NOS Technical Report OCS/CO-OPS 1

THE CHESAPEAKE BAY OPERATIONAL FORECAST SYSTEM (CBOFS): TECHNICAL DOCUMENTATION

Silver Spring, Maryland
December 2000

noaa National Oceanic and Atmospheric Administration

U.S. DEPARTMENT OF COMMERCE
National Ocean Service

National Ocean Service
National Oceanic and Atmospheric Administration
U.S. Department of Commerce

Office of Coast Survey

The Office of Coast Survey (CS) is the national hydrographic office. As the oldest United States scientific organization, dating from 1807, this office has a long history. Today it promotes safe navigation by managing the National Oceanic and Atmospheric Administration's (NOAA) nautical chart and hydrographic survey programs.

There are four components of CS:

The Coast Survey Development Laboratory develops new and efficient techniques to accomplish Coast Survey missions and to produce new and improved products and services for the maritime community and other coastal users.

The Marine Chart Division collects marine navigational data to construct and maintain nautical charts, Coast Pilots, and related marine products for the United States.

The Hydrographic Surveys Division directs programs for ship and shore-based hydrographic survey units and conducts general hydrographic survey operations.

The Navigation Services Division is the focal point for Coast Survey customer service activities, concentrating predominantly on charting issues, fast-response hydrographic surveys and Coast Pilot updates.

Center for Operational Oceanographic Products and Services

The NOS Center for Operational Oceanographic Products and Services (CO-OPS) collects and distributes observations and predictions of water levels and currents to ensure safe, efficient and environmentally sound maritime commerce. CO-OPS provides the set of water level and coastal current products required to support NOS' Strategic Plan mission requirements, and assists in providing operational oceanographic data/products required by NOAA's other Strategic Plan themes. For example, CO-OPS provides data and products required by the National Weather Service to meet its flood and tsunami warning responsibilities. The National Water Level Observation Network (NWLON) and a national network of Physical Oceanographic Real-Time Systems (PORTS) in major US harbors is managed by CO-OPS. This Office also establishes standards for the collection and processing of water level and current data; collects and documents user requirements which serve as the foundation for all resulting program activities; designs new and/or improved oceanographic observing systems; designs software to improve CO-OPS' data processing capabilities; maintains and operates oceanographic observing systems; performs operational data analysis/quality control; and produces/disseminates oceanographic products.

NOS Technical Report OCS/CO-OPS 1

THE CHESAPEAKE BAY OPERATIONAL FORECAST SYSTEM (CBOFS): TECHNICAL DOCUMENTATION

Thomas F. Gross
Office of Coast Survey
Kathryn Thompson Bosley
Center for Operational Oceanographic Products and Services
Kurt W. Hess
Office of Coast Survey

December 2000

noaa National Oceanic and Atmospheric Administration

U.S. DEPARTMENT OF COMMERCE
Norman Y. Mineta, Secretary

National Oceanic and Atmospheric Administration
D. James Baker, Under Secretary

National Ocean Service
Margaret Davidson
Acting Assistant Administrator

Office of Coast Survey
Captain David MacFarland

Center for Operational Oceanographic Products and Services
David M. Kennedy

NOTICE

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

ASCII	American Standard Code for Information Interchange
CAFE	Chesapeake Area Forecast Experiment
CBBT	Chesapeake Bay Bridge Tunnel
CBOFS	Chesapeake Bay Operational Forecast System
CMAN	Coastal Marine Automated Network
CO-OPS	Center for Operational Oceanographic Products and Services (NOS)
CORMS	Continuous Operational Real-Time Monitoring System
CSDL	Coast Survey Development Laboratory (NOS)
DPAS	Data Processing and Analysis Software
ETSS	Extra-Tropical Storm Surge (model)
FTP	File Transfer Protocol
IDL	Interactive Data Language (by Research Systems, Inc.)
ISD	Information Services Division (NOS)
MECCA	Model for Estuarine and Coastal Circulation Assessment
MLLW	Mean Lower Low Water
MSL	Mean Sea level
NWLON	National Water Level Observational Network
NCEP	National Centers for Environmental Prediction (NWS)
NOAA	National Oceanic and Atmospheric Administration
NOS	National Ocean Service
NWS	National Weather Service
ODAAS	Operational Data Acquisition and Archive System
OSO	Office of Systems Operations (NWS)
PORTS	Physical Oceanographic Real-Time Systems
PUFFF	PORTS Universal Flat Format File
SGI	Silicon Graphics, Inc.
SQL	Structured Query Language
TDL	Techniques Development Laboratory (NWS)
URL	Universal Resource Locator

1. INTRODUCTION

Mariners operating in the Chesapeake Bay presently use the National Oceanic and Atmospheric Administration/National Ocean Service (NOAA/NOS) tide tables as aides to navigation. These products are tidal predictions based solely on astronomical forcing. However, nontidal forcing is significant in the Bay and at times completely overwhelms the tidal signal, causing safety problems for mariners who have made decisions based on the tidal predictions (Figure 1). A method of predicting the departures from the astronomical tide would overcome this shortcoming.

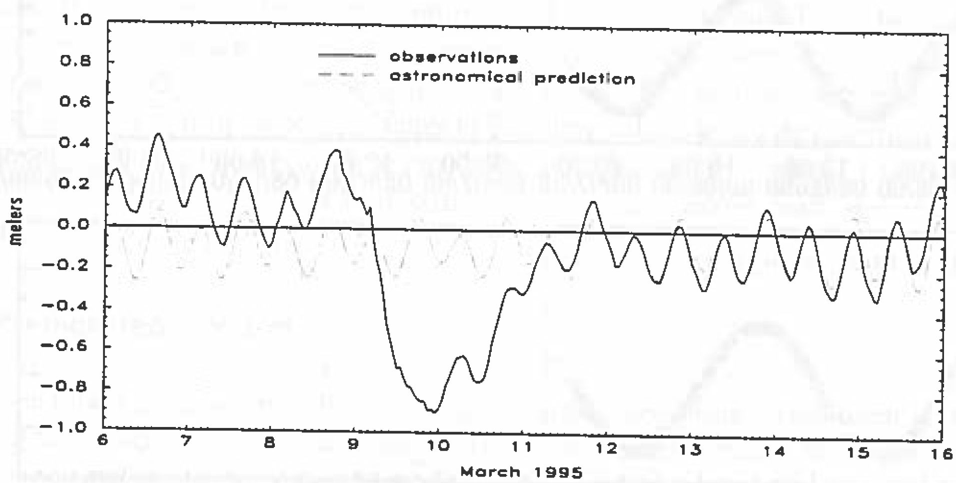

Figure 1. The observed (solid) and astronomically predicted (dashed) water level at Baltimore during a winter storm in March 1995. The rapid draining of the Harbor was caused by strong northwesterly winds, causing the water levels to be as much a 1 meter below the astronomically-predicted level.

To respond to this need, NOS has implemented the Chesapeake Bay Operational Forecast System (CBOFS), which is aimed at enhancing predictions of water level within the Chesapeake Bay. The ultimate objective of CBOFS is to provide enhanced real-time and forecast marine information for use by the commercial, governmental, and recreational maritime communities. The present version of CBOFS produces twice-daily nowcasts and forecasts of total water level in the Bay and distributes them as graphical products. Future enhancements may include the prediction of current velocity, wave, temperature, and salinity information.

1.1. Products

CBOFS runs a model twice a day to produce a nowcast of water levels for the most recent day and a forecast of the water levels for the next 24 hours. A sample plot showing simulated and observed water levels appears in Figure 2. These graphs show the water level height at each of several water level measurement stations throughout the bay. Additional graphics include plots of winds and water levels superimposed over a map of the Bay. All the graphics are updated with each new run of the model and posted on the Website (http://co-ops.nos.noaa.gov/CBOFS).

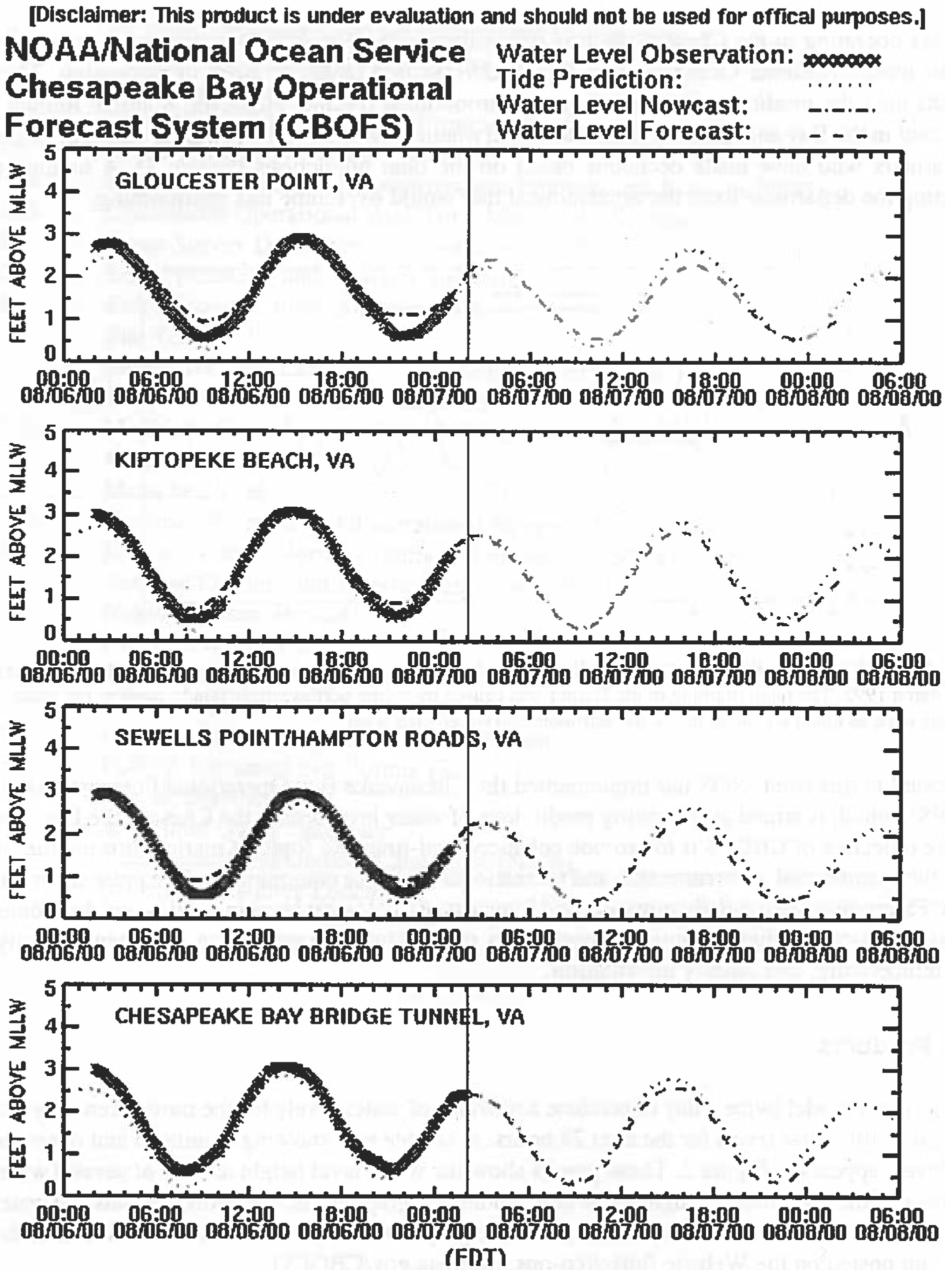

Figure 2. Sample of observed, nowcast, and forecast water levels plots produced by CBOFS.

1.2. Potential Benefits

Perhaps the most important benefit of accurate nowcasts and forecasts of water level in any estuary is the increased margin of safety provided o the maritime community. Over 1,000 groundings were reported in just five of the Nation's estuaries (New York/New Jersey, Boston, San Francisco, Tampa, and Houston/Galveston) from 1981-1995 (Kite-Powell, 1996). Since nearly half of all cargo carried by ships in U.S. waters is either petroleum or hazardous materials, minimizing the likelihood of maritime accidents is essential to the protection of the environment (NOAA, 1994).

Maximizing the efficiency of maritime commerce is another very significant benefit of water level nowcasting and forecasting. The U.S. economy is heavily dependent upon marine transportation. In fact, more than 95 percent (by weight) of all international commerce is transported by water. Many of today's deep-draft vessels can enter U.S. waters only at high tide, and the time spent waiting can be extremely costly. Accurate water level information will allow Chesapeake Bay port authorities and maritime shippers to maximize loading and vessel draft, without compromising vessel safety.

1.3. Project Background

CBOFS is a direct descendent of the Chesapeake Area Forecasting Experiment (CAFE), in which the numerical model and data streams were first developed and tested for Chesapeake Bay. CAFE was developed and run in experimental mode in the Coast Survey Development Laboratory (CSDL) and the output was evaluated (Bosley, 1996; Bosley and Hess, 1997). Following testing, the daily acquisition of observational and forecast data needed to drive the system was implemented on NOS' computers. A computed-based system to handle the access and processing of these data, called the Operational Data Acquisition and Archive System (ODAAS), was developed next (Kelley et al., 2000). Protocols for the development of forecast systems and standards for skill assessment were established (NOS, 1999).

A system development plan was formulated (Bosley et al., 1997) and plans begun for the transition to operational status (Bosley et al., 1999). An updated version of the hydrodynamic model (Hess, 1989) was documented (Hess, 2000). A skill assessment that followed the NOS procedures was completed (Gross, 2000). CAFE, renamed CBOFS to reflect its operational nature, was ported in the fall of 1999 to NOS' Center for Operational Oceanographic Products and Services (CO-OPS), which maintains an operational computer environment.

1.4. Skill Assessment

Before the CBOFS model could be released to operational status, NOS standards (NOS, 1999) required a quantitative assessment of the ability of the model to consistently reproduce observed water levels. This skill assessment was completed (Gross, 2000) and consisted of comparing the model's performance against real data and astronomical tide only simulations. Differences in observed and model simulated water level height at eleven tide stations were statistically examined.

NOS (1999) prescribed conditions were met by the CBOFS model at most stations. Among other conditions it is required that the modeled water levels are within 6 in (15 cm) of the observed values at least 90% of the time. The model was able to pass water level nowcast/forecast accuracy requirements for the test period with the exception of a small number of storm events. The forecast model was shown to be more accurate than both the tide table prediction and a 'persistence' forecast (a prediction that added the observed departure from the mean to the astronomical prediction) and can thus be a useful improvement upon printed tidal tables for mariners.

1.5. About This Document

The purpose of this document is to describe in detail the CBOFS in terms of its NOS-developed component models, operations and functions, programs and scripts, and files. An effort was made to cover all components, but not necessarily in equal detail. Some components are described in more detail elsewhere; reference is made to the supporting documents where appropriate.

The organization of this document is as follows. A system overview section is followed by a section which details the model system which is used in all three modes of operation. The next three sections present the details of the initialization, nowcast, and forecast modes, respectively. Production and dissemination of graphical output is detailed in a separate section because it relies on the results of all three modes. The final section of this document prescribes the tasks involved in maintenance of CBOFS and provides some troubleshooting tips.

A few words about formats used here are required. Variables appearing in the programs and variables in equations are given in italics; for example, the wind components in the hydrodynamic model are *WX* and *WY*. Unix scripts will have the extension 'sh', and computer programs will have the extension 'f' for Fortran77, 'f90' for Fortran90, 'c' for C, and 'pro' for the Interactive Data Language (IDL). The names of computer programs, scripts, and files are given in courier font; for example, the script that runs the nowcast is `CBOFSNOW.sh`.

In September 2000, during the course of preparation of this document, the atmospheric model used to supply the meteorological forcing changed from the Eta32 to the Eta22. Except for the grid (the newer model has a nominal cell size of 22 km), the models are similar, and only minor changes in the CBOFS coding were required. Therefore, we have retained the original references to the Eta32 model in the text and figures.

2. SYSTEM OVERVIEW

CBOFS is based on running a hydrodynamic model to produce simulations of water levels in Chesapeake Bay. Both recent observations and forecasts of water levels and winds are used to drive the model. The hydrodynamic model, observations, data files, and data links comprise the system.

The hydrodynamic model presently used is the Model for Estuarine and Coastal Circulation Assessment (MECCA) (Hess, 1989; 2000). This is a two-dimensional numerical model that requires winds and ocean water levels for forcing. The model is discussed in more detail in Section 3. Observed water levels near the bay's entrance at the Chesapeake Bay Bridge Tunnel (CBBT) are used to represent the water level on the adjacent continental shelf. Predictions of future water levels are obtained from the National Weather Service's (NWS') Techniques Development Laboratory (TDL), which runs the Extra-Tropical Storm Surge (ETSS) model (Chen et al., 1993; Kim et al., 1996). Observed winds are obtained from CBBT and from Thomas Point in the upper Bay. Forecasts of winds are obtained from NWS, which runs the Eta32 atmospheric model (Black, 1994). Locations of the water level and wind gauges are shown in Figure 3.

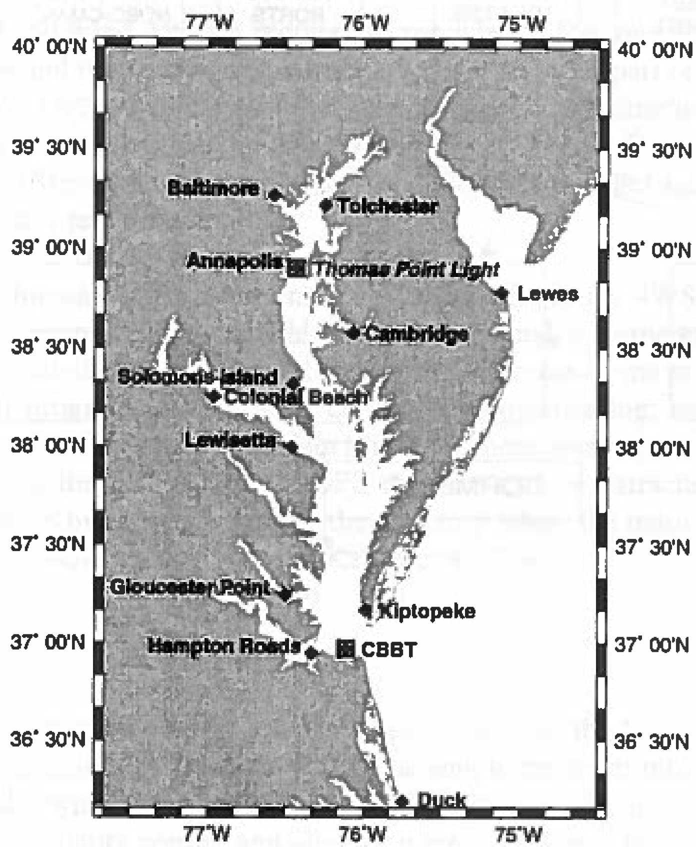

Figure 3. Location of the water level (◆) and wind stations (□) used in CBOFS.

CBOFS is implemented by Unix scripts and programs which assemble the various elements. The system consists of three component operations:

- the initialization hindcast,
- the nowcast, and
- the forecast.

The system must prepare the observed data by gathering them from various sources, quality checking them, filling instrument data gaps, and formatting the data for direct reading by the MECCA program. Similarly, data from the atmospheric forecast model and the coastal sea level forecast model are gathered and formatted for direct reading by the MECCA program. The scripts also execute the hydrodynamic model runs and create the graphical output. An overview of the system appears in Figure 4.

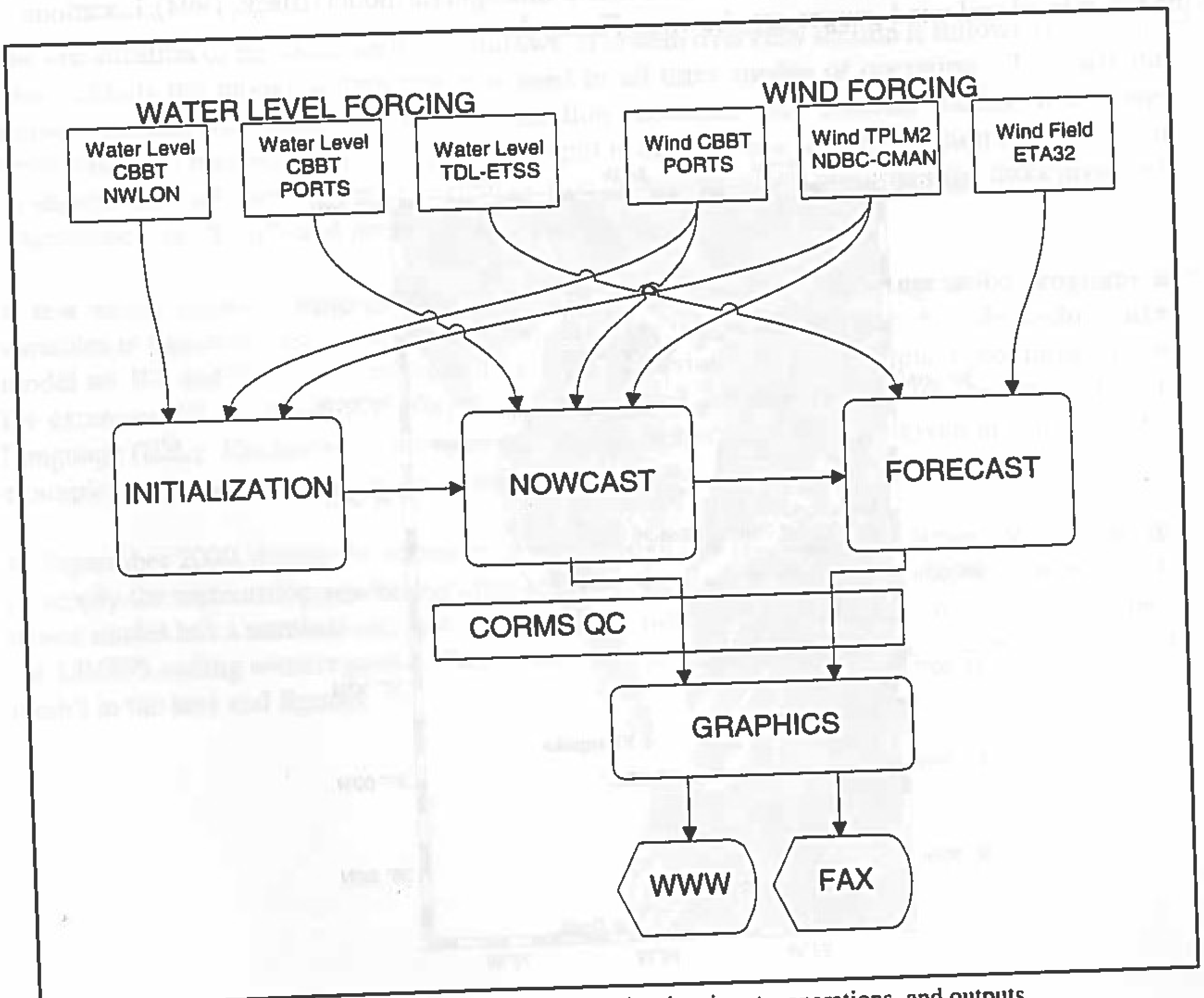

Figure 4. Schematic of the CBOFS system depicting major data inputs, operations, and outputs.

The following four subsections will briefly describe the four functions common to each of the three CBOFS operations. The functions are:

- data ingest,
- generation of forcing fields,
- model runs, and
- data dissemination and archives.

2.1. Data Ingest

A large and complicated part of the CBOFS system is the acquisition of the observational and forecast model data (water levels and winds) needed to force the model runs. Data are downloaded from operational data centers including: the NWS Office of System Operations (OSO), the NWS National Centers for Environmental Prediction (NCEP), the Chesapeake Bay Physical Oceanographic Real Time System (CB PORTS) and the National Water Level Observation Network (NWLON). The OSO and NCEP data centers are accessed by FTP across the Internet. CO-OPS maintains, monitors and archives data from CB PORTS and NWLON, thus the data from these systems are accessed directly from the locally held databases.

Observed water level and wind data are required to provide the forcing conditions for the model. The water level gauge and meteorological station at CBBT provides part of this data. CBBT data are delivered to the CO-OPS computer network every 6 minutes as a function of CB PORTS. The only other required real-time data are the winds observed at Thomas Point Light. Thomas Point Light is a NWS Coastal Marine Automated Network (CMAN) station and data are available through the OSO system in nearly real time.

The Eta32 and ETSS forecast models are run in operational mode on NWS computer systems, to which direct access is commonly unavailable. However, special agreements with the operators of the Eta32 and ETSS models allow ODAAS fast and direct access to the model output files. The ODAAS is a suite of programs which takes care of the downloading, preprocessing and local archiving of these data sets. The raw output data files from these systems are quite large and require some processing before the data which CBOFS requires can be extracted. The CBOFS takes advantage of the ODAAS by simply linking to the directory where the required files will be found. Complete details for ODAAS can be found in Kelley et al. (2000).

2.2. Generation of Forcing Fields

Once all the input data have been collected from remote sources, the fields necessary to force the model runs must be generated. In some cases this is a simple matter of reformatting data to create the ASCII or binary files which are read by the MECCA. But for some data, which may be missing or corrupted, additional quality control and alteration are necessary. The input data are either the water level data or the wind data. The water level data are reformatted, quality controlled, corrected and gap-filled by the Fortran program `gentidenew.f`, which is used in initialization, nowcast, and forecast. The wind data are reformatted and altered into a binary, two-dimensional grid suitable for ingestion by MECCA by two programs; `genwind_2obsB.f` used with Thomas Point and CBBT observed winds for initialization and nowcast, and `genwind_bin.f` used with the Eta32 winds for the forecast.

In nowcast mode, the model is forced by estimated water levels for the continental shelf that are based on the levels measured at the CBBT, and wind measurements from Thomas Point and CBBT. Gaps in water level data are filled with astronomical tidal calculations with a persisted subtidal water level. The subtidal level is based on a linear interpolation of the previous and next valid subtidal value. The subtidal value is the observed water level value minus the predicted tide. A wind field covering the whole Bay is generated from the values at the two points. Gaps in wind data are filled by linearly interpolating the x- and y-components, or if the gap is greater than 12 hours, filled by ramping down over 6 hours to zero values then ramping back up over 6 hours to the next available wind value.

In forecast mode, the model is forced by water level predictions for CBBT from the TDL's ETSS model. The ETSS model is a non-tidal forecast, so an astronomical tidal prediction using NOS accepted constituents for CBBT must be added. The TDL model is run twice daily, providing 0:00 and 12:00 forecasts of hourly water levels for the next 48 hours. The forecast wind fields are obtained from the 32-km Eta wind model run. The wind forecast is read, and wind field for the Bay is created by spatially interpolating values from the forecast grid. Eta32 is run daily at 0000 and 1200 GMT, and produces forecast wind fields for the projections of 3, 6, 9, 12, 15, 18, 21, and 24 hours.

Occasionally either one or both of the forecast models may not have run or may not have reported their results in time for the CBOFS model. When possible, a previous forecast model cycle may be used. Lacking any useable forecast model results, the back-up water level forecast consists of astronomical tides with an added value equal to the last reported non-tidal water level. The back-up wind field is based on the use of the last good observations.

2.3. Model Runs for the Initialization, Nowcast, and Forecast Cycles

Once the required forcing data have been collected and processed, CBOFS runs three operations: initialization, nowcast and forecast. The purpose of the initialization is to provide the initial conditions file describing the state of the model from which the nowcast can be run. CBOFS initialization is produced by running from a state of no motion for 8 days of spin-up to a state of full motion. The input data for the initialization are quality-controlled water level observations from the CO-OPS Data Processing and Analysis Software (DPAS). The nowcast uses the results of the initialization as initial conditions for a run which provides the state of the water level in the Bay for the past 18 or 30 hours, depending on the time of the run. This run uses the latest observed winds and water levels from locations around the Bay. The nowcast produces results for plotting and dissemination as well as providing an initial condition file to be used by the forecast. The forecast uses the nowcast initial condition file, forecast winds from the NWS's Eta32 atmospheric model, and forecast oceanic water levels from the ETSS model to predict water levels 24 hours into the future. Forecast output files are created for plotting and dissemination. After the nowcast and forecast files are created, the results and reformatted observation data are plotted and the graphics disseminated via the Web and fax.

The daily sequence of operations of CBOFS is as follows. The initialization is run once a day, at 1235 UTC. The initialization gathers forcing data from the NWLON archives for the past 8 days. The model is run from a state of no motion and brought up to full motion, providing the initial condition file that is valid for the 0000 UTC hour. The first nowcast run occurs at 1810 UTC following the 1235 UTC initialization and covers 18 hours, thereby providing an initialization field valid for 1800 UTC. The first 24-hour forecast is run immediately following the nowcast. The second nowcast begins at 610 UTC of the next day, and uses the same initial conditions file valid at 0000 UTC. This nowcast covers 30 hours. Immediately after this nowcast, the second 24-hr forecast is run. A schematic appears in Figure 5.

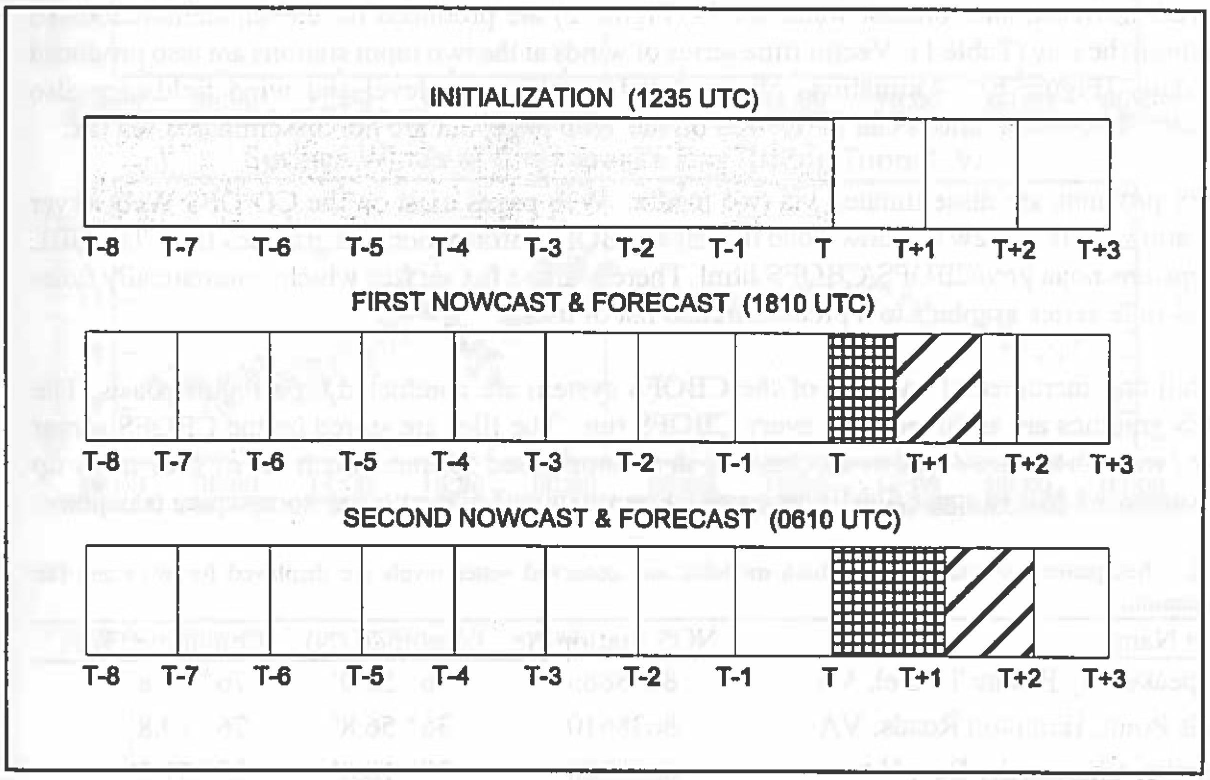

Figure 5. Schematic showing the start and end times for which the daily 8-day-long initialization (solid gray), the 18- or 30- hr nowcast (cross hatching), and the 24-hr forecast runs (diagonal lines) are valid. The horizontal axis represents time, with T representing the start of the day. The UTC time above each bar shows the time that the run begins on the computer.

The initialization-nowcast-forecast sequence described above was chosen to mitigate as many problems as possible. In another common sequence, every nowcast produces an initial conditions file for the next nowcast (as well as for the forecast). Operationally, this is a difficult system to maintain since it requires operator intervention to restart in the event of a failure such as a computer failure or lengthy forcing data dropout. Also, the influence of corrupted data may be propagated through subsequent runs. Corrupted or data with missing records are common due to the real-time nature of CBOFS. However, in the current configuration, restarting following a computer outage is automatic after the computer is restored and requires no operator intervention. Because the initialization simulates the past with non real-time data, the data can be obtained from the NWLON database which contains more thoroughly quality-controlled data and is continuously monitored by the CO-OPS personnel.

2.4. Output Products, Dissemination, and Archives

Two types of graphical output are produced by CBOFS. Time series graphics depicting two days of observed, nowcast, and forecast water levels (Figure 2) are produced for eleven stations located throughout the Bay (Table 1). Vector time series of winds at the two input stations are also produced twice daily (Figure 6). Animations of simulated hourly water level and wind fields are also produced. These animations can be viewed on the Web page, but are not disseminated via fax.

CBOFS products are disseminated via two media. Web pages exist on the CO-OPS Web server which allow users to view and download the latest CBOFS information and graphics files. The URL is co-ops.nos.noaa.gov/CBOFS/CBOFS.html. There is also a fax service which automatically faxes CBOFS time series graphics to a predetermined list of users.

Both full and incremental backups of the CBOFS system are conducted on a regular basis. The CBOFS graphics are archived after every CBOFS run. The files are stored on the CBOFS server (`/var/www/htdocs/cafe/archive`) in a compressed format. Each set of files takes up approximately 1 MB of space (2 MB per day). Currently no off-site archival storage plan is in place.

Table 1. Chesapeake Bay stations for which modeled and observed water levels are displayed for Web and fax dissemination.

Station Name	NOS Station No.	Latitude (N)	Longitude (W)
Chesapeake Bay Bridge Tunnel, VA	8638863	36° 58.0'	76° 6.8'
Sewells Point, Hampton Roads, VA	8638610	36° 56.8'	76° 19.8'
Kiptopeke, Chesapeake Bay, VA	8632200	37° 10.0'	75° 59.3'
Gloucester Point, York River, VA	8637624	37° 14.8'	76° 30.0'
Lewisetta, Potomac River, VA	8635750	37° 59.8'	76° 27.8'
Colonial Beach, Potomac River, VA	8635150	38° 15.1'	76° 57.6'
Solomons Island, Patuxent River, MD	8577330	38° 19.0'	76° 27.2'
Cambridge, Choptank River, MD	8571892	38° 34.5'	76° 4.3'
Annapolis, Severn River, MD	8575512	38° 59.0'	76° 28.8'
Baltimore (Fort McHenry), MD	8574680	39° 16.0'	76° 34.7'
Tolchester Beach, Chesapeake Bay, MD	8573364	39° 12.8'	76° 14.7'

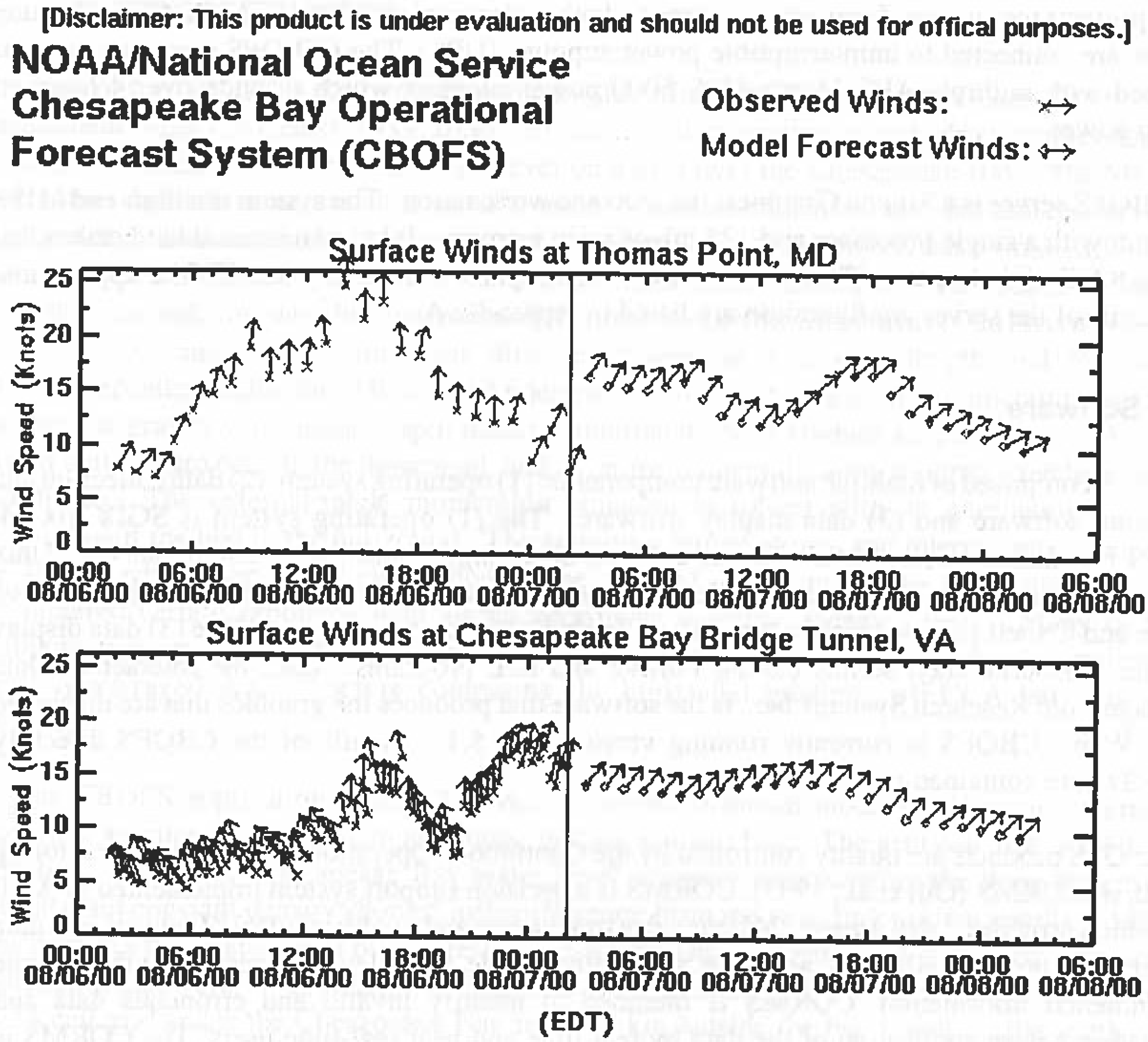

Figure 6. Sample wind time series CBOFS output.

2.5. Operational Environment

2.5.1. Hardware

The CBOFS hardware system must be maintained in a secure environment. The majority of CBOFS computer hardware is located in the CO-OPS' floor computer facility (Room 6247) in SSMC4 at 1305 East West Highway, Silver Spring, Maryland. This room is environmentally controlled; air temperature and humidity are electronically maintained to precise levels by two Liebert System 3 air conditioning units. The electrical power is conditioned by a Liebert Precision Power Center which protects the systems from power surges and other electrical problems. The CBOFS computer systems are connected to uninterruptible power supplies (UPS). The CO-OPS computer room is equipped with multiple APC Matrix-UPS 5000 power supplies which provide over 4 hours of backup power.

The CBOFS server is a Silicon Graphics, Inc., Octane workstation. The system is a high-end MIPS computer with a single processor and 128 mb of main memory. It has two internal hard drives that provide 8 GB of disk space. The system is more than capable of handling the CBOFS application. The details of the server configuration are listed in Appendix A.

2.5.2. Software

CBOFS is comprised of multiple software components: (1) operating system, (2) data collection and processing software and (3) data display software. The (1) operating system is SGI's Irix 64. CBOFS is running version 6.5.3. This is a 64-bit operating system that is compliant with Unix System V, Release 4. The (2) data collection and processing software is comprised of various Bourne and C shell scripts which in turn call multiple Fortran and C programs. The (3) data display software is Bourne shell scripts calling Fortran and IDL programs. IDL, the Interactive Data Language from Research Systems Inc., is the software that produces the graphics that are displayed on the Web. CBOFS is currently running version IDL 5.1. Details of the CBOFS directory (`CBOFS/`) are contained in Appendix B.

All CO-OPS products are quality controlled by the Continuous Operational Real-Time Monitoring System, or CORMS (Gill et al., 1997). CORMS is a decision support system implemented in April 1998 which provides 7 day a week, 24 hour a day monitoring and quality control of sensors and data in order to insure the availability, accuracy, and quality of tide, water level, current, and other marine environmental information. CORMS is intended to identify invalid and erroneous data and information before application of the data by real-time and near real-time users. The CORMS is currently monitored by teams of technicians in the Silver Spring, Maryland, offices of the National Weather Service.

A Standard Operating Procedure (SOP) for CORMS monitoring of CBOFS is available to the Operators. This SOP outlines the steps to be taken twice daily by the Operators to insure the availability and quality of the various CBOFS products. The SOP contains details regarding the products and the persons to be e-mailed regarding failures.

3. THE MODEL SYSTEM

The core of the CBOFS is the hydrodynamic model, Model for Estuarine and Coastal Circulation Assessment (MECCA) (Hess, 1989; 2000). By numerical integration of the hydrodynamic equations of motion, MECCA simulates the water level on a grid over the Chesapeake Bay. The MECCA code solves the hydrodynamic equations of motion for momentum, mass, salt, and heat conservation. The model is three-dimensional in space, uses a vertical sigma coordinate, has a time-varying free surface, and incorporates non-linear horizontal momentum advection. The model includes a three-dimensional time variable, horizontal diffusion based on the Smagorinsky diffusivity (see Tag et al., 1979), and vertical turbulent diffusion based on a mixing length and Richardson number-dependent reduction (Munk and Anderson, 1948). For the horizontal momentum equations, the external gravity wave mode is split from the internal mode. Variables are placed on an Arakawa C-grid with square cells in the horizontal, and at uniform intervals along a sigma-stretched vertical coordinate. The external-mode momentum equation is solved with an alternating-direction, semi-implicit method in the horizontal. The salinities, temperatures, and internal-mode velocities are solved with a semi-implicit method in the vertical. In recent years, some modelers have encountered certain problems with sigma coordinate systems (Haney, 1991). Many of these problems can be ameliorated by using uniformly-spaced sigma levels and by subtracting the spatially-averaged density before computing the horizontal gradient; MECCA has both these features.

For the CBOFS application, MECCA is run in two-dimensional mode with barotropic pressure gradients, solving for momentum and mass, but not salt and heat. The grid cell size is 5.606 km. Sensitivity studies of Chesapeake Bay water level response incorporating the three-dimensional currents and constant density showed little difference from the two-dimensional results. Also, test runs with cell dimensions half of the present size showed little difference in simulated water levels, confirming that water level changes over distances on the order of 5 km are not significant. The 55 by 34 cell grid spans the Chesapeake Bay from 15 km outside the bay mouth in the south to the mouth of the Susquehanna River in the north (Figure 7). The orientation of the MECCA grid is described in Figure 8.

Depths are specified by the bathymetry data set developed at the U.S. Naval Academy (Hoff, 1990) and span depths of 1 m to 19 m. Triangular cells have the same placement of variables but have half the cell area of a square cell. Upper reaches of rivers are modeled as narrow channels and river discharges are included. Modifications to the software for CBOFS include elimination of the three-dimensional temperature and salinity routines and the addition of a routine to save wind and water level fields for post-processing.

The model is forced with (1) stress at the air-water interface due to wind, (2) water levels at the coastal open boundary to the waters of the continental shelf, and (3) river flow. MECCA parameterization, calibration and validation are discussed in the subsequent sub-sections, followed by the logic and equations used to formulate the atmospheric and open boundary conditions.

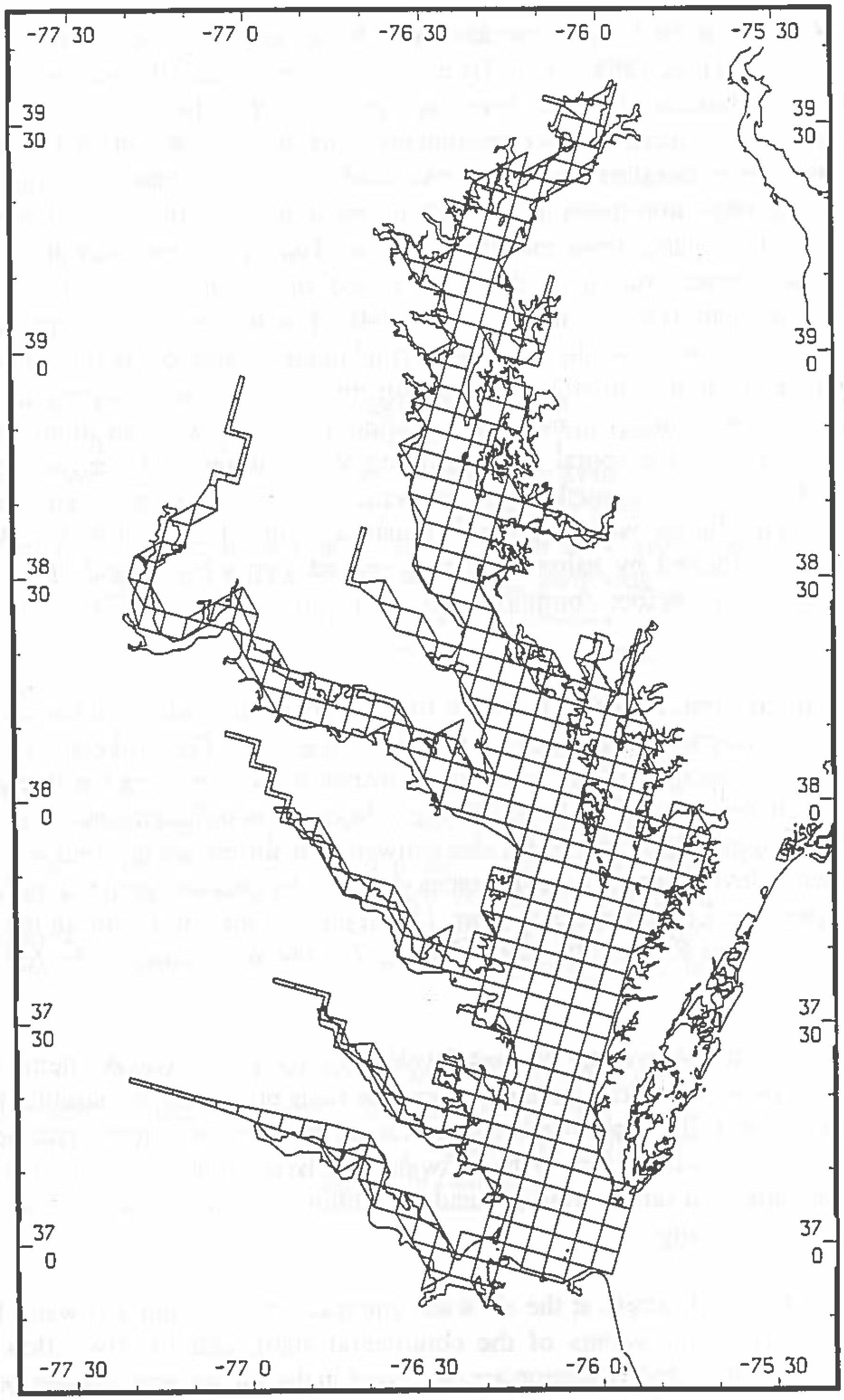

Figure 7. The MECCA model grid for Chesapeake Bay. Grid cells are 5.606 kilometers on a side.

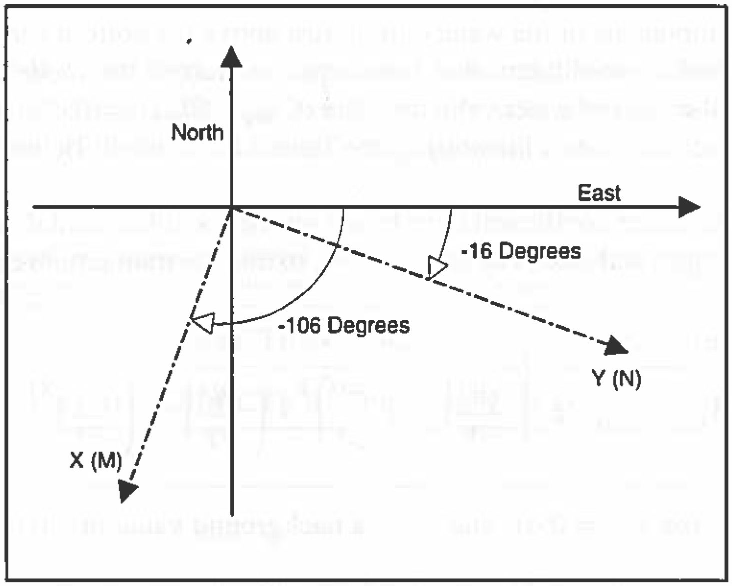

Figure 8. Sketch (not to scale) showing the orientation of MECCA grid coordinates (dashed lines) relative to geographic coordinates (solid lines).

3.1. Model Parameters

A complete description of the MECCA model, underlying equations, boundary conditions, and parameterization of drag coefficients and wind stress calculations are found in the MECCA Program Documentation (Hess, 1989; 2000). The three variables whose formulation often varies from model to model, namely surface wind stress, bottom stress, and horizontal diffusion, are described here.

Wind stress components τ_{sx} and τ_{sy} are calculated by

$$\tau_{sx} = \rho_a(C_{DAW1} + C_{DAW2}(W_x^2 + W_y^2)^{1/2})(W_x^2 + W_y^2)^{1/2} W_x$$

$$\tau_{sy} = \rho_a(C_{DAW1} + C_{DAW2}(W_x^2 + W_y^2)^{1/2})(W_x^2 + W_y^2)^{1/2} W_y$$

where W_x and W_y are components of the wind velocity vector referenced to 10 meters above the water surface and ρ_a is the density of air (1.2 kg/m^3). The drag coefficients are based on those of Large and Pond (1981), but adjusted to better simulate Chesapeake Bay. Here (C_{DAW1} , C_{DAW2}) = (0.0008 , 0.000065) in MKS units.

Bottom stress components τ_{bx} and τ_{by} are calculated by

$$\tau_{bx} = \rho_w(C_{DWB1} + C_{DWB2}(U_b^2 + V_b^2)^{1/2})U_b$$

$$\tau_{by} = \rho_w(C_{DWB1} + C_{DWB2}(U_b^2 + V_b^2)^{1/2})V_b$$

where U_b and V_b are components of the water current just above the bottom and ρ_w is the density of water (1000 kg/m^3). In the two-dimensional barotropic version of the model, the U_b and V_b are components of the depth-averaged water velocity. Here (C_{DWB1}, C_{DWB2}) = (0.0007, 0.0) in MKS units; therefore, the CBOFS version uses a linear drag coefficient for sea bed friction.

Horizontal turbulent exchange coefficients are based on the local horizontal velocity shear and a length scale equal to the grid cell size (Tag et al., 1979), so that the momentum exchange coefficient is

$$A_h = A_{ho} + C_{AH} \Delta L^2 \left[\left(\frac{\partial U}{\partial X} \right)^2 + \left(\frac{\partial V}{\partial Y} \right)^2 + \left(\frac{\partial U}{\partial Y} \right)^2 + \left(\frac{\partial V}{\partial X} \right)^2 \right]^{1/2}$$

with ΔL is the grid cell size, C_{AH} = 0.01, and A_{ho} is a background value of 1.0 m^2/s.

3.2. Atmospheric Boundary Conditions

For the initialization, nowcast and the forecast runs, wind (and potentially other atmospheric data) are read from data files, arrays of wind vectors for the hydrodynamic model are created, and the data are stored in meteorological forcing files for use during the respective runs. This process is completed in several steps:

- First, for the specific CBOFS run, the times that include the date of the model run and the length of the forecast to be made are specified.
- Next, the wind files are read. The wind files consist of either observed winds for the initialization and nowcast, or sets of arrays containing forecast wind components.
- Then, the winds are interpolated to the hydrodynamic model grid. This is because the wind points generally have a larger spacing than cells in the hydrodynamic model. In addition, the wind components must be rotated to be aligned with the axial directions of the hydrodynamic model.
- Finally, the output file is created. This is a single binary file containing winds for the specified model run.

These steps are discussed in more detail below.

3.2.1. Initialization and Nowcast Winds

For the initialization and nowcast runs, wind components at all cells in the hydrodynamic model are generated from the observational data at two locations, CBBT and Thomas Point (TP). The Fortran program that accomplishes this is `genwind_2obsB.f`. The timing variables are the date of the start of the run, including the year (*iyeara*), the GMT day-in-year (*uta*), and the hour (*ihr*), and the length of the run in days (*delut*). Wind fields are generated for each hour (*delhr*=1).

Periodic observations of wind speed, wind direction (from), and atmospheric pressure are usually available to NOS in near real time. The wind data are converted to eastward and northward components. The format of the data files is shown in Table 2.

Table 2. The structure of the observed wind files. Each box represents the contents on one record. *TEMP1* and *TEMP2* are air temperatures. Direction is the direction from which the wind comes.

$YEAR_1$, DAY_1, $SPEED_1$, $DIRECTION_1$, $PRESSURE_1$, $TEMP1_1$, $TEMP2_1$, $IFLAG_1$
$YEAR_2$, DAY_2, $SPEED_2$, $DIRECTION_2$, $PRESSURE_2$, $TEMP1_2$, $TEMP2_2$, $IFLAG_2$
etc.

Missing wind data at either or both stations are filled by temporal interpolation. The temporal interpolation is linear and uses the preceding and succeeding (hourly) values in time to create interpolated values, W^i. However, if at the time a wind value is needed at station A the gap in the data is greater than 4 hours, then interpolated values at the other location, station B, are substituted to create a weighted final value, W^w, as follows:

$$W_A^w = \max(f_1, f_2)W_A^i + (1 - \max(f_1, f_2))W_B^i$$

where

$$f_1 = \max\left(0,\ \min(1,\ 1 - \frac{t - T_m}{D})\right) \qquad for\ t \geq T_m$$

and

$$f_2 = \max\left(0,\ \min(1,\ 1 + \frac{t - T_p}{D})\right) \qquad for\ t \leq T_P$$

Here t is the time, T_m is the time of the previous value, T_p is the time of the next value, and D is half the minimum gap (here D is 2 hours). A plot of this function appears in Figure 9.

Winds are then rotated to the hydrodynamic model coordinates (Figure 8), where in Cartesian coordinates x is toward -106 degrees (Southeast) and y is toward -16 degrees (Southwest). Next, the winds are spatially interpolated to the center of cells in the hydrodynamic model grid. The spatial interpolation is based on using the observed wind at two locations, CBBT and Thomas Point (TP). For example, the x-direction component, *WX*, is:

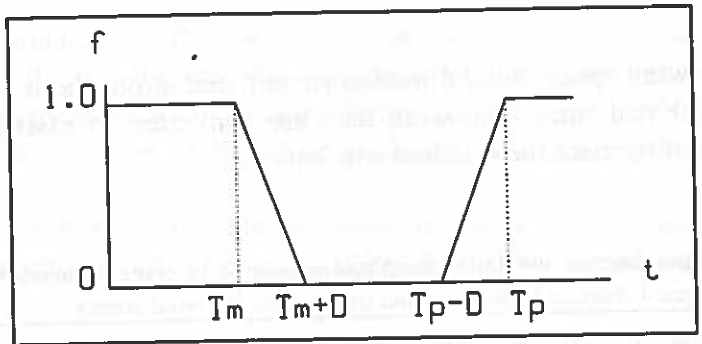

Figure 9. Plot showing the weighting function, *f*, for replacing missing time series values of winds.

for $M \leq M_U$, $WX = WX_{TP}$

for $M \geq M_L$ $WX = WX_{CBBT}$

otherwise, $WX = (M-M_U)/(M_L-M_U)WX_{CBBT} + (M_L-M)/(M_L-M_U)\ WX_{TP}$

where M is the cell index value, $M_U = 18$, and $M_L = 52$. Calculations are similar for the y-direction component, WY. Figure 10 shows the wind vectors generated by applying this spatial interpolation scheme to a northeastward wind of 14 m/s at Thomas Point and a southeastward wind of 14 m/s at CBBT.

3.2.2. Forecast Wind Fields

The Fortran program `genwind_bin.f90` reads the wind forecast files and creates arrays of wind vectors for the hydrodynamic model. The forecast wind files consist of sets of arrays containing forecast wind components for a specific forecast time and at a number of points. The timing requirements are set when the program reads in the date of the start of the model forecast and the length of the forecast to be made. The date includes the four-digit year (*fyra*); the month from 1 to 12, although this value is not used (*fmon*); and the day in year from 1 to 366 (*fday*), the length of the forecast run in hours (*delut*), and the time interval for the output fields (*deltahours*).

The most recent forecast wind file changes daily, so the name must be created for each CBOFS run. Using the date information described above, the script `CBOFSfore.sh` creates the name of the file containing the latest forecast winds in the ODAAS archive. The file name has the general form `%Y%m%d%Heta32.cb.bin`, where `%Y` is the four-digit year, `%m` the two-digit month, `%d` the two-digit day in the month, and `%H` is the two-digit forecast hour. Here `%H` corresponds to the time of the start (here, either 00 or 12 hours) of the forecast, corresponding to the 00-hr projection. If the file containing the latest wind forecast is missing, a new file name is generated. The new name corresponds to the file generated on the previous forecast cycle, which here is 12 hours earlier. If that file is,also missing, a CORMS status message is issued and the run continues but with no winds.

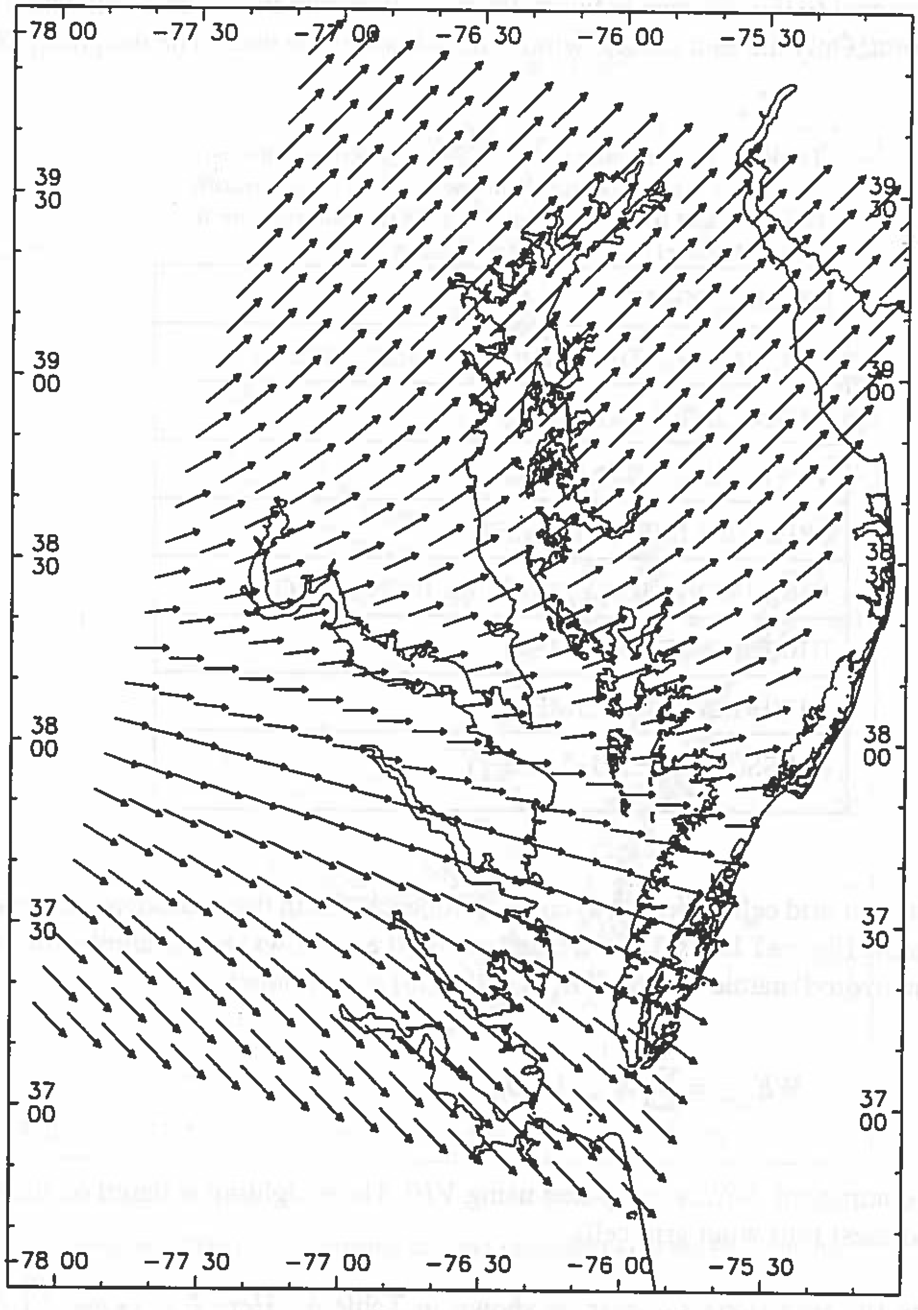

Figure 10. Sample winds over the MECCA grid. The wind vectors are generated by applying the spatial interpolation scheme to a northeastward wind of 10 m/s at Thomas Point and a southeastward wind of 10 m/s at CBBT. Vectors at every other cell are plotted.

The forecast wind file for CBOFS contains a series of records containing grid data, forecast times, and wind components. The form of each file is shown in Table 3. The file is read and searched for 13 sets of wind data, corresponding to the 0-, 3-, 6-, ..., and 36-hour projections. Winds in the NWS' Eta 32 model are spaced at approximately 32 km (Figure 11). The wind at each cell in the hydrodynamic model grid is generated by the use of a weighting function. The weighting is inversely proportional to the distance between the hydrodynamic model grid cell center and the wind forecast point. Only the four closest wind forecast points are used. The weighting function,

Table 3. The structure of the ODAAS-generated forecast wind files. Each box represents the contents on one record. IMODEL and IGRID are the NWS GRIB identifiers for the MECCA model type and grid projection..

IMODEL, IGRID
JYR_1, $JMON_1$, $JDAY_1$, $JHOUR_1$, $JMIN_1$, $JPROJ_1$
U10(I=1,IMET, J=1,JMET)
V10(I=1,IMET, J=1,JMET)
PRESS(I=1,IMET, J=1,JMET)
JYR_2, $JMON_2$, $JDAY_2$, $JHOUR_2$, $JMIN_2$, $JPROJ_2$
U10(I=1,IMET, J=1,JMET)
V10(I=1,IMET, J=1,JMET)
PRESS(I=1,IMET, J=1,JMET)

w(k), and the forecast grid cell indices, *i(k)* and *j(k)*, for each cell in the hydrodynamic model grid are read in from the file `cellwgt32.out` and are used as follows. For example, the eastward wind used in the hydrodynamic model, *WE*, at cell (*n,m*) is computed as

$$WE_{n,m} = \sum_{k=1}^{4} w_{n,m,k} U10_{l(n,m,k)}$$

The northward component, *WN*, is computed using *V10*. The weighting is based on the inverse-distance to the closest four wind grid cells.

The form of the file `cellwgt32.out` is shown in Table 4. Here $L = i + jmet*(j-1)$ is the position of the forecast wind cell in the equivalent, one-dimensional array. By setting the variable *IRDWGT* in `genwind_e32b.f` equal to 0 and recompiling, a new file containing weights and cell indices can be generated.

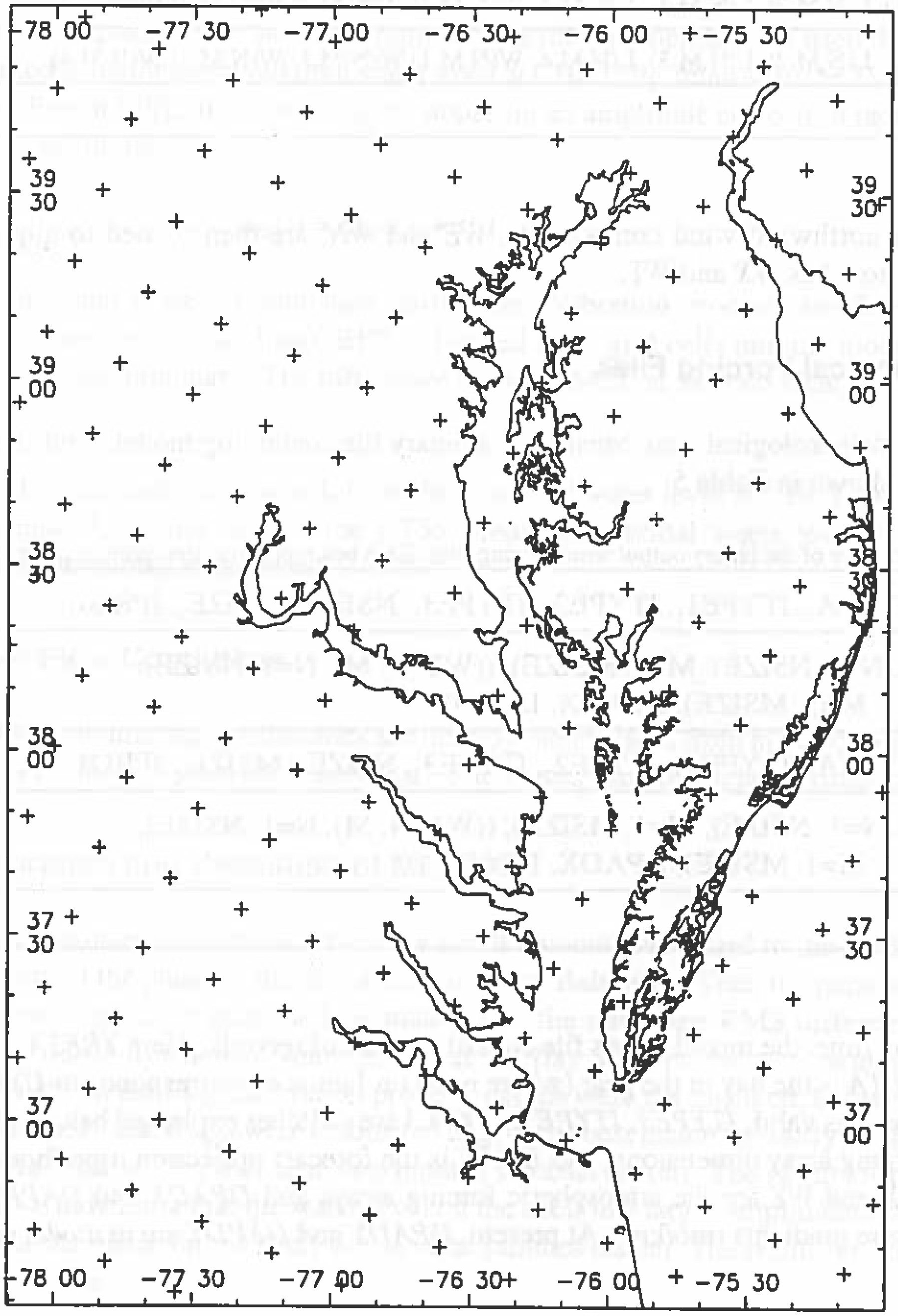

Figure 11. Chesapeake Bay region showing the grid locations (+) of the Eta32 model forecast winds.

Table 4. The structure of the binary weighting function files. Each box represents the contents of one record.

N_1,M_1,L(N,M,1), L(N,M,2), L(N,M,3), L(N,M,4),W(N,M,1),W(N,M,2),W(N,M,3),W(N,M,4)
N_2,M_2,L(N,M,1), L(N,M,2), L(N,M,3), L(N,M,4),W(N,M,1),W(N,M,2),W(N,M,3),W(N,M,4)
etc.

The eastward and northward wind components, *WE* and *WN*, are then rotated to align with the model's grid and stored as *WX* and *WY*.

3.2.3. Meteorological Forcing Files

The output of the meteorological runs consists of a binary file containing model wind forcing. The form of the file is shown in Table 5.

Table 5. The structure of the binary output wind forcing files. Each box represents the contents of one record.

$YRETA_1$, $UTETA_1$, $ITYPE1_1$, $ITYPE2_1$, $ITYPE3_1$, $NSIZE_1$, $MSIZE_1$, $IPROJ_1$
(WX(N, M), N=1, NSIZE), M=1, MSIZE), ((WY(N, M), N=1, NSIZE), M=1, MSIZE), DPADX, DPADY
$YRETA_2$, $UTETA_2$, $ITYPE1_2$, $ITYPE2_2$, $ITYPE3_2$, $NSIZE_2$, $MSIZE_2$, $IPROJ_2$
(WX(N, M), N=1, NSIZE), M=1, MSIZE), ((WY(N, M), N=1, NSIZE), M=1, MSIZE), DPADX, DPADY
etc.

For each projection time, the model winds file consist of pairs of records. Here *YRETA* is the four-digit year and *UTETA* is the day in the year (where noon on January 1 corresponds to *UTETA*=1.5) for which the forecast is valid; *ITYPE1, ITYPE2, ITYPE3* are switches explained below; *NSIZE* and *MSIZE* are the forcing array dimensions; and IPROJ is the forecast projection time (hours). In the second record, *WX* and *WY* are the atmospheric forcing arrays and *DPADX* and *DAPDY* are the atmospheric pressure gradients (mb/km). At present, *DPADX* and *DAPDY* are in model coordinate directions.

The values of the switches (*ITYPE1, ITYPE2*, and *ITYPE3*) are set in the Fortran code and presently have the values (1,0,0). If *ITYPE1* = 1, then *WX* and *WY* are winds; if *ITYPE1* = 2, then *WX* and *WY* are stresses, computed as explained in Section 3.1. If *ITYPE2* = 0, no *DPDX* and *DPDY* values are read; if *ITYPE2* = 1, they are read. If *ITYPE3* = 0, the forcing array variables are in model coordinates; if *ITYPE2* = 1, they are in eastward and northward coordinates, and will be rotated. If *NSIZE* = *MSIZE* = 1, then the single value read in for each of *WX, WY, DPADX*, and *DPADY* are applied to all model grid cells.

3.3. Coastal Boundary Conditions

For the coastal boundary, the water level representing the continental shelf used to drive the model, h_O, is computed from the observed tidal water level at CBBT, h_C, relative to the 1960-1978 National Tidal Datum Epoch MSL. It is computed by applying an amplitude correction factor, a, and a time correction, t', as follows:

$$h_O(t) = a h_C(t + t')$$

The values of a and t' were established during the calibration process (see Section 3.5). These parameters account for the fact that CBBT is located three grid cells into the model domain rather than at the true open boundary. The difference between MSL at the two locations was assumed to be zero.

For the initialization and the nowcast, h_C is the observed water level at CBBT, relative to MSL. For the forecast, h_C is the sum of the ETSS-predicted subtidal water level at CBBT and the astronomically-predicted water level.

3.4. River Flow Conditions

Daily river flows at nine major tributaries are linearly interpolated from historical daily mean values at about every 30 days to get a nowcast value. The values used in the latest runs appear in Table 6.

3.5. Calibration and Validation of MECCA

During the preliminary calibration process, a small amount was added to the MSL water depth at each cell to correct the phase of the astronomical tide at Baltimore. Then the parameters a, t', C_{DWB1} and C_{DWB2} were adjusted to give the best match (i.e., the minimum RMS difference) to predicted (astronomical tide only) hourly water levels at 10 Bay stations for two months (October and November 1994). In the final calibration process, depths were not changed, but winds were added. The parameters a, t', and C_{DWB1} were readjusted to give the best match to hourly total observed water levels at 10 Bay stations for the same two months of observations. The results for the water level were $a = 1.134$, indicating that the water level on the shelf had larger amplitudes, and $t' = 0.283$ hr, indicating that the phase on the shelf was several minutes earlier. The result for bottom stress was $C_{DWB1} = 0.0007$ m/s.

The model was validated using three methods. The first method involved running the model for the entire year of 1994 with only the predicted astronomical tide at the mouth, and harmonically analyzing the water levels at three stations. The tidal constituent amplitudes and phases were then compared to the accepted NOS values. The results showed good agreement between the model and the NOS values for the six most dominant constituents (M_2, S_2, N_2, K_1, O_1, and P_1). At CBBT the largest amplitude difference was only 0.1 cm with a maximum phase difference of 2 degrees. At

Table 6. Sample file of the river flowrate. Data in the columns are the year, day in year, and average daily flow rates (m^3/s) for nine rivers: the Susquehanna, Chester, Choptank, Nanticoke, Patuxent, Potomac, Rappahannock, York, and James Rivers.

2000	1	875.64	3.40	4.42	2.58	10.39	332.10	51.06	57.09	216.51
2000	15	875.64	3.40	4.42	2.58	10.39	332.10	51.06	57.09	216.51
2000	46	1469.19	4.19	5.66	3.23	12.03	502.68	64.14	63.51	285.49
2000	74	1910.76	4.05	8.18	4.53	16.20	725.96	78.18	77.16	363.64
2000	105	2558.03	4.25	6.77	4.47	15.26	711.83	88.60	86.00	374.55
2000	135	1363.65	4.13	4.56	3.45	13.14	493.73	59.41	52.67	258.22
2000	166	873.49	2.86	2.75	2.27	8.78	234.83	34.52	27.55	163.16
2000	196	553.82	2.61	1.56	1.42	5.75	129.63	20.39	17.58	95.03
2000	227	424.50	1.78	1.33	1.36	4.93	105.59	18.52	18.01	95.91
2000	258	343.51	1.81	1.19	1.10	4.67	78.15	16.71	9.60	96.36
2000	288	494.44	1.73	1.27	1.13	5.18	108.54	24.49	14.70	104.86
2000	319	884.98	2.44	1.61	1.27	7.50	222.57	45.11	30.95	170.58
2000	350	1289.35	3.03	3.00	1.78	10.08	319.19	47.54	44.12	197.65

Baltimore the agreement was also good, with maximums of 1.8 cm and 8.8 degrees. Results were only slightly less accurate at Solomons Island. The second method of validation was the comparison of the hourly water levels (from the year-long astronomical tide run) for seven Bay stations. The differences, which are generally small, are shown in Table 7. The difference is the lowest (0.7 cm) at CBBT, near the entrance, and generally increases with distance up the Bay. The largest value (6.4 cm) occurred at Baltimore.

Table 7. The Standard Deviation (cm) of differences between predicted astronomical tide and demeaned modeled tide at several locations in Chesapeake Bay. Predicted tide has a zero mean, so model time series was demeaned. Results are based on 365 days of hourly values for 1994.

Location	Baltimore	Annapolis	Solomons	Lewisetta	Gloucester	Kiptopeke	CBBT
Diff.	6.4	5.0	4.8	3.9	4.7	2.6	0.7

The third method of validating the model was to compare the power spectra of the observed and simulated water levels. The results for CBBT, Solomons Island, and Baltimore for the 1994 hourly water levels are shown below (Figures 12a, b, and c). The model was forced with observed winds and water levels, with gaps filled as described above. Overall, the spectrum of the observed data is closely matched by the model. Small differences at Solomons Island and Baltimore occurring in the

1.5- to 5-day period band suggest that the model is not energetic enough in response to direct meteorological forcing. This may be attributed to the lack of spatial (and temporal) resolution in the driving wind fields.

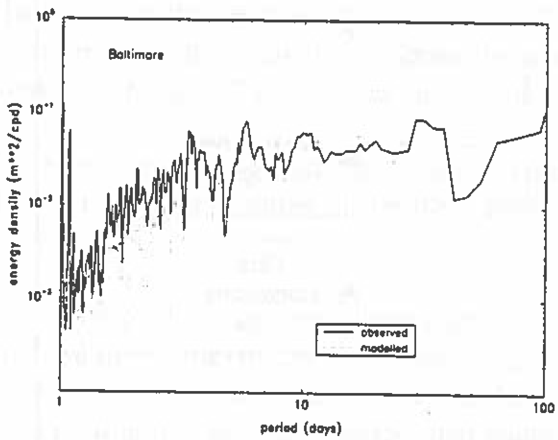

Figure 12a. Spectral energy of the observed and modeled water level at Baltimore.

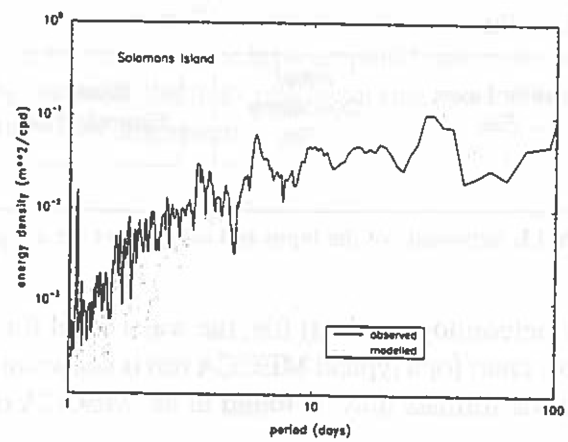

Figure 12b. Spectral energy of the observed and modeled water level at Solomons Island.

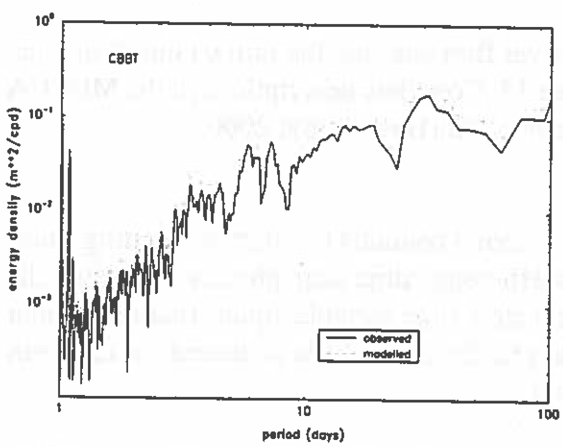

Figure 12c. Spectral energy of the observed and modeled water level at CBBT.

3.6. MECCA Input and Output Files

Execution of the MECCA program is managed by a control file (extension 'con') which contains the names of the input files needed for the run. The additional files include the geography file, the wind (and meteorological data) file, the water level file, the river flowrate file, the initial conditions file. A flow chart for a typical MECCA run is shown in Figure 13. Complete descriptions of the MECCA input file formats may be found in the MECCA documentation (Hess, 1989; 2000).

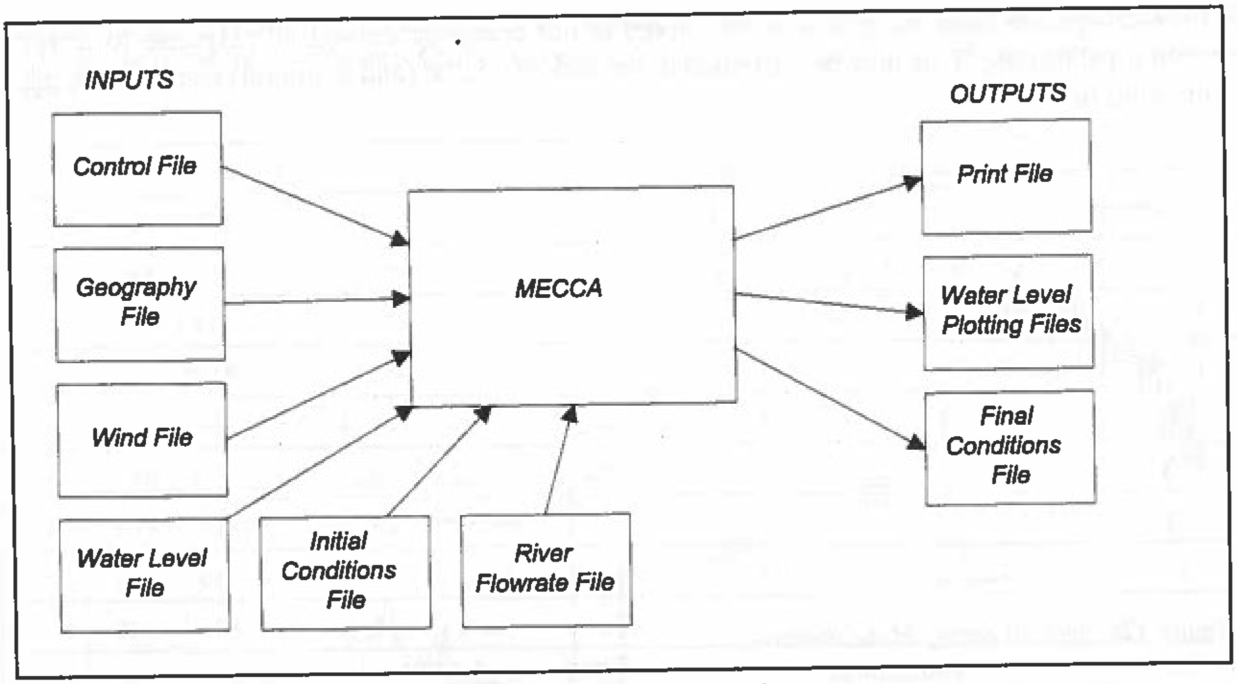

Figure 13. Schematic of the input and output files for a typical MECCA run.

(and meteorological data) file, the water level file, the river flowrate file, the initial conditions file. A flow chart for a typical MECCA run is shown in Figure 13. Complete descriptions of the MECCA input file formats may be found in the MECCA documentation (Hess, 1989; 2000).

The control files (`init.con`, `now.con`, and `fore.con`) contains the start and ending times for the run, the model configuration parameters (drag coefficients, time step, physics switches), the printing parameters (node numbers of locations for output etc.), time variable inputs, (names of input data files) and output file names. Since the timing changes, the control file is altered for each run of the model. For a sample, see Hess (2000), Appendix C.

The geography file describing the grid is `fullbay21c.geo`. It contains the specification of the active water grid location and the boundary. It also contains the enumeration and location of the river inputs, the specification of the grid locations to be used for open water forcing, river width descriptions, corner geometry, and the depth data. The depth data are specified in feet with a conversion factor given to convert feet to meters, used by the rest of the model. A scaling factor has been applied to all depths to calibrate the wave speed and improve the tidal phase at upper bay locations. This scaling is incorporated into the feet to meters conversion factor, so instead of 0.3048 the value read in is 0.2996. For a sample, see Hess (2000), Appendix B.

The wind files (`genwind_now.out` and `genwind_fore.out`) contain the driving winds and atmospheric pressures (see Section 3.3.3.). The coastal boundary water levels (`gentide.out`) file is as described in Section 3.4.

The water level files (`gentide_init.out and gentide_now.out`) contain the time series values that are applied to the model's grid cells representing the Bay's entrance.

The initial conditions file (`HOTSTART.DAT, nowFIN.DAT`) contains the last computed values of water level and velocity components at all grid cells. These data are read in to provide the conditions for a 'hot start' of the model run.

The river flowrate file (`rivers.met`) contains the climatological monthly-averaged observed flows of major tributaries of the Bay. These are used for the initialization, the nowcast, and the forecast.

At the completion of the run, three files are written out. The print file contains run-time output, the final conditions file (`HOTSTART.DAT, nowFIN.DAT`) contains the last computed values of water level and velocity components at all grid cells to be used for the next initialization, and the water level plotting file contains interpolated values for the Web animation.

4. THE INITIALIZATION

The first of the three CBOFS operations is initialization. This operation runs once a day and spins up the model from rest by simulating a 8-day period that ends at hour 00 UTC of that day. Water level and wind data, which have been quality controlled, are acquired from CO-OPS' data base. The output of the run is a 'hot start' file (`HOTSTART.DAT`) that is used as the initial conditions file for each of the two daily nowcasts.

The initialization is managed by the Unix crontab-controlled script, `CBOFSINIT.sh`. Each day at 1235 UTC, the script `CBOFSINIT.sh` gathers and reformats the NWLON water level data, gathers and reformats the wind data, and runs the model to spinup to 00 UTC of the present day. The script is described briefly in Appendix C.

The following subsections detail the data ingest, forcing field generation, model run, and output tasks of the initialization. An overview of the process appears in Figure 14 and a listing of the processes and files appears in Appendix D.

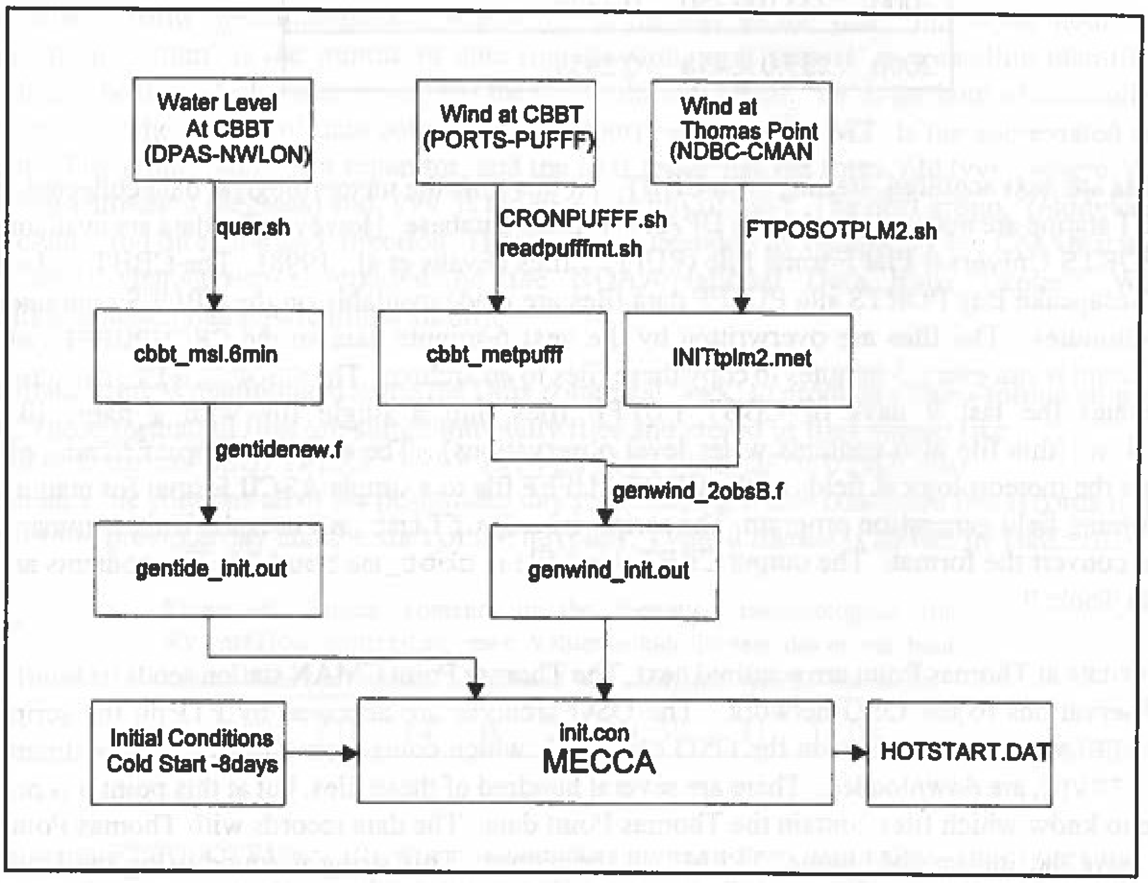

Figure 14. Schematic of the programs, scripts, and processes in the initialization.

4.1. Data Ingest

The first step is to get the water level data for CBBT for forcing the ocean boundary of the MECCA model. The water level forcing data are pulled from the DPAS NWLON data base, which is more likely to have more complete data than the raw PORTS files compiled by the CBOFS cron script. The `CBOFSINIT.sh` script first accesses the NWLON data base to download CBBT water level data. The NWLON database is a Structured Query Language- (SQL)-type database, which is queried for data of the required format using the script `quer.sh`. The quality control flags are used by the query to reject any questionable data. The water level data are originally referenced to MLLW. A MSL datum is required by MECCA, so the value 0.442 m (the difference between MSL and MLLW for the 1960-1978 National Tidal datum Epoch) is subtracted from the NWLON values and the formatted file `cbbt_msl.6min` is generated (Table 8).

Table 8. Typical contents of the formatted file `cbbt_msl.6min`. Each record contains year, day in year, and water level (meters).

2000.	233.031250	0.2748
2000.	233.036458	0.2959

Wind data are next acquired, starting with CBBT. At this time the meteorological data collected at the CBBT station are not contained in the DPAS NWLON database. However, the data are available in the PORTS Universal Flat Format File (PUFFF) files (Evans et al., 1998). The CBBT and all other Chesapeake Bay PORTS site PUFFF data files are made available on the CBOFS computer every 6 minutes. The files are overwritten by the next 6-minute data so the `CRONPUFFF.sh` crontab script is run every 5 minutes to copy these files to an archive. The `CBOFSINIT.sh` script concatenates the last 9 days of CBBT PUFFF files into a single file with a name like `%Y%m%d.wl`(this file also contains water level observations). The script `readpufffmt.sh` reformats the meteorological fields of the CBBT PUFFF file to a simple ASCII format for reading by the forcing field generation program. The script `readpufffmt.sh` uses the Unix command 'awk' to convert the format. The output CBBT wind file is `cbbt_metpufff`. The contents are shown in Table 9.

The wind data at Thomas Point are acquired next. The Thomas Point CMAN station sends its hourly wind observations to the OSO network. The OSO archives are accessed by FTP in the script `FTPOSOTPLM2.sh`. The files on the OSO computer, which come from the USGS data stream (`USG*.SRVR`), are downloaded. There are several hundred of these files, but at this point it is not possible to know which files contain the Thomas Point data. The data records with Thomas Point data all have the station code name 'TPLM2' in the record. This string is found using the Unix command 'grep' to produce a file with just the Thomas Point records.

Table 9. Typical contents of the formatted file `cbbt_metpufff` and `NOWtplm.met`. Data are the year, day in year, speed (m/s), direction from (degrees from N), pressure (mb), air temperature (C), water temperature (C), and a quality code.

2000	233.050000	0.0	120	1019.6	23.0	24.8	11
2000	233.054167	0.0	143	1019.7	23.1	24.8	11
2000	233.058333	0.0	141	1019.8	23.2	24.8	11

The CMAN data contain the date, wind speed, and wind direction characters. The first few characters of a typical CMAN record for Thomas Point are:

2243153A 23114052532196 1400 TPLM2 46/// /029105(/031110) 10229 30142(3////) ...

The first, 8-character group contains a coded station identification number. The second group has the general form 'jjjhhmmssssss', where 'jjj' is the day in the year, 'hh' is the hour of data transmission, 'mm' is the minute of data transmission, and 'ssssss' is a satellite identification number. The third, 4-character group has the form 'hhmm', where 'hh' is the hour of data collection and 'mm' is the minute of data collection. The fourth group, 'TPLM2', is the abbreviated station name. The group '46///' is a separator, and the next group has the form '/dddvvv', where 'ddd' is the wind direction (degrees) and 'vvv' is the speed (tenths of m/s). The next group, '(/dddvvv)', are a second wind direction and direction. The data were decoded by comparing the CMAN data with the wind characteristics reported on the NOAA/National Data Buoy Center's Website (seaboard.ndbc.noaa.gov/realtime.shtml).

The data string is manipulated using the Unix command 'awk' to produce a fixed-format output data file. These formatted files are sorted into daily files and stored in files named like
`/CBOFS/data/OSO/TPLM2/199909/1999091300_OSOtplm2.met`
Each data file contains all of the designated day's records, but it also contains a few records from the end of the previous day and the start of the next day. Typical format is shown in Table 10.

Table 10. Typical contents of the formatted meteorological file, `%Y%m%d00_OSOtplm2.met`. Values include the year, day in year, hour, minute, direction1, speed1, direction2 (not used), and speed2 (not used).

1999	231	14	0	29	10.50	31	11.00

The script `FTPOSOTPLM2.sh` will concatenate new data to existing files, and it uses the Unix command '`sort -u`' to avoid repetitions. The previous 8 days of Thomas Point wind files are read, reformatted, and concatenated to form the new file, `INITtplm.met`. This file has the same format as does `cbbt_metpufff` (Table 9) and is sent to the forcing field generation program.

4.2. Generation of Forcing Fields

The coastal boundary condition is based on the water level measurements from CBBT. The program gentidenew.f reads the data file acquired from the DPAS NWLON database, cbbt_msl.6min. The water level data are gap-filled and then reformatted. Gaps are identified and filled with values interpolated from the beginning and end of the gaps by detiding the original signal, extracting the non-tidal signal, linearly filling the non-tidal signal gap and then finally adding back the tidal signal generated using as many as the full 37 constituents for which accepted harmonic constants exist. The tidal component reconstruction is calculated using subroutines based on Zervas (1999), with values obtained from the DPAS data base. Files bt_m_msl_g.in and yr contain the tidal constituent and other data. The water level data are then adjusted by offsetting the data in time and rescaling the amplitude, where the 0.0118056 is the time shift in days, and 1.134 is the amplitude scaling. The final output is saved in the file gentide_init.out. It has the format shown in Table 8.

The two wind data time series files, cbbt_metpufff and INITtplm.met, which are of identical format (Table 9), are ready for forcing field generation by the program genwind_2obsB.f. The genwind_2obsB.f program fills gaps by interpolating and creates the wind fields as described in Section 3.3.1 The program genwind_2obsB.f uses the geometry and dimensions of the Chesapeake Bay MECCA grid (saved in file fullbay21c.geo), and the positions of CBBT and Thomas Point hardwired in to create the field. Any changes in these geometries (additional or different wind stations etc.) would necessitate reprogramming genwind_2obsB.f. The output file, genwind_init.out, is in binary format and is ready to be read by MECCA.

4.3. Model Run

Once the forcing input files have been generated, the MECCA program is ready to run. The MECCA program is run with init.con, the control file which specifies the files, starting and ending times, and other parameters for the MECCA run. For the initialization, the model is spun up from a state of rest, so no initial conditions file is needed. However, coastal water level data (gentide_init.out) and wind fields (genwind_init.out) are used, along with the Bay's geometry (fullbay21c.geo) and river flow rate data (rivers.met).

4.4. Output and Archives

The only required output of the initialization run is the final conditions file, HOTSTART.DAT. Although other files are produced and may be found in the CBOS directory rundir/init, nothing else is saved, archived or plotted.

5. THE NOWCAST

The second of the three CBOFS operations is the nowcast. This operation runs twice a day and uses the output of the initialization run for starting conditions. The coastal water levels and winds are acquired from the archive of the latest real-time observations, and they may not have been quality controlled. The nowcast's final conditions file become the starting conditions for the forecast run.

The script which runs the MECCA model for the nowcasts is `CBOFSNOW.sh`. The nowcasts are run at approximately 1810 UTC and 610 UTC (the following day). `CBOFSNOW.sh` obtains the starting time from the existing `HOTSTART.DAT` file of initial conditions created by the initialization. The script is described briefly in Appendix C. The final conditions of the nowcast is saved in the file `nowFIN.DAT` for use as the next forecast run's initial conditions. The model-calculated water levels at the locations of the NWLON stations for the past 12 hours are saved in the output data file `now.wl.out` and archived as a dated file in `output/nowcast`. Files containing the water levels are also created for graphical display. A schematic of the process appears in Figure 15 and a listing of the processes and files appears in Appendix D.

Although `CBOFSNOW.sh` was designed to be executed at any time of the day, the actual times are determined by the Unix crontab program which executes the script at specified times each day. The ending time for `CBOFSNOW.sh` is supposed to be a time close to the launch time, which is the moment the script is executed. The actual ending time is the time of the last observation data point contained in the CBBT water level file, which, when all is going well, is within 6 minutes of the launch time. The ending time is rounded to the hour as follows. A last data point at 45 minutes past the hour will allow the model to run to the end of the hour, with only 15 minutes of filled data. A last data point earlier than 44 minutes past the hour will round down to the nearest hour. The nowcast is based on the most recent real-time observation data, limiting extrapolation of observational data to no more than 15 minutes.

5.1. Data Ingest

The nowcast requires the most recent observational data from the CBBT water level gauge, the CBBT wind data, and the Thomas Point wind data. These data are needed for a time as close as possible to the moment the nowcast is run, so they must be acquired from instruments reporting their data in real time. The CBBT wind and water level data are obtained from the PORTS sites as PUFFF files, which are continuously delivered to the CBOFS computer by a simple network connection. The Thomas Point winds are obtained from the OSO data network.

5.1.1. CBBT Water Level Ingest

The latest real-time water level data at CBBT (and all Chesapeake Bay stations) are available in PUFFF files every 6 minutes (Evans et al. 1998). A typical PUFFF file is shown in Table 11.

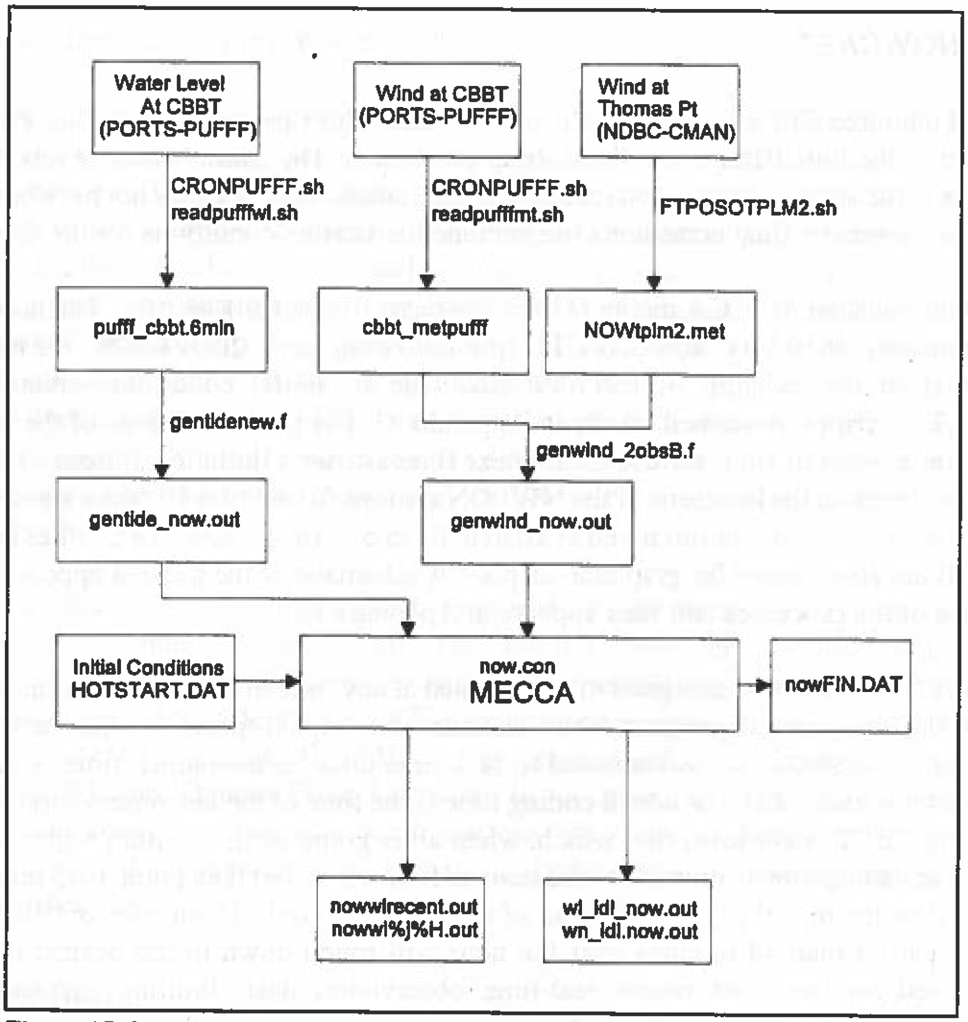

Figure 15. Schematic showing the programs, scripts, and processes in the nowcast run.

Table 11. Records in a typical PUFFF file.

OSTEF1 PORTS
8638863 Bay Bridge Tunnel
(blank)
(blank)
12.721 1.113 0 399 1022 8.602 .042 1 17.7 18.3 6.5 356 7.6 1013.6 3.232 .041 0 9.3 13.6 99.9 9999.9 17.4 0 1.000 .000 .000
1999 05 19 15 18
1019 42 1 000000000000000000000000000000 300
CORMS 00000000000000000000000010000000

The files are overwritten by the next 6 minute's data, so the CRONPUFFF.sh crontab script is run every 5 minutes to copy these files to an archive. The 5-minute timing guarantees that each 6-minute file will be recorded. Duplicates are more easily discarded than missing files are retrieved. The PUFFF files are concatenated together into large daily files and archived with file names like: /CBOFS/data/pufff/wl8636580/19990628.wl. Note that the directory wl8636580 corresponds to the NOS station number (Table 1). These daily files have the format shown in Table 11 and consist of about 300 PUFFF records (i.e., about 24 hours worth, plus some duplications).

The script readpufffwl.sh converts the water level fields of the CBBT PUFFF file YYYYMMDD.wl to a simple ASCII format using the Unix 'awk' command. The PUFFF format has been known to change and since the 'awk' command is fixed, such changes would probably cause problems with readpufffwl.sh. The output of readpufffwl.sh is the water level referenced to MLLW; the conversion to MSL requires the subtraction of 0.442 meters. PUFFF flags bad water level values by replacing the data value with the flag value of 99.999. The bad values and the MLLW to MSL conversion are taken care of with the 'awk' statement in CBOFSNOW.sh which produces the CBBT water level file, pufff_cbbt.6min (format like that in Table 8).

5.1.2. Wind Data Ingest

The CBBT wind data are obtained from the same PUFFF data files used for the CBBT water level, %Y%m%d.wl. The script readpufffmt.sh converts the meteorological fields of the CBBT PUFFF file to a simple ASCII format for reading by the forcing field generation program. The CBBT wind file is saved in cbbt_metpufff (Table 9).

The Thomas Point CMAN station sends its wind data to the OSO network. The OSO archives are accessed by FTP in the script FTPOSOTPLM2.sh. This process is the same one described for the Thomas Point data ingest by the initialization procedure. The previous 2 days of Thomas Point wind files are concatenated to form the file, NOWtplm.met. This file is sent to the forcing field generation program.

5.2. Generation of Forcing Fields

The coastal water level boundary condition is based on the water level measurements from CBBT. The program gentidenew.f reads the CBBT data file constructed by readpufffwl.sh. The data are gap-filled and then reformatted for use by MECCA. Gaps are identified and filled with values interpolated from the beginning and end of the gaps by detiding the original signal, extracting the non-tidal signal, linearly filling the non-tidal signal gap and then finally adding back the tidal signal generated using the full 37-component tidal reconstruction. The tidal component reconstruction is calculated using subroutines described by Zervas (1999), with tidal constituent values obtained from the DPAS data base. Files bt_m_msl_g.in and yr contain the constituent and other data. The program time-shifts and scales the data with values hardwired into the program, where the 0.0118056 is the time shift in days, and 1.134 is the amplitude scaling. The final output is saved to gentide_now.out for use by MECCA (for format, see Table 8).

The two wind data time series, cbbt_metpufff and NOWtplm.met, which are of identical format (Table 9), are ready for forcing field generation by the program genwind_2obsB.f. The genwind_2obsB.f program extrapolates and interpolates the linear wind field described in Section 3.3.1 The program genwind_2obsB.f reads the geometry and dimensions of the Chesapeake Bay MECCA grid from the file fullbay21c.geo, and the positions of CBBT and Thomas Point hardwired in to perform this operation. Any changes in these geometries (additional or different wind stations etc.) would necessitate reprogramming genwind_2obsB.f. The output file, genwind_now.out, is in binary format and is ready to be read by MECCA.

5.3. Model Run

Once the forcing input files have been generated, the MECCA program is ready to run. For the nowcast the model is run from the fully developed state of motion described by the HOTSTART.DAT file generated by the initialization run. The MECCA program is run with now.con, the control file which specifies the input and output filenames, starting and ending times, and other parameters for the MECCA run. Additionally, the coastal water level data (gentide_now.out) and wind fields (genwind_now.out) are used, along with the Bay' geometry (fullbay21c.geo) and river flow rate data (rivers.met).

5.4. Output and Archives

The main output file produced by the nowcast MECCA run is the final conditions file, nowFIN.DAT, which is used as the initial conditions file for the forecast. Other data [hourly water levels from the 11 key station locations (Figure 3, Table 1), the coastal boundary water level, and winds and scaled wind stresses at CBBT and Thomas Point] are also saved. This information is copied to the post- processing file nowwlrecent.out, which always has the same name so that graphics programs can access it easily. The same file is archived to files named like nowwl%j%H.out. These files have the format shown in Table 12. Once or twice a year these files should be moved to yearly directories like: /CBOFS/output/nowcastout/1999now/*. Two dimensional field output files, wl_idl.now.out and wn_idl.now.out, are produced for use by the graphics animation programs. These files are not archived.

Table 12. Typical record from files nowwlrecent.out and fore.wl.out. The record contains the year, the day in year, computed water levels (m) for 12 locations (coastal boundary, CBBT, Hampton Roads, Kiptopeke, Gloucester, Lewisetta, Colonial Beach, Solomons Is., Cambridge, Annapolis, Baltimore, and Tolchester) and meteorological data. The meteorological data consists of the (modeled) x-direction wind stress (N/m^2) times 10^7 at CBBT and Baltimore, the y-direction wind stress times 10^7 at CBBT and Baltimore, the wind speed (m/s) at CBBT and Thomas Point, the wind direction (from North, in degrees) at CBBT and Thomas Point, and the wind speed and direction at Rappahannock Light.

2000. 255.2500	-0.1251	-0.1031	-0.0891	-0.1118	-0.0669	
0.2913	0.3622	0.3180	0.3635	0.2719	0.1958	0.1552
-2.3363	-209.6483	-44.5788	-224.8202	2.0000	4.7997	109.0000
149.0000	2.7314	130.4344				

6. THE FORECAST

The forecast run provides the water level prediction, based on the initial conditions given by the most recent nowcast, and the forecast forcing fields generated from predictions by NCEP for the offshore water level and wind fields for the next 24 hour period.

The forecast script `CBOFSFORE.sh` executes immediately after completion of a nowcast. `CBOFSFORE.sh` reads the starting time from `nowFIN.DAT` and uses the ETSS forecast model of coastal sub-tidal water levels and Eta32 model of forecast wind fields to attempt a forecast. The script is described briefly in Appendix C. The ETSS and Eta32 prediction files are downloaded from the NCEP computers by the ODAAS system. If the expected ETSS and/or Eta32 files are not present for the forecast period, the `CBOFSFORE.sh` script attempts to find the most recent forecast files which have an overlap with the forecast period of interest. If no overlapping old forecast files can be found, `gentidenew.f` and/or `genwind_e32b.f` fill with persisted values of water level or winds, respectively. The forecast runs forward for 24 hours and writes output data files to `fore.wl.out` and to files for plotting. An overview of the system appears in Figure 16 and a listing of the processes and files appears in Appendix D.

6.1. Data Ingest

6.1.1. CBBT Water Level Ingest

The ETSS model is run twice daily by NCEP to provide estimates of the predicted coastal storm surge due to winds over the Atlantic Ocean. ETSS provides hourly predicted water levels at many locations, including a location very close to the CBBT water level station. The ETSS input file is found in `/CBOFS/data/tdl/%Y%m/cbbt/%Y%m%d%H.cbbt` (%Y etc. are year, month, and date formats). This is actually a logical link to the ODAAS directory which produces and archives these files. ODAAS has already simplified the format of these files so that it is ready to be read by the forcing field generation program `gentidenew.f`.

6.1.2. Wind Data Ingest

The Eta32 model is run twice daily by the NCEP to provide predicted wind fields over most of North America, including the Chesapeake Bay region. The original Eta32 output files are quite large, so the ODAAS system has subsetted the Eta32 output and provided binary field files which are ready to be read by the forcing field generation programs. The files are saved on `/CBOFS/data/eta32/%Y%m/cb/%Y%m%d%Heta32.cb.bin`. As with the TDL files, this is a logical link into the ODAAS archive system.

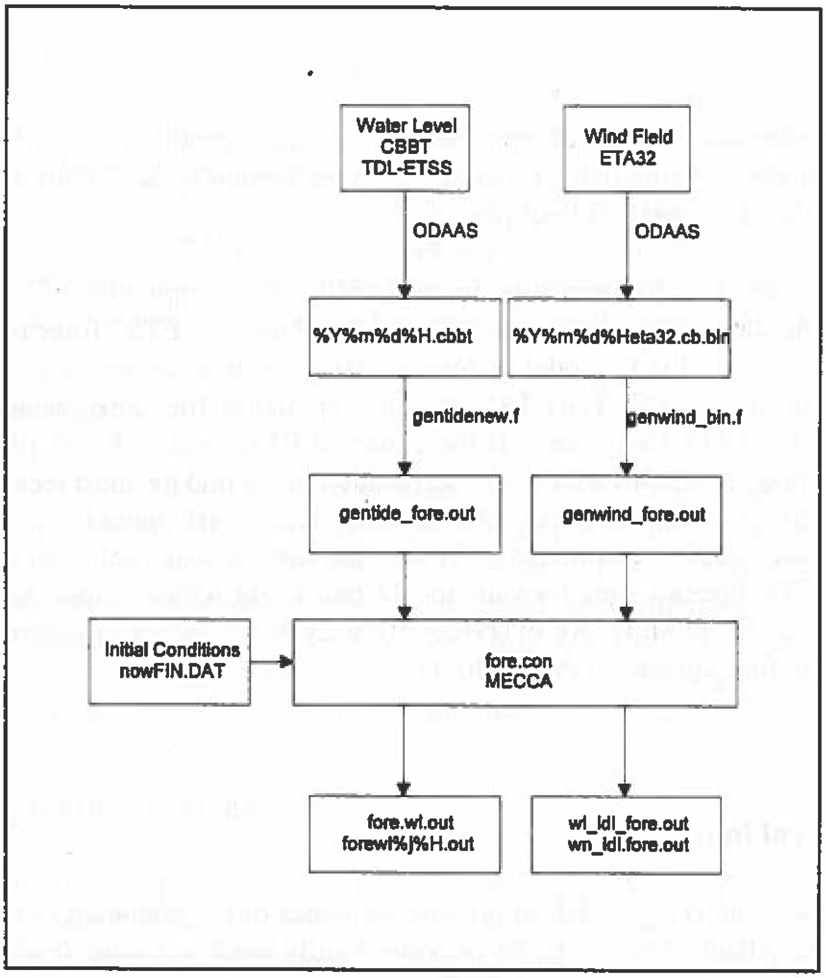

Figure 16. Schematic showing the programs, scripts, and processes in the forecast run.

6.2. Generation of Forcing Fields

6.2.1. Water Level Forcing

A forecast water level time series for the coastal boundary, in the same format as that of the initialization or nowcast, is required by MECCA. The Fortran program `gentidenew.f` constructs this file by reading the ETSS non-tidal water level predictions. A value of 0.2544 m is added to the ETSS predicted value, since it was found during calibration with 1998 data that the ETSS values were lower than the local observed MSL. In addition, because the first value in the ETSS subtidal water level prediction can be very different from the last observation, a linear ramp is created over the first 6 hours. Without the ramp, the sudden transition from the observation value to a forecast value, which might be 10 or more centimeters different, will cause the model to simulate a sharp and powerful wave which, if it does not bring the program to a halt, will create an extremely poor water level forecast. After a ramped and properly interpolated non-tidal water level series is created, the astronomical tide at CBBT is added. The astronomical tidal component reconstruction is calculated using subroutines described by Zervas (1999), with tidal constituent values obtained from the DPAS data base. Files bt_m_msl_g.in and yr in the directory /CBOFS/data/astrotides contain the constituent data and other necessary data. Phase and amplitude offsets (-0.0118056 days, 1.134 height scaling) are applied to these data to transform the CBBT tide to the coastal boundary conditions for the model. The final predicted water level file, data/gentide_fore.out is output by gentidenew.f, and is ready to be read by MECCA.

6.2.2. Wind Forcing

The Eta32 wind file obtained from ODAAS contains winds specified at the nodes of the Eta32 model, with approximately 32 kilometer spacing (Figure 11). The MECCA grid is quite different, with nodes located about 5 kilometers apart on a rotated grid. The generation of a wind field usable by MECCA involves a spatial interpolation of the Eta32 data onto the MECCA grid. The interpolation is performed as described in Section 3. This process is performed by the Fortran program genwind_bin.f which does the interpolation. The program generates an interpolated Eta32 wind field for every 2 hours of the forecast duration, 24 hours. The output binary wind field file to be used by MECCA is genwind_fore.out.

6.3. Model Run

When the forecast forcing input files have been generated, the MECCA program is ready to run. For the forecast, the model is run from the fully developed state of motion described by the nowFIN.DAT file generated at the end of the nowcast run. The MECCA program is run with fore.con, the control file which specifies the input and output filenames, starting and ending times, and other parameters for the MECCA run. Additionally, the coastal water level data (gentide_fore.out) and wind fields (genwind_fore.out) are used, along with the Bay's geometry (fullbay21c.geo) and river flow rate data (rivers.met).

6.4. Output and Archives

The main output file produced by the forecast MECCA run are the hourly water levels from the 11 key station locations (Figure 3, Table 1), the coastal boundary water level, and winds and scaled wind stresses at CBBT and Thomas Point. This information is saved to the post-processing file latestfore, which always has the same name for the graphics programs. The same file is archived to /CBOFS/output/forecastout/forewl%j%H.out. Once or twice a year these files must be moved to yearly directories like: /CBOFS/output/forecastout/1999fore/*. Two-dimensional water level fields in files wl_idl.fore.out and wn_idl.fore.out are also produced for use by the graphics animation programs. These files are not archived.

Some additional graphics programs, which are not currently in use, were created to plot the results of recent forecast runs for an evaluation of how good yesterday's forecast was. The forecast water level files are backed up for this use in a 'first in, last out' stack to file names post/prefore0, prefore1, prefore2, prefore3, and prefore4, where prefore0 is the same as latestfore and prefore1 is the file from the previous forecast run (12 hours ago).

(page 40 is blank)

7. POST-PROCESSING AND DISSEMINATION

At the conclusion of each model system run, CBOFS output is subject to (1) post-processing and the generation of graphical products, and (2) dissemination of products.

7.1. Graphical Products

Two types of graphical output are produced by CBOFS; time series and animations. Time series graphics depicting 2 days of observed, nowcast, and forecast water levels are produced for 11 stations located throughout the Bay. These may be for groups of stations in the upper, middle, and lower Bay (Figure 2) or for individual stations (a sample appears in Figure 17). Also, time series of wind vectors at the two input stations are also produced twice daily (a sample appears in Figure 5). Animations of simulated hourly water level and wind fields are also produced. The water levels appear as color-coded (according to amplitude) areas in a plan view of the Bay with a superimposed coastline (Figure 18). The wind vectors are color-coded (according to speed) in a plan view with a coastline (Figure 19). These animations can be viewed on the Web page.

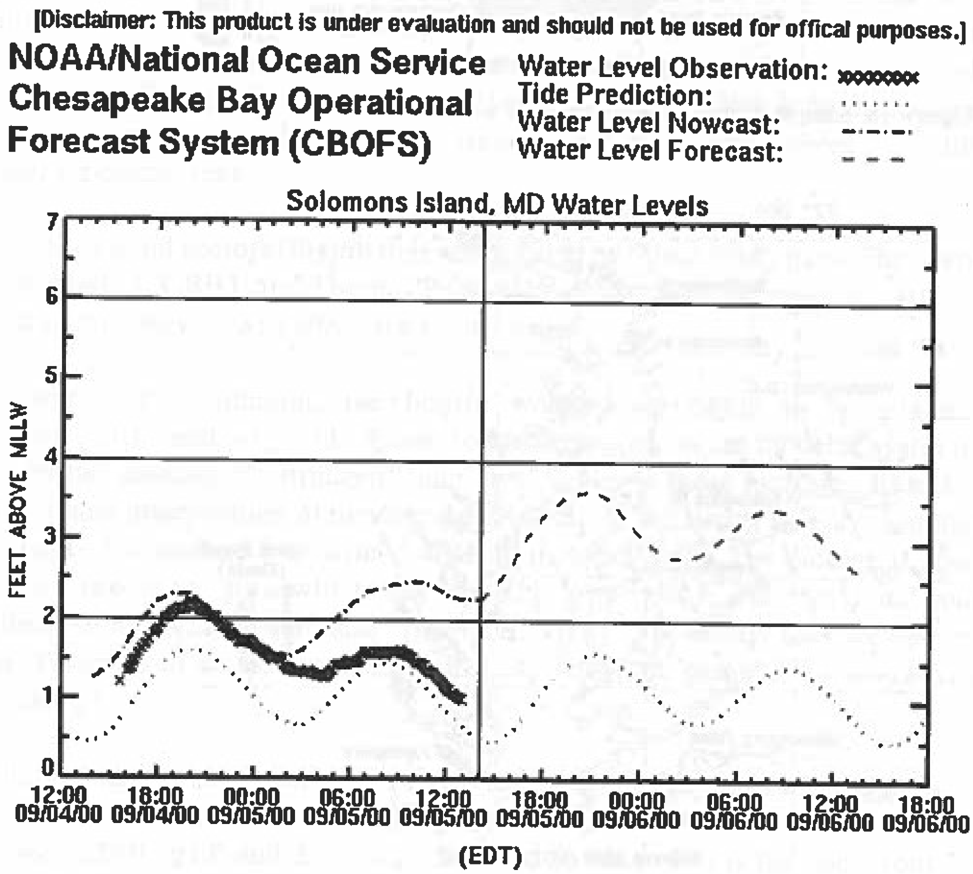

Figure 17. Sample of a time series plot of water levels for a single station.

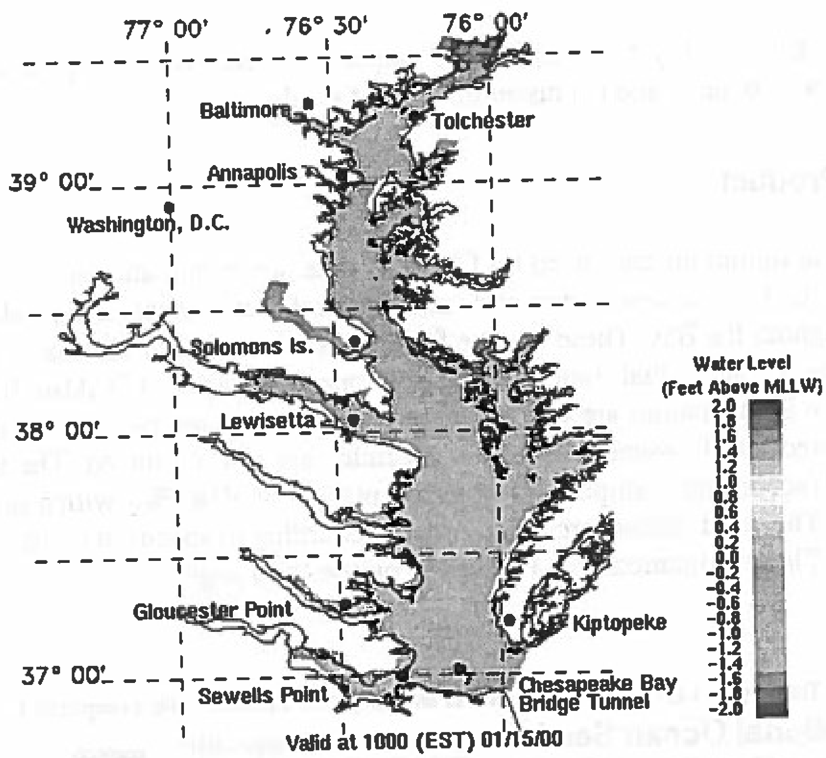

Figure 18. Sample of the animation plot for water levels.

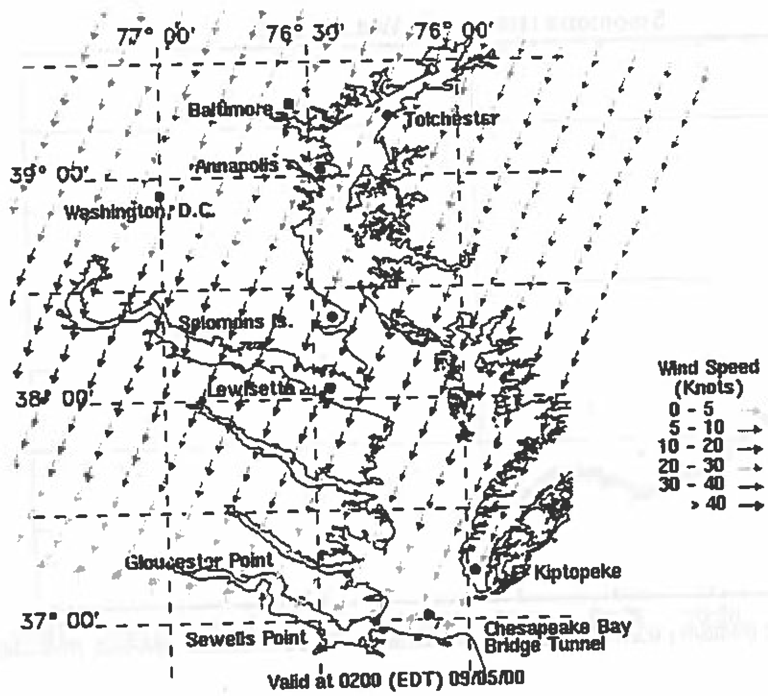

Figure 19. Sample plot showing the animated wind vectors.

The first product is the plot of a time series of water levels at an individual station. The graphics post-processing for all products is managed by the script post/idlprograms/ graphicsall.sh, which is executed after both the nowcast and forecast have finished. The first step is to access the DPAS-NWLON and PUFFF data bases to obtain the observed water levels for the 11 stations in the Bay for the previous 3 days; this is done by the script idlprograms/ stations/getgraphicsdatadpas.sh. Output is saved in files named like bt_mllw.6min. In addition, a prediction of the hourly astronomical tide has been made (at the start of the year) at all the stations and stored in files like bt_astro.dat. The graphics uses those data along with the nowcast and forecast water levels (nowwlrecent.out and fore.wl.out) to create the plots.

The subdirectory /CBOFS/post contains the graphics programs within the subdirectory idlprograms/stations. The programs are written in the IDL language and run with a shell script stage and an IDL batch job stage. This is handled by the execution script stationscript.sh which specifies IDL directory paths, and calls IDL with the batch job specified as IDL_STARTUP=graphics/runstation.pro. Program runstation.pro simply runs cafestation.pro, which reads the most recent data files. The output are .gif images (e.g., cbbt.gif) written to /CBOFS/post/gifs.

The next product - the time series of water levels at groups representing the lower, middle, and upper Bay (Figure 2) - is created in a similar way when the script graphicsall.sh calls the script wlwind.sh. The IDL graphing program is wlwinds.pro, which is described in Appendix E. Output files (e.g., upperbay.gif, upperbay.ps) are saved to the directory /var/www/htdocs/cafe.

The time series of wind vectors (Figure 6) is also created by wlwinds.pro. They depict observed and forecast winds at CBBT and Thomas Point. The output files are saved as winds.gif and winds.ps in directory /var/www/htdocs/cafe.

For the water level animation, the hourly water level fields to be plotted (saved in wl_idl.now.out and wl_idl.fore.out) are based on the modeled water levels but are augmented by the addition of estimated water level values at more locations. The new values are generated by linear interpolation of the original grid values, but taking into account the presence or absence of land. The method is described in detail in Appendix F. The plotting is done by the IDL program nowforecast.pro which is described in Appendix E. The individual hourly plots are saved in files like nowwlHH.gif and forewlHH.gif. The hourly files are then read by the C program gifmerge to create the animation files, which are stored in nowwlanim.gif and nowwlanim.gif.

For the wind animation, the hourly wind fields to be plotted (saved in wn_idl.now.out and wn_idl.fore.out) are used by the IDL program nowforecast.pro. The output is saved in files nowwindHH.gif and forewindHH.gif, where HH is the hour from 01 to 24. The hourly files are then read by the C program gifmerge to create the animation files, which are stored in nowwindanim.gif and nowwindanim.gif.

7.2. Graphics Archive

After graphics programs have created the new .gif images, the script /CBOFS/post/idlgraphics/idlprograms/graphicsall.sh creates an image archive. The .gif images residing in /var/www/htdocs/cafe are collected into an archive file stored in /var/www/htdocs/cafe/archive. Files there have names like cbofs_gifs_%m%d%Y%H%M.tar. To extract a .gif file to a local directory, use the command:

```
tar -xvRRRf /var/www/htdocs/cafe/archive/cbofs_gifs_%m%d%Y%H%M.tar.
```

7.3. Dissemination

CBOFS products are disseminated via two media. Web pages exist on the CO-OPS Web server which allow users to view and download the latest CBOFS information and graphics files. The URL is http://co-ops.nos.noaa.gov/CBOFS/cbofs.html. There is also a fax service which is capable of automatically faxing CBOFS time series graphics to a predetermined list of users.

7.4. CORMS System Status Flags

The quality of the input and output data files is displayed by the Continuous Operational Real-Time Monitoring System (CORMS) (see Section 8.3.1 for a discussion of CORMS procedures). This system visually displays the status of CBOFS data files using color codes, one each for the three possible states (Good, Warning, and Fail).

CORMS system status flags are generated as follows. First, during the nowcast and forecast runs, CBOFS produces a file containing raw data quality information such as numerical values representing presence or absence of data. A typical file (execlog/CORMS/2000090606corms.dat) is shown below in Table 13. Next, the script MAKECORMSFLAGS.sh processes this raw data file, which is interpreted by the C program computecorms to produce an CORMS-ready file, CBOFSFLAGS.txt, which contains the binary flags. A sample is shown in Table 14. This file is sent by FTP to the CORMS computer, where it is read by the CORMS software and displayed to the CORMS Operators.

Each digit represents a binary yes (1) or no (0) for a particular flag. A group of three flags describes the potential state of a single variable or data source. Since only one of the three flags can be active, each three-digit group of numbers gives the status: Good (100), Warning (010) or Fail (001). The order of the flags and the specific variables are shown in Table 15.

Table 13. Typical raw data quality file.

```
CORMSLOG  CBOFSNOW.sh Started at RealTimeClock=  Wed Sep 6 06:10:00 UTC 2000
NNT 0
NNW 0
PNT  99
PCW  99
PTW  98
NNM 0
CORMSLOG  CBOFSNOW.sh Finished at RealTimeClock=  Wed Sep 6 06:10:10 UTC 2000
CORMSLOG  CBOFSFORE.sh Started at RealTimeClock=  Wed Sep 6 06:15:00 UTC 2000
PFT  100
NFT 0
PFW  100
NFW 0
NFM 0
CORMSLOG  CBOFSFORE.sh Finished at RealTimeClock=  Wed Sep 6 06:15:06 UTC 2000
Percentage of ba_mllw.6min from DPAS NWLON SQL
PPT  75
PPR 0
```

Table 14. Typical CORMS-ready flag file.

```
1001001001001001000101001001001001001001001001001
```

Table 15. CORMS flags and criteria. In the criteria, Ran means the program ran to completion, Failed means that it did not. NA means not applicable.

No.	Symbol	Category	Variable	Criterion
0	GPOCL	Good	Percent Observed CBBT Water Level	≥80%
1	WPOCL	Warning	Percent Observed CBBT Water Level	≥60%,<80%
2	FPOCL	Fail	Percent Observed CBBT Water Level	<60%
3	GPOCW	Good	Percent Observed CBBT Wind Data	≥80%
4	WPOCW	Warning	Percent Observed CBBT Wind Data	≥60%,<80%
5	FPOCW	Fail	Percent Observed CBBT Wind Data	<60%
6	GPOTW	Good	Percent Observed Thomas Point Wind Data	≥80%
7	WPOTW	Warning	Percent Observed Thomas Point Wind Data	≥60%,<80%
8	FPOTW	Fail	Percent Observed Thomas Point Wind Data	<60%
9	XXXXX	XXXXX	Reserved for future use	≥80%
10	XXXXX	XXXXX	Reserved for future use	≥60%,<80%
11	XXXXX	XXXXX	Reserved for future use	<60%
12	GPFWL	Good	Percent Forecast Water Level Input Data (TDL)	≥80%
13	WPFWL	Warning	Percent Forecast Water Level Input Data (TDL	≥60%,<80%
14	FPFWL	Fail	Percent Forecast Water Level Input Data (TDL)	<60%
15	GPFW	Good	Percent Forecast Wind Input Data (ETA32)	≥80%
16	WPFW	Warning	Percent Forecast Wind Input Data (ETA32)	≥60%,<80%
17	FPFW	Fail	Percent Forecast Wind Input Data (ETA32)	<60%
18	GPPOL	Good	Percent Plot Observed Water Level (NWLON DPAS)	≥80%
19	WPPOL	Warning	Percent Plot Observed Water Level(NWLON DPAS	≥60%,<80%
20	FPPOL	Fail	Percent Plot Observed Water Level (NWLON DPAS)	<60%
21	GNWLF	Good	Nowcast Water Level Forcing file (gentidenew.x)	Ran
22	WNWLF	Warning	Nowcast Water Level Forcing file	NA
23	FNWLF	Fail	Nowcast Water Level Forcing file	Failed
24	GNWFF	Good	Nowcast Wind Forcing File (genwind_2obsB.f)	Ran
25	WNWFF	Warning	Nowcast Wind Forcing File	NA
26	FNWFF	Fail	Nowcast Wind Forcing File	Failed
27	GNMOF	Good	Nowcast Model Output File (mecca21.f)	Ran
28	WNMOF	Warning	Nowcast Model Output File	NA
29	FNMOF	Fail	Nowcast Model Output File	Failed
30	GFWLF	Good	Forecast Water Level Forcing file(gentidenew.f)	Ran
31	WFWLF	Warning	Forecast Water Level Forcing file	NA
32	WFWLF	Fail	Forecast Water Level Forcing file	Failed
33	GFWFF	Good	Forecast Wind Forcing File (genwindbin.f	Ran
34	WFWFF	Warning	Forecast Wind Forcing File	NA
35	FFWFF	Fail	Forecast Wind Forcing File	Failed
36	GFMOF	Good	Forecast Model Output File (mecca21.f)	Ran
37	WFMOF	Warning	Forecast Model Output File	NA
38	FFMOF	Fail	Forecast Model Output File	Failed
39	GPPR	Good	Plotting Programs Run (graphicsall.sh)	Ran
40	WPPR	Warning	Plotting Programs Run	NA
41	FPPR	Fail	Plotting Programs Run	Failed
42	GCSS	Good	CBOFS Server Status (computercorms.c)	Ran
43	WCSS	Warning	CBOFS Server Status	NA
44	FCSS	Fail	CBOFS Server Status	Failed
45	GFSS	Good	FAX Server Status (fax program)	Ran
46	WFSS	Warning	FAX Server Status	NA
47	FFSS	Fail	FAX Server Status	Failed
48	GWSS	Good	WEB Server Status (Web received graphics)	Ran
49	WWSS	Warning	WEB Server Status	NA
50	FWSS	Fail	WEB Server Status	Failed

8. MAINTENANCE AND TROUBLESHOOTING

8.1. Software Maintenance

CO-OPS' Information System Division (ISD) will be responsible for maintaining and upgrading the SGI Irix operating system (OS). The OS will be updated as patches and newer versions become available. ISD is also responsible for periodic backup of the system and maintaining backup tapes.

8.1.1. System Failure and Recovery

In the event of a hardware failure, the disk system should be reconstructed to as recent a state as possible from backup tapes. Assure that the /CBOFS directory structure is intact and that the programs and static data files exist. Because the system is self initializing the only action required is to assure that the crontab program is executing. While logged on to the CBOFS operations account, issue the crontab listing command:

```
crontab -l
```

The contents of the file /CBOFS/scripts/CBOFSCRON should display. If they do not then issue this command to start the crontab:

```
crontab /CBOFS/scripts/CBOFSCRON
```

The CBOFS programs will begin to execute at their appropriate times of the day. To test the system immediately, issue the main script commands in this order:

```
/CBOFS/scripts/CBOFSINIT.sh
/CBOFS/scripts/CBOFSNOW.sh
/CBOFS/scripts/CBOFSFORE.sh
/CBOFS/post/idlprograms/graphicsall.sh
```

This set of commands runs the entire CBOFS system, so all potential problems will be encountered.

8.1.2. Annual Maintenance

Long term archiving of model results consist only of the storage of the nowcast and forecast water level files /CBOFS/output/nowcastout/nowwl%j%H.out and /CBOFS/output/forecastout/forewl%j%H.out. About twice a year, these files should be moved to year-named directories such as /CBOFS/output/nowcastout/2000/. Simply create such a directory and move the old files to it.

The PORTS PUFFF files are stored with year-named files, but a large number in one directory becomes a nuisance. A script has been provided which is to be run several times a year to move the files to year-named directories. This script is /CBOFS/data/pufff/ARCHIVEYEAR.sh. It moves only PUFFF files older than 2 months to directories named like:

/CBOFS/data/pufff/wl838863/WL2000.

The astronomical tide files which are used by the plotting routines must be created for each new calendar year. In the directory /CBOFS/post/staticdata/astrotides is a 'README' file which describes this simple process. A script, runtidesall2001, exists in the directory /CBOFS/post/staticdata/astrotides/progs which will create the files for the year 2001. For subsequent years, copy the script to a new file and change its mode to executable. Edit the file for all occurrences of '2001' and change them to the new year. Copy the file INPUT2001.template to a file of the same name, but new year. Edit it to change 2001 to the new year. Execute the runtidesall script for the new year. This will create the new directory such as /CBOFS/post/staticdata/astrotides/2002 and fill it with the data files required by the plotting programs.

For the climatological river discharge data, rivers.met, the file contains dates with the year specified. Since the data are actually year independent, updating the file to the new year is only a matter of copying an old file and editing it for the new year. The file /CBOFS/staticdata/rivers19992000.met should be copied to an updated name and the years edited appropriately. The resultant file is copied to rivers.met, the file read by MECCA.

8.2. Hardware Maintenance

Maintenance of the CBOFS hardware will be handled by CO-OPS' ISD. All hardware computer components are covered under ISD's umbrella of maintenance contracts with Silicon Graphics. ISD continually monitors the CBOFS hardware through the use of automated procedures.

8.3. Troubleshooting

8.3.1. CORMS

All CO-OPS products are quality controlled by the Continuous Operational Real-Time Monitoring System (CORMS). CORMS is a decision support system implemented in April 1998 which provides 7 day-a-week, 24 hour-a-day monitoring and quality control of sensors and data in order to insure the availability, accuracy, and quality of tide, water level, current, and other marine environmental information. CORMS is intended to identify invalid and erroneous data and information before application of the data by real-time and near real-time users. The CORMS is currently monitored by teams of NOS technicians in the Silver Spring, Maryland, offices of the National Weather Service. See Section 7.3 for a discussion of error flags.

A Standard Operating Procedure (SOP) for CORMS Monitoring of CBOFS is available for the Operators. This SOP outlines the steps to be taken twice daily by the Operators to insure the availability and quality of the various CBOFS products. The SOP contained details regarding the products and the persons to be e-mailed regarding failures.

8.3.2. Software

The CBOFS system has been tested extensively and found to continue operations with little maintenance. However, many of the programs are sensitive to file formats and locations of input data streams. When another agency which supplies data to CBOFS changes its data formats, corrections must be made to the CBOFS software. Several of these possible changes are described here, along with recommendations for remedial action.

PUFFF Files

PORTS is a CO-OPS operation on which CBOFS relies, so changes in PORTS-PUFFF files can have an impact. The two programs `readpufffwl.sh` and `readpufffmt.sh` read the current PUFFF format. They use the Unix 'awk' command, which depends on counting numbers of lines and expect data to be separated by spaces. The addition of new lines or extra data or words on a line can cause problems. Corrective action requires carefully working through those scripts to account for the changes.

Because the scripts work on concatenated collections of PUFFF files, the period of time when the new format and old format exist together in the data stream will be very hard to work through. It is recommended that changes in PUFFF files be mirrored for a period of time to produce both the old and new simultaneously. Otherwise CBOFS operations may be interrupted for 8 days (the length of the Initialization phase.)

Thomas Point Winds

One problem in the past has been the decision of OSO to change the location of the Thomas Point data record. For example, once it was switched to the `USG*.SRVR` files when previously it had been found in the `WAL*.SRVR` collection. Another possible problem with OSO may be a change in the FTP computer address (currently 140.90.88.142, with a backup on 140.90.6.103). If either address changes, the Thomas Point winds will disappear from the CBOFS input data stream. A problem with this naming convention or the FTP link will be indicated if the data still appear on the NDBC Web page (http://www.ndbc.noaa.gov/station_page?$station=tplm2) proving that the station is still operational. Both problems can be solved by changes to `FTPOSOTPLM2.sh`. The FTP computer address appears on the lines with the FTP command. The problem with a name change of the file is solved by changing the line:

```
echo "mget USG*.SRVR">> ftp1.oso
```

Figuring out what to change `USG*.SRVR` to may be more difficult. (On one occasion, CO-OPS' ISD personnel downloaded dozens of possible files and ran the Unix command '`grep TPLM2 *`' to search for the occurrence of the Thomas Point code.)

ETSS Model

Changes to the format of NCEP ETSS model files can cause the ODAAS programs to fail. See ODAAS documentation for help. If ODAAS changes the format of the subsequent TDL files, the CBOFS programs `gentidenew.f` will crash. `Gentidenew.f` has a subroutine dedicated to reading TDL format files. Inspection of the new file format and the subroutine should solve the problem.

Eta32 Wind Forecasts

NCEP changes the grid spacing of their Eta model about once every 2 years. A new change is anounced well in advance and NWS usually runs models with old and new grids concurrently for a period to allow smooth transition. A change in grid results in a number of problems. For example, the subsetting of the grid by ODAAS will be affected. Once the ODAAS operator has solved the problem, a new Eta subsetted file be available for ingestion by `genwind_bin.f`. First, it will be of different dimension than the old one, so changes in the Fortran code of `genwind_bin.f` will need to address this. Secondly the new grid will require a new set of cell weights to be generated; `genwind_bin.f` has provisions for doing this task; see the `genwind_bin.f` documentation for details. After the new weights are generated, the new name of the cell weights file (`cellwgt32.out`) will have to be inserted into the `CBOFSFORE.sh` script where `genwind_bin.f` is called.

9. ACKNOWLEDGMENTS

Many people have contributed to CBOFS. The authors wish especially to thank CO-OPS' Thomas Bethem who headed the team which oversaw the transition of CBOFS to the operational environment. CSDL's John Kelley has worked tirelessly to build and maintain the ODAAS which provides the system with the NWS's atmospheric model forecasts. CSDL's John Cassidy developed most of the original graphical products, and CO-OPS' Michael Evans and Zhong Li refined the graphics and created the Web site. CO-OPS' Janet Burton has assisted in accessing the wind data files.

REFERENCES

Black, T. L., 1994. The new NMC mesoscale Eta model: Description and forecast examples. **Weather and Forecasting**, 9, pp.265-278.

Bosley, K. T., 1996. Toward a Nowcast/Forecast system for water levels in the Chesapeake Bay. **Proceedings of the Oceans 96/Marine Technology Society Meeting** September 23-26, 1996, Ft. Lauderdale, FL.

Bosley, K. T., T. Bethem, and M. J. Evans, 1999. Migration of the Chesapeake Area Forecasting Experiment (CAFE) to an Operational Nowcast/Forecast System: the Chesapeake Bay Operational Forecast System. NOS, Center for Operational Oceanographic Products and Services. 42 pp.

___, 1997. Chesapeake Area Forecasting Experiment (CAFE): Migration to an Operational Product. NOS, Center for Operational Oceanographic Products and Services. 27 pp + appendices

Bosley, K. T., and K. W. Hess, 1997. Development of an experimental Nowcast/Forecast system for Chesapeake Bay water levels. In **Estuarine and Coastal Modeling. Proceedings of the Fifth International Conference**, M.L. Spaulding and A. F. Blumberg, eds. American Society of Civil Engineers, N.Y. pp. 413-426.

Chen, J., W. Shaffer, and S. Kim, 1993. A forecast model for extratropical storm surge. **Advances in HydroScience and Engineering**. Vol. 1, pp. 1437-1444.

Gill, S., W. Stoney, and T. Bethem, 1997. System Development Plan, CORMS: Continuous Operational Real-Time Monitoring System. **NOAA Technical Report** NOS OES14. 41 pp.

Evans, M., G. French, and T. Bethem, 1998. PORTS Uniform Flat File Format (PUFFF). CO-OPS Internal technical report. 28 pp.

Gross, T. F., 2000. Skill assessment of the CAFE system. **NOAA Technical Report** NOS CS 7. 46 pp.

Haney, R. L., 1991: On the pressure gradient force over steep topography in sigma coordinate ocean models. **Journal of Physical Oceanography**, 21, 610 - 619.

Hess, K. W., 2000. MECCA 2 Program Documentation. US Department of Commerce NOAA, **NOAA Technical Report** NOS CS 5. 49 pp.

___, 1989. MECCA Program Documentation. US Department of Commerce, **NOAA Technical Report** NESDIS 46, 258 pp.

Hoff, M., 1990. A Chesapeake Bay Circulation Model. US Naval Academy, Annapolis, pp 26.

Johnson, D. F., and K. W. Hess, 1990. Numerical simulations of blue crab larval dispersal and recruitment, **Bulletin of Marine Science**, 46(1), 195-213.

Kelley, J. G. W., E. Wei, S. Maxwell, A. Thompson, and M. Westington, 2000. Description of the Operational Data Acquisition and Data Archive System (ODAAS) to Support the NOS Chesapeake Bay Operational Forecast System (CBOFS). **NOAA Technical Report** (In Preparation).

Kim, S.-C., J. Chen, and W. A. Shaffer, 1996. An operational forecast model for extratropical storm surges along the U.S. east coast. **Preprints Conference on Coastal Oceanic and Atmospheric Prediction,** Atlanta, Amer. Meteor. Soc., 281-286.

Kite-Powell, H. L.,1996. Formulation of a Model for Ship Transit Risk. Massachusetts Institute of Technology **Sea Grant College Program Report** No. 96-19.

Large, W. G., and S. Pond, 1981: Open ocean momentum flux measurements in moderate to strong winds. **Journal of Physical Oceanography** 11, 324 - 336.

Munk, W. H., and E. R. Anderson, 1948: Notes on the theory of the thermocline. Journal of Marine Research, 7, 276 - 295.

NOAA, 1994. Safeguarding Our Nation's Waterways With PORTS. A Report to the House of Representatives Committee on Appropriations in Response to House Report 103-57.

NOS, 1999. NOS procedures for developing and implementing operational nowcast and forecast systems for PORTS. **NOAA Technical Report** NOS CO-OPS 0020.

Tag, P. M., F. W. Murray, and L. R. Koenig, (1979) A comparison of several forms of eddy viscosity parameterization in a two-dimensional cloud model. **J. Applied Meteorology**, 18, 1429-1441.

Zervas, C., 1999. Tidal Current Analysis Procedures and Associated Computer Programs. **NOAA Technical Report** NOS CO-OPS 0021.

APPENDIX A. CBOFS SERVER CONFIGURATION

The specifications of the CBOFS server, which is a Silicon Graphics, Inc., Octane workstation.

One 195 MHZ IP30 Processor
CPU: MIPS R10000 Processor Chip Revision: 2.7
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Main memory size: 128 Mbytes
Instruction cache size: 32 Kbytes
Data cache size: 32 Kbytes
Secondary unified instruction/data cache size: 1 Mbyte
Integral SCSI controller 0: Version QL1040B (rev. 2), single ended
 Disk drive: unit 1 on SCSI controller 0
 Disk drive: unit 2 on SCSI controller 0
Integral SCSI controller 1: Version QL1040B (rev. 2), single ended
 Tape drive: unit 4 on SCSI controller 1: DAT
 CDROM: unit 6 on SCSI controller 1
IOC3 serial port: tty1
IOC3 serial port: tty2
IOC3 parallel port: plp1
Graphics board: SI
Integral Fast Ethernet: ef0, version 1, pci 2
Iris Audio Processor: version RAD revision 12.0, number 1

APPENDIX B. CBOFS DIRECTORY STRUCTURE

The CBOFS directory is located on the CO-OPS SGI. The CBOFS sub-directories and all files are found in the working CBOFS home directory (e.g. /CBOFS). The ODAAS system is located physically of the CBOFS directory but logically in the directory /ODAAS. The subdirectories of CBOFS are listed below.

Directory	Contents
data	Most of the input data files for runs (also see staticdata)
docs	Documentation files
execlog	CORMS and other files created during execution
output/nowcastout	Output of MECCA nowcast runs
output/forecastout	Output of MECCA forecast runs
post	Post-processing graphics, output scripts, and output files
progs	Fortran code and compiled executable
rundir	Temporary output files for the running programs are stored here. All are overwritten, so this is only useful to analyze the most recent runs.
scripts	The collection of scripts for running the programs, plus template files which get edited by the scripts and then serve as input files to the programs. The crontab files which runs these scripts at specified times are also here.
staticdata	Additional input data files describing model grid geometry and other fixed values

APPENDIX C. CRONTABS

The CBOFS collection of programs are run from the CBOFS directory on a single computer, controlled by Unix crontab files which execute routines at specified times each day. The following Unix crontab scripts (in `scripts/CBOFSCRON`) are enabled at specified times each day to do the described tasks:

1. CRONPUFFF

Continuously monitor and download the PORTS data files made available every 6 minutes for the real time data from the CBBT station. This is the source of the real-time data for the CBBT water level and wind information.

```
# PUFFF every 5 minute grab PUFFF files
1,6,11,16,21,26,31,36,41,46,51,56 * * * *
(/CBOFS/scripts/CRONPUFFF.sh > /CBOFS/execlog/pufff.log) 2>
/CBOFS/execlog/puffferr.log
```

2. FTPOSOTPLM2.sh

Access the OSO computer files to copy and format the wind data for Thomas Point CMAN station.

```
# FTPOSOTPLM2.sh Grabs and archives Thomas Point wind from the
OSO files
2 6,18  * * * (/CBOFS/scripts/FTPOSOTPLM2.sh >
/CBOFS/execlog/logFTPOSOTPLM2) 2> /CBOFS/execlog/ERRORFTPOSOTPLM2
#
```

3. CBOFSINIT.sh

Initialize (once a day) the hot start files based on best data available. Avoids the use of hot start files which were created with incomplete and filled data, which used forecasts to fill gaps.

```
# CBOFSINIT.sh Daily reinitialization: PUFFF CBBT wind and NWLON
water levels
35 12 * * * (/CBOFS/scripts/CBOFSINIT.sh >
/CBOFS/execlog/logcbofsINIT) 2> /CBOFS/execlog/ERRORcbofsINIT
```

4. CBOFSNOW.sh

Run the nowcast for at least the last 12 hours up to the present. This script uses the current date and time to determine what input data files are available and then uses `gentidenew.f` and `genwind_obs2.f` to create input data files for the running of MECCA. Finally writes output files to archive and makes them available for plotting.

```
# CBOFSNOW.sh Nowcast Thomas Point and CBBT winds and PUFFF water
levels
10 6,18  * * * (/CBOFS/scripts/CBOFSNOW.sh  >
/CBOFS/execlog/logcbofsNOW) 2> /CBOFS/execlog/ERRORcbofsNOW
```

5. `CBOFSFORE.sh`

Run the forecast for the next 24 hours starting with the end of the nowcast. This script determines the ending time of the most recently run nowcast and assembles input data files from TDL water level predictions and ETA32 wind predictions. MECCA is then run from the last nowcast hotstart file for initial conditions and the output data is archived and made available for plotting.

```
# CBOFSFORE.sh ETA32 winds and TDL water levels
15 6,18  * * * (/CBOFS/scripts/CBOFSFORE.sh  >
/CBOFS/execlog/logcbofsFORE) 2> /CBOFS/execlog/ERRORcbofsFORE
```

6. `graphicsall.sh`

Create graphics from the output of the most recent run of the nowcast and forecast.

```
# Create ALL the graphics files
18 6,18  * * * (/CBOFS/post/idlprograms/graphicsall.sh >
/CBOFS/execlog/loggraphicsall) 2> /CBOFS/execlog/ERRORgraphicsall
```

7. `MAKECORMSFLAG.sh`

Make CORMS flags. After all the other scripts have run in one cycle, the CORMS flag generation program is run. This reads the CORMS flag data in file `/CBOFS/execlog/CORMS/%Y%m%d%Hcorms.dat` and processes it into the set of flags for the CORMS operators, `/CBOFS/execlog/CORMS/CBOFSFLAGS.txt`.

```
#  MAKECORMSFLAG.sh
#  After everything else has run this hour
#  Read the corms data and produce the flags
25 6,18  * * * (/CBOFS/scripts/MAKECORMSFLAG.sh >
/CBOFS/execlog/logmakecorms) 2> /CBOFS/execlog/ERRORmakecorms
#
```

8. Crontab Control Commands

Additional cron commands are available for special purposes. To remove or stop all of existing crontabs owned by the operator:

```
crontab -r
```

Start the necessary cron scripts:

```
crontab /CBOFS/scripts/CBOFSCRON
```

List jobs to confirm that they are running:

```
crontab -l
```

The crontabs will remain running, even after the computer is rebooted, unless something catastrophic occurs to the computer disk system.

APPENDIX D. SUMMARY OF CBOFS PROCESSES, INPUT, AND OUTPUT FILES

Table D.1. Processes and files, listed by task. Data file names are given in Table D.2 and program directories are in Table D.3. **Notes:** [1] I= Initialization Run, N= Nowcast Run, and F= Forecast Run. [2] % indicates a date format and is identical to Unix 'date' command syntax: %Y = four digit year; %m = two digit month; %d = two digit day of month; %H = two digit hour (00 to 23), always GMT; %j = three digit day of the year (001 to 366). All with leading zeros. [3] (a) The full file name is: %Y%m%d00_OSOtplm2.met. (b) The full file name is %Y%m%d%Heta32.cb.bin. (c) All MECCA runs require fullbay21c.geo and rivers.met as static inputs. (d) Similarly-named files exist for other NOS stations. (e) Similar files exist, called middlebay and lowerbay.

1a. Ingest (Acquisition)

Data Type	Source	Controlling Software	Supporting Software	Output Files
CBBT wl obs (I)	DPAS	CBOFSINIT.sh	quer.sh	cbbt_msl.6min
CBBT wl obs (N)	PUFFF	CRONPUFFF.sh		%Y%m%d.wl
CBBT met obs (I,N)	PUFFF	CRONPUFFF.sh		%Y%m%d.wl
TP met obs (I,N)	OSO	FTPOSOTPLM2.sh		OSOtplm2.met (a)
TDL-ETSS fcst (F)	NCEP Cray	ODAAS		%Y%m%d%H.cbbt
Eta32 wind fcst (F)	NCEP Cray	ODAAS		eta32.cb.bin (b)

1b. Ingest (Reformatting)

Data Type	Input Data	Controlling Software	Reformatting Software	Output Files
CBBT wl obs (N)	%Y%m%d.wl	CBOFSNOW.sh	readpuffwl.sh	pufff_cbbt.6min
CBBT met obs (I)	%Y%m%d.wl	CBOFSINIT.sh	readpuffmt.sh	cbbt_metpufff
CBBT met obs (N)	%Y%m%d.wl	CBOFSNOW.sh	readpuffmt.sh	cbbt_metpufff
TP met obs (I)	OSOtplm2.met (a)	CBOFSINIT.sh		INITtplm.met
TP met obs (N)	OSOtplm2.met (a)	CBOFSNOW.sh		NOWtplm.met

2. Generate Forcing Fields

Field	Input Data	Controlling Software	BC Production Software	Output Files
Coastal wl (I)	cbbt_msl.6min	CBOFSINIT.sh	gentidenew.f	gentide_init.out
Wind fields (I)	INITtplm.met cbbt_metpufff fullbay21c.geo	CBOFSINIT.sh	genwind_2obsB.f	genwind_init.out
Coastal wl (N)	pufff_cbbt.6min	CBOFSNOW.sh	gentidenew.f	gentide_now.out
Wind fields (N)	NOWtplm.met cbbt_metpufff fullbay21c.geo	CBOFSNOW.sh	genwind_2obsB.f	genwind_now.out

Coastal wl (F)	bt_m_msl_g.in, yr %Y%m%d%H.cbbt		gentidenew.f	gentide_fore.out
Wind fields (F)	eta32.cb.bin (b)		genwind_bin.f90	genwind_fore.out

3. Model Runs

Operations	Input Files(c)	Controlling Software	Model Code	Output Files
Initialization	init.con gentide_init.out genwind_init.out	CBOFSINIT.sh readinitspace.f90	mecca21.f	HOTSTART.DAT
Nowcast	now.con HOTSTART.DAT gentide_now.out genwind_now.out	CBOFSNOW.sh readinitspace.f90	mecca21.f	nowFIN.DAT nowwlrecent.out wl_idl.now.out wn_idl.now.out nowwl%j%H.out
Forecast	fore.con nowFIN.DAT gentide_fore.out genwind_fore.out	CBOFSFORE.sh	mecca21.f	fore.wl.out forewl%j%H.out wl_idl.fore.out wn_idl.fore.out latestfore.out

4. Post Processing

Function	Input Data	Controlling Program	Auxiliary Programs	Output
Get WL Obs	DPAS	graphicsall.sh	getgraphicsdatadpas.sh	bt_mllw.6min (d)
Get WL Preds	8638863 (d)	graphicsall.sh	runalltides prefabs.f, greg2yday wlpreds.f	bt_astro.dat (d)
Plot WL Time Series, Station	bt_mllw.6min (d) bt_astro.dat (d) nowwlrecent.out fore.wl.out	graphicsall.sh	cafestation.pro	cbbt.gif (d)
Plot WL Time Series, Group	bt_mllw.6min (d) bt_astro.dat (d) nowwlrecent.out fore.wl.out	graphicsall.sh	wlwinds.pro	upperbay.gif (e) upperbay.ps (e)
Plot Wind Time Series	bt_mllw.6min (d)	graphicsall.sh	wlwinds.pro	winds.gif winds.ps
Plot Water Level Animations	wl_idl.now.out wl_idl.fore.out	graphicsall.sh	nowforecast.pro	nowwlanim.gif forewlanim.gif
Plot Wind Animations	wn_idl.now.out wn_idl.fore.out	graphicsall.sh	nowforecast.pro	nowanimwinds.gif foreanimwinds.gif

Table D.2. Data file names, with directories. All files are in directory `/CBOFS`. Notes: % is used for date notation as in previous Table. (a) additional files exist for other water level stations with a 4-character naming convention. (b) additional files exist for other water level stations with a 2-character naming convention. (c) Additional files named `middlebay.gif` and `lowerbay.gif` exist.

File Name	Directory
cbbt.gif (a)	post/gifs
cbbt_msl.6min	rundir
cbbt_mllw.6min (a)	data/wlnet/STATIONS
cbbt_metpufff	rundir/init , rundir/now
bt_mllw.6min (b)	data/wlnet/STATIONS
bt_m_msl_g.in	data/astrotides
bt_astro.dat (b)	post/staticdata/astrotides/2000
pufff_cbbt.6min	rundir/now
fore.con	rundir/fore
fore.wl.out	rundir/fore
forewl%j%H.out	output/forecastout
forewl%H.gif	post/idlprograms/nowforecast/pics
forewind%H.gif	post/idlprograms/nowforecast/pics
gentide_init.out	data
gentide_now.out	data
gentide_fore.out	data
genwind_init.out	data
genwind_now.out	data
genwind_fore.out	data
HOTSTART.DAT	data/HOTSTART.DAT
init.con	rundir/init
INITtplm.met	rundir/init
latestfore	post
NOWtplm.met	rundir/now
nowFIN.DAT	data
now.con	rundir
nowwwlrecent.out	post
nowwl%j%H.out	output
nowwl%H.gif	post/idlprograms/nowforecast/pics
nowwind%H.gif	post/idlprograms/nowforecast/pics
rivers.met	staticdata
upperbay.gif (c)	/var/www/htdocs/cafe
wl_idl.now.out	post
wn_idl.now.out	post
wl_idl.fore.out	post
wn_idl.fore.out	post
winds.gif	var/www/htdocs/cafe
winds.ps	var/www/htdocs/cafe
yr	data/astrotides
%Y%m%d.wl	data/pufff/wl8638863 (d)
%Y%m%d00_OSOtplm2.met	data/OSO/TPLM2
%Y%m%d%H.cbbt	data/tdl/%Y%m/cb
%Y%m%d%Heta32.cb.bin	data/eta32/%Y%m/cb
8638663 (d)	post/idlprograms/wlwinds/wlpreddata

Table D.3. Program file names with directories. All files are in directory /CBOFS. Fortran source code (.f) and executable code (.x) reside in the same directories.

File Name	File name with directory
CBOFSINIT.sh	scripts
CBOFSNOW.sh	scripts
CBOFSFORE.sh	scripts
FTPOSOTPLM2.sh	scripts
gen_corn.f	progs
gifmerge	post/idlprograms/nowforecast
graphicsall.sh	post/idlprograms
getgraphicsdatadpas.sh	post/idlprograms/stations
gentidenew.f	progs
genwind_2obsB.f	progs
genwind_bin.f90	progs
greg2yday	astrotides/progs
mecca21.f	progs
prefabs.f	astrotides /progs
readinitspace.f90	progs
runalltides	post/staticdata/astrotides/progs
wlpred.f	post/idlprograms/wlwinds/wlpreddata

APPENDIX E. DOCUMENTATION OF THE CBOFS GRAPHICS SOFTWARE

This section contains documentation for the IDL plotting programs `wlwinds.pro`, which creates the time series of water levels and winds, and `nowforecast.pro`, which creates the animations fields for water levels and winds. The programs reside in the `/CBOFS/post` directory.

1. Program Filename: wlwinds.pro

1.1. Purpose

The program is implemented for the CBOFS to produce graphical output of water level time series, which include observed, predicted, nowcast and forecast water levels (Figure 2). The program also provides winds time series display (Figure 6). The outputs are formatted in both Postscript and GIF; the Postscript files will be used for fax-on-demands and GIF files will be for Web pages.

1.2. Input Data Files

A. Observed water levels

The program module ObsWLdata is used to read the observed water levels data. Data files are listed as follows. The type of the data files is ASCII text, as shown in Table 8.

Station	Data sources
Gloucester Point	data/wlnet/STATIONS/gp_mllw.6min
Kiptopeke	data/wlnet/STATIONS/kp_mllw.6min
Hampton Road	data/wlnet/STATIONS/ha_mllw.6min
CBBT	data/wlnet/STATIONS/bt_mllw.6min
Lewisetta	data/wlnet/STATIONS/lw_mllw.6min
Solomons Island	data/wlnet/STATIONS/so_mllw.6min
Colonial Beach	data/wlnet/STATIONS/co_mllw.6min
Cambridge	data/wlnet/STATIONS/ca_mllw.6min
Baltimore	data/wlnet/STATIONS/ba_mllw.6min
Tolchester	data/wlnet/STATIONS/to_mllw.6min
Annapolis	data/wlnet/STATIONS/an_mllw.6min

B. Predicted water levels

The module PredWLdata is used to read the prediction water levels data. The prediction water levels data for each station is produced on-the-fly by running a Fortran program `/CBOFS/post/idlprograms/wlwinds/wlpreddata/wlpred` (See Zervas, 1999). A shell program `wlpred.sh` is used as follows:

/CBOFS/post/idlprograms/wlwinds/wlpred.sh stationid startdate enddate wlpred.dat,

This finally results in the prediction data file /CBOFS/post/idlprograms/wlwinds/wlpreddata/wlpred.dat.

Additional input consists of the tidal constituents for each station (e.g., /CBOFS/post/idlprograms/wlwinds/8638863).
The type of the output data files is ASCII text. The contents are the date, time (hr:min), and water level.

07/08/2000 00:00	0.791
07/08/2000 00:06	0.817
07/08/2000 00:12	0.842

C. Nowcast and forecast water levels

The module NowWLdata is used to read the nowcast water levels data. The data file is /CBOFS/post/nowwlrecent.out (See Table 12). The module ForeWLWinddata is used to read the forecast water levels data and winds data. The data file is /CBOFS/post/latestfore (see Table 12).

D. Observed winds

The module ObsWindsdata is used to read the observed winds data. The type of the data files is ASCII text, such as in Table 8. Data files are listed as following:

Station	Data sources
CBBT	rundir/now/cbbt_metpufff
Thomas Point	rundir/now/NOWtplm.met

1.3. Program Module Description

All program modules are briefly described in the following table.

Module Name	Brief Description
Color_Setup	Sets up a color table with 8 colors.
Check_Year	Checks whether a year is a non-leap or leap year.
Cal_Jul	Returns a value of Julian day after converting from the calendar date.
Cal_Greg	Converts Julian date to Calendar date.
LABEL_DATE_TIME	Labels axes with date and time (hour:min).

Init_Arrays	Initializes all data as 0 in data arrays.
Read_OWL	Reads the observed water levels data from files into variables.
ObsWLdata	Gets observed water levels data.
Read_PWL	Reads the predicted water levels data from files into variables.
PredWLdata	Gets prediction water levels data.
NowWLdata	Gets nowcast water levels data.
ForeWLWinddata	Gets forecast water levels and winds data, and also determines the time range and time zone (EDT or EST) for the time series plots.
Read_ObsWinds	Reads the observed winds data from files into variables.
ObsWindsdata	Gets observed winds data.
Plot_Legends	Plots legend labels for grouped water levels time series plots.
Plot_LowBay	Plots Lower Bay water levels time series.
Plot_MidBay	Plots Middle Bay water levels time series.
Plot_UpperBay	Plots Upper Bay water levels time series.
Plot_Station	Plots water levels time series for each station.
Plot_EachSta	Makes the gif file for each station water levels time series.
Plot_Winds	Plots winds time series.
Plot_NoData	Displays text saying data unavailable.
CAFEMain	Is main module and produces postscript and gif output files.

1.4. Other Related Files

Other related files are listed as the following:

File Name	Usage
post/idlprograms/wlwinds/wlwinds.sh	Sets up IDL environment parameters and runs wlwinds.pro.
post/idlprograms/wlwinds/runwlwinds	IDL commands to run wlwinds.pro.
post/idlprograms/wlwinds/wlpred.sh	Runs to get on fly prediction water levels data.
post/idlprograms/wlwinds/wlpreddata/wlpred.f	Fortran programs to produce the on fly prediction water levels data.
post/idlprograms/wlwinds/wlpreddata/wlpred	Execution program from the above Fortran program wlpred.f.
post/idlprograms/wlwinds/wlpreddata/[station id]	Constant file for each station used by the above wlpred.

1.5. Output Files

Postscript and gif files produced in the program are as follows:

Filename	File Size (Bytes)	Made By
/var/www/htdocs/cafe/lowerbay.ps.	850940	Plot_LowBay
/var/www/htdocs/cafe/lowerbay.gif	20556	"
/var/www/htdocs/cafe/middlebay.ps	850940	Plot_MidBay
/var/www/htdocs/cafe/middlebay.gif	19240	"
/var/www/htdocs/cafe/upperbay.ps	850940	Plot_UpperBay
/var/www/htdocs/cafe/upperbay.gif	15311	"
/var/www/htdocs/cafe/winds.ps	602487	Plot_Winds
/var/www/htdocs/cafe/winds.gif	14167	"
/var/www/htdocs/cafe/anna.gif	9732	Plot_EachSta
/var/www/htdocs/cafe/balt.gif	9693	"
/var/www/htdocs/cafe/camb.gif	10155	"
/var/www/htdocs/cafe/cbbt.gif	10027	"
/var/www/htdocs/cafe/colo.gif	10121	"
/var/www/htdocs/cafe/glou.gif	10604	"
/var/www/htdocs/cafe/hamp.gif	10757	"
/var/www/htdocs/cafe/kipt.gif	10714	"
/var/www/htdocs/cafe/lewi.gif	9848	"
/var/www/htdocs/cafe/solo.gif	9910	"
/var/www/htdocs/cafe/tolc.gif	9753	"
/var/www/htdocs/cafe/nodata.gif	4643	Plot_NoData

2. Program Filename: nowforecast.pro

2.1. Purpose

The program `nowforecast.pro` is implemented to plot nowcast and forecast water levels and winds of Chesapeake Bay. Several gif image files are produced and then they are used to make an animation gif file for the Web page.

2.2. Input Data Files

Input data files of the program are listed here and described in detail later.

Data Sources	Description
post/idlprograms/nowforecast/data/mecca_grd.corsid	Lat. and long. for water levels
post/idlprograms/nowforecast/data/mecca.lat_lon	Lat. and long. for winds
post/idlprograms/nowforecast/data/ches_co.dat	Lat. and long. for coastline
post/idlprograms/nowforecast/data/elevcol.dat	Color table for WL contour
post/wl_idl.now.out	Nowcast water levels
post/wl_idl.fore.out	Forecast water levels
post/wn_idl.now.out	Nowcast winds
post/wn_idl.fore.out	Forecast winds

A. File /CBOFS/post/idlprograms/nowforecast/data/mecca_grd.corsid

This file contains latitude and longitude data arrays (68 by 110) for water level contour display. The file type is ASCII text. The data are column index (M), row index (N), latitide, and longitude of the points.

1	1	39.87495	-77.12904
1	1	39.87495	-77.12904
1	3	39.86140	-77.06749

It is read using:

```
openr,10,'/CBOFS/post/idlprograms/nowforecast/data/mecca_grd.corsid'
          rlatwl = fltarr(68,110)
          rlonwl = fltarr(68,110)
          while (not eof(10)) do begin
            readf, 10, m, n, lat, lon
            rlatwl(n-1,m-1) = lat
            rlonwl(n-1,m-1) = lon
          endwhile
          close, 10
```

B. File /CBOFS/post/idlprograms/nowforecast/data/mecca.lat_lon

This file contains latitude and longitude array (34 by 55) data for winds vector display. The file type is unformatted data made by Fortran77. It is read using:

```
rlatwind = fltarr(34,55)
rlonwind = fltarr(34,55)
openr,10,'/CBOFS/post/idlprograms/nowforecast/data/mecca.lat_lon',
/f77_unformatted
readu, 10, rlatwind, rlonwind
```

C. File /CBOFS/post/idlprograms/nowforecast/data/ches_co.dat

This file contains latitude and longitude data for plotting the coastline of Chesapeake Bay. The file type is ASCII text and contains latitude, longitude, and a pen code.

36.996948	76.581108	1	0	1
37.000000	76.580276	1	1	2
36.931114	76.013611	1	0	1

The file is read using

```
openr,10,'/CBOFS/post/idlprograms/nowforecast/data/ches_co.dat'
readf, 10, lat, long, ipencode
```

D. File /CBOFS/post/idlprograms/nowforecast/data/elevcol.dat

This file (below, right) contains the color scale for color-filled contouring of water levels. Its type is ASCII text and is listed below.

It is read using

```
openr,11,'/CBOFS/post/idlprograms/nowforecast/dat
a/elevcol.dat'
readf,11,format='(i3,1x,i3,1x,i3)',red,green,blue
```

Color scale file, showing intensity for red, green, and blue.

255	255	255
0	0	0
255	0	0
255	46	0
255	92	0
255	138	0
255	184	0
255	230	0
232	255	0
186	255	0
140	255	0
94	255	0
48	255	0
1	255	0
0	255	44
0	255	90
0	255	136
0	255	182
0	255	229
0	234	255
0	188	255
0	142	255
0	96	255

E. File /CBOFS/post/wl_idl.now.out (or wl_idl.fore.out)

These files contain water levels (68*110) data for two dimensional fields of latitude and longitude at several specific times. The file type is unformatted data made by Fortran77. It is read using:

```
openr, 30, datafile, /f77_unformatted
wl = fltarr(68,110)
readu, 30, year, rjd, wl
```

F. File /CBOFS/post/wn_idl.now.out (or wn_idl.fore.out)

These files contain wind speed data arrays (34 by 55) in x direction and y direction for two dimensional fields of latitude and longitude at several specific times. The file type is unformatted data made by Fortran77. It is read using:

```
wx = fltarr(34,55) & wy = fltarr(34,55)
openr, 30, datafile, /f77_unformatted
readu, 30, year, rjd, wx, wy
```

2.3. Program Module Description

Program modules are briefly described in the following table.

Module Name	Brief Description
Check_Year	Checks whether a year is a non-leap or leap year.
Cal_Greg	Converts Julian date to Calendar date.
shoreline_labels	Plots the coastline of Chesapeake Bay and labels of cities and grid lines.
set_color_contour_levels	Establishes color scale for color-filled contouring of water levels.
set_color_winds	Establishes color scale for winds.
windscale	Plots the color scale labels for winds.
wlscale	Plots the color scale bar and labels for water levels.
Plot_Water_Levels	Plots nowcast and forecast water level contour.
Plot_Winds	Plots nowcast and forecast wind vector.
NowForeMain	The main module that calls other modules.

2.4. Other Related Files

Other related files are listed as the following:

File Name	Usage
post/idlprograms/nowforecast/nowforecast.sh	Sets up IDL environment parameters, runs nowforecast.pro, and combines the output gif files to produce animation gif files.
post/idlprograms/nowforecast/runnowforecast	IDL commands to run nowforecast.pro.
post/idlprograms/nowforecast/gifmerge.*	C program and execution files used to merge gif files to produce animation gif files.
/var/www/htdocs/cafe/nowwlanim.gif	Output animation gif file.
/var/www/htdocs/cafe/forewlanim.gif	Output animation gif file.
/var/www/htdocs/cafe/nowwindanim.gif	Output animation gif file.
/var/www/htdocs/cafe/forewindanim.gif	Output animation gif file.

2.5. Output Files

Output image gif files of the program are listed in the following table.

Output Filenames	In Module	Description
post/idlprograms/nowforecast/pics/nowwl!!.gif*	Plot_Water_Levels	WL contour
post/idlprograms/nowforecast/pics/forewl!!.gif	Plot_Water_Levels	WL contour
post/idlprograms/nowforecast/pics/nowwind!!.gif	Plot_Winds	Wind vector
post/idlprograms/nowforecast/pics/forewind!!.gif	Plot_Winds	Wind vector

* Note: !! in filenames means time series number 01, 02, ..., 12, etc.

APPENDIX F. GENERATION OF THE WATER LEVELS FOR ANIMATION

For graphing purposes, a field of new water levels (Figure 18) is created by spatially-interpolating values from the existing grid in subroutine `merge` in `mecca21.f`. Values in the new grid are created at positions in the computational grid corresponding to the cell center, the cell corners, and the middle of cell sides, as shown in Figure E.1.

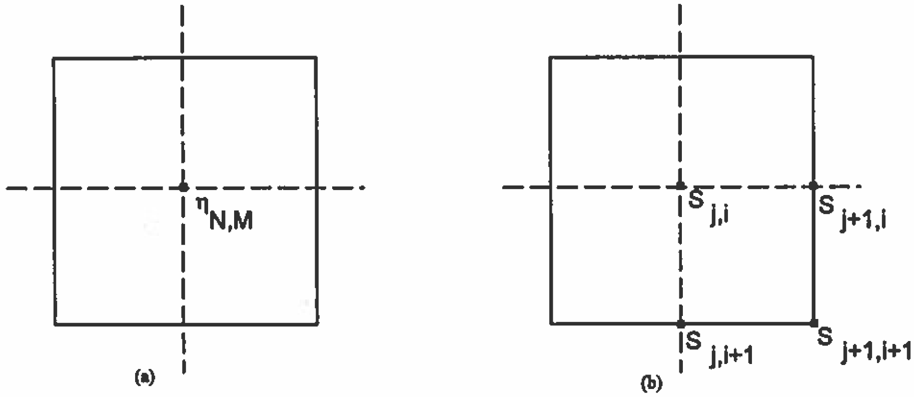

Figure E.1. Definition sketch showing (a) a cell (N, M) in the original grid and the water level η. In (b), the cell is shown with positions of new values (denoted by an '*S*') created by interpolation. Values shown with subscripts: $j=1+2(N-1)$ and $i=1+2(M-1)$.

Values of the interpolated water level field, *S*, are found from the water levels computed on the original grid, η, in several ways, depending on position. For water cells, *S* at the center of the old cell has the value of η in that cell because they occupy the same position. Values midway along cell sides are computed by a linear interpolation of the values in the cells that have that side in common, and values at cell corners are an average of four surrounding values. If some of the adjacent cells are not water, their water levels values are not used in the interpolation.

For example, $S_{j+1,i+1}$ is computed as follows:

$$S_{j+1,i+1} = \frac{I_{3,N,M}\eta_{N,M} + I_{2,N+1,M}\eta_{N+1,M} + I_{1,N+1,M+1}\eta_{N+1,M+1} + I_{4,N,M+1}\eta_{N,M+1}}{I_{3,N,M} + I_{2,N+1,M} + I_{1,N+1,M+1} + I_{4,N,M+1}}$$

where the cell corner index *I* has a value of either 0 or 1. Also, for the value at the center of the right side of the cell,

$$S_{j+1,i} = \frac{min(I_{3,N,M}, I_{4,N,M})\eta_{N,M} + min(I_{1,N+1,M}, I_{2,N+1,M})\eta_{N+1,M}}{min(I_{3,N,M}, I_{4,N,m}) + min(I_{1,N+1,M}, I_{2,N+1,M})}$$

The four indices, *I*, for each cell are generated in program `gen_corn.f`, which reads the geography file (set to `fullbay21c.geo` in the code).