj
/*
/* Finally, a polygon coverage of the hydrologic or political boundary of
/* interest is used to "clip" out the hydrologic features specific to that
/* region.
/*
clip hydrocov border reghydro line
/*
&return
/*****end of AML*****/

```



## Appendix C-2

### Procedure to Obtain USGS Land Use/Land Cover Files from USEPA Ftp Site

Source: Maidment, 1996

#### Introduction

Land Use/Land Cover (LULC) files are developed by the USGS and are available via internet from either USGS or the EPA. In both cases, the user must specify the 1:250,000-scale mapsheet name(s) corresponding to the region(s) of interest, and then download the coverage(s). Accessing the actual LULC files from the USGS has proven problematic to date, as the files exist in a "modified" UTM projection that is not clearly defined in the users guide. Given these difficulties and the relative ease with which the same files are obtained from EPA, the EPA Home Page is currently the preferred source of this data. However, the 1:250,000-scale mapsheet names for the United States are still accessible at the USGS Home Page.

The following procedure is a THREE step process and may not sound very efficient. However, it is the only way we have found, so far, to successfully obtain the LULC files.

#### STEP 1 Accessing the 1:250,000-scale mapsheet names:

- Once at the initial Home Page of a particular data browser (Mosaic, Netscape, etc.), enter the address of the USGS EROS Data Center Home Page - <http://sun1.cr.usgs.gov/eros-home.html>.

- At the Home Page, scroll down to the 1:250,000-Scale and 1:100,000-Scale LULC section of the page and select 250K FTP via Graphics. A map of the continental United States will appear.

- Using the cursor, click in the general area of the particular region of interest, and a finer resolution map will appear with the 1:250,000-scale maps and mapnames superimposed.

- Note the mapnames that correspond to the hydrologic or political region of interest.

#### STEP 2 Obtaining the Filenames from EPA:

- An alternate source for LULC data is the EPA Home Page - <http://www.epa.gov>.

Retrieval of LULC files from this location still requires mapsheet names to be known ahead of time (see step 1 above).

- Once at this Home Page, select the Search the EPA Server hypertext. When you click on this hypertext, a query box will appear; within the query box, type *land use* and then press 'start query'. The result of the query should appear after a minute or two.

- Scroll down to and select the EPA EPAGIRAS hypertext and then enter a mapsheet name of a LULC coverage to be downloaded. This will lead to a page where an export file (.e00.gz) should be available for selection.

- write down the actual name of the e00 file -- it should be a series of three letters and 5 numbers, followed by .e00.gz.

#### STEP 3 Obtaining the file from the EPA ftp site

Since downloading the file through Netscape or Mosaic is not always successful (especially in the middle of the day). It is best to go to the EPA ftp site. Now that you know the actual filename of what you are looking for, this process is pretty straightforward.

- At a command prompt, type `ftp earth1.epa.gov`

- Logon anonymously by typing anonymous at the user name prompt and your e mail address for the password

- Change directories to `/pub/EPAGIRAS/mgiras` (note that ftp is case sensitive!)

- Once in this directory, you can type `get filename`. The download process will take some time. Typically, these files are between 1.5 and 2 megs. When you are done getting the files you need, exit out of the ftp prompt by typing `bye`. - At the Unix prompt of the host workstation, unzip the compressed file:

**\$: gunzip filename.e00.gz**

Invoke Arc/Info and import the file as an Arc/Info coverage:

**Arc: import cover filename.e00 filename**

The process is now complete as the LULC coverage exists in an Albers projection that can easily be manipulated with other coverages and LULC files.

## Appendix C-3

### FORTRAN Program Used to Estimate Baseflow from Measured Flow

Source: Olivera, *pers. comm.*, 1996

```

*****
* Author: Francisco Olivera
* CRWR -- University of Texas at Austin
* Date: February 5, 1996
* Revised: Jennifer Benaman
* CRWR -- University of Texas at Austin
* Date: February 16, 1996
* Purpose: Given a flow time series, the program generates
* base flow and direct runoff time series.
* Given the plot of the flow time series, the base flow is
* obtained by pivoting a straight line on a point and
* connecting it with the lowest part of the flow curve. The
* length of the straight line is defined by the user.
*
* Input: (1) Length of the straight line (in time steps)
* (2) Flow time series
*****
PROGRAM BFLOW3

* Declaring Parameters
PARAMETER (NMAX=10000)

* Declaring variables
REAL FLOW(NMAX), BASEFLOW(NMAX), MINSLOPE, SLOPE(NMAX)
INTEGER L,N

* Open input files for reading
OPEN(UNIT=10, FILE='FLOW.IN', STATUS='OLD')

* Open output file for writing
OPEN(UNIT=30, FILE='BASEFLOW.OUT', STATUS='UNKNOWN')

* Reading input file
READ(10,*) L
DO 101 I=1,NMAX
  READ(10,*,END=901) FLOW(I)

          N=I
101 CONTINUE
901 CONTINUE

* Generating the baseflow

  BASEFLOW(1)=FLOW(1)
  I=1
501 CONTINUE
  MINSLOPE=1000000

  DO 102 J=1,L
    IF (I+J .LE. N) THEN
      SLOPE(J) = (FLOW(I+J)-BASEFLOW(I))/J
      IF (SLOPE(J) .LT. MINSLOPE) THEN
        MINSLOPE=SLOPE(J)
      END IF
    END IF
  102 CONTINUE

  BASEFLOW(I+1)=BASEFLOW(I)+MINSLOPE

  I=I+1
  IF(I-N) 501,502,502
502 CONTINUE

* Echo the output to the screen
  DO 104 I=1,N
    WRITE (*,*) I, FLOW(I), BASEFLOW(I)
  104 CONTINUE

* Writing the Flow and the Baseflow
  DO 105 I=1,N
    WRITE(30,*) I, FLOW(I), BASEFLOW(I)
  105 CONTINUE

END

```

## Appendix C-4

### FORTRAN Programs Used to Format Text Files for WASP5 Input File

#### Program: calgen.for Generates calibration input file for TOXIS

```

*****
*      Program:  Used to take ten text files written from
*                ArcView and write the WASP5 input file.
*                Presently, this program only writes the file
*                for TOXIS -- model calibration w/salinity.
*      Input:    Twelve text files
*      Author:   Jennifer Benaman
*                Graduate Research Assistant
*      Date:     June, 1996
*****

* Main Program
  PROGRAM INPUTCAL

* Declare Variables

  REAL A(20), B(100,5), C(100,4),D(300,2),E(100,2),F(100,2),
&  G(100,4),H(10),I(10),J(300,2),TPRINT
  INTEGER NOSEG,MNSEG
  CHARACTER*12 FNAME

* Explanation of Variables

*      A(20): the array which holds the input file A
*             variables: model paramters
*      B(100,5): the array which holds the input file
*               B variables: exchange functions
*      C(100,4): the array which holds the input file
*               C variables: volumes
*      D(300,2): the array which holds the input file
*               D variables: flows
*      E(100,2): the array which holds the input file
*               E variables: boundaries
*      F(100,4): the array which holds the input file
*               F variables: loads
*      G(100,4): the array which holds the input file
*               G variables: parameters
*      H(10):    the array which holds the input file
*               H variables: constants
*      I(10):    the array which holds the input file
*               I variables: times functions
*      J(300,2): the array which holds the input file

*      G variables: initial conditions
*      TPRINT: the number of days the model is to be
*              run -- obtained in subroutine INPUTA
*      NOSEG:  the number of segments in the system -- obtained
*              in Subroutine INPUTA

* Open Files

  OPEN (UNIT=6, FILE='INPTIME.TXT',STATUS='OLD')
  READ (6,16) FNAME
  16 FORMAT (A12)

  OPEN (UNIT=3, FILE='TITLE.TXT',STATUS='OLD')
  OPEN (UNIT=5, FILE='CALA.TXT',STATUS='OLD')
  OPEN (UNIT=10, FILE='B.TXT',STATUS='OLD')
  OPEN (UNIT=15, FILE='C.TXT',STATUS='OLD')
  OPEN (UNIT=13, FILE='DYN.TXT',STATUS='OLD')
  OPEN (UNIT=20, FILE='D.TXT',STATUS='OLD')
  OPEN (UNIT=25, FILE='CALE.TXT',STATUS='OLD')
  OPEN (UNIT=30, FILE='CALF.TXT',STATUS='OLD')
  OPEN (UNIT=35, FILE='CALG.TXT',STATUS='OLD')
  OPEN (UNIT=40, FILE='CALH.TXT',STATUS='OLD')
  OPEN (UNIT=45, FILE='I.TXT',STATUS='OLD')
  OPEN (UNIT=50, FILE='CALJ.TXT',STATUS='OLD')
  OPEN (UNIT=55, FILE=FNAME,STATUS='UNKNOWN')

* Call Subroutines

  CALL INPUTA(A,TPRINT,NOSEG,MNSEG)
  CALL INPUTB(B,TPRINT,NOSEG)
  CALL INPUTC(C,NOSEG)
  CALL INPUTD(D,TPRINT)
  CALL INPUTE(E,TPRINT)
  CALL INPUTF(F,TPRINT,MNSEG)
  CALL INPUTG(G,NOSEG)
  CALL INPUTH(H)
  CALL INPUTI(I)
  CALL INPUTJ(J,NOSEG)

  STOP
  END

***** Subroutines

```

```

***Subroutine INPUTA: Creates the A Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****Backward differencing is always used (ADFAC=0.0)
*****A transport file is always generated (TFLG = 0)
*****The first six segments are those which solns are displayed
*   on the screen (Record 5)
*****The same maximum time step is used throughout the model-run
*****The same print interval is used throughout the model-run
***This subroutine also reads the time the model will run (TPRINT)

```

```

SUBROUTINE INPUTA(A,TPRINT,NOSEG,MNSEG)

```

```

INTEGER NOSEG, NOSYS, ICFL, MFLAG, JMASS, NEGSLN, INTYP,
& TPLG,ZMIN, ZHR,
& MNSEG
REAL A(11), ADFAC, ZDAY, DTS, TPRINT, PRTINV
CHARACTER*80 TITLE
CHARACTER*75 LEVEL

```

```

READ (3,1) TITLE
READ (5,*) A

```

```

* Declaring Variables

```

```

NOSEG = INT(A(1))
NOSYS = 1
ICFL = INT(A(2))
MFLAG = INT(A(3))
JMASS = 1
NEGSLN = INT(A(4))
ZDAY = A(5)
ZHR = INT(A(6))
ZMIN = INT(A(7))
INTYP = 1
TFLG = 0
ADFAC = 0.0
DTS = A(8)
PRTINV = A(10)
TPRINT = A(9)
MNSEG = INT(A(11))
LEVEL = 'Calibration with Salinity'

```

```

* Format statements

```

```

1  FORMAT (A80)
2  FORMAT (A5,1X,A)
3  FORMAT (11A5,A25)
4  FORMAT (7I5,2F5.0,I3,I2,I5)
5  FORMAT (8I5)
6  FORMAT (I5)
7  FORMAT (2(F10.1, F10.1))
9  FORMAT (F10.5,F10.1)

```

```

* Writing the input file

```

```

WRITE (55,1) TITLE
WRITE (55,2) "TOXI5 ",LEVEL
WRITE (55,3) "NSEG", "NSYS", "ICRD", "MFLG", "IDMP", "NSLN",
& "INTY", "ADFC", "DD", "HHMM", "TFLG", "A: MODEL OPTIONS"
WRITE (55,4) NOSEG, NOSYS, ICFL, MFLAG, JMASS, NEGSLN,
& INTYP, ADFAC, ZDAY, ZHR, ZMIN, TPLG
WRITE (55,5) 1,2,3,4,5,6
WRITE (55,6) 1
WRITE (55,9) DTS,TPRINT
WRITE (55,6) 2
WRITE (55,7) PRTINV,0.,PRTINV,TPRINT
WRITE (55,5) 0,1,1,1,1,1

RETURN
END

```

```

***Subroutine INPUTB: Creates the B Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****All exchanges are steady state
*****The scaling and conversion factors are set to 1.0 (Areas should
*   be in m^2 and exchange coeff. in m^2/sec)

```

```

SUBROUTINE INPUTB(B,TPRINT,NOSEG)

```

```

INTEGER NRFLD,NTEX,NORS,
+ TEMP,S, N
REAL B(100,5),TPRINT

COUNT = 1
DO WHILE (TEMP2 .NE. 555)
  READ (10,*) B(COUNT,1)
  TEMP2 = B(COUNT,1)
  COUNT = COUNT + 1
END DO

REWIND (UNIT = 10)
N = COUNT-2
DO I=1,N
  READ (10,*) (B(I,J),J=1,5)
END DO

NRFLD = INT(B(1,2))

WRITE (55,10) NRFLD,"B:EXCHANGES"

```

```

10 FORMAT (I5,5X,A)
11 FORMAT (I5,2F10.1)
12 FORMAT (I5)
13 FORMAT (2F10.1,2I5)
14 FORMAT (4(F10.1,F10.1))
15 FORMAT (8I5)

```

```

NTEX = 1
TEMP = 2
DO I = TEMP, N-1
  IF (B(I,1).EQ.B(I+1,1)) THEN
    GO TO 500
  ELSE
    NTEX = NTEX + 1
  END IF
500 END DO

WRITE (55,11) NTEX,1.0,1.0

DO S = 1,NTEX

  NORS = 1
  DO I = TEMP,N-1
    IF (B(I,1) .EQ. B(I+1,1)) THEN
      NORS = NORS + 1
    ELSE
      GO TO 510
    END IF
  END DO

510  WRITE (55,12) NORS

      DO I = TEMP,TEMP+NORS-1
        WRITE (55,13) B(I,2),B(I,3),INT(B(I,4)),INT(B(I,5))
      END DO
    WRITE (55,12) 2
    WRITE (55,14) B(TEMP,1),0.0,B(TEMP,1),TPRINT
    TEMP = TEMP + NORS

  END DO

  WRITE (55,15) 0,1,1,1,1,1,1,1
  RETURN
END

***Subroutine INPUTC:  Creates the C Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****The scaling and conversion factors are set to 1.0 (Volumes should
*   be in m^3)
*****The hydraulic coefficients used to calculate reaeration and
*   volitilization do not spatially vary

SUBROUTINE INPUTC(C,NOSEG)

  INTEGER NOSEG, N
  REAL C((NOSEG+5),4)

  N = NOSEG + 5

  DO I=1,N
    READ (15,*) (C(I,J),J=1,4)

```

```

  END DO

20  FORMAT (2I5,F10.4,5X,A)
21  FORMAT (2F10.1)
22  FORMAT (3I10,F10.1,F10.4,3F10.1)

  WRITE (55,20) INT(C(1,1)),INT(C(1,2)),C(1,3),"C·VOLUMES"
  WRITE (55,21) 1.0,1.0

  DO I = 6,N
    WRITE (55,22) INT(C(I,1)),INT(C(I,2)),INT(C(I,3)),C(I,4),
    & C(2,1),C(3,1),C(4,1),C(5,1)
  END DO

  RETURN
END

***Subroutine INPUTD:  Creates the D Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****The scaling and conversion factors are set to 1.0 (Flows should
*   be in m^3/sec)
*****The number of flow fields is set to 1:  Water column only (no pore
*   water flows
*****The flow is steady state

SUBROUTINE INPUTD(D,TPRINT)

  INTEGER I,IQOPT,NFIELD,K,J,
  & W,COUNT,X,FLW,H
  REAL D(300,2),TPRINT,TEMP
  CHARACTER*12 HYDFILE

  NFIELD = 1
  FLW = 0
  N = 2
  TEMP = 0.0
  DO WHILE (TEMP.NE.555.0)
    READ (20,*) (D(N,I),I=1,2)
    IF (D(N,2).EQ.999.0) THEN
      FLW = FLW+1
      PRINT *, FLW
    END IF
    TEMP = D(N,1)
    N = N+1
  END DO
  REWIND (UNIT = 20)

  DO I = 1,1
    READ(20,*) (D(I,J),J=1,2)
  END DO

  READ (13,35) HYDFILE

```

```

      IQOPT = INT(D(1,1))

      IF (IQOPT .EQ. 3) THEN
        WRITE (55,30) IQOPT,NFIELD,HYDFILE,"D: FLOWS"
        RETURN
      ELSE
        WRITE (55,30) IQOPT,NFIELD,HYDFILE,"D: FLOWS"
        WRITE (55,31) FLW, 1.0, 1.0
      END IF

      Z = 1
      X = 2
      TEMP = 0.0
      W = 1
      DO WHILE (TEMP.NE.555.0)
        COUNT = 0
        DO I = X,300
          READ(20,*) (D(I,J),J=1,2)
          IF (D(I,2).EQ.999) THEN
            GO TO 800
          ELSE IF (D(I,1).EQ.555.0) THEN
            GO TO 810
          ELSE
            COUNT = COUNT + 1
            W = W+1
          END IF
        END IF
      END DO

800 WRITE (55,32) COUNT

      DUMMY1 = INT(COUNT/4)
      DUMMY2 = COUNT - DUMMY1*4

      IF (DUMMY1.EQ.0) THEN
        DUMMY3 = X
      END IF

      DO K = X,(4*DUMMY1)+X-1,4
        WRITE (55,33) 1.0,INT(D(K,1)),INT(D(K,2)),
          & 1.0,INT(D(K+1,1)),INT(D(K+1,2)),
          & 1.0,INT(D(K+2,1)),INT(D(K+2,2)),
          & 1.0,INT(D(K+3,1)),INT(D(K+3,2))
        DUMMY3 = K+4
      END DO

      IF (DUMMY2.NE.0) THEN
        WRITE (55,33) (1.0,INT(D(H,1)),INT(D(H,2)),H=DUMMY3,DUMMY2+
          & DUMMY3-1)
      END IF

      WRITE (55,32) 2
      WRITE (55,34) D(W+Z,1),0.0,D(W+Z,1),TPRINT

      X = COUNT+X+1
      Z = Z+1
810 TEMP = D(I,1)
      END DO

      30 FORMAT (2I5,A12,5X,A)
      31 FORMAT (I5,2F10.1)
      32 FORMAT (I5)
      33 FORMAT (4(F10.2,2I5))
      34 FORMAT (4F10.2)
      35 FORMAT (A12)
      36 FORMAT (8I5)

      WRITE (55,36) 0,1,1,1,1,1,1,1,1

      RETURN
      END

      ***Subroutine INPUTE: Creates the E Block to the WASP5 input file
      ***The following defaults are set in this subroutine:
      *****The scaling and conversion factors are set to 1000 (Boundary conditions
      * should be in ppt)
      *****Only Salinity is considered
      *****The bc's are steady state

      SUBROUTINE INPUTE(E,TPRINT)

      INTEGER NOBC,I,
      & N
      REAL E(100,2),TPRINT

      DO I = 1,1
        READ(25,*) (E(I,J),J=1,2)
      END DO

      NOBC = INT(E(1,1))
      N = NOBC + 1

      IF (NOBC .EQ. 0) THEN
        WRITE (55,40) 0,"*", "E:BOUNDARIES"
        RETURN
      END IF

      DO I = 2,N
        READ(25,*) (E(I,J),J=1,2)
      END DO

      40 FORMAT (I10,2X,A5,5X,A)
      42 FORMAT (2I5)
      43 FORMAT (2(2F10.2))

```

```

WRITE (55,40) NOBC,"SAL","E:BOUNDARIES"
WRITE (55,43) 1000.0,1.0
DO I = 2,N
  WRITE (55,42) INT(E(I,1)),2
  WRITE (55,43) E(I,2),0.0,E(I,2),TPRINT
END DO

RETURN
END

***Subroutine INPUTF: Creates the F Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****There are no point or nonpoint source salinity loadings

SUBROUTINE INPUTF(F,TPRINT,MNSEG)

INTEGER N,
& MNSEG
REAL F(25,2),TPRINT

READ (30,*) F(1,1)

N = INT(F(1,1))

IF (N .EQ. 0) THEN
  WRITE (55,60) N,"F:LOADS -- NO LOADS"
  WRITE (55,60) 0,"NO NPS LOADS"
  RETURN
END IF

60 FORMAT (I10,5X,A)

RETURN
END

***Subroutine INPUTG: Creates the G Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****There are no paramters for TOXIS complexity level 1

SUBROUTINE INPUTG(G,NOSEG)

INTEGER NOSEG,NOPAR
REAL G((NOSEG+1),4)

READ(35,*) G(1,1)

NOPAR = INT(G(1,1))

50 FORMAT (I10,5X,A)

WRITE (55,50) NOPAR,"G:PARAMETERS -- NO PARAMETERS"

RETURN
END

***Subroutine INPUTH: Creates the H Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****The constants do not spatially vary
*****Partition coefficient in L/kg
*****Water column biodegradation
*****Molecular Weight in g/mole

SUBROUTINE INPUTH(H)

REAL H(3),PIXC,KBW,MOLWT

READ(40,*) H

PIXC = H(1)
KBW = H(2)
MOLWT = H(3)

70 FORMAT (10X,A)
71 FORMAT (A10,I10)
72 FORMAT (2(A10,I10,F10.2))
73 FORMAT (A10,I10,F10.2)

WRITE (55,70) "H:CONSTANTS"
WRITE (55,71) "GLOBALS",0
WRITE (55,71) "SALINITY",1
WRITE (55,72) "GENERAL",3
WRITE (55,72) "PIXC",111,PIXC,"KBW",141,KBW
WRITE (55,73) "MOLWT",81,MOLWT

RETURN
END

***Subroutine INPUTI: Creates the I Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****Since this model is considered steady state (right now), there
* are no time functions presently input into the model

SUBROUTINE INPUTI(I)

REAL I(1)

READ (45,*) I

WRITE (55,80) INT(I(1)),"I:TIME FUNCTIONS"
80 FORMAT (I10,20X,A)

RETURN

```

```

END

***Subroutine INPUTJ: Creates the J Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****This program sets all i.c.'s to 0, except for those boundary
*   conditions set in ArcView
*****The dissolved fraction of salinity is 1.0
*****The maximum value of all variables is set at 35000 mg/L
*****The density of salinity is set at 0.0 -- EUTRO does not
*   use those numbers
*****All initial conditions are in mg/L

```

```

SUBROUTINE INPUTJ(J,NOSEG)

INTEGER NOSEG,K,NOBC,N,L
REAL J(NOSEG*8+8,2),DENBOD,DENDO,MAX,
& FRAC,TEMP(30,2)

DO K = 1,2
  READ (50,*) (J(K,L),L=1,2)
END DO

FRAC = J(1,1)
NOBC = INT(J(2,1))
DO K = 3,2+NOBC
  READ (50,*) (J(K,L),L=1,2)
END DO

N = 3
DO K = 1,NOSEG
  IF (K.EQ.INT(J(N,1))) THEN
    TEMP(K,1) = J(N,1)

```

```

TEMP(K,2) = J(N,2)*1000
N = N+1
ELSE
TEMP(K,1) = REAL(K)
TEMP(K,2) = 0.0
END IF

END DO

MAX = 35000.0
DENBOD = 1.0
DENDO = 1.0

90 FORMAT (A10,30X,I5,F5.1,E10.2,A)
91 FORMAT (3(I5,2F10.2))

DUMMY1 = INT(NOSEG/3)
DUMMY2= NOSEG - 3*DUMMY1

WRITE (55,90) "SAL",3,0.0,MAX,"J:INITIAL CONDITIONS"
DO K = 1,3*DUMMY1,3
  WRITE (55,91) INT(TEMP(K,1)),TEMP(K,2),FRAC,
& INT(TEMP(K+1,1)),TEMP(K+1,2),FRAC,
& INT(TEMP(K+2,1)),TEMP(K+1,2),FRAC
DUMMY3 = K+3
END DO
WRITE (55,91) (INT(TEMP(K,1)),TEMP(K,2),FRAC,K=DUMMY3,DUMMY2
& +DUMMY3-1)

RETURN
END

```

Program: outgen.for  
Generates BOD/DO model input file for EUTRO5

```

*****
*   Program: Used to take ten text files written from
*             ArcView and write the WASP5 input file.
*             Presently, this program only writes the file
*             for EUTRO5.
*   Input:   Ten text files
*   Author:   Jennifer Benaman

```

```

*   Graduate Research Assistant
*   Date:   June, 1996
*****
*   Main Program
*   PROGRAM INPUTWASP

```



```

* Declare Variables

REAL A(20), B(100,5), C(100,4),D(300,2),E(100,2),F(100,2),
& G(100,4),H(10),I(10),J(300,2),TPRINT
INTEGER NOSEG,MNSEG
CHARACTER*12 FNAME

```

\* Explanation of Variables

```

* A(20): the array which holds the input file A
*       variables: model parameters
* B(100,5): the array which holds the input file
*          B variables: exchange functions
* C(100,4): the array which holds the input file
*          C variables: volumes
* D(300,2): the array which holds the input file
*          D variables: flows
* E(100,2): the array which holds the input file
*          E variables: boundaries
* F(100,4): the array which holds the input file
*          F variables: loads
* G(100,4): the array which holds the input file
*          G variables: parameters
* H(10): the array which holds the input file
*        H variables: constants
* I(10): the array which holds the input file
*        I variables: times functions
* J(300,2): the array which holds the input file
*          G variables: initial conditions
* TPRINT: the number of days the model is to be
*         run -- obtained in subroutine INPUTA
* NOSEG: the number of segments in the system -- obtained
*        in Subroutine INPUTA

```

\* Open Files

```

OPEN (UNIT=6, FILE='INPTNME.TXT',STATUS='OLD')

READ (6,16) FNAME
16 FORMAT (A12)

OPEN (UNIT=3, FILE='TITLE.TXT',STATUS='OLD')
OPEN (UNIT=5, FILE='A.TXT',STATUS='OLD')
OPEN (UNIT=10, FILE='B.TXT',STATUS='OLD')
OPEN (UNIT=15, FILE='C.TXT',STATUS='OLD')
OPEN (UNIT=13, FILE='DYN.TXT',STATUS='OLD')
OPEN (UNIT=20, FILE='D.TXT',STATUS='OLD')
OPEN (UNIT=25, FILE='E.TXT',STATUS='OLD')
OPEN (UNIT=30, FILE='F.TXT',STATUS='OLD')
OPEN (UNIT=35, FILE='G.TXT',STATUS='OLD')
OPEN (UNIT=40, FILE='H.TXT',STATUS='OLD')
OPEN (UNIT=45, FILE='I.TXT',STATUS='OLD')
OPEN (UNIT=50, FILE='J.TXT',STATUS='OLD')
OPEN (UNIT=55, FILE=FNAME,STATUS='UNKNOWN')

```

\* Call Subroutines

```

CALL INPUTA(A,TPRINT,NOSEG,MNSEG)
CALL INPUTB(B,TPRINT,NOSEG,MODEL)
CALL INPUTC(C,NOSEG)
CALL INPUTD(D,TPRINT,MODEL)
CALL INPUTE(E,TPRINT,MODEL)
CALL INPUTF(F,MODEL,TPRINT,MNSEG)
CALL INPUTG(G,NOSEG)
CALL INPUTH(H,MODEL)
CALL INPUTI(I)
CALL INPUTJ(J,NOSEG)

```

```

STOP
END

```

\*\*\*\*\* Subroutines

```

***Subroutine INPUTA: Creates the A Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****Backward differencing is always used (ADFAC=0.0)
*****A transport file is always generated (TFLG = 0)
*****The first six segments are those which solns are displayed
*       on the screen (Record 5)
*****The same maximum time step is used throughout the model-run
*****The same print interval is used throughout the model-run
***This subroutine also reads the time the model will run (TPRINT)

```

```

SUBROUTINE INPUTA(A,TPRINT,NOSEG,MNSEG)

```

```

INTEGER NOSEG, NOSYS, ICFL, MFLAG, JMASS, NEGSLN, INTYP,
& MODEL, TFLG, NH, NO, PO, CHL, CBOD, DO, ON, OP, ZMIN, ZHR,
& MNSEG
REAL A(14), ADFAC, ZDAY, DTS, TPRINT, PRTINV
CHARACTER*80 TITLE
CHARACTER*75 LEVEL

READ (3,1) TITLE
READ (5,*) A

```

\* Declaring Variables

```

MODEL = INT(A(1))
NOSEG = INT(A(2))
NOSYS = INT(A(3))
ICFL = INT(A(4))
MFLAG = INT(A(5))
JMASS = INT(A(6))+1
NEGSLN = INT(A(7))
ZDAY = A(8)
ZHR = INT(A(9))
ZMIN = INT(A(10))

```

```

INTYP = 1
TFLG = 0
ADFAC = 0.0
DTS = A(11)
PRTINV = A(13)
TPRINT = A(12)
MNSEG = INT(A(14))

IF (MODEL .EQ. 0) THEN
  LEVEL = 'Simple Streeter-Phelps with SOD'
  NH = 1
  NO = 1
  PO = 1
  CHL = 1
  CBOD = 0
  DO = 0
  ON = 1
  OP = 1
END IF

* Format statements

1  FORMAT (A80)
2  FORMAT (A5,1X,A)
3  FORMAT (11A5,A25)
4  FORMAT (7I5,2F5.0,I3,I2,I5)
5  FORMAT (8I5)
6  FORMAT (I5)
7  FORMAT (2(F10.1, F10.1))
9  FORMAT (F10.5,F10.1)

* Writing the input file

WRITE (55,1) TITLE
WRITE (55,2) "EUTRO ",LEVEL
WRITE (55,3) "NSEG","NSYS","ICRD","MFLG","IDMP","NSLN",
& "INTY","ADFC","DD","HHMM","TFLG","A: MODEL OPTIONS"
WRITE (55,4) NOSEG, NOSYS, ICFL, MFLAG, JMASS, NEGSLN,
& INTYP, ADFAC, ZDAY, ZHR, ZMIN, TFLG
WRITE (55,5) 1,2,3,4,5,6
WRITE (55,6) 1
WRITE (55,9) DTS,TPRINT
WRITE (55,6) 2
WRITE (55,7) PRTINV,0.,PRTINV,TPRINT
WRITE (55,5) NH,NO,PO,CHL,CBOD,DO,ON,OP

RETURN
END

***Subroutine INPUTB: Creates the B Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****All exchanges are steady state
*****The scaling and conversion factors are set to 1.0 (Areas should
*   be in m^2 and exchange coeff. in m^2/sec)

```

```

SUBROUTINE INPUTB(B,TPRINT,NOSEG,MODEL)

INTEGER MODEL,NRFLD,NTEX,NH,NO,PO,CHL,CBOD,DO,ON,OP,NORS,
+ TEMP,S, N, COUNT
REAL B(100,5),TPRINT

COUNT = 1
DO WHILE (TEMP2 .NE. 555)
  READ (10,*) B(COUNT,1)
  TEMP2 = B(COUNT,1)
  COUNT = COUNT + 1
END DO

REWIND (UNIT = 10)
N = COUNT-2
DO I=1,N
  READ (10,*) (B(I,J),J=1,5)
END DO

MODEL = INT(B(1,1))
NRFLD = INT(B(1,2))

IF (MODEL.EQ.0) THEN
  NH = 1
  NO = 1
  PO = 1
  CHL = 1
  CBOD = 0
  DO = 0
  ON = 1
  OP = 1
END IF

WRITE (55,10) NRFLD,"B:EXCHANGES"

10 FORMAT (I5,5X,A)
11 FORMAT (I5,2F10.1)
12 FORMAT (I5)
13 FORMAT (2F10.1,2I5)
14 FORMAT (4(F10.3,F10.1))
15 FORMAT (8I5)

NTEX = 1
TEMP = 2
DO I = TEMP, N-1
  IF (B(I,1).EQ.B(I+1,1)) THEN
    GO TO 500
  ELSE
    NTEX = NTEX + 1
  END IF
500 END DO

```

```

WRITE (55,11) NTEX,1.0,1.0
DO S = 1,NTEX
  NORS = 1
  DO I = TEMP,N-1
    IF (B(I,1) .EQ. B(I+1,1)) THEN
      NORS = NORS + 1
    ELSE
      GO TO 510
    END IF
  END DO
510 WRITE (55,12) NORS
  DO I = TEMP,TEMP+NORS-1
    WRITE (55,13) B(I,2),B(I,3),INT(B(I,4)),INT(B(I,5))
  END DO
  WRITE (55,12) 2
  WRITE (55,14) B(TEMP,1),0.0,B(TEMP,1),TPRINT
  TEMP = TEMP + NORS
END DO
WRITE (55,15) NH,NO,PO,CHL,CBOD,DO,ON,OP
RETURN
END
***Subroutine INPUTC: Creates the C Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****The scaling and conversion factors are set to 1.0 (Volumes should
* be in m^3)
*****The hydraulic coefficients used to calculate reaeration and
* volitilization do not spatially vary
SUBROUTINE INPUTC(C,NOSEG)
  INTEGER NOSEG, N
  REAL C((NOSEG+5),4)
  N = NOSEG + 5
  DO I=1,N
    READ (15,*) (C(I,J),J=1,4)
  END DO
20 FORMAT (2I5,F10.4,5X,A)
21 FORMAT (2F10.1)
22 FORMAT (3I10,F10.1,F10.4,3F10.1)
  WRITE (55,20) INT(C(1,1)),INT(C(1,2)),C(1,3),"C:VOLUMES"
  WRITE (55,21) 1.0,1.0
  DO I = 6,N

```

```

WRITE (55,22) INT(C(I,1)),INT(C(I,2)),INT(C(I,3)),C(I,4),
& C(2,1),C(3,1),C(4,1),C(5,1)
END DO
RETURN
END
***Subroutine INPUTD: Creates the D Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****The scaling and conversion factors are set to 1.0 (Flows should
* be in m^3/sec)
*****The number of flow fields is set to 1: Water column only (no pore
* water flows)
*****The flow is steady state
SUBROUTINE INPUTD(D,TPRINT,MODEL)
  INTEGER I,IQOPT,NFIELD,K,J,MODEL,H,FLW,
& NH,NO,PO,CHL,CBOD,DO,ON,OP,W,COUNT,X
  REAL D(300,2),TPRINT,TEMP
  CHARACTER*12 HYDFILE
  IF (MODEL.EQ.0) THEN
    NH = 1
    NO = 1
    PO = 1
    CHL = 1
    CBOD = 0
    DO = 0
    ON = 1
    OP = 1
  END IF
  NFIELD = 1
  FLW = 0
  N = 2
  TEMP = 0.0
  DO WHILE (TEMP.NE.555.0)
    READ (20,*) (D(N,I),I=1,2)
    IF (D(N,2).EQ.999.0) THEN
      FLW = FLW+1
      PRINT *, FLW
    END IF
    TEMP = D(N,1)
    N = N+1
  END DO
  REWIND (UNIT = 20)
  DO I = 1,1
    READ(20,*) (D(I,J),J=1,2)
  END DO
  READ (13,35) HYDFILE

```

```

      WRITE (55,34) D(W+Z,1),0.0,D(W+Z,1),TPRINT

      X = COUNT+X+1
      Z = Z+1
810  TEMP = D(I,1)
      END DO

30  FORMAT (2I5,A12,5X,A)
31  FORMAT (I5,2F10.1)
32  FORMAT (I5)
33  FORMAT (4(F10.2,2I5))
34  FORMAT (4F10.2)
35  FORMAT (A12)
36  FORMAT (8I5)

      WRITE (55,36) NH,NO,PO,CHL,CBOD,DO,ON,OP

      RETURN
      END

***Subroutine INPUTE:  Creates the E Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****The scaling and conversion factors are set to 1.0 (Boundary conditions
*   should be in mg/L)
*****Only BOD and DO are considered
*****The bc's are steady state

      SUBROUTINE INPUTE(E,TPRINT,MODEL)

      INTEGER  NOBC,I,MODEL,NH,NO,PO,CHL,CBOD,DO,ON,OP,
& N
      REAL  E(100,2),TPRINT

      DO I = 1,1
          READ(25,*) (E(I,J),J=1,2)
      END DO

      NOBC = INT(E(1,1))
      N = 2*NOBC + 1

      IF (NOBC .EQ. 0) THEN
          WRITE (55,40) 0,"**","E:BOUNDARIES"
          RETURN
      END IF

      DO I = 2,N
          READ(25,*) (E(I,J),J=1,2)
      END DO

40  FORMAT (I10,2X,A5,5X,A)
41  FORMAT (I10,2X,A)
42  FORMAT (2I5)

      IQOPT = INT(D(1,1))

      IF (IQOPT .EQ. 3) THEN
          WRITE (55,30) IQOPT,NFIELD,HYDFILE,"D:FLWS"
          RETURN
      ELSE
          WRITE (55,30) IQOPT,NFIELD,HYDFILE,"D:FLWS"
          WRITE (55,31) FLW, 1.0, 1.0
      END IF

      Z = 1
      X = 2
      TEMP = 0.0
      W = 1
      DO WHILE (TEMP.NE.555.0)
          COUNT = 0
          DO I = X,300
              READ(20,*) (D(I,J),J=1,2)
              IF (D(I,2).EQ.999) THEN
                  GO TO 800
              ELSE IF (D(I,1).EQ.555.0) THEN
                  GO TO 810
              ELSE
                  COUNT = COUNT + 1
                  W = W+1
              END IF
          END DO
      END DO

800  WRITE (55,32) COUNT

      DUMMY1 = INT(COUNT/4)
      DUMMY2 = COUNT - DUMMY1*4

      IF (DUMMY1.EQ.0) THEN
          DUMMY3 = X
      END IF

      DO K = X,(4*DUMMY1)+X-1,4
          WRITE (55,33) 1.0,INT(D(K,1)),INT(D(K,2)),
& 1.0,INT(D(K+1,1)),INT(D(K+1,2)),
& 1.0,INT(D(K+2,1)),INT(D(K+2,2)),
& 1.0,INT(D(K+3,1)),INT(D(K+3,2))
          DUMMY3 = K+4
      END DO

      IF (DUMMY2.NE.0) THEN
          WRITE (55,33)(1.0,INT(D(H,1)),INT(D(H,2)),H=DUMMY3,DUMMY2+
& DUMMY3-1)
      END IF

      WRITE (55,32) 2

```

```

43 FORMAT (2(2F10.2))

IF (MODEL.EQ.0) THEN
  NH = 0
  NO = 0
  PO = 0
  CHL = 0
  CBOD = NOBC
  DO = NOBC
  ON = 0
  OP = 0
END IF

WRITE (55,40) NH,"NH3","E:BOUNDARIES"
WRITE (55,41) NO,"NO3"
WRITE (55,41) PO,"PO4"
WRITE (55,41) CHL,"CHLa"
WRITE (55,41) CBOD,"CBOD"
WRITE (55,43) 1.0,1.0
DO I = 2,NOBC+1
  WRITE (55,42) INT(E(I,1)),2
  WRITE (55,43) E(I,2),0.0,E(I,2),TPRINT
END DO
WRITE (55,41) DO,"DO"
WRITE (55,43) 1.0,1.0
DO I = NOBC+2,2*NOBC+1
  WRITE (55,42) INT(E(I,1)),2
  WRITE (55,43) E(I,2),0.0,E(I,2),TPRINT
END DO
WRITE (55,41) ON,"ON"
WRITE (55,41) OP,"OP"

RETURN
END

***Subroutine INPUTF: Creates the F Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****All point sources are steady state
*****At the present time, all non-point sources are also steady state
*****Loads are in kg/day

```

```

SUBROUTINE INPUTF(F,MODEL,TPRINT,MNSEG)

```

```

INTEGER MODEL,NH,NO,PO,CHL,CBOD,DO,ON,OP,N,I,J,X,W,
& MNSEG
REAL F(25,2),TPRINT,TEMP(26,2),LOAD,NPS

```

```

G = 2*MNSEG+1
DO I = 1,G
  READ (30,*) (F(I,J),J=1,2)
END DO

```

```

N = INT(F(1,1))

```

```

IF (MODEL.EQ.0) THEN
  NH = 0
  NO = 0
  PO = 0
  CHL = 0
  CBOD = N
  DO = 0
  ON = 0
  OP = 0
END IF

60 FORMAT (I10,5X,A)
61 FORMAT (2F10.1)
62 FORMAT (2I5)
63 FORMAT (2(2F10.2))

WRITE (55,60) NH,"NH3" F:LOADS"
WRITE (55,60) NO, "NO3"
WRITE (55,60) PO, "PO4"
WRITE (55,60) CHL,"CHLa"
WRITE (55,60) CBOD,"CBOD"
WRITE (55,61) 1.0,1.0

X = 1
DO J = 1,N
DO I = 2,N+1
  W = INT(F(I,1))
  IF (W.EQ.X) THEN
    TEMP(I,1) = F(I,1)
    TEMP(I,2) = F(I,2)
    GO TO 700
  END IF
END DO
END DO
700 X = X+1
END DO

DO I = 2,N+1
  NPS = F(I+N,2)
  LOAD = TEMP(I,2) + NPS
  WRITE (55,62) INT(TEMP(I,1)),2
  WRITE (55,63) LOAD,0.0,LOAD,TPRINT
END DO

WRITE (55,60) DO,"DO"
WRITE (55,60) ON,"ON"
WRITE (55,60) OP,"OP"
WRITE (55,60) 0

RETURN
END

```

```

***Subroutine INPUTG: Creates the G Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****The scaling and conversion factors are set to 1.0
*   Temperatures are in °C
*   Sediment Oxygen Demand are in g/m^2-day
*   Salinity is in ppt
*****The theta used to correct SOD for temperature does not
*   spatially vary

```

```

SUBROUTINE INPUTG(G,NOSEG)

INTEGER NOSEG
REAL G((NOSEG+1),4), SODTA

DO I = 1,NOSEG+1
  READ(35,*) (G(I,J),J=1,4)
END DO

SODTA = G(1,1)

50 FORMAT (I10,5X,A)
51 FORMAT (4(A5,I5,F10.3))
52 FORMAT (I10)

WRITE (55,50) 4,"G:PARAMETERS"
WRITE (55,51) "TMPSG",3,1.0,"SOD1D",9,1.0,"SODTA",
& 12,1.0,"SAL",2,1.0

DO I = 2,NOSEG+1
  WRITE (55,52) INT(G(I,1))
  WRITE (55,51) "TMPSG",3,G(I,2),"SOD1D",9,G(I,3),"SODTA",
& 12,SODTA,"SAL",2,G(I,4)
END DO

RETURN
END

```

```

***Subroutine INPUTH: Creates the H Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****The reaeration rate (/day) does not spatially vary
*****The deoxygenation coefficient (/day) does not spatially vary

```

```

SUBROUTINE INPUTH(H,MODEL)

INTEGER MODEL,NH,NO,PO,CHL,CBOD,DO,ON,OP
REAL H(2),KD,KA

READ(40,*) H

KD = H(1)

```

```

KA = H(2)

70 FORMAT (10X,A)
71 FORMAT (A10,I10)
72 FORMAT (A10,I10,F10.2)

```

```

IF (MODEL.EQ.0) THEN
  NH = 0
  NO = 0
  PO = 0
  CHL = 0
  CBOD = 1
  DO = 1
  ON = 0
  OP = 0
END IF

WRITE (55,70) "H:CONSTANTS"
WRITE (55,71) "GLOBALS",0
WRITE (55,71) "NH3",NH
WRITE (55,71) "NO3",NO
WRITE (55,71) "PO4",PO
WRITE (55,71) "CHLa",CHL
WRITE (55,71) "CBOD",CBOD
WRITE (55,72) "deoxygenation",1
WRITE (55,72) "KD",71,KD
WRITE (55,71) "DO",DO
WRITE (55,72) "oxygenation",1
WRITE (55,72) "K2",82,KA
WRITE (55,71) "ON",ON
WRITE (55,71) "OP",OP

RETURN
END

```

```

***Subroutine INPUTI: Creates the I Block to the WASP5 input file
***The following defaults are set in this subroutine:
*****Since this model is considered steady state (right now), there
*   are no time functions presently input into the model

```

```

SUBROUTINE INPUTI(I)

REAL I(1)

READ (45,*) I

WRITE (55,80) INT(I(1)),"I:TIME FUNCTIONS"
80 FORMAT (I10,5X,A)

RETURN
END

```

```

***Subroutine INPUTJ: Creates the J Block to the WASP5 input file

```

```

***The following defaults are set in this subroutine:
*****This program, at present only considers the bod and do i.c.'s
*****The dissolved fraction of BOD is set at 0.5
*****The dissolved fraction of DO is always 1.0
*****The maximum value of all variables is set at 1.0e8
*****Solids Field 3 is what transports the system in its particulate
* form for BOD and Solids Field 5 is used for DO
*****The densities of BOD and DO are set at 1.0 -- EUTRO does not
* use those numbers
*****All initial conditions are in mg/L

```

```

SUBROUTINE INPUTJ(I,NOSEG)

```

```

INTEGER NOSEG,J
REAL I(NOSEG*8+8,2),DENBOD,DENDO,MAX

```

```

DO J = 1,NOSEG*8+8
  READ (50,*) (I(J,K),K=1,2)
END DO

```

```

MAX = 100000000.0
DENBOD = 1.0
DENDO = 1.0

```

```

90 FORMAT (A10,30X,I5,F5.1,E10.2,A)
91 FORMAT (3(I5,2F10.2))

```

```

DUMMY1 = INT(NOSEG/3)
DUMMY2= NOSEG - 3*DUMMY1

```

```

WRITE (55,90) "NH3",3,1.2,MAX,"J:INITIAL CONDITIONS"

```

```

DO J = 2,3*DUMMY1+1,3
  WRITE (55,91) INT(I(J,1)),I(J,2),I(1,1),
& INT(I(J+1,1)),I(J+1,2),I(1,1),
& INT(I(J+2,1)),I(J+1,2),I(1,1)
  DUMMY3 = J+3
END DO

```

```

WRITE (55,91) (INT(I(J,1)),I(J,2),I(1,1),J=DUMMY3,DUMMY2
& +DUMMY3-1)

```

```

X = DUMMY3+DUMMY2+1
WRITE (55,90) "NO3",5,1.2,MAX
DO J = X,6*DUMMY1+DUMMY2+2,3
  WRITE (55,91) INT(I(J,1)),I(J,2),I(NOSEG+2,1),
& INT(I(J+1,1)),I(J+1,2),I(NOSEG+2,1),
& INT(I(J+2,1)),I(J+1,2),I(NOSEG+2,1)
  DUMMY3 = J+3
END DO

```

```

WRITE (55,91) (INT(I(J,1)),I(J,2),I(1,1),J=DUMMY3,DUMMY2
& +DUMMY3-1)

```

```

X = DUMMY3+DUMMY2+1

```

```

WRITE (55,90) "PO4",5,1.2,MAX
DO J = X,9*DUMMY1+2*DUMMY2+3,3
  WRITE (55,91) INT(I(J,1)),I(J,2),I(2*NOSEG+3,1),
& INT(I(J+1,1)),I(J+1,2),I(2*NOSEG+3,1),
& INT(I(J+2,1)),I(J+1,2),I(2*NOSEG+3,1)
  DUMMY3 = J+3
END DO
WRITE (55,91) (INT(I(J,1)),I(J,2),I(1,1),J=DUMMY3,DUMMY2
& +DUMMY3-1)

```

```

X = DUMMY3+DUMMY2+1
WRITE (55,90) "PHYT",4,1.2,MAX
DO J = X,12*DUMMY1+3*DUMMY2+4,3
  WRITE (55,91) INT(I(J,1)),I(J,2),I(3*NOSEG+4,1),
& INT(I(J+1,1)),I(J+1,2),I(3*NOSEG+4,1),
& INT(I(J+2,1)),I(J+1,2),I(3*NOSEG+4,1)
  DUMMY3 = J+3
END DO
WRITE (55,91) (INT(I(J,1)),I(J,2),I(1,1),J=DUMMY3,DUMMY2
& +DUMMY3-1)

```

```

X = DUMMY3+DUMMY2+1
WRITE (55,90) "CBOD",3,DENBOD,MAX
DO J = X,15*DUMMY1+4*DUMMY2+5,3
  WRITE (55,91) INT(I(J,1)),I(J,2),I(4*NOSEG+5,1),
& INT(I(J+1,1)),I(J+1,2),I(4*NOSEG+5,1),
& INT(I(J+2,1)),I(J+1,2),I(4*NOSEG+5,1)
  DUMMY3 = J+3
END DO
WRITE (55,91) (INT(I(J,1)),I(J,2),I(1,1),J=DUMMY3,DUMMY2
& +DUMMY3-1)

```

```

X = DUMMY3+DUMMY2+1
WRITE (55,90) "DO",5,DENDO,MAX
DO J = X,18*DUMMY1+5*DUMMY2+6,3
  WRITE (55,91) INT(I(J,1)),I(J,2),I(5*NOSEG+6,1),
& INT(I(J+1,1)),I(J+1,2),I(5*NOSEG+6,1),
& INT(I(J+2,1)),I(J+1,2),I(5*NOSEG+6,1)
  DUMMY3 = J+3
END DO
WRITE (55,91) (INT(I(J,1)),I(J,2),I(1,1),J=DUMMY3,DUMMY2
& +DUMMY3-1)

```

```

X = DUMMY3+DUMMY2+1
WRITE (55,90) "ON",3,1.2,MAX
DO J = X,21*DUMMY1+6*DUMMY2+7,3
  WRITE (55,91) INT(I(J,1)),I(J,2),I(6*NOSEG+7,1),
& INT(I(J+1,1)),I(J+1,2),I(6*NOSEG+7,1),
& INT(I(J+2,1)),I(J+1,2),I(6*NOSEG+7,1)
  DUMMY3=J+3
END DO
WRITE (55,91) (INT(I(J,1)),I(J,2),I(1,1),J=DUMMY3,DUMMY2
& +DUMMY3-1)

```

```

X = DUMMY3+DUMMY2+1
WRITE (55,90) "OP",3,1.2,MAX
DO J = X,24*DUMMY1+7*DUMMY2+8,3
  WRITE (55,91) INT(I(J,1)),I(J,2),I(7*NOSEG+8,1),
&      INT(I(J+1,1)),I(J+1,2),I(7*NOSEG+8,1),
&      INT(I(J+2,1)),I(J+1,2),I(7*NOSEG+8,1)
  DUMMY3 = J+3
END DO
WRITE (55,91) (INT(I(J,1)),I(J,2),I(1,1),J=DUMMY3,DUMMY2
&      +DUMMY3-1)
RETURN
END

```

## Appendix C-5 FORTRAN Programs Used to Format WASP5 Output for Avenue Processing

### Program: calout.for Formats TOXI5 output

```

*****
* Program:   calout.for
* Purpose:   This program will take WASP5 output
             (specifically TOXI5, Level One Complexity
             and create an array, written to a txt
             File (salinity.txt). This text file will
             then be read by Avenue and imported into
             ArcView for viewing with Charts and graphs
* Hardware:  IBM Pentium 100
* Software:  Microsoft FORTRAN
* Author:    Jennifer Benaman
             Graduate Research Assistant,
             University of Texas at Austin
* Date:      June, 1996
*****

* Program Declaration
  PROGRAM CALOUT

* Variable Declaration
  REAL TPRINT,DTS,PRINTINV,INT(4),OUT(20000,6),
& TEMP(20000,100)

  INTEGER NSEG,N,M
  CHARACTER*12 FNAME

* TPRINT:    the final time (days) the model was run
* DTS:       the model maximum time step (days)
* PRINTINV:  the print interval (days) that the output
             was printed to the file
* NSEG:      the total number of segments in the system
* OUT(20000,6): an array that holds the initial output file

*****
* TEMP(20000,100):an array that holds the values for the text
*                   file for the final output

OPEN (UNIT=6, FILE='OUTINME.TXT',STATUS='OLD')

READ (6,16) FNAME
16 FORMAT (A12)

OPEN (UNIT=30, FILE="START.TXT" ,STATUS="OLD")
OPEN (UNIT=40, FILE=FNAME, STATUS="OLD")
OPEN (UNIT=50, FILE="SALINITY.TXT", STATUS="UNKNOWN")

CALL INITIAL(TPRINT,DTS,PRINTINV,NSEG,INT,N,M)
CALL OUTPUT(NSEG,OUT,TEMP,N,M)

STOP
END

*****Subroutines

****Subroutine: initial
*****This subroutine will read the intial information from
*****a text file created by Avenue

SUBROUTINE INITIAL(TPRINT,DTS,PRINTINV,NSEG,I,N,M)

REAL TPRINT,DTS,PRINTINV,I(4)
INTEGER NSEG

READ (30,*) I

TPRINT = I(1)

```



```

DTS = I(2)
PRINTINV = I(3)
NSEG = INT(I(4))

N = INT((TPRINT/PRINTINV)+1)
M = INT(4*NSEG*(TPRINT/PRINTINV+1)+22)

RETURN
END

****Subroutine: output
*****This subroutine will take the output file generated by TOXI5 and
*****just read the salinity values at each time step and create a
*****new array which has just the time steps and salinity for each
*****segment

SUBROUTINE OUTPUT(NSEG,OUT,TEMP,N,M)

INTEGER NSEG,N,M,T,X
REAL TEMP(N,NSEG+1),OUT(M,6)
CHARACTER*20 JUNK

DO I = 1,22
  READ (40,10) JUNK
END DO
10 FORMAT (A20)
20 FORMAT (F6.0,F10.0)

```

```

30 FORMAT (6E11.3)
40 FORMAT (F10.3,50(F7.3))

X = 1
DO I = 23,M,4
  READ (40,20) (OUT(I,J),J=1,2)
  READ (40,30) ((OUT(K,J),J=1,6),K=I+1,I+3)
  X = X+1
END DO

T = 1
DO I = 23,M,4*NSEG
  TEMP(T,1) = OUT(I,2)
  DO J = 2,NSEG+1
    IF (J.EQ.2) THEN
      TEMP(T,J) = OUT(I+2,1)/1000000
    ELSE
      TEMP(T,J) = OUT((I+2+4*(J-2)),1)/1000000
    END IF
  END DO
  WRITE (50,40) (TEMP(T,J),J=1,NSEG+1)
  T = T+1
END DO

RETURN
END

```

### Program: modout.for Formats EUTRO5 output

```

*****k*****
* Program: modout.for
* Purpose: This program will take WASP5 output
*           (specifically EUTRO5, Level One Complexity
*           and create an array, written to txt
*           Files (bod.txt and do.txt). These text file will
*           then be read by Avenue and imported into
*           ArcView for viewing with Charts and graphs
* Hardware: IBM Pentium 100
* Software: Microsoft FORTRAN
* Author: Jennifer Benaman
*           Graduate Research Assistant,
*           University of Texas at Austin
* Date: June, 1996
*****

* Program Declaration
PROGRAM MODOUT

```

```

* Variable Declaration
REAL TPRINT,DTS,PRINTINV,INT(4),OUT(20000,6),
& TEMP1(20000,100),TEMP2(20000,100)

INTEGER NSEG,N,M
CHARACTER*12 FNAME

* TPRINT: the final time (days) the model was run
* DTS: the model maximum time step (days)
* PRINTINV: the print interval (days) that the output
* was printed to the file
* NSEG: the total number of segments in the system
* OUT(20000,6): an array that holds the initial output file
* TEMP1(20000,100):an array that holds the do values for the text
* file for the final output (do.txt)
* TEMP2(20000,100):an array that holds the bod values for the text
* file for the final output(bod.txt)

OPEN (UNIT=6, FILE="EOUTNME.TXT",STATUS="OLD")

```

```

      READ (6,16) FNAME
16  FORMAT (A12)

      OPEN (UNIT=30,FILE="ESTART.TXT",STATUS="OLD")
      OPEN (UNIT=40,FILE=FNAME,STATUS="OLD")
      OPEN (UNIT=50,FILE="DO.TXT",STATUS="UNKNOWN")
      OPEN (UNIT=60,FILE="BOD.TXT",STATUS="UNKNOWN")

      CALL INITIAL(TPRINT,DTS,PRINTINV,NSEG,INT,N,M)
      CALL OUTPUT(NSEG,OUT,TEMP1,TEMP2,N,M)
      STOP
      END

*****Subroutines

****Subroutine: initial
*****This subroutine will read the intial information from
*****a text file created by Avenue

      SUBROUTINE INITIAL(TPRINT,DTS,PRINTINV,NSEG,I,N,M)

      REAL TPRINT,DTS,PRINTINV,I(4)
      INTEGER NSEG

      READ (30,*) I

      TPRINT = I(1)
      DTS = I(2)
      PRINTINV = I(3)
      NSEG = INT(I(4))

      N = INT((TPRINT/PRINTINV)+1)
      M = INT(8*NSEG*(TPRINT/PRINTINV+1)+48)

      RETURN
      END

****Subroutine: output
*****This subroutine will take the output file generated by EUTRO5 and
*****just read the bod and do values at each time step and create a
*****two new arrays which have just the time steps and do and bod for each
*****segment

      SUBROUTINE OUTPUT(NSEG,OUT,TEMP1,TEMP2,N,M)

      INTEGER NSEG,N,M,T,X
      REAL TEMP1(N,NSEG+1),TEMP2(N,NSEG+1),OUT(M,6)
      CHARACTER*20 JUNK

      DO I = 1,48
         READ (40,10) JUNK
      END DO

```

```

10  FORMAT (A20)
20  FORMAT (F6.0,F12.0)
30  FORMAT (6E11.3)
40  FORMAT (F10.3,50(F7.3))

      DO I = 49,M,8
         READ (40,20) (OUT(I,J),J=1,2)
         READ (40,30) ((OUT(K,J),J=1,6),K=I+1,I+7)
      END DO

      T = 1
      DO I = 49,M,8*NSEG
         TEMP1(T,1) = OUT(I,2)
         TEMP2(T,1) = OUT(I,2)
         DO J = 2,NSEG+1
            IF (J.EQ.2) THEN
               TEMP1(T,J) = OUT(I+1,5)
               TEMP2(T,J) = OUT(I+3,2)
            ELSE
               TEMP1(T,J) = OUT((I+1+8*(J-2)),5)
               TEMP2(T,J) = OUT((I+3+8*(J-2)),2)
            END IF
         END DO
         WRITE (50,40) (TEMP1(T,J),J=1,NSEG+1)
         WRITE (60,40) (TEMP2(T,J),J=1,NSEG+1)
         T = T+1
      END DO

      RETURN
      END

```

**Appendix D**  
**WASP5 Input Files**

## Appendix D-1

### Input File for WASP5 for Model Calibration, Created by Avenue and FORTRAN Formatting Program (calgen.for)

Salinity Calibration									3538.5	4.6	2	9								
TOXI5 Calibration with Salinity									1989.9	4.6	3	9								
NSEGS NSYS ICRD MPLG IDMP NSLN INTY ADPC DD HHMM TFLG									2											
17 1 0 0 1 0 1 0. 0. 0 0 0									2	.0	.0	.0	100.0							
1 2 3 4 5 6									0	1	1	1	1	1	1	1				
1									2	0	1.0000						C:VOLUMES			
.05000 100.0									1	0	1.0									
2									8	9		1	8412370.0	.0040	.4	1.2	.6			
.5 .0 .5 100.0									6	9		1	6010670.0	.0040	.4	1.2	.6			
0 1 1 1 1 1 1									7	9		1	6153160.0	.0040	.4	1.2	.6			
1 B:EXCHANGES									5	9		1	6565760.0	.0040	.4	1.2	.6			
5 1.0 1.0									1	9		1	5265120.0	.0040	.4	1.2	.6			
7									4	9		1	4278080.0	.0040	.4	1.2	.6			
1625.8 3093.5 6 7									2	9		1	5752860.0	.0040	.4	1.2	.6			
2471.2 2947.1 7 8									3	9		1	3235260.0	.0040	.4	1.2	.6			
1625.8 3867.8 5 6									10	0		1	3745030.0	.0040	.4	1.2	.6			
1625.8 3388.5 1 2									12	0		1	541385.0	.0040	.4	1.2	.6			
1625.8 3334.9 4 5									11	0		1	1626730.0	.0040	.4	1.2	.6			
1625.8 2764.2 2 3									13	0		1	598034.0	.0040	.4	1.2	.6			
1625.8 2310.7 3 4									14	0		1	1701400.0	.0040	.4	1.2	.6			
2									15	0		1	2308450.0	.0040	.4	1.2	.6			
7 704.5 .0 704.5 100.0									16	0		1	837184.0	.0040	.4	1.2	.6			
7									9	0		3	2503.8	.0040	.4	1.2	.6			
465.4 5642.7 10 1									2	nothing										
168.2 1609.8 12 2									8	1.0		1.0								
505.4 1609.8 11 3									11											
185.8 1609.8 13 4									1.00	0	10	1.00	10	1	1.00	1	2	1.00	2	3
528.6 1609.8 14 5									1.00	3	4	1.00	4	5	1.00	5	6	1.00	6	7
717.2 1609.8 15 6									1.00	7	8	1.00	8	17	1.00	17	0			
260.1 1609.8 16 8									2			14.35	.00	14.35	100.00					
2									10			1.00	0	12	1.00	12	2	1.00	2	3
119.9 .0 119.9 100.0									1.00	4	5	1.00	5	6	1.00	6	7	1.00	7	8
1									1.00	8	17	1.00	17	0						
2471.2 3311.4 17 8									2			5.70	.00	5.70	100.00					
2									9			1.00	0	11	1.00	11	3	1.00	3	4
704.5 .0 704.5 100.0									1.00	5	6	1.00	6	7	1.00	7	8	1.00	8	17
1									1.00	17	0									
2471.2 1609.3 17 0									2			3.54	.00	3.54	100.00					
2									8			1.00	0	13	1.00	13	4	1.00	4	5
959.3 .0 959.3 100.0									1.00	6	7	1.00	7	8	1.00	8	17	1.00	17	0
8																				
3404.2 4.0 8 9																				
3697.1 3.1 6 9																				
2489.9 4.0 7 9																				
4038.5 4.6 5 9																				
3238.5 4.6 1 9																				
2631.4 3.1 4 9																				

```

2       1.00      .00      1.00     100.00
7       1.00     0  14      1.00    14   5      1.00    5   6      1.00    6   7
       1.00     7   8      1.00    8   17     1.00   17  0
2       2.06      .00      2.06     100.00
6       1.00     0  15      1.00    15   6      1.00    6   7      1.00    7   8
       1.00     8  17      1.00    17   0
2       7.59      .00      7.59     100.00
4       1.00     0  16      1.00    16   8      1.00    8  17      1.00   17  0
2       1.20      .00      1.20     100.00
2       1.00     0  17      1.00    17   0
2       42.40     .00      42.40     100.00
0  1  1  1  1  1  1  1
  8  SAL  E:BOUNDARIES
1000.00      1.00
10  2
       1.82      .00      1.82     100.00
17  2
       10.90     .00      10.90     100.00
12  2

```

```

       .20      .00      .20     100.00
11  2
       .20      .00      .20     100.00
13  2
       .20      .00      .20     100.00
14  2
       .20      .00      .20     100.00
15  2
       .20      .00      .20     100.00
16  2
       .20      .00      .20     100.00
0       F:LOADS -- NO LOADS
0       NO NPS LOADS
0       G:PARAMETERS -- NO PARAMETERS

```

```

H:CONSTANTS
GLOBALS      0
SALINITY     1
GENERAL      3
PIXC        111      .00      KBW      141      .00
MOLWT       81      78.50
0
I:TIME FUNCTIONS
SAL          3  .0  .35E+05J:INITIAL CONDITIONS
1  .00  1.00  2  .00  1.00  3  .00  1.00
4  .00  1.00  5  .00  1.00  6  .00  1.00
7  .00  1.00  8  .00  1.00  9  .00  1.00
10 1820.00 1.00 11  .00  1.00 12  .00  1.00
13  .00  1.00 14  .00  1.00 15  .00  1.00
16  .00  1.00 17 10900.00 1.00

```

## Appendix D-2

### Input File for WASP5 for BOD/DO Model,

### Created by Avenue and FORTRAN Formatting Program (outgen.for)

#### Average Year Scenario

```

Average Year Conditions -- Initial
EUTRO Simple Streeter-Phelps with SOD
NSEG NSYS ICRD MPLG IDMP NSLN INTY ADPC DD HHMM TFLG
17  8  0  0  0  5  1  1  0.  0.  0  0  0
  1  2  3  4  5  6
1
.05000      100.0
2
.5          .0          .5     100.0
1  1  1  1  0  0  1  1
1  B:EXCHANGES
5  1.0      1.0
7
1625.8    3093.5    6  7
2471.2    2947.1    7  8
1625.8    3867.8    5  6

```

```

A: MODEL OPTIONS
1625.8    3388.5    1  2
1625.8    3334.9    4  5
1625.8    2764.2    2  3
1625.8    2310.7    3  4
2
704.500      .0    704.500     100.0
7
465.4     5642.7    10  1
168.2     1609.8    12  2
505.4     1609.8    11  3
185.8     1609.8    13  4
528.6     1609.8    14  5
717.2     1609.8    15  6
260.1     1609.8    16  8
2
119.900      .0    119.900     100.0

```



3.03	.00	3.03	100.00
12 2			
5.00	.00	5.00	100.00
11 2			
5.00	.00	5.00	100.00
13 2			
5.00	.00	5.00	100.00
14 2			
5.00	.00	5.00	100.00
15 2			
5.00	.00	5.00	100.00
16 2			
5.00	.00	5.00	100.00
17 2			
3.63	.00	3.63	100.00
0	ON		
0	OP		
0	NH3	F:LOADS	
0	NO3		
0	PO4		
0	CHLa		
8	CBOD		
1.0	1.0		
1 2			
8010.42	.00	8010.42	100.00
2 2			
3709.71	.00	3709.71	100.00
3 2			
2080.65	.00	2080.65	100.00
4 2			
2474.68	.00	2474.68	100.00
5 2			
1432.67	.00	1432.67	100.00
6 2			
4718.60	.00	4718.60	100.00
7 2			
146.86	.00	146.86	100.00
8 2			
973.81	.00	973.81	100.00
0	DO		
0	ON		
0	OP		
4	G:PARAMETERS		
TMPSG 3	1.000SOD1D	9	1.000 SAL 2 1.000
8			
TMPSG 3	24.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 9.850
6			
TMPSG 3	24.200SOD1D	9	1.500SODTA 12 1.065 SAL 2 9.450
7			
TMPSG 3	24.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 9.850
5			
TMPSG 3	26.400SOD1D	9	1.500SODTA 12 1.065 SAL 2 9.960
1			

TMPSG 3	28.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 5.820
4			
TMPSG 3	23.800SOD1D	9	1.500SODTA 12 1.065 SAL 2 7.870
2			
TMPSG 3	23.800SOD1D	9	1.500SODTA 12 1.065 SAL 2 7.870
3			
TMPSG 3	23.800SOD1D	9	1.500SODTA 12 1.065 SAL 2 7.870
10			
TMPSG 3	23.600SOD1D	9	1.500SODTA 12 1.065 SAL 2 1.820
12			
TMPSG 3	20.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 .200
11			
TMPSG 3	20.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 .200
13			
TMPSG 3	20.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 .200
14			
TMPSG 3	20.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 .200
15			
TMPSG 3	20.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 .200
16			
TMPSG 3	20.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 .200
17			
TMPSG 3	25.300SOD1D	9	1.500SODTA 12 1.065 SAL 2 10.900
9			
TMPSG 3	20.000SOD1D	9	1.500SODTA 12 1.065 SAL 2 .000

H:CONSTANTS

GLOBALS	0																			
NH3	0																			
NO3	0																			
PO4	0																			
CHLa	0																			
CBOD	1																			
deoxygenta	1																			
KD	71	.10																		
DO	1																			
oxygenatio	1																			
K2	82	.10																		
ON	0																			
OP	0																			

I:TIME FUNCTIONS

NH3																				
1	.00	1.00	2	.00	3	1.2	.10E+09J:INITIAL CONDITIONS													
4	.00	1.00	5	.00	6	1.00	.00	1.00	3	.00	1.00									
7	.00	1.00	8	.00	9	1.00	.00	1.00	6	.00	1.00									
10	.00	1.00	11	.00	12	1.00	.00	1.00	9	.00	1.00									
13	.00	1.00	14	.00	15	1.00	.00	1.00	12	.00	1.00									
16	.00	1.00	17	.00	16	1.00	.00	1.00	15	.00	1.00									
NO3																				
1	.00	1.00	2	.00	3	1.2	.10E+09													
4	.00	1.00	5	.00	6	1.00	.00	1.00	3	.00	1.00									
7	.00	1.00	8	.00	9	1.00	.00	1.00	6	.00	1.00									
10	.00	1.00	11	.00	12	1.00	.00	1.00	9	.00	1.00									
13	.00	1.00	14	.00	15	1.00	.00	1.00	12	.00	1.00									
16	.00	1.00	17	.00	16	1.00	.00	1.00	15	.00	1.00									

PO4				5	1.2	.10E+09		
1	.00	1.00	2	.00	1.00	3	.00	1.00
4	.00	1.00	5	.00	1.00	6	.00	1.00
7	.00	1.00	8	.00	1.00	9	.00	1.00
10	.00	1.00	11	.00	1.00	12	.00	1.00
13	.00	1.00	14	.00	1.00	15	.00	1.00
16	.00	1.00	17	.00	1.00			
PHYT				4	1.2	.10E+09		
1	.00	1.00	2	.00	1.00	3	.00	1.00
4	.00	1.00	5	.00	1.00	6	.00	1.00
7	.00	1.00	8	.00	1.00	9	.00	1.00
10	.00	1.00	11	.00	1.00	12	.00	1.00
13	.00	1.00	14	.00	1.00	15	.00	1.00
16	.00	1.00	17	.00	1.00			
CBOD				3	1.0	.10E+09		
1	7.18	.50	2	5.04	.50	3	5.04	.50
4	5.04	.50	5	6.25	.50	6	6.25	.50
7	5.06	.50	8	5.06	.50	9	5.06	.50
10	8.14	.50	11	8.40	.50	12	8.40	.50
13	8.00	.50	14	8.60	.50	15	8.60	.50
16	6.00	1.00	17	7.42	1.00			
DO				5	1.0	.10E+09		
1	1.36	1.00	2	1.81	1.00	3	1.81	1.00
4	1.81	1.00	5	.68	1.00	6	.68	1.00
7	1.64	1.00	8	1.64	1.00	9	1.64	1.00
10	3.03	1.00	11	5.00	1.00	12	5.00	1.00
13	5.00	1.00	14	5.00	1.00	15	5.00	1.00
16	5.00	1.00	17	3.63	1.00			
ON				3	1.2	.10E+09		
1	.00	1.00	2	.00	1.00	3	.00	1.00
4	.00	1.00	5	.00	1.00	6	.00	1.00
7	.00	1.00	8	.00	1.00	9	.00	1.00
10	.00	1.00	11	.00	1.00	12	.00	1.00
13	.00	1.00	14	.00	1.00	15	.00	1.00
16	.00	1.00	17	.00	1.00			
OP				3	1.2	.10E+09		
1	.00	1.00	2	.00	1.00	3	.00	1.00
4	.00	1.00	5	.00	1.00	6	.00	1.00
7	.00	1.00	8	.00	1.00	9	.00	1.00
10	.00	1.00	11	.00	1.00	12	.00	1.00
13	.00	1.00	14	.00	1.00	15	.00	1.00
16	.00	1.00	17	.00	1.00			



**Appendix E**  
**Examples of Free Form Text Files Generated by Avenue**

*The following is an example of the text files written by Avenue from the "All Input Blocks" option under "BOD/DO Input Block" Menu*

a.txt -- created by script inputa

0 17 8 0 0 4 1 0 0 0 0.05 100.0 0.5 8

inptnme.txt -- created by script inputa

aveyr.inp

title.txt -- created by script inputa

Average Year Model Run

b.txt -- created by script inputb

0 1 0 0 0  
704.5 1625.8 3093.5 6 7  
704.5 2471.2 2947.06 7 8  
704.5 1625.8 3867.77 5 6  
704.5 1625.8 3388.48 1 2  
704.5 1625.8 3334.92 4 5  
704.5 1625.8 2764.21 2 3  
704.5 1625.8 2310.66 3 4  
119.9 465.4 5642.69 10 1  
119.9 168.2 1609.85 12 2  
119.9 505.4 1609.85 11 3  
119.9 185.8 1609.85 13 4  
119.9 528.6 1609.85 14 5  
119.9 717.2 1609.85 15 6  
119.9 260.1 1609.85 16 8  
704.5 2471.2 3311.43 17 8  
959.3 2471.2 1609.35 17 0  
0.001 3404.16 4.01 8 9  
0.001 3697.06 3.1 6 9  
0.001 2489.95 4.01 7 9  
0.001 4038.48 4.62 5 9  
0.001 3238.48 4.62 1 9  
0.001 2631.37 3.1 4 9  
0.001 3538.48 4.62 2 9  
0.001 1989.95 4.62 3 9  
555

c.txt -- created by script inputc

2 0 1.0 0  
0.004 0 0 0  
0.4 0 0 0  
1.2 0 0 0  
0.6 0 0 0  
8 9 1 8.41237e+006  
6 9 1 6.01067e+006

7 9 1 6.15316e+006  
5 9 1 6.56576e+006  
1 9 1 5.26512e+006  
4 9 1 4.27808e+006  
2 9 1 5.75286e+006  
3 9 1 3.23526e+006  
10 0 1 3.74503e+006  
12 0 1 541385  
11 0 1 1.62673e+006  
13 0 1 598034  
14 0 1 1.7014e+006  
15 0 1 2.30845e+006  
16 0 1 837184  
17 0 1 7.95405e+006  
9 0 3 250.28

d.txt -- created by script inputd

2 0  
0 10  
10 1  
1 2  
2 3  
3 4  
4 5  
5 6  
6 7  
7 8  
8 17  
17 0  
14.35 999  
0 12  
12 2  
2 3  
3 4  
4 5  
5 6  
6 7  
7 8  
8 17  
17 0  
5.7 999  
0 11  
11 3  
3 4  
4 5  
5 6  
6 7  
7 8  
8 17

```
17 0
3.54 999
0 13
13 4
4 5
5 6
6 7
7 8
8 17
17 0
1 999
0 14
14 5
5 6
6 7
7 8
8 17
17 0
2.06 999
0 15
15 6
6 7
7 8
8 17
17 0
7.59 999
0 16
16 8
8 17
17 0
1.2 999
0 17
17 0
42.4 999
555 0
```

e.txt -- created by script inpute

```
8 0
10 1.92
12 1.37
11 1.87
13 1.68
14 1.67
15 1.48
16 1.35
17 7.42
10 3.03
12 5
11 5
13 5
14 5
15 5
16 5
17 3.63
```

f.txt -- created by script inputf

```
8 0
1 3.25666
2 24.1639
3 0
4 1827.28
5 70.0928
6 291.247
7 30.814
8 281.801
1 8007.16
2 3685.55
3 2080.65
4 647.397
5 1362.58
6 4427.35
7 116.044
8 692.011
```

g.txt -- created by script inputg

```
1.065 0 0 0
8 24 1.5 9.85
6 24.2 1.5 9.45
7 24 1.5 9.85
5 26.4 1.5 9.96
1 28 1.5 5.82
4 23.8 1.5 7.87
2 23.8 1.5 7.87
3 23.8 1.5 7.87
10 23.6 1.5 1.82
12 20 1.5 0.2
11 20 1.5 0.2
13 20 1.5 0.2
14 20 1.5 0.2
15 20 1.5 0.2
16 20 1.5 0.2
17 25.3 1.5 10.9
9 20 1.5 0
```

h.txt -- created by script inputh

```
0.1
0.1
```

i.txt -- created by script inputi

```
0
```

j.txt -- created by script inputj

```
1 0
1 0
```

2 0  
3 0  
4 0  
5 0  
6 0  
7 0  
8 0  
9 0  
10 0  
11 0  
12 0  
13 0  
14 0  
15 0  
16 0  
17 0  
1 0  
1 0  
2 0  
3 0  
4 0  
5 0  
6 0  
7 0  
8 0  
9 0  
10 0  
11 0  
12 0  
13 0  
14 0  
15 0  
16 0  
17 0  
1 0  
1 0  
2 0  
3 0  
4 0  
5 0  
6 0  
7 0  
8 0  
9 0  
10 0  
11 0  
12 0  
13 0  
14 0  
15 0  
16 0  
17 0  
1 0  
1 0  
2 0

3 0  
4 0  
5 0  
6 0  
7 0  
8 0  
9 0  
10 0  
11 0  
12 0  
13 0  
14 0  
15 0  
16 0  
17 0  
0.5 0  
1 7.18  
2 5.04  
3 5.04  
4 5.04  
5 6.25  
6 3.53  
7 5.06  
8 5.06  
9 0  
10 8.14  
11 8.4  
12 6.9  
13 8  
14 8.6  
15 5.9  
16 6  
17 7.42  
1 0  
1 1.36  
2 1.81  
3 1.81  
4 1.81  
5 0.68  
6 2.25  
7 1.64  
8 1.64  
9 0  
10 3.03  
11 5  
12 5  
13 5  
14 5  
15 5  
16 5  
17 3.63  
1 0  
1 0  
2 0  
3 0

4 0  
5 0  
6 0  
7 0  
8 0  
9 0  
10 0  
11 0  
12 0  
13 0  
14 0  
15 0  
16 0  
17 0  
1 0  
1 0  
2 0  
3 0  
4 0  
5 0  
6 0  
7 0  
8 0  
9 0  
10 0  
11 0  
12 0  
13 0  
14 0  
15 0  
16 0  
17 0  
1 0  
1 0

2 0  
3 0  
4 0  
5 0  
6 0  
7 0  
8 0  
9 0  
10 0  
11 0  
12 0  
13 0  
14 0  
15 0  
16 0  
17 0  
1 0  
1 0  
2 0  
3 0  
4 0  
5 0  
6 0  
7 0  
8 0  
9 0  
10 0  
11 0  
12 0  
13 0  
14 0  
15 0  
16 0  
17 0

**Appendix F**  
**Avenue Scripts Used for ArcView/WASP5 Connection**

### Scripts Created for ArcView/WASP5 Connection

Script Name	Function
all	Runs all scripts to create texts file for EUTRO5 input file
btn	Controls the "bug" icon on the "Segmentation" view to plot a chart
calinputa	Writes the text files for Input Block A in the model calibration
calinputall	Runs scripts which create text files for TOXI5 input file
cal_parchk	Checks the parameters on the output file to process for the calibration model
eutrorun	Runs the EUTRO5 model from ArcView
frame1 - frame4	Produces a new theme on the "Segmentation" to observe the change in concentration over time; each script is time delayed
gen_file_eutro	Executes outgen.exe from ArcView
gen_file_toxi	Executes calgen.exe from ArcView
help	Executes the help file for the ArcView/WASP5 connection
inputa	Writes the text files for Input Block A for the BOD/DO model input file
inputb	Writes the text files for Input Block B for the BOD/DO or calibration model input file
inputc	Writes the text files for Input Block C for the BOD/DO or calibration model input file
inputcale	Writes the text files for Input Block E for the calibration input file
inputcalf	Writes the text files for Input Block F for the calibration input file
inputcalg	Writes the text files for Input Block G for the calibration input file
inputcalh	Writes the text files for Input Block H for the calibration input file
inputcalj	Writes the text files for Input Block J for the calibration input file
inputd	Writes the text files for Input Block D for the BOD/DO or calibration input file
inpute	Writes the text files for Input Block E for the BOD/DO input file
inputf	Writes the text files for Input Block F for the BOD/DO input file
inputg	Writes the text files for Input Block G for the BOD/DO input file
inputh	Writes the text files for Input Block H for the BOD/DO input file
inputi	Writes the text files for Input Block I for the BOD/DO or calibration input file
inputj	Writes the text files for Input Block J for the BOD/DO input file
mod_parchk	Checks the parameters on the output file to process for the BOD/DO model
run_calout	Executes calout.exe and creates a dbf table for salinity at each segment over time
run_modout	Executes modout.exe and creates dbf tables for DO and BOD at each segment over time
toxirun	Executes TOXI5 from ArcView
vwout	Controls the output presentation processes -- all five options

**'Script: all**

'this script runs all of the ten input blocks for  
'the model run of eutro  
theProject = av.GetProject

```
_dummy = 0
aSEd = av.GetProject.FindDoc("inputa")
aSEd.Run(_dummy)
bSEd = av.GetProject.FindDoc("inputb")
bSEd.Run(_dummy)
cSEd = av.GetProject.FindDoc("inputc")
cSEd.Run(_dummy)
dSEd = av.GetProject.FindDoc("inputd")
dSEd.Run(_dummy)
eSEd = av.GetProject.FindDoc("inpute")
eSEd.Run(_dummy)
fSEd = av.GetProject.FindDoc("inputf")
fSEd.Run(_dummy)
gSEd = av.GetProject.FindDoc("inputg")
gSEd.Run(_dummy)
hSEd = av.GetProject.FindDoc("inputh")
hSEd.Run(_dummy)
iSEd = av.GetProject.FindDoc("inputi")
iSEd.Run(_dummy)
jSEd = av.GetProject.FindDoc("inputj")
jSEd.Run(_dummy)
```

**'Script: btn**

'This script controls the 'bug' button that graphs a chart of salinity vs. Time for  
'a chosen segment

theProject = av.GetProject

```
'setting the Tables and views
salTable = av.GetProject.FindDoc(_tblname)
salVTab = salTable.GetVTab
salFields = salVTab.GetFields
timeField = salVTab.FindField("time")
theView = av.GetProject.FindDoc("Segmentation")
segTheme = theView.FindTheme("Main Segmentation")
bcTheme = theView.FindTheme("Boundary Segmentation")
segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab
```

'Getting the User to select a point

```
theView = av.GetProject.FindDoc("Segmentation")
p = theView.GetDisplay.ReturnUserPoint
```

```
theThemes = theView.GetActiveThemes
t = theThemes.get(0)
theFTab = t.GetFTab
if (t.CanSelect.Not) then
  exit
end
t.SelectByPoint(p,#VTAB_SELTYPE_NEW)
recs=t.FindByPoint(p)
if (recs.count = 0) then
  MsgBox.Info("No segment selected", "")
  exit
end
for each rec in recs
  numrec = rec
end
gcField = theFTab.FindField("grid-code")
seg = theFTab.ReturnValue(gcField,numrec)
plotField = salFields.Get(seg)
fldList = {plotField}
```

' Creates a list of tables and allows the user to pick which one they  
' want to view the output for -- these steps execute only if the user has not  
' yet chosen a table.

```
if (timeField = nil) then
docList = theProject.GetDocs
tabList = List.Make
numdocs=docList.count
for each i in 0..(numdocs-1)
  dtype=(docList.get(i)).GetClass.GetClassName
  if (dtype="Table") then
    tabList.Add(docList.Get(i).GetName)
  end
end
```

```
_tblname = MsgBox.ChoiceAsString(tabList,"Choose the output table you want to work with","View Output")
salTable = av.GetProject.FindDoc(_tblname)
salVTab = salTable.GetVTab
salFields = salVTab.GetFields
timeField = salVTab.FindField("time")
end
```

'Allows the user to choose a color for the chart

```
colorlist=List.Make
colorlist.Add("blue")
colorlist.Add("yellow")
colorlist.Add("green")
colorlist.Add("red")
col=MsgBox.ChoiceAsString(colorlist,"Select a color for the chart","View Output")
if (col="blue") then
  _chcolor = Color.GetBlue
elseif (col="yellow") then
```



```

_chcolor = Color.GetYellow
elseif (col="green") then
_chcolor = Color.GetGreen
else (col="red")
_chcolor = Color.GetRed
end

step = salVTab.ReturnValue(timeField,1) - salVTab.ReturnValue(timeField,0)
'Makes the chart
xChart = Chart.Make(salVTab fldLst)
xChart.SetSeriesFromRecords(false)
' xChart.SetRecordLabelField(TimeField)
xchartname=xchart.getname
theProject.setActive(xchart)
xChartDisp = xchart.GetChartDisplay
xChartDisp.setType(#CHARTDISPLAY_line)
xChartDisp.SetSeriesColor(0,_chcolor)
the_x=xChart.GetAxis
the_y=xChart.GetAxis
the_x.SetTickLabelsVisible(false)
the_x.SetMajorGridVisible(false)
the_y.SetMajorGridVisible(True)
the_x.SetCrossValue(0)
the_y.SetCrossValue(0)
the_x.SetLabelVisible(true)
the_y.SetLabelVisible(true)
xLegend=xChart.GetChartLegend
xLegend.SetVisible(False)
xChart.GetTitle.SetName("Segment "+seg.AsString)
ylst = {"Salinity (ppt)", "DO (mg/L)", "BOD (mg/L)"}
yname = MsgBox.ChoiceAsString(ylst, "Please enter the y-axis", "View Output")
the_x.SetName("Time -- Step = "+step.AsString++"days")
the_y.SetName(yname)
xchart.GetWin.Open
keep = MsgBox.YesNo("Would you like to keep the chart?", "ArcView", true)
if (keep=false) then
av.GetProject.removeDoc(xchart)
exit
else
chname = MsgBox.Input("Name the Chart Window", "View Output", "")
xchart.setname(chname)
end
.....
'Script: calinputa
theProject = av.GetProject
_dir = MsgBox.Input("Enter the working directory.", "Working Directory", "c:\benaman\wasp")
_dir.asFileName.setCWD

```

This program sets the initial model options. Note, that presently, there are a few model options which are preset within the input file. These include:

- '1. Backward differencing used in finite differencing solver
- '2. A transport file is always generated
- '3. The first 6 segments are those solutions which are printed to the screen while the model is running
- '4. The same maximum time step is used throughout the model.
- '5. The same print interval is used throughout the model.
- '6. The model parameters are all steady state at the present time.

To change these settings, one must physically go into the input file which is created in this interface and change the variables. For more information, please consult the WASP5 User's Manual B.

The following Tables are needed:

- ' 1. Main Segmentation (segment arc attribute table)
- ' 2. Main Segment Parameters
- ' 3. Boundary Segments (Parameters of boundary segments)

```

tle = MsgBox.Input("Please write a title for the model run (No more than 60 characters)", "Input File A: Model Options", "Test Run")

```

Choosing the tables that are important

```

segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab
parTable = av.GetProject.FindDoc("Main Segment Parameters")
parVTab = parTable.GetVTab
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab

```

Joining the main segmentation table with its parameters

```

segFields = segVTab.GetFields
parFields = parVTab.GetFields
jtofield = segFields.Get(5)
jfromfield = parFields.Get(0)
segVTab.Join(jtofield, parVTab, jfromfield)

```

Determining the number of segments (main and boundary)

```

_seg = (segVTab.GetNumRecords)+(bcVTab.GetNumRecords)

```

Asking user for other preferences dealing with WASP file

```

icfl = MsgBox.Input("Do you want the model to read or write to a restart file?", "Input File A: Model Options", "N")
if (icfl = "N") then
icfl = 0 else
icfl = 1
end

```

mflag = MsgBox.Input("Do you want all error messages printed to the screen?", "Input File A: Model Options", "Y")

```

if (mflag = "Y") then
mflag = 0 else
mflag = 1
end

```

```

negsln = MsgBox.Input("Do you want to prevent negative solutions?", "Input File A: Model Options", "Y")
if (negsln = "Y") then
  negsln = 0 else
  negsln = 1
end

zlst = {"Day", "Hour", "Minute"}
zdef = {"0", "0", "0"}
z = MsgBox.MultiInput("Please enter the start time.", "Input File A: Model Options", zlst, zdef)

_dts = MsgBox.Input("What is the maximum time step allowed (/day)?", "Input File A: Model Options", "0.001")

_tend = MsgBox.Input("How many days would you like to run the model?", "Input File A: Model Options", "100.0")

_prn = MsgBox.Input("At what interval (in days) would you like the results printed to the output file?", "Input File A: Model Options", "1.0")

mnsegs = segVTab.GetNumRecords

fname=MsgBox.Input("Please enter the filename for the model input file", "Input File A: Model Options", "*.inp")

fnametxt = LineFile.Make("inptnme.txt".AsFileName, #FILE_PERM_WRITE)
fnametxt.WriteElt(fname)
titlefile = LineFile.Make("title.txt".AsFileName, #FILE_PERM_WRITE)
titlefile.WriteElt(tle)
atxt = LineFile.Make("cala.txt".AsFileName, #FILE_PERM_WRITE)
otpta = _seg.AsString++icfl.asString++mflag.asString
otpta = otpta++negsln.asString

for each i in z
  otpta = otpta++i.AsString
end
otpta = otpta++_dts.AsString++_tend.AsString++_prn.AsString++mnsegs.AsString
atxt.WriteElt(otpta)

MsgBox.Info("Done Writing Input Block A", "")
. . . . .

'Script: calinputall
This script runs all ten input block scripts for the calibration input

theProject = av.GetProject

_dummy = 0
aSEd = av.GetProject.FindDoc("calinputa")
aSEd.Run(_dummy)
bSEd = av.GetProject.FindDoc("inputb")
bSEd.Run(_dummy)
cSEd = av.GetProject.FindDoc("inputc")

```

```

cSEd.Run(_dummy)
dSEd = av.GetProject.FindDoc("inputd")
dSEd.Run(_dummy)
eSEd = av.GetProject.FindDoc("inputcale")
eSEd.Run(_dummy)
fSEd = av.GetProject.FindDoc("inputcalf")
fSEd.Run(_dummy)
gSEd = av.GetProject.FindDoc("inputcalg")
gSEd.Run(_dummy)
hSEd = av.GetProject.FindDoc("inputcalh")
hSEd.Run(_dummy)
iSEd = av.GetProject.FindDoc("inputi")
iSEd.Run(_dummy)
jSEd = av.GetProject.FindDoc("inputcalj")
jSEd.Run(_dummy)

. . . . .

'Script: cal_parchk
This script checks the parameters of the calibration output file, before you
run the processing output step

theProject = av.GetProject

_dir = MsgBox.Input("Enter the working directory.", "Working Directory", "c:\benaman\wasp")
_dir.AsFilename.setCWD
segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab
starttxt = LineFile.Make("start.txt".AsFileName, #FILE_PERM_WRITE)

_seg = (segVTab.GetNumRecords)+(bcVTab.GetNumRecords)

_dts = MsgBox.Input("What was the maximum time step allowed (/day)?", "Input File A: Model Options", "0.001")

_tend = MsgBox.Input("How many days did you run the model?", "Input File A: Model Options", "100.0")

_prn = MsgBox.Input("At what interval (in days) did you have the results printed to the output file?", "Input File A: Model Options", "1.0")

outname = MsgBox.Input("Enter the name of the file you want to process", "Calibration Output Processing", "*.tdf")
outtxt = LineFile.Make("outnme.txt".AsFileName, #FILE_PERM_WRITE)
outtxt.WriteElt(outname)

qu = MsgBox.YesNo("Is the output file called"++outname+"?", "Calibration Output Processing", TRUE)
if (qu=TRUE) then
  lst = "Is the following correct?" + nl + "Number of Days Run:" + _tend.AsString + "days" + nl + "Print Interval for output
file:" + _prn.AsString + "days" + nl + "Number of Segments:" + _seg.AsString
  qu2 = MsgBox.YesNo(lst, "Calibration Output Processing", TRUE)
  if (qu2=TRUE) then

```

```

starttxt.WriteElt(_tend.AsString++_dts.AsString++_prn.AsString++_seg.AsString)
else
  MsgBox.Info("Please see the Help for how to set up the output for processing","")
end
else
  MsgBox.Info("Please see the Help for how to set up the output for processing","")
exit
end

```

**Script: eutrorun**

This script executes the WASP model EUTRO5  
 It allows you to choose any input file, but the one  
 generated by this connection will be called "test.inp"

```

_dir.asFileName.setCWD
system.execute("eutro5.exe")
MsgBox.Info("Click OK when EUTRO5 is done running","")

```

**Script: frame1**

This script produces the first frame of the "movie"  
 It is called by, and dependent on, the script, vwout

```
theProject = av.GetProject
```

'getting parameters passed by the main script

```

segSrcName = SELF.Get(0)
concname = SELF.Get(1)
time = SELF.Get(2)
theView = SELF.Get(3)
movVTab = SELF.Get(4)
x = SELF.Get(5)
n = SELF.Get(6)
inv = SELF.Get(7)

```

'Setting the intervals for the classifications of the legend

```

x1 = n
x2 = inv + n
x3 = 2*inv + n
x4 = 3*inv + n
x5 = 4*inv + n
x6 = 5*inv + n
x7 = 6*inv + n
x8 = 7*inv + n
x9 = x

```

'Making the theme

```
newTheme = Theme.Make(segSrcName)
```

```

newTheme.SetVisible(true)
theView.AddTheme(newTheme)
theView.Invalidate
newFTab = newTheme.GetFTab
movFields = movVTab.GetFields

```

'Gets the FTab for the new theme and joins the newly made  
 'temp.dbf to the aat of the theme

```

newFields = newFTab.GetFields
jtoField = newFields.Get(8)
jfromField = movFields.Get(0)
newFTab.Join(jtoField,movVTab.jfromField)

```

'setting the classifications of the legend

```

a = Classification.Make(x1,x2)
b = Classification.Make(x2,x3)
c = Classification.Make(x3,x4)
d = Classification.Make(x4,x5)
e = Classification.Make(x5,x6)
f = Classification.Make(x6,x7)
g = Classification.Make(x7,x8)
h = Classification.Make(x8,x9)

```

```
ClassLst = {a,b,c,d,e,f,g,h}
```

```
newTheme.SetName(concname.AsString++" at "++time.AsString)
```

```

lookField = newFTab.FindField("conc1")
theLegend = newTheme.GetLegend
theLegend.Quantile(newFTab,lookField,8)
theClassList = theLegend.GetClassifications
cnt = 0

```

```

for each i in theClassList
  theClassList.Set(cnt,ClassLst.Get(cnt))
  cnt = cnt + 1
end
newTheme.UpdateLegend

```

'ramping the colors and making it a larger line

```

theLegend.RampColors(Color.GetGray,Color.GetBlue)
c = 0

```

for each i in theLegend.GetSymbols

```

if (c < 4) then
  i.SetWidth(2)
else
  i.SetWidth(3)
end

```

```

c = c+1
end

```

```

newTheme.UpdateLegend
theView.Invalidate

```

.....

**'Script: frame2**  
This script produces the second frame of the "movie"  
It is called by, and dependent on, the script, vwout

theProject = av.GetProject

'getting parameters passed by the main script

segSrcName = SELF.Get(0)  
concname = SELF.Get(1)  
time = SELF.Get(2)  
theView = SELF.Get(3)  
movVTab = SELF.Get(4)  
x = SELF.Get(5)  
n = SELF.Get(6)  
inv = SELF.Get(7)

'Setting the intervals for the classifications of the legend

x1 = n  
x2 = inv + n  
x3 = 2\*inv + n  
x4 = 3\*inv + n  
x5 = 4\*inv + n  
x6 = 5\*inv + n  
x7 = 6\*inv + n  
x8 = 7\*inv + n  
x9 = x

'Making the theme

newTheme = Theme.Make(segSrcName)  
newTheme.SetVisible(true)  
theView.AddTheme(newTheme)  
theView.Invalidate  
newFTab = newTheme.GetFTab  
movFields = movVTab.GetFields

'Gets the FTab for the new theme and joins the newly made  
'temp.dbf to the aat of the theme

newFields = newFTab.GetFields  
jtoField = newFields.Get(8)  
jfromField = movFields.Get(0)  
newFTab.Join(jtoField,movVTab,jfromField)

'setting the classifications of the legend

a = Classification.Make(x1,x2)  
b = Classification.Make(x2,x3)  
c = Classification.Make(x3,x4)  
d = Classification.Make(x4,x5)

e = Classification.Make(x5,x6)  
f = Classification.Make(x6,x7)  
g = Classification.Make(x7,x8)  
h = Classification.Make(x8,x9)

ClassLst = {a,b,c,d,e,f,g,h}  
newTheme.SetName(concname.AsString++" at "+time.AsString)  
lookField = newFTab.FindField("conc2")  
theLegend = newTheme.GetLegend  
theLegend.Quantile(newFTab.lookField,8)  
theClassList = theLegend.GetClassifications  
cnt = 0  
for each i in theClassList  
  theClassList.Set(cnt,ClassLst.Get(cnt))  
  cnt = cnt + 1  
end  
newTheme.UpdateLegend  
'ramping the colors and making it a larger line  
theLegend.RampColors(Color.GetGray,Color.GetBlue)  
c = 0  
for each i in theLegend.GetSymbols  
  if (c < 4) then  
    i.SetWidth(2)  
  else  
    i.SetWidth(3)  
  end  
  c = c + 1  
end  
newTheme.UpdateLegend  
theView.InValidate

.....  
**'Script: frame3**  
This script produces the third frame of the "movie"  
It is called by, and dependent on, the script, vwout

theProject = av.GetProject

'getting parameters passed by the main script

segSrcName = SELF.Get(0)  
concname = SELF.Get(1)  
time = SELF.Get(2)  
theView = SELF.Get(3)  
movVTab = SELF.Get(4)  
x = SELF.Get(5)  
n = SELF.Get(6)  
inv = SELF.Get(7)

'Setting the intervals for the classifications of the legend

```

x1 = n
x2 = inv + n
x3 = 2*inv + n
x4 = 3*inv + n
x5 = 4*inv + n
x6 = 5*inv + n
x7 = 6*inv + n
x8 = 7*inv + n
x9 = x

'Making the theme
newTheme = Theme.Make(segSrcName)
newTheme.SetVisible(true)
theView.AddTheme(newTheme)
theView.Invalidate
newFTab = newTheme.GetFTab
movFields = movVTab.GetFields

'Gets the FTab for the new theme and joins the newly made
'temp.dbf to the aat of the theme

newFields = newFTab.GetFields
jtoField = newFields.Get(8)
jfromField = movFields.Get(0)
newFTab.Join(jtoField,movVTab,jfromField)

'setting the classifications of the legend
a = Classification.Make(x1,x2)
b = Classification.Make(x2,x3)
c = Classification.Make(x3,x4)
d = Classification.Make(x4,x5)
e = Classification.Make(x5,x6)
f = Classification.Make(x6,x7)
g = Classification.Make(x7,x8)
h = Classification.Make(x8,x9)

ClassLst = {a,b,c,d,e,f,g,h}
newTheme.SetName(concname.AsString++" at "+time.AsString)
lookField = newFTab.FindField("conc3")
theLegend = newTheme.GetLegend
theLegend.Quantile(newFTab,lookField,8)
theClassList = theLegend.GetClassifications
cnt = 0
for each i in theClassList
  theClassList.Set(cnt,ClassLst.Get(cnt))
  cnt = cnt + 1
end
newTheme.UpdateLegend
ramping the colors and making it a larger line
theLegend.RampColors(Color.GetGray,Color.GetBlue)

```

```

c = 0
for each i in theLegend.GetSymbols
  if (c < 4) then
    i.SetWidth(2)
  else
    i.SetWidth(3)
  end
  c = c+1
end
newTheme.UpdateLegend
theView.InValidate

.....

'Script: frame4
'This script produces the fourth frame of the "movie"
'It is called by, and dependent on, the script, vwout

theProject = av.GetProject

'getting parameters passed by the main script
segSrcName = SELF.Get(0)
concname = SELF.Get(1)
time = SELF.Get(2)
theView = SELF.Get(3)
movVTab = SELF.Get(4)
x = SELF.Get(5)
n = SELF.Get(6)
inv = SELF.Get(7)

'Setting the intervals for the classifications of the legend

x1 = n
x2 = inv + n
x3 = 2*inv + n
x4 = 3*inv + n
x5 = 4*inv + n
x6 = 5*inv + n
x7 = 6*inv + n
x8 = 7*inv + n
x9 = x

'Making the theme
newTheme = Theme.Make(segSrcName)
newTheme.SetVisible(true)
theView.AddTheme(newTheme)
theView.Invalidate
newFTab = newTheme.GetFTab
movFields = movVTab.GetFields

'Gets the FTab for the new theme and joins the newly made

```

'temp.dbf to the aat of the theme

```
newFields = newFTab.GetFields
jtoField = newFields.Get(8)
jfromField = movFields.Get(0)
newFTab.Join(jtoField,movVTab,jfromField)
```

'setting the classifications of the legend

```
a = Classification.Make(x1,x2)
b = Classification.Make(x2,x3)
c = Classification.Make(x3,x4)
d = Classification.Make(x4,x5)
e = Classification.Make(x5,x6)
f = Classification.Make(x6,x7)
g = Classification.Make(x7,x8)
h = Classification.Make(x8,x9)
```

```
ClassLst = {a,b,c,d,e,f,g,h}
newTheme.SetName(concname.AsString++" at "++time.AsString)
lookField = newFTab.FindField("conc4")
theLegend = newTheme.GetLegend
theLegend.Quantile(newFTab,lookField,8)
theClassList = theLegend.GetClassifications
cnt = 0
for each i in theClassList
  theClassList.Set(cnt,theClassLst.Get(cnt))
  cnt = cnt + 1
end
newTheme.UpdateLegend
```

'ramping the colors and making it a larger line

```
theLegend.RampColors(Color.GetGray,Color.GetBlue)
c = 0
for each i in theLegend.GetSymbols
  if (c < 4) then
    i.SetWidth(2)
  else
    i.SetWidth(3)
  end
  c = c + 1
end
newTheme.UpdateLegend
theView.InValidate
```

**'Script: gen\_file\_eutro**

This script executes the FORTRAN program which takes all 12  
'of the text files created from Scripts inputa through  
'inputj and formats them into one large input file for EUTRO  
The input file is ALWAYS called test.inp

```
_dir.asFileName.setCWD
system.execute("outgen.exe")
MsgBox.Info("Done generating WASP5 input file", "")
```

**'Script: gen\_file\_toxi**

theProject = av.GetActiveDoc  
'This script executes the FORTRAN program which takes all 12  
'of the text files created from Scripts inputa through  
'inputj and formats them into one large input file for TOXI  
'for the model calibration

```
_dir.asFileName.setCWD
system.execute("calgen.exe")
MsgBox.Info("Done generating WASP5 input file", "")
```

**'Script: help**

'this script executes the help file, written for winhelp.exe

```
system.execute("winhelp.exe c:\benaman\winhelp\wasptm.hlp")
```

**'Script: inputa**

```
theProject = av.GetProject
_dir = MsgBox.Input("Enter the working directory.", "Working Directory", "c:\benaman\wasp")
_dir.asFileName.setCWD
```

'This program sets the initial model options. Note, that presently, there are  
'a few model options which are preset within the input file. These include:

- '1. Backward differencing used in finite differencing solver
- '2. A transport file is always generated
- '3. The first 6 segments are those solutions which are printed to the screen while the model is running
- '4. The same maximum time step is used throughout the model.
- '5. The same print interval is used throughout the model.
- '6. The model parameters are all steady state at the present time.

'To change these settings, one must physically go into the input file which is created in this interface  
'and change the variables. For more information, please consult the WASP5 User's Manual B.

'The following Tables are needed:

- ' 1. Main Segmentation (segment arc attribute table)
- ' 2. Main Segment Parameters
- ' 3. Boundary Segments (Parameters of boundary segments)

```
mdlList = {"Simple Streeter-Phelps with SOD","Modified Streeter Phelps with NBOD","Linear DO Balance with  
Nitrification","Nonlinear DO Balance","Simple Eutrophication","Intermediate Eutrophication","Intermediate  
Eutrophication with Benthos" }
```

```
for each i in 1..6
```

```
mdl = MsgBox.ChoiceAsString(mdlList,"What type of model would you like to run?","Input File A: Model Options")
```

```

_imdl=mdlList.Find(mdl)
if (_imdl > 0) then
  MsgBox.Info("Unable to run that Level at this time.", "Sorry")
else
  break
end
end
tle = MsgBox.Input("Please write a title for the model run (No more than 60 charaters)", "Input File A: Model Options", "Test Run")

'Choosing the tables that are important
segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab
parTable = av.GetProject.FindDoc("Main Segment Parameters")
parVTab = parTable.GetVTab
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab

'Joining the main segmentation table with its parameters

segFields = segVTab.GetFields
parFields = parVTab.GetFields
jtofield = segFields.Get(5)
jfromfield = parFields.Get(0)
segVTab.Join(jtofield, parVTab, jfromfield)

'Determining the number of segments (main and boundary)
seg = (segVTab.GetNumRecords)+(bcVTab.GetNumRecords)

'Asking for the number of systems (i.e. constiuents to be modeled -- default for EUTRO is 8)
for each i in 1..3
  sys = MsgBox.Input("Please enter the number of systems.", "Input File A: Model Options", "8")
  sys = sys.AsNumber
  if (sys = 8) then
    break
  else
    MsgBox.Info("EUTRO5 always requires 8 systems", "")
    continue
  end
end
'Asking user for other preferences dealiing with WASP file
icfl = MsgBox.Input("Do you want the model to read or write to a restart file?", "Input File A: Model Options", "N")
if (icfl = "N") then
  icfl = 0 else
  icfl = 1
end

mflag = MsgBox.Input("Do you want all error messages printed to the screen?", "Input File A: Model Options", "Y")
if (mflag = "Y") then
  mflag = 0 else

```

```

  mflag = 1
end

massList = {"NH3", "NO3", "PO4", "Chla", "CBOD", "DO", "ON", "OP"}
massys = MsgBox.ChoiceAsString(massList, "Choose the system for which the mass balance will be performed", "Input File A: Model Options")
imassys = massList.Find(massys)

negsln = MsgBox.Input("Do you want to prevent negative solutions?", "Input File A: Model Options", "Y")
if (negsln = "Y") then
  negsln = 0 else
  negsln = 1
end

zlst = {"Day", "Hour", "Minute"}
zdef = {"0", "0", "0"}
z = MsgBox.MultiInput("Please enter the start time.", "Input File A: Model Options", zlst, zdef)

_dts = MsgBox.Input("What is the maximum time step allowed (/day)?", "Input File A: Model Options", "0.001")

_tend = MsgBox.Input("How many days would you like to run the model?", "Input File A: Model Options", "100.0")

_prn = MsgBox.Input("At what interval (in days) would you like the results printed to the output file?", "Input File A: Model Options", "1.0")

mnsegs = segVTab.GetNumRecords
fname=MsgBox.Input("Please enter the filename for the model input file", "Input File A: Model Options", "*.inp")

flnmetxt = LineFile.Make("inptnme.txt".AsFileName, #FILE_PERM_WRITE)
flnmetxt.WriteElt(fname)
titlefile = LineFile.Make("title.txt".AsFileName, #FILE_PERM_WRITE)
titlefile.WriteElt(tle)
atxt = LineFile.Make("a.txt".AsFileName, #FILE_PERM_WRITE)
opta = _imdl.AsString++seg.AsString++sys.AsString++icfl.AsString++mflag.AsString++imassys.AsString
opta = opta++negsln.AsString

for each i in z
  opta = opta++i.AsString
end
opta = opta+_dts.AsString+_tend.AsString+_prn.AsString+mnsegs.AsString
atxt.WriteElt(opta)

MsgBox.Info("Done Writing Input Block A.", "")

. . . . .

'Script: inputb
'This script generates the free format text file for Input Block B
'The following defaults are set:
' 1. All exchanges are steady state
'Tables needed: Same as inputa

```

```

theProject = av.GetProject
_dir.asFileName.setCWD
'_imdl = the type of model being run

btxt = LineFile.Make("b.txt".AsFileName,#FILE_PERM_WRITE)

'get the initial model information

exList = {"Water Column Only","Water Column and Pore Water"}
nrfld = MsgBox.ChoiceAsString(exList,"What fields undergo exchange?","Input File B: Exchange Coefficients")
inrfld=exList.Find(nrfld)
if (inrfld = 0) then
  nrfld = 1 else
  nrfld = 2
end
if (_imdl=nil) then
  _imdl = 0
end

lst = _imdl.AsString++nrfld.AsString++"0"+"0"+"0"
btxt.WriteElt(lst)

'identify the two tables that have the segment information in them

segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab

'Now concentrate on the main segments first: Get the exchange coefficients,
'characteristic length, and neighboring segment number

srt1Field = segTable.GetVTab.FindField("Grid-Code")
if (segTable.GetWin.IsOpen) then
  segTable.Sort( srt1Field, FALSE )
end

gcField = segVTab.FindField("grid-code")
dwnField = segVTab.FindField("dwnstr_seg")
araField = segVTab.FindField("x_area")
exField = segVTab.FindField("ex_coeff")
lnField = segVTab.FindField("length")

For Each rec in segVTab
  seg = segVTab.ReturnValue(gcField,rec)
  dwnstr = segVTab.ReturnValue(dwnField,rec)
  area1 = segVTab.ReturnValue(araField,rec)
  ex1 = segVTab.ReturnValue(exField,rec)
  length1=segVTab.ReturnValue(lnField,rec)

```

Finding the same values for the downstream segment

```

For Each reca in segVTab
  dwnseg = segVTab.ReturnValue(gcField,reca)
  if (dwnseg = dwnstr) then
    tmprec = reca.AsString
    itmprec = tmprec.AsNumber
    break
  else
    tmprec = "999"
    itmprec = tmprec.AsNumber
  end
end

if (itmprec = 999) then
  MsgBox.Info("The segment downstream of segment"++seg.AsString++"is a boundary segment.", "")
else
  area2 = segVTab.ReturnValue(araField,itmprec)
  ex2 = segVTab.ReturnValue(exField,itmprec)
  length2=segVTab.ReturnValue(lnField,itmprec)

'comparison of the sgement values and its downstream segment values
if (area1 > area2) then
  area = area2
else
  area = area1
end
if (ex1 > ex2) then
  ex = ex1
else
  ex = ex2
end
chlth = (length1 + length2)/2
'writing the information to the file
exclst = ex.AsString++area.AsString++chlth.AsString++seg.AsString++dwnstr.AsString
btxt.WriteElt(exclst)
end
end

```

'Now concentrate on the boundary segments

```

bcVTab = bcTable.GetVTab
perField = bcVTab.FindField("perpin")
bcgcField = bcVTab.FindField("grid-code")
bcdwnField = bcVTab.FindField("dwnstr_seg")
upField = bcVTab.FindField("upstr_seg")
bcaraField = bcVTab.FindField("x_area")
bcexField = bcVTab.FindField("ex_coeff")
bclnField = bcVTab.FindField("act_length")

```



```
wdField = segVTab.FindField("width")
typField = bcVTab.FindField("type")
```

```
srt1Field = bcTable.GetVTab.FindField("Grid-Code")
if (bcTable.GetWin.IsOpen) then
  bcTable.Sort( srt1Field, FALSE )
end
```

```
For Each rec in bcVTab
  seg = bcVTab.ReturnValue(bcgcField,rec)
  dwnstr = bcVTab.ReturnValue(bcdwnField,rec)
  type = bcVTab.ReturnValue(typField,rec)
  if (type <> 1) then
    break
  end
  if (dwnstr = 0) then
    'allows the most downstream segment to look at its upstream segment
    dwnstr = bcVTab.ReturnValue(upField,rec)
    'assumes the most downstream segment has a dispersive boundary with 0
    lastseg = seg
    lastex = bcVTab.ReturnValue(bcexField,rec)
    lastarea = bcVTab.ReturnValue(bcaraField,rec)
    lastchlth = (bcVTab.ReturnValue(bcInField,rec)/2)
  end
  area1 = bcVTab.ReturnValue(bcaraField,rec)
  ex1 = bcVTab.ReturnValue(bcexField,rec)
  length1 = bcVTab.ReturnValue(bcInField,rec)
  pi = bcVTab.ReturnValue(perField,rec)
  p = pi.AsString
```

Finding the same values for the downstream segment

```
For Each reca in segVTab
  dwnseg = segVTab.ReturnValue(gcField,reca)
  if (dwnseg = dwnstr) then
    tmprec = reca.AsString
    itmprec = tmprec.AsNumber
    break
  else
    tmprec = "999"
    itmprec = tmprec.AsNumber
  end
end
```

```
if (itmprec = 999) then
  dummy = 0
else
  area2 = segVTab.ReturnValue(araField,itmprec)
  length2 = segVTab.ReturnValue(lnField,itmprec)
  wdth = segVTab.ReturnValue(wdField,itmprec)
```

```
ex2 = segVTab.ReturnValue(exField,itmprec)
end
```

'comparison of the segegment values and its downstream segment values

```
if (area1 > area2) then
  area = area2
else
  area = area1
end
```

'checks to see if the boundary is perpindicular

```
if (p = "N")
  then
  chlth = (length1 + length2)/2
  if (ex1 > ex2) then
    ex = ex2
  else
    ex = ex1
  end
else
  chlth = (wdth + length1)/2
  area = area1
  ex = ex1
end
```

'writing the information to the file

```
if (ex=0) then
  break
end
bcexclst = ex.AsString++area.AsString++chlth.AsString++seg.AsString++dwnstr.AsString
btxt.WriteElt(bcexclst)
```

end

```
bcexlst = lastex.AsString++lastarea.AsString++lastchlth.AsString++lastseg.AsString++"0"
btxt.WriteElt(bcexlst)
```

'Now look at the exchange between the sediment layer and the overlying water body

```
bcdpField = bcVTab.FindField("depth")
'Find the record the sediment layer is in
For each rec in bcVTab
  type = bcVTab.ReturnValue(typField,rec)
  if (type = 3) then
    sedex = bcVTab.ReturnValue(bcexField,rec)
    sedgc = bcVTab.ReturnValue(bcgcField,rec)
    seddth = bcVTab.ReturnValue(bcdpField,rec)
  end
end
```

```
dpField = segVTab.FindField("depth")
```

```

For each rec in segVTab
  area = (segVTab.ReturnValue(lnField,rec))*(segVTab.ReturnValue(wdField,rec))
  dp = segVTab.ReturnValue(dpField,rec)
  chln = (dp+seddth)/2
  seg = segVTab.ReturnValue(gcField,rec)
  bcexlst = sedex.AsString++area.AsString++chln.AsString++seg.AsString++sedgc.AsString
  btxt.WriteElt(bcexlst)
end

btxt.WriteElt("555")
MsgBox.Info("Done writing Input Block B", "")

. . . . .

Script: inputc
This script writes the free form text file used to create
Input Block C
The following defaults are set in this file:
' 1. The hydraulic coefficients used, in WASP, to calculate volatilization
' and reaeration do not spatially vary
Tables Needed: Same as inputa

theProject = av.GetProject
_dir.asFileName.setCWD

cctx = LineFile.Make("c.txt".AsFileName,#FILE_PERM_WRITE)

'get the initial model information

volList = {"Constant Water Column Volume", "Volumes Adjusted to Maintain Flow Continuity"}
vopt= MsgBox.ChoiceAsString(volList, "Choose a water column volume option.", "Input File C: Volumes")
ivopt=(volList.Find(vopt))+1

for each i in 1..2
  bedList = {"Constant Bed Volume", "Volumes Adjusted to Respond to Sediment Transport"}
  bedv= MsgBox.ChoiceAsString(bedList, "Choose a benthic volume option", "Input File C: Volumes")
  ibedv=(bedList.Find(bedv))
  if (ibedv < 0) then
    MsgBox.Info("Sediment transport is currently not simulated in this model", "Input File C: Volumes")
  else
    break
  end
end
end

tdints = MsgBox.Input("Please enter the benthic time step for recomputing porosity (/day).", "Input File
C: Volumes", "")

lst = ivopt.AsString++ibedv.AsString++tdints.AsString++"0"
cctx.WriteElt(lst)

```

```

'get the hydraulic coefficients

hydlist = {"a", "b", "c", "d"}
hyddef = {"0.004", "0.4", "1.2", "0.6"}

hydcoef = MsgBox.MultiInput("Enter the coef. for the eqns: v=aQ^b and d=Q^c", "Input File
C: Volumes", hydlist, hyddef)

for each i in hydcoef
  cctx.WriteElt(i.AsString++"0"+"0"+"0")
end

'identify the two tables that have the segment information in them

segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab

gcField = segVTab.FindField("grid-code")
lnField = segVTab.FindField("length")
araField = segVTab.FindField("x_area")
botField = segVTab.FindField("btm_seg")
typField = segVTab.FindField("type")

'Get the volume, segment type, and bottom segment for each segment
First the main segments, then the boundary segments

for each rec in segVTab
  seg = segVTab.ReturnValue(gcField,rec)
  len = segVTab.ReturnValue(lnField,rec)
  area = segVTab.ReturnValue(araField,rec)
  botseg = segVTab.ReturnValue(botField,rec)
  type = segVTab.ReturnValue(typField,rec)

  volume = len * area

  lst = seg.AsString++botseg.AsString++type.AsString++volume.AsString
  cctx.WriteElt(lst)
end

gcField = bcVTab.FindField("grid-code")
lnField = bcVTab.FindField("act_length")
araField = bcVTab.FindField("x_area")
botField = bcVTab.FindField("btm_seg")
typField = bcVTab.FindField("type")

for each rec in bcVTab
  seg = bcVTab.ReturnValue(gcField,rec)
  len = bcVTab.ReturnValue(lnField,rec)

```

```

area = bcVTab.ReturnValue(araField,rec)
botseg = bcVTab.ReturnValue(botField,rec)
type = bcVTab.ReturnValue(typField,rec)

volume = len * area

lst = seg.AsString++botseg.AsString++type.AsString++volume.AsString
ctxt.WriteElt(lst)
end

MsgBox.Info("Done writing Input Block C", "")

. . . . .

'Script: inputcale
This script writes the input block e for
the calibration file
Tables needed:
1. Boundary Segments

theProject = av.GetProject
_dir.asFileName.setCWD

etxt = LineFile.Make("cale.txt".AsFileName,#FILE_PERM_WRITE)
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab

numrecs = bcVTab.GetNumRecords
typField = bcVTab.FindField("type")
gcField = bcVTab.FindField("grid-code")
salField = bcVTab.FindField("sal")

'Find the number of water segments
for each rec in bcVTab
  type = bcVTab.ReturnValue(typField, rec)
  if (type = 3) then
    numrecs = numrecs - 1
  end
end

etxt.WriteElt(numrecs.AsString+"0")

lst = list.Make
'get the salinity conc for each boundary
nbc = MsgBox.Input("How many boundaries are you going to set", "Input File E: Boundaries", "2")
nbc = nbc.AsNumber
For Each i in 1..nbc
  x =MsgBox.Input("Enter the segment number", "Input File J: Initial Conditions", "")
'obtain the salinity initial conditions and write them
  for each rec in bcVTab
    seg = bcVTab.ReturnValue(gcField,rec)

```

```

sal = bcVTab.ReturnValue(salField,rec)
if (seg = x.AsNumber) then
  lst.Add(rec)
  etxt.WriteElt(seg.AsString++sal.AsString)
  break
else
  continue
end
end

temp = 555
for each rec in bcVTab
  seg = bcVTab.ReturnValue(gcField,rec)
  sal = bcVTab.ReturnValue(salField,rec)
  type = bcVTab.ReturnValue(typField,rec)
  if (type = 3)
  then
  break
end
  for each i in lst
    if (i = rec) then
      temp = 999
      break
    else
      temp = 555
    end
  end
  if (temp = 999) then
  continue
else
  etxt.WriteElt(seg.AsString+"0.2")
end
end

. . . . .

'Script: inputcalf
This script writes the input block f, for
the calibration file
'Presently, there are no salinity loads into the
'system

theProject = av.GetProject
_dir.asFileName.setCWD

fxt = LineFile.Make("calf.txt".AsFileName,#FILE_PERM_WRITE)
fxt.WriteElt("0")
MsgBox.Info("There are no loads of salinity for the calibration input file.", "Input File F: Loads")

'Script:inputcalfg

```

```
'this script writes the input block g for the
'calibration file, presently, there are no
'parameters needed for the level one complexity
'of TOX15

theProject = av.GetProject
_dir.asFileName.setCWD

gtxt = LineFile.Make("calg.txt".AsFileName,#FILE_PERM_WRITE)
gtxt.WriteElt("0")
MsgBox.Info("There are no parameters need for Level One Calibration", "Input File G: Parameters")

. . . . .

'Script: inputcalh
'This script writes the input block h for the
'calibration output file
'Tables needed: NONE

theProject = av.GetProject
_dir.asFileName.setCWD

htxt = LineFile.Make("calh.txt".AsFileName,#FILE_PERM_WRITE)

'Get the constants Kd and Ka from the user and write them to a file

lst = {"Solids-Independent Partition Coefficient(L/kg)", "Water Column Biodegradation", "Molecular Weight(g/mol)"}
lstdef = {"0.0", "0.0", "78.5"}
cnst = MsgBox.MultiInput("Please Enter the Following Constants", "Input File H: Constants", lst, lstdef)

for each i in cnst
    htxt.WriteElt(i.AsString)
end

MsgBox.Info("Done writing Input Block H", "")

. . . . .

'Script: inputcalj
'This script writes the input block j
'for the calibration input file
'Tables Needed:
'1.Main Segmentation
'2.Boundary Segmentation

theProject = av.GetProject
_dir.asFileName.setCWD

jtxt = LineFile.Make("calj.txt".AsFileName,#FILE_PERM_WRITE)

'get the necessary tables
segTable = av.GetProject.FindDoc("Main Segmentation")
```

```
segVTab = segTable.GetVTab
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab

bcgcField = bcVTab.FindField("grid-code")
bcсалField = bcVTab.FindField("sal")

nbc = MsgBox.Input("How many boundaries are you going to set", "Input File J: Initial Conditions", "2")
nbc = nbc.AsNumber
jtxt.WriteElt("1.0"+"0") 'writing the dissolved fraction
jtxt.WriteElt(nbc.AsString+"0") 'writing the number of boundary conditions
For Each i in 1..nbc
    x =MsgBox.Input("Enter the segment number", "Input File J: Initial Conditions", "")
'obtain the salinity initial conditions and write them
    for each rec in bcVTab
        seg = bcVTab.ReturnValue(bcgcField,rec)
        sal = bcVTab.ReturnValue(bcсалField,rec)
        if (seg = x.AsNumber) then
            jtxt.WriteElt(seg.AsString++sal.AsString)
            break
        else
            continue
        end
    end
end

MsgBox.Info("Done Writing Input Block J", "")

. . . . .

'Script: inputd
'This script generates the free form text file used to create
'Input Block D
'The following defaults are set in this script:
' 1. Presently, this program only considers Field 1 flows -- it does not
'    account for second water column flow or pore water flow
' 2. The flow is steady state
'Tables needed: Same as inputa and Flow Accumulation Values

theProject = av.GetProject
_dir.asFileName.setCWD

dinptxt = LineFile.Make("d.txt".AsFileName,#FILE_PERM_WRITE)
dyntxt = LineFile.Make("dyn.txt".AsFileName,#FILE_PERM_WRITE)

'Get the flow option desired
lst = {"Flow Option One", "Flow Option Two", "Flow Option Three"}
iqpt = MsgBox.ChoiceAsString(lst, "Choose your flow option (see Help for explanation)",
    "Input File D: Flows")
iiqpt = (lst.Find(iqpt))+1
if (iiqpt = 3) then
```

```

ttitle = MsgBox.Input("Enter the name of the DYNHYD File to be read", "Input File D: Flows",
"*.*HYD")
else
ttitle = "nothing"
end
dyntxt.WriteElt(ttitle)
dinptxt.WriteElt(iiqpt.asString++"0")
'Ask if they are going to simulate dry weather conditions

_drywthr = MsgBox.MiniYesNo("Do you want to simulate dry weather conditions?",TRUE)

'identify the two tables that have the segment information in them
segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab
flTable = av.GetProject.FindDoc("Flow Accumulation Values")
flVTab = flTable.GetVTab
runTable = av.GetProject.FindDoc("Runoff Accumulation Values")
runVTab = runTable.GetVTab

srt1Field = flTable.GetVTab.FindField("grid-code")
if (flTable.GetWin.IsOpen) then
flTable.Sort( srt1Field, FALSE )
end

'Create a new table from the Flow Accumulation values which
holds the segment flow in m^3/sec

nwflVTab=VTab.MakeNew("flow.dbf".asfilename,dBase)
flList=List.Make
flList.Add(Field.Make("grid-code",#FIELD_BYTE,4,0))
flList.Add(Field.Make("cumm_flow",#FIELD_FLOAT,10,2))
flList.Add(Field.Make("int_flow",#FIELD_FLOAT,10,2))
flList.Add(Field.Make("runoff",#FIELD_FLOAT,10,2))
flList.Add(Field.Make("baseflow",#FIELD_FLOAT,10,2))
if (nwflVTab.CanEdit)then
nwflVTab.SetEditable(true)
else
MsgBox.Info("Can't Edit the Table", "Exit")
exit
end

flListclone = flList.DeepClone
nwflVTab.AddFields(flListclone)
nwflFields = nwflVTab.getFields
nwflTable = Table.Make(nwflVTab)
nwflTable.SetName(nwflVTab.GetName)

```

```

gcField = flVTab.FindField("grid-code")
facField = flVTab.FindField("flow accumulation")
nwgcField = nwflVTab.FindField("grid-code")
nwfacField = nwflVTab.FindField("cumm_flow")
nwintflField = nwflVTab.FindField("int_flow")
newrunField = nwflVTab.FindField("runoff")
newbflField = nwflVTab.FindField("baseflow")

' add the cumulative flow in m^3/sec to the new table
nrec = 0
For Each rec in flVTab
seg = flVTab.ReturnValue(gcField,rec)
fac = flVTab.ReturnValue(facField,rec)
flw = fac*(0.00000317098) 'Conversion of mm/yr-cell to m^3/sec
nwflVTab.AddRecord
nwflVTab.SetValue(nwgcField,rec,seg)
nwflVTab.SetValue(nwfacField,rec,flw)
nrec = nrec + 1
end

'determine the incremental flow and add it to the new
table
temp = 0
s = 1
For each i in 1..nrec
For each rec in nwflVTab
seg = nwflVTab.ReturnValue(nwgcField,rec)
flw = nwflVTab.ReturnValue(nwfacField,rec)
if (seg = s) then
int_flw = flw - temp
s = s+1
temp = flw
nwflVTab.SetValue(nwintflField,rec,int_flw)
end
end
end

'determine the incremental runoff and add that to the new table
runcgField = runVTab.FindField("grid-code")
racField = runVTab.FindField("rfac")

temp = 0
s = 1
for each i in 1..nrec
For each rec in runVTab
seg = runVTab.ReturnValue(runcgField,rec)
rac = runVTab.ReturnValue(racField,rec)
if (seg = s) then
runoff = (rac-temp)*(0.00000317098)

```

```

s = s+1
temp = rac
nwflVTab.SetValue(newrunField,rec,runoff)
end
end
end

'determine the baseflow into each segment from the total flow
'minus the runoff

For each rec in nwflVTab
flow = nwflVTab.ReturnValue(nwintflField,rec)
runoff = nwflVTab.ReturnValue(newrunField,rec)
baseflow = flow - runoff
nwflVTab.SetValue(newbfField,rec,baseflow)
end

'join the tables (the new flow table with the main
'segmentation table)
segFields = segVTab.GetFields
jtofield = segFields.Get(5)
jfromfield = nwflFields.Get(0)
segVTab.Join(jtofield,nwflVTab,jfromfield)

typField = bcVTab.FindField("type")
bcdwnField = bcVTab.FindField("dwnstr_seg")
bcgcField = bcVTab.FindField("grid-code")
bcflField = bcVTab.FindField("flow")
bcupField = bcVTab.FindField("upstr_seg")

mngcField = segVTab.FindField("grid-code")
dwnField = segVTab.FindField("dwnstr_seg")
upField = segVTab.FindField("upstr_seg")
inflowField = segVTab.FindField("int_flow")

if (_drywthr = TRUE) then
wrtField = segVTab.FindField("baseflow")
else
wrtField = segVTab.FindField("int_flow")
end

numrecs = segVTab.GetNumRecords
temp = 0
ultdwnstr = 0
For each rec in bcVTab
seg = bcVTab.ReturnValue(bcgcField,rec)
dwnstrseg = bcVTab.ReturnValue(bcdwnField,rec)

```

```

if (dwnstrseg = 0) then
temp = rec
ultdwnstr = seg
break
end
upstrseg = bcVTab.ReturnValue(bcupField,rec)
dinptxt.WriteElt(upstrseg.AsString++seg.AsString)
dinptxt.WriteElt(seg.AsString++dwnstrseg.AsString)
'Find the corresponding flow for this input
For each flrec in segVTab
newseg = segVTab.ReturnValue(mngcField,flrec)
if (newseg = dwnstrseg) then
flow = segVTab.ReturnValue(wrtField,flrec)
else
continue
end
end
'Find the flow path segment to downstream segment and so on
For each i in 1..numrecs
For each n in segVTab
seg = segVTab.ReturnValue(mngcField,n)
if (seg = dwnstrseg) then
dwnstrseg = segVTab.ReturnValue(dwnField,n)
dinptxt.WriteElt(seg.AsString++dwnstrseg.AsString)
else
continue
end
end
dinptxt.WriteElt(dwnstrseg.AsString++"0")
dinptxt.WriteElt(flow.AsString++"999")
end

'Now, take care of the very last segment
flow = bcVTab.ReturnValue(bcflField,temp)
if (flow <> 0) then
dinptxt.WriteElt("0"++ultdwnstr.AsString)
dinptxt.WriteElt(ultdwnstr.AsString++"0")
dinptxt.WriteElt(flow.AsString++"999")
end

dinptxt.WriteElt("555"++"0")
MsgBox.Info("Done writing Input Block D","")

```

```

.....
'Script: inpute
'This script generates the free form text file needed to complete
'the Input Block E in EUTRO
'The following defaults are set:
' 1. Since, presently, only Simple Streeter Phelps is possible,

```

```

' only BOD and DO are considered in the b.c.
' 2. The b.c.'s are steady state
'Tables needed: same as inputa

theProject = av.GetProject
_dir.asFileName.setCWD

etxt = LineFile.Make("e.txt".AsFileName,#FILE_PERM_WRITE)

if (_imdl = 0)
then
nh3 = 0
n03 = 0
po4 = 0
chla = 0
on = 0
op = 0
end

bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab

numrecs = bcVTab.GetNumRecords
typField = bcVTab.FindField("type")
gcField = bcVTab.FindField("grid-code")
doField = bcVTab.FindField("int_do")
bodField = bcVTab.FindField("int_bod")

'Determine the boundary conditions which are water boundary segments

for each rec in bcVTab
type = bcVTab.ReturnValue(typField, rec)
if (type = 3) then
numrecs = numrecs - 1
end
end

etxt.WriteElt(numrecs.asString++"0")

'get the BOD conc for each boundary

for each rec in bcVTab
seg = bcVTab.ReturnValue(gcField,rec)
bod = bcVTab.ReturnValue(bodField,rec)
type = bcVTab.ReturnValue(typField,rec)
if (type = 3)
then
break
end
etxt.WriteElt(seg.asString++bod.asString)

```

```

end

'get the DO for each boundary

for each rec in bcVTab
seg = bcVTab.ReturnValue(gcField,rec)
do = bcVTab.ReturnValue(doField,rec)
type = bcVTab.ReturnValue(typField,rec)
if (type = 3)
then
break
end
etxt.WriteElt(seg.asString++do.asString)
end

MsgBox.Info("Done writing Input Block E", "")

.....

'Script: inputf
'This script creates the free form text file needed to generate
'Input Block F
'The following defaults are set:
' 1. All point and non-point sources are steady state
' 2. Since presently, the loads are steady state, the nps
' loads are added to the point source loads and written
' in the main input block; if unsteady nps loads are
' determined, a new file must be created, separate from
' the main input file in WASP -- note that this fact would
' entail some changes in the FORTRAN code which formats the
' input file (ougen.for)

'Tables Needed: same as inputa and Runoff Accumulation Values,
' Point Source BOD, and BOD Loading Values

if (_drywthr = nil) then
_drywthr = MsgBox.MiniYesNo("Do you want to simulate dry weather conditions?",TRUE)
end

theProject = av.GetProject
_dir.asFileName.setCWD

ftxt = LineFile.Make("f.txt".AsFileName,#FILE_PERM_WRITE)

'identify the two tables that have the segment information in them
psTable = av.GetProject.FindDoc("Point Source BOD")
psVTab = psTable.GetVTab
npsTable = av.GetProject.FindDoc("BOD Loading Values")
npsVTab = npsTable.GetVTab
runTable = av.GetProject.FindDoc("Runoff Accumulation Values")
runVTab = runTable.GetVTab

```

```

'Create a Table which will hold the Loadings in kg/day
ldVTab=VTab.MakeNew("load.dbf".asfilename,dBase)
ldList=List.Make
ldList.Add(Field.Make("grid-code",#FIELD_BYTE,4,0))
ldList.Add(Field.Make("bod_ps",#FIELD_FLOAT,10,2))
ldList.Add(Field.Make("bod_nps",#FIELD_FLOAT,10,2))

if (ldVTab.CanEdit)then
  ldVTab.SetEditable(true)
else
  MsgBox.Info("Can't Edit the Table","Exit")
  exit
end

ldListclone = ldList.DeepClone
ldVTab.AddFields(ldListclone)
ldFields = ldVTab.getFields
ldTable = Table.Make(ldVTab)
ldTable.SetName(ldVTab.GetName)
ldgcField = ldVTab.FindField("grid-code")
psField = ldVTab.FindField("bod_ps")
bodnpsField = ldVTab.FindField("bod_nps")

'Find the number of main segments that get loading and
'write it to the file

numrecs = npsVTab.GetNumRecords
fxtt.WriteElt(numrecs.asString+"0")

'Find the loadings from each table:
' 1. BOD Point Sources
' 2. BOD NPS from BOD Flow Accumulation
'Write these loads to the table "load.dbf" and
'to the file f.txt

gcField = psVTab.FindField("segment")
lbspsField = psVTab.FindField("bod")
runField = runVTab.FindField("rfac")
bodField = npsVTab.FindField("accumulated bod loading")

for each rec in psVTab
  seg = psVTab.ReturnValue(gcField,rec)
  load = psVTab.ReturnValue(lbpsField,rec)
  load = load * 1.243
  fxtt.WriteElt(seg.asString++load.asString)
  ldVTab.AddRecord
  ldVTab.SetValue(ldgcField,rec,seg)
  ldVTab.SetValue(psField,rec,load)
end

```

```

bodlst = List.Make
for each rec in npsVTab
  load = npsVTab.ReturnValue(bodField,rec)
  bodlst.Add(load)
end
bodlst.Sort(TRUE)

p = numrecs+1
n = 0
s = 1
temp = 0
pre = 0
for each i in bodlst
  temp = bodlst.Get(n)
  load = temp - pre
  load = load/365
  if (_drywthr = TRUE) then
    fxtt.WriteElt(s.AsString++"0")
  else
    fxtt.WriteElt(s.AsString++load.AsString)
  end
  for each i in 1..p
    for each rec in ldVTab
      x = ldVTab.ReturnValue(ldgcField,rec)
      if (x=s) then
        ldVTab.SetValue(bodnpsField,rec,load)
        break
      end
    end
  end
  pre = temp
  n = n+1
  s = s+1
end

```

```

MsgBox.Info("Done writing Input Block F","")

```

```

.....
'Script: inputg
'This script creates a the free form text file used to generate
'the input block G
'The following defaults are set in this file:
' 1. Theta does not spatially vary
'Tables needed: Same as inputa

theProject = av.GetProject
_dir.asFileName.setCWD

```



```

gtxt = LineFile.Make("g.txt".AsFileName,#FILE_PERM_WRITE)

'get the important tables
segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab

gcField = segVTab.FindField("grid-code")
salField = segVTab.FindField("sal")
tempField = segVTab.FindField("temp")
sodField = segVTab.FindField("sod")

bcgcField = bcVTab.FindField("grid-code")
bcSalField = bcVTab.FindField("sal")
bcTempField = bcVTab.FindField("temp")
bcSodField = bcVTab.FindField("sod")

'Get the theta used for the SOD temperature correction (spatially constant)

sodta = MsgBox.Input("Please enter the Theta used to SOD temperature correction","Input File G:
Parameters","1.065")
gtxt.WriteElt(sodta.asString++"0"+"0"+"0")

'Get the temperature, salinity, and sediment oxygen demand for each segment
'and write it to the input file

for each rec in segVTab
  seg = segVTab.ReturnValue(gcField,rec)
  tmp = segVTab.ReturnValue(tempField,rec)
  sal = segVTab.ReturnValue(salField,rec)
  sod = segVTab.ReturnValue(sodField,rec)
  gtxt.WriteElt(seg.asString++tmp.asString++sod.asString++sal.asString)
end

for each rec in bcVTab
  seg = bcVTab.ReturnValue(bcgcField,rec)
  tmp = bcVTab.ReturnValue(bcTempField,rec)
  sal = bcVTab.ReturnValue(bcSalField,rec)
  sod = bcVTab.ReturnValue(bcSodField,rec)
  gtxt.WriteElt(seg.asString++tmp.asString++sod.asString++sal.asString)
end

MsgBox.Info("Done writing Input Block G","")

.....

'Script: inputh
This file obtains the constants necessary to run the WASP program
for the Input Block H
Defaults:

```

```

'Since only Simple Streeter Phelps is possible, the only two
constants needed are the deoxygenation rate and the
reaeration rate
NOTE: This connection assumes that the reaeration rate is constant
over time and space
Tables Needed: NONE

theProject = av.GetProject
_dir.asFileName.setCWD

htxt = LineFile.Make("h.txt".AsFileName,#FILE_PERM_WRITE)

'Get the constants Kd and Ka from the user and write them to a file

lst = {"CBOD Deoxygenation Coefficient (/day @ 20°C)","Reaeration Rate (/day)"}
lstdef = {"0.1","0.1"}
cnst = MsgBox.MultiInput("Please Enter the Following Constants","Input File H: Constants",lst,lstdef)

for each i in cnst
  htxt.WriteElt(i.AsString)
end

MsgBox.Info("Done writing Input Block H","")

.....

'Script: inputi
This script writes the free from text file used to
create the Input Block I
NOTE: Since this connection is presently set up for
just steady state, no Time Functions are necessary
Tables needed: NONE

theProject = av.GetProject
_dir.asFileName.setCWD

ixt = LineFile.Make("i.txt".AsFileName,#FILE_PERM_WRITE)

MsgBox.Info("There are no time functions needed for this current model","Input File I: Time Functions")
ixt.WriteElt("0")

.....

'Script: inputj
This script writes the text file used to create
the Input Block J for WASP
This script presently only reads initial conditions for bod and do
Tables needed: same as inputa

theProject = av.GetProject
_dir.asFileName.setCWD

```

```
jtxt = LineFile.Make("j.txt".AsFileName.#FILE_PERM_WRITE)
```

```
'get the necessary tables  
segTable = av.GetProject.FindDoc("Main Segmentation")  
segVTab = segTable.GetVTab  
bcTable = av.GetProject.FindDoc("Boundary Segments")  
bcVTab = bcTable.GetVTab
```

```
gcField = segVTab.FindField("grid-code")  
doField = segVTab.FindField("int_do")  
bodField = segVTab.FindField("int_bod")  
bcgcField = bcVTab.FindField("grid-code")  
bcdoField = bcVTab.FindField("int_do")  
bcbodField = bcVTab.FindField("int_bod")
```

'Write 0 for initial condition for systems 1 through 4

```
for each i in 1..4 'nh3,n03,po4,chl  
jtxt.WriteElt("1"+"0")  
n=1  
totrec = segVTab.GetNumRecords  
p = totrec + 1  
for each i in 1..p  
for each rec in segVTab  
seg = segVTab.ReturnValue(gcField,rec)  
if (seg = n) then  
jtxt.WriteElt(seg.asString++"0")  
n= n+1  
break  
end  
end  
end
```

```
totrec = bcVTab.GetNumRecords  
p = totrec + 1  
for each i in 1..p  
for each rec in bcVTab  
seg = bcVTab.ReturnValue(bcgcField,rec)  
if (seg = n) then  
jtxt.WriteElt(seg.asString++"0")  
n= n+1  
break  
end  
end  
end
```

end

'obtain the BOD initial conditions and write them, in

```
'ascending order into the text file  
jtxt.WriteElt("0.5"+"0") 'writing the dissolved fraction  
n=1  
totrec = segVTab.GetNumRecords  
p = totrec + 1  
for each i in 1..p  
for each rec in segVTab  
seg = segVTab.ReturnValue(gcField,rec)  
bod = segVTab.ReturnValue(bodField,rec)  
if (seg = n) then  
jtxt.WriteElt(seg.asString++bod.asString)  
n= n+1  
break  
end  
end  
end
```

```
totrec = bcVTab.GetNumRecords  
p = totrec + 1  
for each i in 1..p  
for each rec in bcVTab  
seg = bcVTab.ReturnValue(bcgcField,rec)  
bod = bcVTab.ReturnValue(bcbodField,rec)  
if (seg = n) then  
jtxt.WriteElt(seg.asString++bod.asString)  
n= n+1  
break  
end  
end  
end
```

'obtain the DO initial conditions and write them, in  
'ascending order into the text file

```
jtxt.WriteElt("1"+"0")  
n=1  
totrec = segVTab.GetNumRecords  
p = totrec + 1  
for each i in 1..p  
for each rec in segVTab  
seg = segVTab.ReturnValue(gcField,rec)  
do = segVTab.ReturnValue(doField,rec)  
if (seg = n) then  
jtxt.WriteElt(seg.asString++do.asString)  
n= n+1  
break  
end  
end  
end
```

totrec = bcVTab.GetNumRecords

```

p = totrec + 1
for each i in 1..p
  for each rec in bcVTab
    seg = bcVTab.ReturnValue(bcgcField,rec)
    do = bcVTab.ReturnValue(bcdoField,rec)
    if (seg = n) then
      jtxt.WriteElt(seg.asString++do.asString)
      n= n+1
      break
    end
  end
end
end

'writing 0 for initial conditons of system 7 and 8
for each i in 1..4 'on,op
jtxt.WriteElt("1"+"0")
n=1
totrec = segVTab.GetNumRecords
p = totrec + 1
for each i in 1..p
  for each rec in segVTab
    seg = segVTab.ReturnValue(gcField,rec)
    if (seg = n) then
      jtxt.WriteElt(seg.asString++"0")
      n= n+1
      break
    end
  end
end
end

totrec = bcVTab.GetNumRecords
p = totrec + 1
for each i in 1..p
  for each rec in bcVTab
    seg = bcVTab.ReturnValue(bcgcField,rec)
    if (seg = n) then
      jtxt.WriteElt(seg.asString++"0")
      n= n+1
      break
    end
  end
end
end

end

MsgBox.Info("Done writing Input Block J","")
. . . . .

'Script: mod_parchk
'this script checks the model parameters for the eutro

```

```

'output file, beofre the user can run the output
'processer

theProject = av.GetProject

starttxt = LineFile.Make("estart.txt".AsFileName,#FILE_PERM_WRITE)
_dir = MsgBox.Input("Enter the working directory.", "Working Directory", "c:\benaman\wasp")
_dir.AsFilename.setCWD

segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab
bcTable = av.GetProject.FindDoc("Boundary Segments")
bcVTab = bcTable.GetVTab

_seg = (segVTab.GetNumRecords)+(bcVTab.GetNumRecords)

_dts = MsgBox.Input("What was the maximum time step allowed (/day)?", "Input File A: Model Options", "0.001")

_tend = MsgBox.Input("How many days did you run the model?", "Input File A: Model Options", "100.0")

_prn = MsgBox.Input("At what interval (in days) did you have the results printed to the output file?", "Input File A: Model Options", "1.0")

outname = MsgBox.Input("Enter the name of the file you want to process", "Model Output Processing", "*.edf")
outtxt = LineFile.Make("eoutme.txt".AsFileName,#FILE_PERM_WRITE)
outtxt.WriteElt(outname)

qu = MsgBox.YesNo("Is the output file called"++outname+"?", "Calibration Output Processing", TRUE)
if (qu=TRUE) then
  lst = "Is the following correct?" + nl + "Number of Days Run:" + _tend.AsString + "days" + nl + "Print Interval for output
file:" + _prn.AsString + "days" + nl + "Number of Segments:" + _seg.AsString
  qu2 = MsgBox.YesNo(lst, "Model Output Processing", TRUE)
  if (qu2=TRUE) then
    starttxt.WriteElt(_tend.AsString++_dts.AsString++_prn.AsString++_seg.AsString)
  else
    MsgBox.Info("Please see the Help for how to set up the output for processing", "")
  end
else
  MsgBox.Info("Please see the Help for how to set up the output for processing", "")
end
end

. . . . .

'Script: run_calout
'This Script will process the TOX15 tdf file into a
'text file, containing an array of time vs. salinity
'by segment number

'executing the fortran program which extracts the salinity measurements

```

```

'from the tdf file
_dir.asFileName.setCWD

MsgBox.Info("Press 'OK' to begin", "")
system.execute("calout.exe")
MsgBox.Info("Click 'OK' when done processing calibration output", "")

'making a table, named by the user, which will hold the output
name = MsgBox.Input("Name the Table (no more than 8 characters)", "Calibration View Output", "salinity")

theProject=av.GetProject

outfile = linefile.make(("salinity.txt").asFileName, #FILE_PERM_READ)

salVTab=VTab.MakeNew((name+".dbf").asFileName, dBase)
salList=List.Make
salList.Add(Field.Make("Time", #FIELD_FLOAT, 15, 4))
for each i in 1.._seg
salList.Add(Field.Make(i.asString, #FIELD_FLOAT, 10, 4))
end

if (salVTab.CanEdit) then
salVTab.SetEditable(true)
else
MsgBox.Info("Can't Edit", "Exit")
exit
end

salcIneList = salList.DeepClone
salVTab.AddFields(salcIneList)
salFields=salVTab.getFields
salTable=Table.Make(salVTab)
salTable.SetName(salVTab.GetName)

'reading the text file, calout.txt, and
'importing the values into salinity.dbf

while (outfile.IsAtEnd=FALSE)
newrec=salVTab.AddRecord
inline = outfile.ReadElT
count = 0
for each f in salFields
val=inline.Extract(count).AsNumber
salVTab.SetValue(f, newrec, val)
count = count + 1
end
end

MsgBox.Info("Done making salinity table", "")

```

```

. . . . .
'Script: run_modout
'This Script will process the EUTRO5 edf file into a
'text file, containing an array of time vs. salinity
'by segment number

'executing the fortran program which extracts the salinity measurements
'from the tdf file
_dir.asFileName.setCWD

MsgBox.Info("Press 'OK' to begin", "")
system.execute("modout.exe")
MsgBox.Info("Click 'OK' when done processing calibration output", "")

'making a table, named by the user, which will hold the output
bodname = MsgBox.Input("Name the Table for BOD (no more than 8 characters)", "View Output", "bod")
doname = MsgBox.Input("Name the Table for DO (no more than 8 characters)", "View Output", "do")

theProject=av.GetProject

outfile1 = linefile.make(("bod.txt").asFileName, #FILE_PERM_READ)
outfile2 = linefile.make(("do.txt").asFileName, #FILE_PERM_READ)

'Making bod table
bodVTab=VTab.MakeNew((bodname+".dbf").asFileName, dBase)
salList=List.Make
salList.Add(Field.Make("Time", #FIELD_FLOAT, 15, 4))
for each i in 1.._seg
salList.Add(Field.Make(i.asString, #FIELD_FLOAT, 10, 4))
end

if (bodVTab.CanEdit) then
bodVTab.SetEditable(true)
else
MsgBox.Info("Can't Edit", "Exit")
exit
end

bodcIneList = salList.DeepClone
bodVTab.AddFields(bodcIneList)
bodFields=bodVTab.getFields
bodTable=Table.Make(bodVTab)
bodTable.SetName(bodVTab.GetName)

'reading the text file, bod.txt, and
'importing the values into bod.dbf

while (outfile1.IsAtEnd=FALSE)
newrec=bodVTab.AddRecord
inline = outfile1.ReadElT

```

```

count = 0
for each f in bodFields
  val=inline.Extract(count).AsNumber
  bodVTab.SetValue(f,newrec,val)
  count = count + 1
end
end

MsgBox.Info("Done making BOD table","")

'Making do table
doVTab=VTab.MakeNew((doname+".dbf").asFileName,dBase)
doList=List.Make
doList.Add(Field.Make("Time",#FIELD_FLOAT,15,4))
for each i in 1.._seg
doList.Add(Field.Make(i.asString,#FIELD_FLOAT,10,4))
end

if (doVTab.CanEdit) then
doVTab.SetEditable(true)
else
  MsgBox.Info("Can't Edit", "Exit")
  exit
end

docIneList = doList.DeepClone
doVTab.AddFields(docIneList)
doFields=doVTab.getFields
doTable=Table.Make(doVTab)
doTable.SetName(doVTab.GetName)

'reading the text file, do.txt, and
'importing the values into do.dbf

while (outfile2.IsAtEnd=FALSE)
  newrec=doVTab.AddRecord
  inline = outfile2.ReadEl
  count = 0
  for each f in doFields
    val=inline.Extract(count).AsNumber
    doVTab.SetValue(f,newrec,val)
    count = count + 1
  end
end

MsgBox.Info("Done making Dissolved Oxygen table","")

. . . . .

'Script: toxirun
This script executes the WASP model TOX15

```

```

'it allows you to choose any input file, but the one
'generated by this connection will be called "caltest.inp"

_dir.asFileName.setCWD
system.execute("tox15.exe")
MsgBox.Info("Click 'OK' when TOX15 is done running","")

. . . . .

'Script: vwout
'This script controls the output options

theProject = av.GetProject

'Creates a list of tables and allows the user to pick which one they
'want to view the output for.
docList = theProject.GetDocs
tabList = List.Make
numdocs=docList.count
for each i in 0..(numdocs-1)
  dtype=(docList.get(i)).GetClass.GetClassName
  if (dtype="Table") then
    tabList.Add(docList.Get(i).GetName)
  end
end
_tblname = MsgBox.ChoiceAsString(tabList,"Choose the output table you want to work with","View Output")

'Sets the tables and important themes needed for this script
salTable = av.GetProject.FindDoc(_tblname)
salVTab = salTable.GetVTab
salFields = salVTab.GetFields
timeField = salVTab.FindField("time")
theView = av.GetProject.FindDoc("Segmentation")
segTheme = theView.FindTheme("Main Segmentation")
bcTheme = theView.FindTheme("Boundary Segmentation")
segTable = av.GetProject.FindDoc("Main Segmentation")
segVTab = segTable.GetVTab

'Checks to make sure the table selected has a time index to plot
if (timeField=nil) then
  MsgBox.Info("This table does not include a time index","")
  exit
end

mnsegs = segVTab.GetNumRecords

'Prompts the user to choose which type of output they want to view
for each rec in salVTab
  tnd = SalVTab.ReturnValue(timeField,rec)

```

```

end

lst = {"Table: Conc vs. t","Chart: Conc vs. t for one Segment","Chart: Conc vs. Seg# for a given Time","Coverage: Conc
at t = "+tnd.AsString++" days","Movie: Concentration over time"}
vw = MsgBox.ChoiceAsString(lst,"Choose the output you wish to view","View Output")
ivw = lst.Find(vw)

myBitMap = salVTab.GetSelection
myBitMap.ClearAll

'open the table selected
if (ivw = 0) then
    salTable.GetWin.Open

'creates a chart of conc vs. time for a segment chosen from the view
elseif (ivw = 1) then
    myBitMap.ClearAll
    ms = "Have the bug icon active,"
    ms = ms+nl+"Have the segmentation theme active,"
    ms = ms+nl+"Select the segment to graph"
    MsgBox.Info(ms,"View Output")
    theView.GetWin.Open
    theView.GetWin.Maximize

'creates a chart of concentration vs. Segment Number for a given time
elseif (ivw = 2) then

'Allows the user to choose the color of the chartcolorlist=List.Make
colorlist = List.Make
colorlist.Add("blue")
colorlist.Add("yellow")
colorlist.Add("green")
colorlist.Add("red")
col=MsgBox.ChoiceAsString(colorlist,"Select a color for the chart","View Output")
if (col="blue") then
    _chcolor = Color.GetBlue
elseif (col="yellow") then
    _chcolor = Color.GetYellow
elseif (col="green") then
    _chcolor = Color.GetGreen
else (col="red")
    _chcolor = Color.GetRed
end

timelst = List.Make
For each rec in salVTab
    time = salVTab.ReturnValue(timeField,rec)
    timelst.Add(time.AsString)
end
t = MsgBox.ChoiceAsString(timelst,"Choose the time (in days) you would like to graph.", "View Output")

```

```

timerec = timelst.Find(t)
sallst = List.Make
For each i in 1.._seg
    sallst.Add(salFields.Get(i))
end

myBitMap.Set(timerec)
salVTab.UpdateSelection
xChart = Chart.Make(salVTab,sallst)
xchartname=xchart.getname
theProject.setActive(xchart)
xChartDisp = xchart.GetChartDisplay
xChartDisp.setType(#CHARTDISPLAY_column)
xChartDisp.SetSeriesColor(0,_chcolor)
the_x=xChart.GetAxis
the_y=xChart.GetYAxis
the_x.SetTickLabelsVisible(True)
the_x.SetMajorGridVisible(True)
the_y.SetMajorGridVisible(True)
the_x.SetCrossValue(0)
the_y.SetCrossValue(0)
the_x.SetLabelVisible(true)
the_y.SetLabelVisible(true)
xLegend=xChart.GetChartLegend
xLegend.SetVisible(False)
xChart.GetTitle.SetName("Day = "+t.AsString)
ylst = {"Salinity (ppt)","DO (mg/L)","BOD (mg/L)"}
yname = MsgBox.ChoiceAsString(ylst,"Please enter the y-axis","View Output")
the_x.SetName("Segment Number")
the_y.SetName(yname)
xchart.GetWin.Open
keep = MsgBox.YesNo("Would you like to keep the chart?","ArcView",true)
if (keep=false) then
    av.GetProject.removeDoc(xchart)
    exit
else
    cname = MsgBox.Input("Please name the chart window","View Output","")
    xchart.setname(cname)
end

'creates a coverage of concentration and puts it on the active view
elseif (ivw=3) then
'Creates a new theme on the View
myBitMap.ClearAll
theView.GetWin.Open
theView.GetWin.Maximize
segSrcName = SrcName.Make("c:\wasp\cover\segarc arc")
newTheme = Theme.Make(segSrcName)
newTheme.SetVisible(true)
theView.AddTheme(newTheme)

```

```

theView.Invalidate
newFtab = newTheme.GetFTab
'sets the name of the Theme
thmename = MsgBox.Input("Name the new coverage", "View Output", "")
newTheme.SetName(thmename+"@ Time = "+tnd.AsString)

'counts the number of records in the salVTab
  cntr = 0
for each rec in salVTab
  cntr = cntr + 1
end

'create a table that will be joined to the aat
'of the new theme and holds the concentration values
cncVTab = VTab.MakeNew("temp.dbf".asFileName, dBase)
cncLst = List.Make
cncLst.Add(Field.Make("grid-code", #FIELD_BYTE, 4, 0))
cncLst.Add(Field.Make("conc", #FIELD_FLOAT, 10, 2))
cncVTab.SetEditable(true)

cncLstClone = cncLst.DeepClone
cncVTab.AddFields(cncLstClone)
cncFields = cncVTab.getFields

cncTable = Table.Make(cncVTab)
cncTable.SetName(cncVTab.GetName)
cncgcField = cncVTab.FindField("grid-code")
cncField = cncVTab.FindField("conc")
cntr = cntr - 1
for each i in 1..mnsesgs
  rec = i - 1
  cncVTab.AddRecord
  cncVTab.SetValue(cncgcField, rec, i.AsString)
  newField = salFields.Get(i)
  val = salVTab.ReturnValue(newField, cntr)
  cncVTab.SetValue(cncField, rec, val)
end

newFields = newFtab.GetFields
jtofield = newFields.Get(8)
jfromfield = cncFields.Get(0)
newFtab.Join(jtofield, cncVTab, jfromfield)
lookField = newFtab.FindField("conc")
theLegend = newTheme.GetLegend
theLegend.Quantile(newFtab, lookField, 6)
theLegend.RampColors(Color.GetBlue, Color.GetRed)
for each i in theLegend.GetSymbols
  i.SetWidth(3)
end
newTheme.UpdateLegend

```

```

av.GetProject.removeDoc(cncTable)

'Creates a 'movie' of four frames at chosen times
elseif (ivw=4) then
theView = theProject.FindDoc("Segmentation")
m = "This option allows you to choose four times and view how the concentrations change, over time."
MsgBox.Info(m, "View Output")
timelst = List.Make
For each rec in salVTab
  if (rec = 0) then
    continue
  else
    time = salVTab.ReturnValue(timeField, rec)
    timelst.Add(time.AsString)
  end
end
for each i in 1..4
  t = MsgBox.ChoiceAsString(timelst, "Choose the time" ++ i.AsString ++ "(days)", "View Output")
  if (i = 1) then
    t1 = timelst.Find(t) + 1
    time1 = salVTab.ReturnValue(timeField, t1)
  elseif (i = 2) then
    t2 = timelst.Find(t) + 1
    time2 = salVTab.ReturnValue(timeField, t2)
  elseif (i = 3) then
    t3 = timelst.Find(t) + 1
    time3 = salVTab.ReturnValue(timeField, t3)
  elseif (i = 4) then
    t4 = timelst.Find(t) + 1
    time4 = salVTab.ReturnValue(timeField, t4)
  end
end

segSrcName = srcName.Make("c:\wasp\cover\segarc arc")
concname = MsgBox.Input("Name the concentration you are viewing", "View Output", "")
s = MsgBox.Input("Choose a step time (in sec) for the movie frames", "View Output", "5")
s = s.AsNumber

'creates a table which will hold the conc values for
'all four times

movVTab = VTab.MakeNew("temp.dbf".asFileName, dBase)
movLst = List.Make
movLst.Add(Field.Make("grid-code", #FIELD_BYTE, 4, 0))
movLst.Add(Field.Make("conc1", #FIELD_FLOAT, 10, 2))
movLst.Add(Field.Make("conc2", #FIELD_FLOAT, 10, 2))
movLst.Add(Field.Make("conc3", #FIELD_FLOAT, 10, 2))
movLst.Add(Field.Make("conc4", #FIELD_FLOAT, 10, 2))
movVTab.SetEditable(true)

```

```

movLstClone = movLst.DeepClone
movVTab.AddFields(movLstClone)
movFields = movVTab.getFields

movTable = Table.Make(movVTab)
movTable.SetName(movVTab.GetName)
movgcField = movVtab.FindField("grid-code")
c1Field = movVTab.FindField("conc1")
c2Field = movVtab.FindField("conc2")
c3Field = movVtab.FindField("conc3")
c4Field = movVtab.FindField("conc4")
minLst = List.Make
for each x in 1..mnsesg
  rec = x-1
  movVTab.AddRecord
  movVTab.SetValue(movgcField,rec.x.AsString)
  newField = salFields.Get(x)
  val1 = salVTab.ReturnValue(newField,t1)
  val2 = salVTab.ReturnValue(newField,t2)
  val3 = salVTab.ReturnValue(newField,t3)
  val4 = salVTab.ReturnValue(newField,t4)
  movVTab.SetValue(c1Field,rec,val1)
  movVTab.SetValue(c2Field,rec,val2)
  movVTab.SetValue(c3Field,rec,val3)
  movVTab.SetValue(c4Field,rec,val4)
  minLst.Add(val1)
  minLst.Add(val2)
  minLst.Add(val3)
  minLst.Add(val4)
end

n = 35
x = 0
for Each m in MinLst
  if (m < n) then
    n = m
  end
end

for Each m in MinLst
  if (m > x) then
    x = m
  end
end

inv = (x - n)/8

theView.GetWin.Open
theView.GetWin.Maximize
av.delayedrun("frame1",{segSrcName,concname,time1,theView,movVtab,x,n,inv},25)

```

```

av.delayedrun("frame2",{segSrcName,concname,time2,theView,movVtab,x,n,inv},27+s)
av.delayedrun("frame3",{segSrcName,concname,time3,theView,movVtab,x,n,inv},s+s+29)
av.delayedrun("frame4",{segSrcName,concname,time4,theView,movVtab,x,n,inv},s+s+s+31)
av.GetProject.removeDoc(movTable)
end

```



**Appendix G**  
**Help File for ArcView/WASP5 Connection**  
*created using winhelp.exe standards*

# WASP5/ArcView Connection

## Information and Help

### Table of Contents

General Information and Concept
Setting up the WASP5/ArcView Connection
Tables Needed for Processing
Coverages Needed for Processing
Creating an Input File
Changing an Input File
An Overview of Each Input Block
Model Calibration
Executing a Model Run
Viewing the Output
About WASP5
About this Connection
Limitations and Important Notes Concerning this Connection
Troubleshooting
Helpful References

This ArcView/WASP5 connection was created by Jennifer Benaman, Research Assistant, Department of Civil Engineering, The University of Texas at Austin. Last revised: August 1996.

## General Information and Concept

This connection is an ArcView Project which has compiled Avenue Scripts to perform the following functions:

1. Obtains the necessary information needed to run WASP5 and writes it to free form text files. This project will write a text file for each input block. As a result, ten separate text files will be written in the working directory. In addition, three character text files, containing filenames, are created.
2. Runs a FORTRAN program which takes these text files and formats them into the input file for WASP5. The user will be prompted to name the input file.
3. Executes WASP5 (EUTRO5 or TOXI5).
4. Processes the model output.
5. Allows the user to view the output in the form of charts and tables.

Presently, this connection is set up to run a model with the following characteristics:

- Level 1 complexity, EUTRO5 model (Simple Streeter-Phelps Model for BOD/DO)
- Steady-state conditions (Flow conditions: Average Year or Dry Weather)
- System should resemble river or stream (i.e. main model segments have dominating flow in one direction and have exchange with other main segments at the upstream or downstream ends).
- Only water column flow is considered presently. This connection is not set up to deal with pore water column flow, secondary water column flow, or sediment transport. The user can have a single benthic layer under the water segments as a boundary condition to deal with settling.

In addition, this connection can run TOXI5 to **calibrate** the system, by using salinity as the conservative substance. It is recommended that the model be calibrated, before an actual BOD/DO model run is performed.

If the user has additional questions concerning the model parameters or variables used in the input file, please refer to the [WASP5 User's Manuals \(A and B\)](#).

### Related Topics:

[About WASP5](#)

[About this Connection](#)

[Limitations and Important Notes Concerning this Connection](#)

[Table of Contents](#)

## Setting up the WASP5/ArcView Connection

The ArcView/WASP5 model connection and a demo which shows the Houston Ship Channel study discussed in this report can be set up on any computer which has ArcView 2.1 or higher installed on the machine. In order to set up the demonstration on a personal computer, the following steps are taken:

1. Install WASP5 onto the computer. WASP5 is available from the USEPA Homepage ([ftp://ftp.epa.gov/epa\\_ceam/wwwhtml/wasp.htm](ftp://ftp.epa.gov/epa_ceam/wwwhtml/wasp.htm))
2. Download the demonstration files, in zipped format, from the University of Texas, Center for Research in Water Resources Homepage for the ArcView/WASP5 connection demo (<http://www.ce.utexas.edu/prof/maidment/GISHydro/>) and unzip them into the directory that the WASP5 executables are located.
3. Open the project `hsc_wasp.apr` in ArcView version 2.1 or higher. ArcView may initially ask the user to location of some coverages and tables. All necessary coverages and tables should be located with in the directory in which the downloaded file was unzipped. Most tables are in dbf format, while the coverages are in a folder entitled *cover*. Open the script *vwout* and locate the two references to the arc coverage *segarc*. Both of the lines in the script read:  

```
segSrcName = SrcName.Make("c:\wasp\cover\segarc arc")
```

This line informs Avenue on where to locate the main segmentation coverage. Be sure that the drive and directory name in this line is correct. Also, if this script is being recompiled for a new modeling system, this line should reference the correct location and name for the main segmentation coverage in the new system that has been developed.
4. If the *vwout* script was changed, in any way, recompile it, by clicking on the checkmark icon on the bottom toolbar of the ArcView script tools.
5. Model input file creation, model runs, and output viewing can then be performed as described previously in this help file.

### Related Topics:

[Coverages Needed for Processing](#)

[Table of Contents](#)

## Tables Needed for Processing

It is extremely important that all tables listed here are included and opened in the project, before the model input file can be created. It is also very important that all tables and table fields are named (either by alias or real name) EXACTLY as they are written here.

### NOTES:

- Tables are listed here in alphabetical order.
- The order of the records is not important.
- The order of the fields should be adhered to in the Main Segmentation Table (grid-code should always be the 6th field) and in the Main Segment Parameters Table (grid-code should always be the first field).
- Weighted flow accumulation values are based on flow accumulations run on 100m x 100m cell grids. The unit conversion set in the scripts are also based on this cell-size.

*Click on the table name to get a list of fields and their corresponding units.*

### BOD Loading Values

A table (typically a value attribute table -- INFO format) of the BOD flow accumulation values for each "outlet" (i.e. its most downstream point) of the main water segments. These values are usually obtained by running a flow accumulation over the watershed area, weighted by a grid of BOD load, in ArcInfo's subprogram, Grid. This weighted flow accumulation is then "combined" with a grid of the outlet points to obtain this table. This table accounts for the non-point source loading from the watershed land surface.

### Boundary Segments

A dBase file (dbf) table which contains all boundary segments (water and sediment), and their corresponding parameters. This file can be created directly in dBase, or in ArcInfo and exported out of ArcView to a dBase format. If the table is not a dBase format, it will not be possible to edit and change parameters in the table. *The numbering of the boundary segments should start which the next number after the last main segment.*

### Flow Accumulation Values

A table (typically a value attribute table -- INFO format) of the flow accumulation values for each "outlet" (i.e. its most downstream point) of the main water segments. These values are usually obtained by running a flow accumulation over the watershed area, weighted by a grid of flow depth, in ArcInfo's subprogram, Grid. This weighted flow accumulation is then "combined" with a grid of the outlet points to obtain this table.

### Main Segment Parameters

A dbf file which contains the attributes of the main segments for the water quality model. This file can be created directly in dBase, or in ArcInfo and exported out of ArcView to a dBase format. If the table is not a dBase format, it will not be possible to edit and change parameters in the table. This table will be joined, during the generation of Input Block A, to the "Main Segmentation" table below. *The numbering of the main segments should start with "1" and continue, in order, until the last main segment is numbered. Then, number the boundary segments.*

### Main Segmentation

The aat of the main segmentation coverage. *The numbering of the main segments should start with "1" and continue, in order, until the last main segment is numbered. Then, number the boundary segments.*

### Point Source BOD

A dbf file which contains the annual BOD loading into each segment from point sources. This file can be created directly in dBase, or in ArcInfo and exported out of ArcView to a dBase format. If the table is not a dBase format, it will not be possible to edit and change parameters in the table.

### Runoff Accumulation Values

A table (typically a value attribute table -- INFO format) of the flow accumulation values for each "outlet" (i.e. its most downstream point) of the main water segments. These values are usually obtained by running a flow accumulation over the watershed area, weighted by a grid of runoff depth, in ArcInfo's subprogram, Grid. This weighted flow accumulation is then "combined" with a grid of the outlet points to obtain this table.

### Water Boundary Segmentation

The arc attribute table (aat) of the water boundary segment reaches. It is important NOT to join this table with the "Boundary Segments" table, since the boundary segments table may have segments which are not Type 1 (i.e. Water Column). If joined, segments shown on the Boundary Segments table which are not represented in the coverage as water reaches will be "lost". *The numbering of the boundary segments should start with the next number after the last main segment.*

### Related Topics:

[Coverages Needed for Processing](#)

[Table of Contents](#)

## Coverages Needed for Processing

Although the user may wish to add other coverages for informational purposes, there are only two arc coverages needed to run this model connection. Both should be located within the same view.

The view should be entitled "Segmentation". The two themes are described below:

### Boundary Segmentation

An arc coverage of the water boundary segments. The arc attribute table attached to this theme should be named "**Water Boundary Segmentation**".

### Main Segmentation

An arc coverage of the main segmentation being modelled by WASP. The arc attribute table to this theme should be named "**Main Segmentation**". This coverage should be named "segarc" within ArcInfo and aliased as "Main Segmentation" on the view.

### Related Topics:

[Tables Need for Processing](#)

[Table of Contents](#)



## Creating an Input File

The way this connection works is by reading necessary table information and querying the user for needed model options, during the input file generation. WASP5 has ten input blocks (A through J) and a "free form" text file is created for each block. In addition, three text files containing the model run description, DYNHYD file name (if necessary), and the input filename are created. Once the input file generation is performed, these text files will be in the working directory as a.txt, b.txt, etc. It is then possible to change just one input block (i.e. just the model constants -- Input Block H), while the rest of the parameters will stay as they were originally generated.

To create your first input file for your system, follow the following steps:

1. Make sure all FORTRAN programs (calout.exe, calgen.exe, modout.exe, outgen.exe), WASP5 executables, and their related files are located in your working directory.
2. Check the script, vwout, to ensure that the correct coverage and directory is being referenced (see [Setting up the ArcView/WASP5 Connection](#) for more information on this step).
3. Have all of the necessary [Tables](#) and [Views](#) open. They can be minimized to icon views, but they must be open.
4. Have the "Project" window active so that the model connection menu items are shown on the Main Menu Bar. They are: BOD/DO Input Blocks, BOD/DO Model, and Model Calibration.
5. Choose BOD/DO Input Blocks: All Input Blocks and allow the project to run. The entire process will take about 2-3 minutes. The scripts will ask you for some information for some blocks, while other blocks will not require any user input. The project will give you a message box each time it is done writing a particular Input Block. Your working directory should ALWAYS be the directory which holds eutro5.exe, toxi5.exe, outgen.exe, calgen.exe, calout.exe, and modout.exe.
6. Once all ten blocks are written, choose BOD/DO Model: Generate Input File.
7. Be sure you have [calibrated](#) your model before running the BOD/DO model.

### IMPORTANT NOTE:

If you have already run this connection for your system and want to change a few parameters, go to [Changing an Input File](#). Be aware, if you have exited out of the project, even if you have saved your system and the related tables, *you MUST ALWAYS run Input Block A, before generating a new input file and running the model.*

### Related Topics:

[Changing an Input File](#)

[An Overview of Each Input Block](#)

[Executing a Model Run](#)

## Changing an Input File

Once you have initially established your input blocks (see Creating an Input File), it is possible to change just one block of the main input file. This ability is an advantage if you want to either correct a possible mistake or see the changes that may occur in the results, if a constant or a parameter changes (i.e. investigate model sensitivity).

To accomplish this task, first you must have run the entire Input File process at some point. It is not necessary to have run this process during the active session. You could have run it in an earlier session. If so, the free form text file created by Avenue (a.txt, b.txt, etc.) will still be present in your working directory. It will then only be necessary to rerun those blocks which you have changed information to. Important things to remember:

- You must *ALWAYS* run Input Block A at the beginning of a session. This Input Block establishes your working directory, which is linked to the other subroutines that create Input Blocks B through J.
- The text files which are created and used by the formatting FORTRAN program are always named a.txt, b.txt, c.txt, etc. If you change an input block, but want preserve the first input file, be sure to give the new input file a different name, when prompted for a name in the input file generation. However, with the project tables, if you change some parameters, unless you create an entirely new table in dBase or ArcInfo, your old information will not be preserved.
- If you change some parameters in the tables, be sure to rerun all input blocks which are affected by those characteristics you have changed. For example, if you change the cross sectional areas, it will be necessary to rerun both Input Block B: Exchanges, and Input Block C: Volumes, since the cross sectional area is used in both of these blocks. When in doubt, consult the **WASP5 User's Manual B** supplied with the program.

To change the an Input Block:

1. If the parameters to be changed are contained in a table, you must first edit the table. Make the table which you want to edit active and choose Table: Start Editing. Then, choose the editing icon from the tool bar and change the values. When complete, select Table: Stop Editing from the menu. Only dbf Tables can be edited. If you are trying to edit an INFO table, you must first export it as a dbf file, open back up in the project, and then rename it to the correct name. NOTE: To edit the main segment parameters table, you must first unjoin the table from the main segmentation table. To do this, have the main segmentation table active and choose Table: Remove All Joins. You should then run Input Block A to rejoin the tables.

2. Once you are done editing the table (if necessary), set the project window active and select Input File from the menu bar. A menu will appear showing you all ten input blocks. Choose the input block you wish to recreate and Avenue will regenerate the necessary text files.
3. When complete, choose BOD/DO Model: Generate Input File, to recreate the input file with the new information.

### Related Topics

[Creating an Input File](#)

[An Overview of Each Input Block](#)

[Executing a Model Run](#)

[Table of Contents](#)

## An Overview of Each Input Block

### Input Block A: Model Identification and Simulation Control

This Input Block contains basic simulation information and model preferences. The script reads how many segments are present and queries the user to choose model preferences such as length of model run, preferred time step, and print intervals. This script also asks for a model title, which will be printed on the first line of the input file, and the input file name (\*.inp). Be sure to use 8-3 convention when naming the input file and use the extension ".inp".

The following defaults are set in the generation of this input block:

- \* Presently, the connection set to handle just Simple Street-Phelps Modelling (BOD/DO Process)
- \* Backward differencing is always used.
- \* A transport file is always generated.
- \* The first six segments' solutions are those which are displayed on the screen, during a model run.
- \* The same maximum time step is used throughout the model run.
- \* The same print interval is used throughout the model run.

### Input Block B: Exchange Coefficients

This Input Block describes the exchange coefficients for surface water (pore water exchanges are not set in the connection, as of yet). The script reads the lengths of the segments to calculate the characteristic length between segments. It also determines if the boundary segments are perpendicular to the main segmentation and compensates in the characteristic length, if it is. It also reads the exchange coefficients and cross-sectional areas for two neighbouring segments. It chooses the smaller area and the exchange coefficient for printing to the file. Finally, if a benthic sediment segment is set, the water column exchange between each overlying water segment and the pore water in the sediment segment are printed to the text file.

The following defaults are set:

- \* All exchange coefficients are steady-state
- \* All exchange coefficients are written to the text file in  $\text{m}^2/\text{sec}$  and lengths are in m.

### Input Block C: Volumes

This Input Block describes the segment volumes for the system. It also sets the hydraulic geometry parameters for calculating segment depth and velocity. These geometry parameters are used to calculate reaeration (if necessary) or volatilization from the segments; they are not used in transport calculations. For this connection, a constant (in time and space) reaeration coefficient has been assumed -- therefore, these geometry parameters are not used to calculate reaeration in this particular set up. The script reads the cross-sectional areas (in  $\text{m}^2$ ) and multiplies it by that segment's length (m)

to obtain the volume. The script also asks the user to choose the following: 1) water column volume option, 2) benthic volume option, and 3) benthic time step.

The following defaults are set:

- \* Volumes are written to the text file in  $m^3$
- \* Geometry parameters do not spatially vary
- \* Only the first benthic volume option is possible.

### **Input Block D: Flows**

This Input Block provides the advective transport flows that are used in the model. Presently, only flows for WASP Flow Field 1 (Water Column) are used in the connection. First, the script asks the user to choose a flow option. These options are described below:

Flow Option 1: Field one flows are specified directly by the user. Individual flows at each segment interface are summed by the model, and the net flow is applied across the interface.

Flow Option 2: Field one flows are specified directly by the user. Individual flows at each segment interface are applied directly by the model.

Flow Option 3: Flows are read from a formatted file created by DYNHYD or other hydrodynamic model. If this option is chosen, the user will be asked to enter the file name of the text file which holds this information.

The script in ArcView assumes that each boundary condition given has an associated flow and each main segment has just one flow input. The script will track each flow input from its upper-most boundary to the most downstream segment. In order to do this process, each boundary segment and its corresponding upstream and downstream segments are read from the "Boundary Segments" Table. The downstream segment should be a main segment. The main segment is then found in the "Main Segmentation" Table, with its downstream segment. The script will continue to look for the successive downstream segments, until the most downstream main segment is reached. This flow route is printed to the text file and, at the end of the flow path, the actual flow from the original boundary segment is printed.

The table, "flow.dbf" is created during this input block determination, by doing a units conversion on the "Flow Accumulation Values" Table. This table will give the total cumulative flow, the incremental flow, the runoff (from the Runoff Accumulation Table) and the baseflow into each main segment in  $m^3/sec$ . These flows are joined to the "Main Segmentation" table and the incremental flows are used for Input Block D.

If "Dry Weather Conditions" is selected during the generation of this input block, only the baseflow will be printed to the text file as the flow in the system.

The following defaults are set:

- \* All flows are in m<sup>3</sup>/sec. The percentages of the total flow are written to the file, along with the total flow value.
- \* The flow is steady state
- \* The number of flow fields is set to 1: Water column only (no pore water flows)

### **Input Block E: Boundaries**

This Input Block describes the boundary segments and their concentrations. Model boundaries consist of those segments that import, export, or exchange water with locations outside the main network. A boundary is either a tributary inflow, a downstream outflow, or an open water end of the model network across which dispersive mixing can occur. The boundary concentrations are read from the "Boundary Segments" Table.

The following defaults are set:

- \* Since the model is set just for Simple Streeter-Phelps Model, only BOD and DO are considered in the boundary concentrations
- \* All concentrations are read and written in units of mg/L
- \* The boundary conditions are steady-state

### **Input Block F: Waste Loads**

This Input Block writes the BOD and DO loads into each main network segment. The script does two things:

1. Reads the point source BOD loads into each segment and converts the value to kg/day
2. Calculates the non-point source BOD from the "BOD Loading Values" Table in kg/day

Since, presently, the model is set for steady state, the non-point BOD loads are added to the point source BOD loads to get a total load to the segment. Once the non-point source loads are set to time varying, a separate non-point source file will have to be generated.

This Input Block also generates a table called "load.dbf". This table gives the non-point source loads in kg/day for BOD. It will also give the total point source BOD loading in kg/day for each main segment. If "Dry Weather Conditions" is selected during Input Block D generation, only the point source loads are written to the text file. Non-point source loads will be ignored.

The following defaults are set:

- \* All loads are steady state
- \* Only BOD and Do are considered

### **Input Block G: Parameters**

This input block reads the necessary parameters for the Level One EUTRO model. As the complexity level increases, the number of parameters needed will increase. Presently, only four parameters are needed:

1. Temperature, read from the segment tables in °C -- temperature is used to correct for deviations from the standard (20°C) and DO saturation.
2. Sediment oxygen demand (SOD), read from the tables in g/m<sup>2</sup>-day
3. SOD theta correction, input by user -- used to correct SOD for temperatures deviating from 20°C.
4. Salinity, read from the tables -- used to calculate DO saturation

The following defaults are set:

- \* Temperature does not vary in time
- \* SOD theta does not vary in time or space

### **Input Block H: Constants**

This Input Block queries the user for the necessary constants needed to run the Simple Streeter-Phelps model. The definition of the constants will vary, depending upon the structure and kinetics of the systems comprising each model. For the present model, only two constants are needed:

1. CBOD deoxygenation rate at 20°C, per day.
2. **Reaeration** rate constant at 20°C for entire water body, per day.

The following defaults are set:

- \* The constants do not vary in time or space

### **Input Block I: Time Functions**

If the model were non-steady state for any parameter, this Input Block would use time functions to vary the specific parameter. Presently, none of the parameters are set to vary in time, so this block is default to 0.

### **Input Block J: Initial Conditions**

This Input Block describes the initial conditions for each system in the model. Presently, only DO and BOD are considered in this script. The initial conditions are read from the segment tables. If the model is going to run until equilibrium is reached, the initial conditions will not affect the results.

NOTE: If time variant results are desired, additional programming should be done to set the initial conditions in the boundary segments to unique values. Presently, the initial conditions in the boundary segments are set to the boundary concentrations.

The following defaults are set:

- \* The dissolved fraction of BOD is set at 0.5
- \* The dissolved fraction of DO is always 1.0
- \* The maximum value for all systems is 1.0e8
- \* Solids Field 3 transports BOD in its particulate form
- \* Solids Field 5 transports DO
- \* All densities are set to 1.0 (EUTRO does not use those values)
- \* All initial conditions are in mg/L

Related Topics:

[Creating an Input File](#)

[Changing an Input File](#)

[Table of Contents](#)



## Model Calibration

Before running the DO/BOD model in EUTRO, the system being modelled must first be calibrated. The user can perform this calibration without the help of the ArcView connection, or utilize the ArcView to assist in the model calibration.

For calibration, the TOXI5 is used to model a conservative substance in the system. Typically, at least two boundary conditions are set to a known concentration value of the substance; the rest of the segments are set to initial conditions of 200 ppt. The model is run and once equilibrium in the system is reached, the calculated values of the conservative substance can be compared to known values.

For the calibration in this connection, salinity is used as the conservative substance. To run the model calibration perform the following steps:

1. Have all **tables** and **views** necessary for a normal model run open and named correctly.
2. From the menu bar choose Model Calibration: Write Input Information. During this input generation you will be asked the number of boundary conditions and then asked to input the segment numbers which are the boundary conditions, one at a time.
3. When complete choose Model Calibration: Generate Input File.
4. Once the input file is done, choose Model Calibration: Run Model Calibration. This choice will execute TOXI5, when a list of files appears on the screen, choose the name of the input file you just created.
5. You can then review the results by **processing and viewing the output**.

As in the regular input file generation, the avenue program creates thirteen text files. A number of them are identical to the blocks needed for a BOD/DO model run. Specifically, input blocks B,C,D, and I are the same. For this reason, the same free form text files (b.txt,c.txt,d.txt, and i.txt) are written to the working directory. For all other input blocks, the free form text files have the prefix "cal" (i.e. cala.txt).

### Related Topics:

[Table of Contents](#)

## Executing a Model Run

Once an input file is created in ArcView, you can execute a model run by simply choosing BOD/DO Model: Run EUTRO5. Be sure that you have generated the input file before performing this step.

When the model is executed, a DOS window will appear and the EUTRO interface will be shown.

When a list of input files is shown, choose the input file you have created. Do not press the "ok" button on the message box until the model is completed in running. By pressing 'OK' you are informing ArcView that the model is done and it can now exit from DOS and return to the ArcView interface.

NOTE: The EUTRO5 or TOXI5 models do not have to be executed through ArcView. It is possible to get ArcView to write numerous input files for these models and then run the model separately, through DOS. The final output can then be processed and viewed through ArcView.

### Related Topics:

[Table of Contents](#)

[Creating an Input File](#)

## Viewing the Output

There are three main steps to processing and viewing the output from a model run. Note that the file you want to process did not have to be created or executed in ArcView to process. But, if charts and coverages want to be created, the associated original tables and coverages to the model run should be present in ArcView. The three main steps are:

1. Checking the model parameters
2. Processing the output file
3. Viewing the output

All of these steps can be executed from either the BOD/DO Model menu or the Calibration Model menu. Be sure you choose the correct menu, corresponding to the model you are presently working with.

### **Step 1: Checking the model parameters**

This item just reinitializes the model options and file information for ArcView to process the output. With this option, it is possible to create a number of output files with numerous model runs and then process them, one at a time, without having to go back and rerun the model. Be sure that you give the correct output file name for the model run you are interested in. If you are working with the calibration model, the output file will be named the same as the input file name you gave it, with a ".tdf" extension. The BOD/DO model will follow the same convention, with an ".edf" extension.

This step should ALWAYS be executed before processing and viewing an output file.

In addition, you should check the script, vwout, to be sure that the proper coverage and directory name is being referenced in the code (see [Setting up the ArcView/WASP5 Connection](#) for more information).

### **Step 2: Processing the output file**

This option executes a FORTRAN program which will write a text file with either salinity, or BOD and DO measurements, for every segment at each time step. The text file will then be imported into ArcView as a dBase Table. You will be prompted to name each table, as it is processed. Be sure, if you are processing a number of output files, that you give the tables descriptive names so that they can be differentiated in the project.

### **Step 3: Viewing the output**

The first thing you must do, after executing this command is to choose a table with which you want to work. Once selected, all charts or coverages created will be linked to that table. Although the table choice box will show you a list of all available tables, only those that were created by the above process step can be viewed using this menu option. If you choose a table that does not have the proper format, ArcView will exit you out of the view output script. Be sure that the table you select corresponds to the parameters you set in Step 1, above. If it does not, then go back and reset the parameters correctly. This need is because the output viewing steps (such as coverage creation) uses some of these parameters in order to execute some Avenue script commands.

Within this step, there are five options. There are as follows:

1. View the table you have chosen
2. Create a chart of concentration vs. time for a chosen segment
3. Create a chart of concentration vs. segment number for a chosen time
4. Create a coverage of the concentration at the last time step in the table
5. Create a "movie" of four coverages which display concentration at four chosen times

#### Option 1: Viewing the Table

By choosing this option, the table you have selected will open and become the active view. You can then view the output or create your own charts manually from this table, if desired.

#### Option 2: Concentration vs. Time

This option will open the view, "Segmentation" for you and prompt you to activate the "bug" icon and choose a segment. If, when the view opens, you do not see the bug icon, try resizing the ArcView window. The "bug" icon should be on the far right bottom tool bar. Once it is activated, be sure that the proper theme (either Main Segmentation or Boundary Segmentation) is highlighted and click on the segment you want to graph. You will then be asked a few chart options, including color and the name of the y-axis. Afterwards, a chart, showing concentration vs time ( for all time steps) will appear. If you want just particular times plotted, you should manually go in and select those times on the linked table. The chart will then change to show the selected times, accordingly.

#### Option 3: Concentration vs. Segment Number

This option just prompts you for the time at which you want to display the concentration values. A bar chart showing the concentration for each segment (main and boundary) will then appear. If you want to

see a different time, just select a different time on the linked table and the chart will change to reflect the new selection.

#### Option 4: Coverage at Final Time

This option will create an ArcView coverage of the concentration at the final time in the model. The script automatically brings the coverage up on the view and shows the concentration values for that time, in a ramped arc coverage (from blue to red).

#### Option 5: Movie of Four Chosen Times

This option allows you to choose four given times from all possible times in a chosen table. The script will then open a view and create a coverage, at a time delay chosen by the user. Each coverage will have eight intervals of concentration. The script will determine the min and max values within the time steps you have chosen and ramp the coverage from gray to blue in eight intervals. These intervals will stay constant for all four times, so that changes in concentrations can be viewed consistently. After the script is complete, the user will have four new coverages of concentration at each time step.

NOTE: All charts which you create are ALWAYS "linked" to a given table. If you make changes to that table or to your selections within the table, the chart will change, accordingly. For example, if you have created a chart of concentration vs. segment at time = 10 days. If you choose to create another chart of concentration vs. segment at time = 19 days, and link it to the same table, the first chart will also change. To avoid this problem, you can create or add multiple copies of the same table to the project and link a chart to each table. You can do this by either adding the dbf file numerous times and renaming the table so that it is more descriptive. Or, you can process the same output a number of times and just keep changing the table name when prompted for a name. Then, only link one chart to each table.

#### Related Topics:

[Table of Contents](#)

## About WASP5

WASP5 stands for Water Quality Analysis Simulation Program Modelling System. It is developed by the Environmental Protection Agency (EPA) at the Center for Exposure Assessment Modelling (CEAM) in Athens, Georgia. The program consists of a main program, WASP, and three subprograms: EUTRO, TOXI, and DYNHYD. EUTRO is used to model BOD/DO and eutrophication; TOXI, toxic chemicals and model calibration; and DYNHYD, system hydrodynamics.

The program is available, shareware, from the [EPA ftp site](#) and on the [World Wide Web](#). Although the current version of WASP (5.10), has a user interface entitled WISP (WASP Interactive Support Program), the ArcView connection discussed here does NOT utilize this interface. Due to present memory constraints, it is not possible to run WISP, while running the Windows environment necessary for ArcView. As a result, only the EUTRO and TOXI executables (with their related error and message files) are needed for this ArcView connection. The executables, along with the other necessary files, should be located in the working directory designated during Input Block A execution.

For questions or problems regarding model execution and parameters, the user should refer to the [WASP User's Manual A and B](#), available with the model. Further questions regarding the model should be addressed to:

Center for Exposure Assessment Modelling  
U.S. Environmental Protection Agency  
Office of Research and Development  
Environmental Research Laboratory  
960 College Station Road  
Athens, GA 30605-2720  
706 543 3549

### Related Topics:

[Table of Contents](#)

WASP v.5.1 September 1993

EPA ftp site: [earth1.epa.gov](ftp://earth1.epa.gov)

Directory: [/pub/athens/](#)

EPA www site: <ftp://earth1.epa.gov/pub/athens/wwwhtml/wasp.htm>

## About this Connection

This connection was created as part of a research project at the Department of Civil Engineering at University of Texas at Austin. The project dealt with dissolved oxygen modelling of a ship channel, located off of Galveston Bay, Texas. Partial funding for this project was provided by the National Science Foundation.

The GIS/WASP connection is written in Avenue, under ArcView 2.1. The FORTRAN programs used to format the input files (outgen.exe: general input and calgen.exe: calibration input) and process the output (calout.exe and modout.exe) were compiled with Microsoft FORTRAN. The version of WASP which was connected to ArcView was 5.10 (WASP5).

If a CD ROM is supplied with this project, all coverages were generated using ArcInfo 7.03 and ArcView 2.1. To execute this program and view the example on the CD ROM, refer to [Setting up the ArcView/WASP5 Connection](#) in order to view the demo from the CD.

For a complete description on the methodology behind the coverage creation and model connection refer to:

Benaman, J. *Modeling of Dissolved Oxygen in the Houston Ship Channel, using WASP and Geographic Information Systems*. Master's Thesis. Department of Civil Engineering. The University of Texas at Austin. December 1996.

### Related Topics:

[Helpful References](#)

[Table of Contents](#)

## Limitations and Important Notes Concerning this Connection

Presently, this model connection has the following limitations:

- Only steady-state input files can be run
- The connection is best set up for a river system, or tidally influenced river system
- Simple Streeter-Phelps Model is the complexity level -- considers just Biochemical Oxygen Demand (BOD) and Dissolved Oxygen (DO)
- Only water column flow is considered in this connection. The connection is not presently set up for 2-layer water systems or sediment transport.
- Specific assumptions and limitations with each input block are discussed in [An Overview of Each Input Block](#)

Other Notes on this Connection:

- Input Block A must always be run at the beginning of each session in order to set the working directory.
- The working directory should be that which holds the model executables (eutro5.exe and toxi5.exe) and the FORTRAN formatting programs (outgen.exe, calgen.exe, calout.exe, modout.exe).
- Segment number "1" should be the first main segment in your model. The main segments should all be numbered first, followed then by the boundary segments.
- All unit conversions encrypted in the scripts are based on the units given in the [tables](#) descriptions. It is important to note that the tables generated from grid used a 100m x 100m cell-size. If different units are used in the tables, then the user must go into the scripts and change the unit conversions.
- Each time you run Input Block A, the main segment parameters table is joined to the main segmentation table. To avoid successive joining, be sure to unjoin the tables before rerunning Input Block A. Have the main segmentation table active and choose Table: Remove all Joins.
- Be sure that each time you process or view an output file, you check the parameters. This step is done by choosing "Check Model Parameters" under either BOD/DO Model or Calibration Model, depending on which out file you will be processing. This step tells ArcView what output file you want to process and reinitializes some of the model options (i.e. time step, print interval, etc) in order to read the output file correctly.
- All charts which you create are ALWAYS "linked" to a given table. If you make changes to that table or to your selections within the table, the chart will change, accordingly. For example, if you have



created a chart of concentration vs. segment at time = 10 days. If you choose to create another chart of concentration vs. segment at time = 19 days, and link it to the same table, the first chart will also change. To avoid this problem, you can create or add multiple copies of the same table to the project and link a chart to each table. You can do this by either adding the dbf file numerous times and renaming the table so that it is more descriptive. Or, you can process the same output a number of times and just keep changing the table name when prompted for a name. Then, only link one chart to each table.

Related Topics:

[Table of Contents](#)

## Troubleshooting

Most errors will occur because the tables or fields are not named correctly. Be sure that the table and field names (or aliases) correspond to those outlined in the [Tables Needed for Processing](#).

Other errors which may occur:

Error: Message Box reading A Nil Object does not recognize the request AsFileName.

Soln: You need to run Input Block A to set the working directory.

Error: The parameters written in the input file are not correct when spot checked.

Soln: The units in your tables may not correspond to the units necessary for the conversions set in the scripts. If they are different, you must either convert them and create a new table with units that agree to those outlined in the [Tables Needed for Processing](#), or you must go into the scripts and change the unit conversions to match those needed for WASP.

Error: Message Box reading: Segmentation Violation!

Soln: For some unknown reason, occasionally, Avenue will print this error. It is unknown why, but usually it does not interfere with the input file generation. Just click the "OK" button and the program should continue without problems. It is smart, though, to save you project often, especially if you are making many changes.

Error: Repeated Fields in the "Main Segmentation" Table

Soln: Each time you run the entire input file, the calibration input file, or Input Block D, the table "flow.dbf" is joined to the "Main Segmentation" Table. Also, the Input Block A joins the Main Segmentation Table to its corresponding parameters table. To correct multiple fields, select the Main Segmentation Table and Choose Table: Remove all Joins from the menu bar. You would then have to rerun Input Blocks A and D (or all the input blocks at once) to rejoin the tables.

Error: "Bug" Icon used to choose a segment for chart creation is not shown on the View Tool Bar.

Soln: Resize the entire ArcView Window and the icon should appear on the second tool bar, on the far right.

Error: Error Box reading "AV Script Out of Range 0-1"

Soln: This message box probably appears when you are trying to select a segment from the view for creating a chart. Be sure that the "Main Segmentation" or "Boundary Segmentation" theme is active on

the view and highlighted before clicking on the segment. If this error box occurs other than this instance, try saving your project, exiting ArcView, and then reopening ArcView and your project.

Error: The FORTRAN output formatting program does not execute or you get a message similar to "out of memory" or "stack overflow".

Soln: The maximum length of the output file to be processed is dependent on the amount of available memory of the computer running the connection. For EUTRO5, 18 lines of text are written for each segment at every time step. In the same way, 7 lines of text are written for each segment at each time step in the TOXI5 output file. If an "out of memory" or "stack overflow" error occurs when trying to process the output data file, the number of times steps written to the output file can be reduced to decrease the number of lines in the WASP5 output files.

Error : Error Box reading nil object segSrcName does not understand the command Make.

Soln: The script, *vwout*, is not accessing the correct directory and/or coverage name. Refer to [Setting up the ArcView/WASP5 Connection](#) on how to correct this problem.

#### Related Topics:

[Table of Contents](#)

## Helpful References

Ambrose, R.B., T.A. Wool, and J.L. Martin. *The Water Quality Analysis Simulation Program, WASP5 Part A: Model Documentation*. Environmental Research Laboratory. Athens, Georgia. September 1993.

Ambrose, R.B., T.A. Wool, and J.L. Martin. *The Water Quality Analysis Simulation Program, WASP5 Part B: The WASP Input Data Set*. Environmental Research Laboratory. Athens, Georgia. September 1993.

Benaman, J. *Modelling Dissolved Oxygen in the Houston Ship Channel using WASP and Geographic Information Systems*. Master's Thesis. Department of Civil Engineering. University of Texas at Austin. December 1996.

Environmental Systems Research Institute, Inc. *Avenue: Customization and Application Development for ArcView*. Redlands, California. 1994.

Environmental Systems Research Institute, Inc. *Understanding GIS: The ArcInfo Method*. Redlands, California. 1995.

[Related Topics:](#)

[Table of Contents](#)

**Appendix H**  
**Houston Ship Channel Water Quality Data**

Station Name	Marker * (km)	1978 Data			1982 Data			Modeled Salinity (ppt)	1992 Avgs			1978 Data			1982 Data	
		Sal High (ppt)	Sal Low (ppt)	Avg Sal (ppt)	Sal High (ppt)	Sal Low (ppt)	Avg Sal (ppt)		Sal High (ppt)	Sal Low (ppt)	Avg Sal (ppt)	DO High (mg/L)	DO Low (mg/L)	Avg DO (mg/L)	DO High (mg/L)	DO Low (mg/L)
Buffalo Bayou @																
West Belt Drive	-42.20															
Shepherd Drive	-14.60	0.40	0.30	0.30	0.40	0.30	0.40					4.20	2.70	3.40	3.90	3.00
IH 45	-10.50				0.40	0.30	0.30								3.20	2.40
US 59	-7.50				0.40	0.20	0.30								4.00	0.80
Lockwood Drive	-4.00				1.00	0.20	0.40								3.70	0.70
69th Street	-1.60	2.90	2.20	2.60	6.40	0.50	3.80					0.00	0.00	0.00	1.90	0.10
HSC @																
Turning Basin	0.00	14.70	7.10	11.80	10.00	3.00	6.00					2.70	0.00	0.20	2.20	0.00
Wharf 20	1.50				9.40	3.80	6.80								1.30	0.00
Brays Bayou	2.90				10.00	1.90	7.00								1.90	0.00
IH 610	3.90	16.70	9.40	13.00	10.00	4.90	7.40					1.70	0.00	0.10	2.40	0.00
Sims Bayou	6.50	16.40	9.40	13.70	10.60	3.20	7.40					1.70	0.00	0.20	2.20	0.00
Vince Bayou	8.50				11.90	6.20	8.20								2.00	0.00
Washburn Tunnel	9.70	17.10	10.90	13.70	13.00	7.70	9.40					2.10	0.00	0.30	2.30	0.00
Hunting Bayou	11.50	13.60	11.30	12.40	13.60	7.40	9.50					1.40	0.00	0.40	2.20	0.00
Greens Bayou	14.80	10.20	14.30	12.90	14.00	7.70	10.00					2.10	0.00	0.50	2.30	0.00
Beltway 8	17.40				14.30	8.50	10.80								2.40	0.00
Patrick Bayou	20.80	14.70	14.00	14.30	18.80	9.70	11.90					7.10	0.30	0.70	1.90	0.00
Carpenter Bayou	23.20	15.40	14.00	14.80	16.00	10.00	11.40					3.20	0.60	1.50	2.50	0.40
Lynchburg Ferry	25.00				19.40	10.60	12.50								4.70	1.30
10	-4.00							3.90								
1	1.60							6.90								
2	5.00							7.13								
3	7.80							7.41								
4	10.10							7.70								
5	13.40							8.17								
6	17.30							8.81								
7	20.40							9.55								
8	23.30							10.10								
17	26.60							10.70								
H12	26.00								15.70	6.10	10.90					
H13	24.00								15.06	3.26	9.16					
H14	22.00								15.45	4.25	9.85					
H15	17.00								14.15	4.75	9.45					
H16	13.00								14.05	5.76	9.96					
H17	8.00								14.65	3.07	7.87					
H18	2.00								13.25	2.32	5.72					
H19	0.00								14.65	1.65	6.45					
H20	-4.00								12.85	0.00	1.82					

\* 0 = Turning Basin

Sources: 1978 and 1982 Data from (TDWR, 1984)  
1992 Data from (Ward and Armstrong, 1992)

279  
1971 Data from (Espey, *et al.*, 1971)

Station Name	Marker * (km)	Avg DO (mg/L)	Modelled				1992 Avgs		1978	1982	Modelled			1992 Avgs			1978	1982	1971
			DO (mg/L)	DO High (mg/L)	DO Low (mg/L)	Avg DO (mg/L)	BOD (mg/L)	BOD (mg/L)	BOD (mg/L)	BOD High (ppt)	BOD Low (ppt)	Avg BOD (ppt)	Flow (cms)	Flow (cms)	Flow (cms)				
Buffalo Bayou @																			
West Belt Drive	-42.20																1.67		
Shepherd Drive	-14.60	3.70					2.00	2.00										3.28	
IH 45	-10.50	3.10						1.00											
US 59	-7.50	1.50						1.50											
Lockwood Drive	-4.00	1.90						4.00											
69th Street	-1.60	0.30					10.00	5.00											
HSC @																			
Turning Basin	0.00	0.20					5.00	1.50											
Wharf 20	1.50	0.20						1.50											
Brays Bayou	2.90	0.10						1.00											4.67
IH 610	3.90	0.20					2.00	1.00											
Sims Bayou	6.50	0.30					4.00	1.00											1.25
Vince Bayou	8.50	0.40						1.00											2.38
Washburn Tunnel	9.70	0.30					4.00	1.00											
Hunting Bayou	11.50	0.40					2.00	1.00											0.99
Greens Bayou	14.80	0.60					2.00	1.00											3.37
Beltway 8	17.40	0.40						1.00											
Patrick Bayou	20.80	0.70					2.00	1.00											
Carpenter Bayou	23.20	1.30					2.00	1.00											0.99
Lynchburg Ferry	25.00	2.40						1.50											
10	-4.00		2.84							2.83									15.83
1	1.60		2.49							5.38									15.83
2	5.00		2.53							5.39									20.5
3	7.80		2.55							5.44									21.75
4	10.10		2.59							5.51									24.13
5	13.40		2.68							5.65									25.12
6	17.30		2.88							5.97									28.49
7	20.40		3.07							6.34									28.49
8	23.30		3.23							6.67									29.48
17	26.60		3.51							7.18									71.98
H12	26.00			6.04	1.24	3.64					12.12	2.72	7.42						
H13	24.00			8.11	1.91	5.01					8.39	0.59	4.49						
H14	22.00			3.44	-0.16	1.64					10.26	-0.14	5.06						
H15	17.00			4.35	0.15	2.25					6.53	0.53	3.53						
H16	13.00			1.78	-0.42	0.68					11.45	1.05	6.25						
H17	8.00			3.71	-0.09	1.81					9.24	0.84	5.04						
H18	2.00			3.86	-1.14	1.36	280				11.08	3.28	7.18						
H19	0.00			3.37	-0.63	1.37					13.87	0.47	7.17						
H20	-4.00			5.63	0.43	3.03					17.34	-1.06	8.14						

\* 0 = Turning Basin

Sources: 1978 and 1982 Data from (TDWR, 1984)  
1992 Data from (Ward and Armstrong, 1992)

1971 Data from (Espey, *et al.*, 1971)

**Appendix I**  
**List of Acronyms and Nomenclature**



## Acronyms

---

aat	arc attribute table
AGNPS	Agricultural Non-Point Source
aml	ArcInfo macro language
ARS	Agricultural Research Service
BOD	biochemical oxygen demand
CBOD	carbonaceous biochemical oxygen demand
CEAM	Center for Exposure Assessment Modeling
CSO	combined sewer overflow
dbf	dBase file
DEM	digital elevation model
DLG	digital line graph
DO	dissolved oxygen
DYNHYD5	WASP5's hydrodynamics subprogram
EH&A	Espey, Huston, and Associates
EPIC	Erosion Productivity Impact Calculator
EROS	Earth Resources Observation System
ESRI	Environmental Systems Research, Inc.
EUTRO4	WASP4's eutrophication subprogram
EUTRO5	WASP5's eutrophication subprogram
ftp	file transfer protocol
GBNEP	Galveston Bay National Estuary Program
GEO-WAMS	Geographically-based Watershed Analysis and Modeling System
GIS	Geographic Information Systems
GLEAMS	Ground Water Loading Effects of Agricultural Management Systems
GRASS	Geographic Resource Analysis System
GRS80	Global Reference System Spheroid 1980
HSC	Houston Ship Channel
HSPF	Hydrologic Simulation Program FORTRAN
ILWIS	Integrated Land and Water Information System
IWMM	Integrated Watershed Management Model
MICRO-FEM	A European groundwater model
MODFLOW	US Geological Service groundwater model

---

### Acronyms (cont.)

---

NaCl	Sodium Chloride (salt)
NAD83 (27)	North American Datum 1983 (1927)
NPS	non-point source
pat	polygon or point attribute table (depends on the type of coverage)
ppt	parts per thousand
SOD	sediment oxygen demand
SWRRBWQ	A Basin Scale Simulation Model for Soil and Water Resources Management
TDWR	Texas Department of Water Resources
TNRCC	Texas Natural Resource Conservation Commission
TOX15	WASP5's toxic chemical subprogram
TWC	Texas Water Commission
USEPA	United States Environmental Protection Agency
USGS	United States Geological Survey
USGS-Albers	US Geological Service - Albers Equal Area Map Projection
vat	value attribute table
WASP	Water Quality Analysis Simulation Program
WASP4	WASP program, version 4 (1983)
WASP5	WASP program, version 5.1 (1993)
WGS84 (72)	World Geodetic System Datum 1984 (1972)

---

### Nomenclature

---

a,b,c,and d	empirical coefficients or exponents
$A_{ij}$	interfacial area shared by segments "i" and "j" ( $m^2$ )
$C_5$	concentration of carbonaceous biochemical oxygen demand (mg/L) (interpreted as total BOD for level one in EUTRO5)
$C_6$	concentration of dissolved oxygen (mg/L)
$C_{bik}$	concentration in boundary segment, "i" (mg/L)
$C'_{bik}$	adjusted concentration for boundary segment "i" (mg/L)
$C_{ik}, C_{jk}$	concentration of chemical "k" in segments "i" and "j" (mg/L)
$C_s$	dissolved oxygen saturation (mg/L)
D	depth of the overlying water column (m)

---

### Nomenclature (cont.)

---

$E_{ij}(t)$	dispersion coefficient time function for exchange "ij" ( $m^2/day$ )
$f_{DS}$	fraction of dissolved CBOD
$k_2$	reaeration rate (/day)
$K_{BOD}$	half saturation constant for oxygen limitation ( $mg\ O_2/L$ )
$k_d$	deoxygenation rate @ 20 °C (/day)
$L_{cij}$	characteristic mixing length between segments "i" and "j" (m)
$M_{ik}$	mass of chemical "k" in segment I (g)
$Q$	channel flow ( $m^3/sec$ )
$Q_{0i}$	upstream inflow into boundary segment, "i" ( $m^3/day$ )
$Q_{bf}$	steady state baseflow upstream of segment "i" ( $m^3/day$ )
$Q_{tot}$	total flow upstream of segment "i"
$R_x$ available	average yearly runoff depth for partially gauged station, averaged over the record available (mm/yr)
$R_x$ 1961-1990	average yearly runoff depth for a given gauge, x, adjusted to represent the entire period, 1961 to 1990 (mm/yr)
$R_y$ available	average yearly runoff depth of four fully gauged stations, averaged over the record available for gauge x (mm/yr)
$R_y$ 1961-1990	average yearly runoff depth of for fully gauged stations, averaged over the entire period of record, 1961 - 1990 (mm/yr)
$S_{bik}$	boundary loading rate response of chemical "k" in segment, "i" ( $g/m^3-day$ )
SOD	sediment oxygen demand @ 20 °C ( $g/m^2-day$ )
T	temperature ( °C)
V	channel velocity (m/sec)
$V_i$	volume of segment i ( $m^3$ )
$v_{s3}$	organic matter settling velocity (m/day)
$\Theta_D$	deoxygenation temperature coefficient (--)
$\Theta_S$	temperature coefficient (--)

---

## Bibliography

- Al-Abed, N. and H.R. Whiteley. 1995. "Modelling Water Quality and Quantity in the Lower Portion of the Grand River Watershed, Ontario." Published in proceedings: *The International Symposium on Water Quality Modeling*, Sponsored by ASAE - The Society for Engineering in Agricultural, Food, and Biological Systems, Orlando, Florida, pp. 213-223.
- Ambrose, R.B., T.A. Wool, and J.L. Martin. 1993. *The Water Quality Analysis Simulation Program, WASP5*. Environmental Research Laboratory, US Environmental Protection Agency, Athens, Georgia.
- Anderson, J.R., E.E. Hardy, J.T. Roach, and R.E. Witmer. 1976. *A Land Use and Land Cover Classification System for Use with Remote Sensor Data*. U.S. Geological Survey, Professional Paper, Reston, Virginia.
- Armstrong, N.E. and G.H. Ward. 1994. *Point Source Loading Characterization of Galveston Bay*. Publication GBNEP-36, Galveston Bay National Estuary Program, Webster, Texas.
- Bales, J.D. and E.R. Holley. 1992. *Field and Numerical Studies of Tracer Gas Transport and Surface Gas Transfer in Laterally Uniform, Partially Stratified Estuaries*. CRWR Technical Report No. 208, Center for Research in Water Resources, Bureau of Engineering Research, The University of Texas at Austin, Austin, Texas.
- Biesheuvel, A. and C.J. Hemker. 1993. "Groundwater modelling and GIS: integrating MICRO-FEM and ILWIS." Published in proceedings: *HydroGIS 93: Application of Geographic Information Systems and Water Resources*, IAHS, Vienna, Austria, pp. 289-296.
- Brown, M.C., D.H. Davis, R.M. Greenwald, P. Srinivassan, and D.S. Ward. 1996. "ModelGIS, An Interface for the USGS Three Dimensional Finite Difference Model MODFLOW." Published in proceedings: *1996 ESRI's User Conference*, Redlands, California.
- Browne, F.X. 1990. "Stormwater Management" in *Standard Handbook of Environmental Engineers*. Corbitt. R.A. ed., McGraw Hill, New York, New York, pp. 7.1-7.135.
- Chen, C.L, L.E. Gomez, C.W. Chen, C.M. Wu, J.J. Lin, and I.L. Cheng. 1995. "An Integrated Watershed Management Model with GIS and Windows Application." Published in proceedings: *The International Symposium on Water Quality Modeling*, Sponsored by ASAE - The Society for Engineering in Agricultural, Food, and Biological Systems, Orlando, Florida, pp. 242-250.

- Chow, V.T., D.R. Maidment, and L. Mays. 1988. *Applied Hydrology*. McGraw Hill, New York, New York.
- Clarke, D. 1993. *Modeling Toxic Materials in Galveston Bay, Texas*. Master's Thesis, Department of Civil Engineering, University of Texas at Austin, Austin, Texas.
- Daly, C., R.P. Neilson, and D.L. Phillips. 1994. A Statistical-Topographic Model for Mapping Climatological Precipitation over Mountainous Terrain. *Journal of Applied Meteorology*, 33: 140-158.
- DePinto, J.V., J.F. Atkinson, H.W. Calkins, P.J. Densham, W. Guan, H. Lin, F. Xia, P.W. Rodgers, T. Slawicki, and W.L. Richardson. 1993. "Development of Geo-WAMS: A Modeling Support System for Integrating GIS with Watershed Analysis Models." Published in Proceedings: *Second International Conference on Integrating GIS with Environmental Modeling*, Breckenridge, Colorado.
- DePinto, J.V., H.W. Calkins, P.J. Densham, J.F. Atkinson, W. Guan, H. Lin and P.W. Rodgers. 1994. "An Approach for Integrating GIS and Watershed Analysis Models." *Microcomputers in Civil Engineering*, Blackwell Publishers, Cambridge, Massachusetts, 9: 251-262.
- Environmental Systems Research Institute, Inc. (ESRI). 1994. *Avenue: Customization and Application Development for ArcView*. Redlands, California.
- Espey, W.H., A.J. Hays, Jr., W.D. Bergman, J.P. Buckner, R.J. Huston, and G.H. Ward, Jr. 1971. *Galveston Bay Project Water Quality Modeling and Data Management: Phase II Technical Report*, Tracor, Inc, Austin, Texas.
- Espey, Huston, and Associates, Inc. (EH&A). 1994. *Analysis of Sanitary Sewer Overflow Effects on Surface Water Quality*, Three Volumes, Houston, Texas.
- Furst, J., G. Girstmair, and H.P. Nachtnebel. 1993. "Application of GIS in Decision Support Systems for groundwater management." Published in proceedings: *HydroGIS 93: Application of Geographic Information Systems and Water Resources*, IAHS, Vienna, Austria, pp. 13-21.
- Galveston Bay Nation Estuary Program (GBNEP). 1996. *Galveston Bay NEP Fact Sheet*. World Wide Web Homepage, Address: <http://www.epa.gov/nep/gulf/gb/>.
- Geter, W.F., P. Smith, C. Drungil, R. Shepherd, and B. Kuenstler. 1995. "Hydrologic Unit Water Quality Model GIS Interface to Four ARS Water Quality Models for Use by Soil Conservation Service." Published in proceedings: *The International Symposium on Water Quality Modeling*, Sponsored by ASAE - The Society for Engineering in Agricultural, Food, and Biological Systems, Orlando, Florida, pp. 341-347.

- Gloyna, E.F. and J.F. Malina. 1964. *Galveston Bay Water Quality Study: Historical and Recent Data*. Center for Research in Water Resources, Report No. 1, Environmental Health Engineering Laboratories, The University of Texas at Austin, Austin, Texas.
- Holley, E.R. 1996. *Personal Communication*. Professor, Department of Civil Engineering, The University of Texas at Austin, Austin, Texas.
- Hydroscience, Inc. 1968. *Galveston Bay Study*. Unpublished report obtained from Dr. G.H. Ward, Center for Research in Water Resources, University of Texas at Austin, Austin, Texas.
- LaWare, P. 1996. *Personal Communication*. Graduate Student, Rice University, Houston, Texas.
- Maidment, D.R. 1992. *A Grid-Network Procedure for Hydrologic Modelling*. Report prepared for the Hydrologic Engineering Center, US Army Corps of Engineers, Davis, California.
- Maidment, D.R. 1993. "Developing a spatially distributed unit hydrograph by using GIS." Published in proceedings: *Application of Geographic Information Systems in Hydrology and Water Resources Management* -- an international conference organized by the International Association of Hydrological Sciences and the United Nations Educational, Scientific, and Cultural Organization, Vienna, Austria.
- Maidment, D.R. 1996. *GIS in Water Resources*. World Wide Web Homepage, Address: <http://www.ce.utexas.edu/prof/maidment/gishydro/>.
- Maidment, D.R., F. Olivera, Z. Ye, S. Reed, and D.C. McKinney. 1996. "Water Balance of the Niger Basin." Published in Proceedings: *ASCE North American Water and Environment Congress '96*, Anaheim, California.
- Mizgalewicz, P. 1996. *Agrichemical Transport in the Midwest Rivers: A GIS Approach*. Ph.D. Dissertation, University of Texas at Austin, Department of Civil Engineering. Austin, Texas, In preparation.
- Newell, C.J., H.S. Rifai, and P.B. Bedient. 1992. *Characterization of Non-Point Sources and Loadings to Galveston Bay*. Publication GBNEP-15, Galveston Bay National Estuary Program, Webster, Texas.
- Olivera, F. 1996. *Personal Communication*. Graduate student, University of Texas at Austin, Department of Civil Engineering: Environmental and Water Resources Engineering, Austin, Texas.

- Olivera, F. and D.R. Maidment. 1996. "Runoff Computation Using Spatially Distributed Terrain Parameters." Published in Proceedings of the *ASCE - North American Water and Environment Congress '96*, Anaheim, California.
- Pilgrim, D.H. and I. Cordery. 1993. "Flood Runoff" in *Handbook of Hydrology*. Maidment, D.R. ed., McGraw Hill, New York, New York, pp. 9.1-9.42.
- Plover, T. 1996. *Personal Communication*. Operational Systems Support Specialist, Learning Resources Center, University of Texas at Austin, Austin, Texas.
- Rindahl, B. 1996. "Groundwater Modelling and Analysis: Using the USGS MODFLOW program and ArcView." Published in proceedings: *1996 ESRI User's Conference*, Redlands, California.
- Saunders, W.K. 1996. *A GIS Assessment of Non-point Source Pollution in the San Antonio-Nueces Coastal Basin*. Masters Report, University of Texas at Austin, Department of Civil Engineering, Austin, Texas.
- Shipley, F.S. and R.W. Kiesling, eds. 1994. *The State of the Bay: A Characterization of the Galveston Bay Ecosystem*. Publication GBNEP-44, Galveston Bay National Estuary Program, Webster, Texas.
- Sternberg, H. 1996. "PAMS-GIS Permit Application Management System." Published in proceedings: *1996 ESRI User's Conference*, Redlands, California.
- Stuart, N. and C. Stocks. 1993. "Hydrological modelling within GIS: an integrated approach." Published in proceedings: *HydroGIS 93: Application of Geographic Information Systems and Water Resources*, IAHS, Vienna, Austria, pp. 319-329.
- Texas Department of Water Resources (TDWR). 1984. *Waste Load Evaluation for the Houston Ship Channel System in and San Jacinto River Basin*. Department Report, Austin, Texas.
- Texas United States Geological Survey (Texas USGS). 1996. *US Geological Survey in Texas*. World Wide Web Homepage, Address: <http://txwww.cr.usgs.gov/>.
- Thomann, R.V. and J.A. Mueller. 1987. *Principles of Surface Water Quality Modeling and Control*. Harper Collins Publishers, New York, New York.
- Tim, U.S. and R. Jolly. 1994. "Evaluating Agricultural Nonpoint-Source Pollution Using Integrated Geographic Information Systems and Hydrologic/Water Quality Model." *Journal of Environmental Quality*, 23: 25-35.

- U.S. Environmental Protection Agency (USEPA). 1980. *A Water Quality Success Story: Lower Houston Ship Channel and Galveston Bay, Texas*. Office of Water Planning and Standards, Washington, D.C. 20460.
- U.S. Environmental Protection Agency (USEPA). 1996. *US Environmental Protection Agency*. World Wide Web Homepage, Address: <http://www.epa.gov>.
- United States Geological Survey -- US GeoData (USGS). 1993. *1:100,000 Scale Digital Line graphs (DLG) Data: Hydrography and Transportation - Optional Format - Area 8 - Texas and Oklahoma*. CD ROM, Reston, Virginia.
- United States Geological Survey. 1996. *EROS Home Page*. World Wide Web Homepage, Address: <http://sun1.cr.usgs.gov/eros-home.html>.
- Visnovsky, K. 1996. *Personal Communication*. Texas Natural Resource Conservation Commission, Toxicity Evaluation Team, Research and Environmental Assessment Section, Water Planning and Assessment Division, Austin, Texas.
- Ward, G.H. 1993. *Dredge and Fill Activities in Galveston Bay*. Publication GBNEP-28, Galveston Bay National Estuary Program, Webster, Texas.
- Ward, G.H. and N.E. Armstrong. 1992. *Ambient Water and Sediment Quality of Galveston Bay: Present Status and Historical Trends*. Publication GBNEP-22, Galveston Bay National Estuary Program, Webster, Texas.
- Ye, Z. 1996. *Map-Based Surface and Subsurface Flow Simulation Models: An Object-Oriented and GIS Approach*. Ph.D. Dissertation, University of Texas at Austin, Department of Civil Engineering, Austin, Texas, In preparation.