Towards an Operational Model for the West Iberian Atlantic based on the Mercator and MOHID
Draft v0.2

Guillaume Riflet
MARETEC IST

March 13, 2007

Contents

Chapter 1

Introduction

For the past eight months, since mid-May 2006, Maretec-IST has been receiving weekly extractions of U (zonal velocity), V (meridional velocity), T (temperature), S (salinity) and η (water level) of Mercator's PSY2V2R1 operational solution for the western Iberia coast. Being each constituted by 14 days of reanalysis and 14 days of predictions, these extractions are the finest product, regarding operational oceanography, Mercator-Océan has to offer. Assimilating altimetry remote-sensing data, *in-situ* data and forced by a 10 day atmospheric prediction given by the ECMWF[1], the $1/15^{o}$ horizontal resolution model, product from the MERSEA [2] project, presents itself amongst the most robust meddy-permitting operational forecasting system for the North Atlantic area [18]. IST has got a one year contract with Mercator-Océan in exchange of quarterly reports and a final annual report.

Since late November 2006, Maretec-IST has been receiving atmospheric 7 day forecasts from MM5 model, managed by professor J. J. Delgado Domingo's group at IST[3][16]. This exchange is an agreement that allows several PhD students to work on their thesis related with ocean-atmosphere model coupling. The 9 km horizontal resolution model is capable of inducing upwelling scenarios in an hydrodynamic coastal model as well as reproducing fairly accurately the fate of surface tracers such as oil spills [29].

Since November 2006, a simple pre-operational model of the hydrodynamics for the continental portuguese coast started producing weekly 4 day forecasts available at an internet web-site[4], and ready to inspect with the

[1] European Centre for Medium-range Weather Forecast
[2] http://w3.mersea.eu.org/html/ocean_modelling/global_tep.html
[3] http://meteo.ist.utl.pt
[4] http://data.mohid.com/data.xml

aid of a graphical tool such as `Ncbrowse`[5]. The pre-operational model makes use of Mercator solution for oceanic forcing and MM5 model for atmospheric forcing. As Mercator provides only daily means, tide is enforced by a 2D barotropic model forced with the FES2004[31] solution. Furthermore, a downscalling for the Estremadura bank region is also simulated, which encompasses the INSEA project area. Although we can expect to accurately simulate upwelling events and the consequent nutrients transport to the photic zone in the Tagus region of influence, a higher wind resolution ($\sim$ 1 km) would be necessary to reproduce realistic wind-induced surface current patterns near Tagus estuary's mouth.

This report aims at describing best the work done so far for building this pre-operational system, and at examining more closely the outputs from the models in order to assess their capability for primary production forecasting near the Tagus mouth region of influence, when coupled with ecological models. In particular, methodologies to plan the building of an operational system are proposed (and were used) as well as a generic downscalling method of the Mercator PSY2V2R1 solution with a 3D baroclinic hydrodynamic model. It can be seen as a product of the INSEA, EASY and ECOOP projects. Probably, some parts of this document, and some results, will embed one or more of the INSEA, EASY and ECOOP project deliverables.

[5] http://www.epic.noaa.gov/java/ncBrowse/

Chapter 2

A Pre-Operational System

The goals of this operational system are twofold:

- Provide offline model results to force local marine environment models such as the ones created to cope with Maretec or Hidromod projects.
- Provide forecasts of sea salinity, sea temperature and sea currents, for any determined group of end-users and end-user model.

If these goals are met, the coupling of ecological models becomes possible. This enables algae and phytoplankton growth forecasting off western Iberia coast. Fisheries and environmental monitoring would then have an appreciable input.

In this section, we will first enounce what we conveyed as being a sound method to plan and define a generic operational model, after which we will describe its implementation to our particular system and goals.

The original plan to build the current pre-operational system was depicted in a state-process model as shown in figure 2.1, where a set of states would characterize steps in the operational framework, and processes would characterize the work required to change from a state to another state, step-by-step. The basic idea depicted in figure 2.1 is that each forcing model (atmosphere, ocean and tide) provides raw data (each encapsulated in a state which is renewed every week, or not,) that must be transformed into "cooked" data (encapsulated in a state) ready to feed the Mohid model as input. Once the running process - or processes, since we may want to run several concurrent runs at once, a prediction mode and an analyzed mode for example, - is done, the output data (representing the results state) goes through the publishing process, which makes the results available to the de-

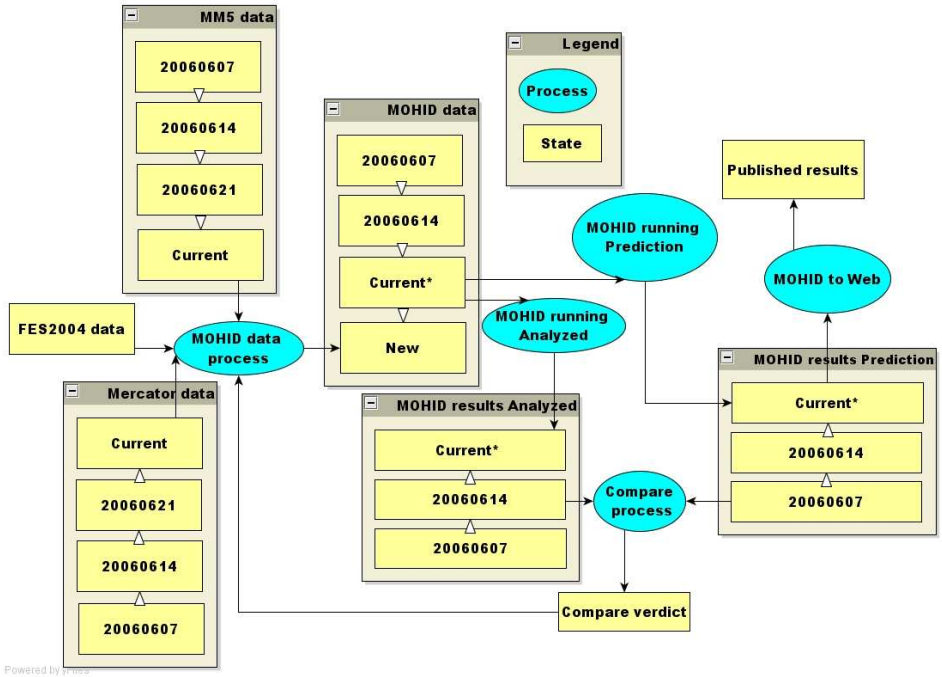

Figure 2.1: Processes-States diagrams of all the information required to develop a coastal oceanographical operational system. Square boxes represent the system's states while elipses represent the processes. Black arrows describe the information flow between states and processes. White arrows flow chronologically along the states' causal relations.

sired group of end-users. During the publishing process, the results may be processed into end-user ready input.

2.1 Methodology

2.1.1 The Transfer-Transform schema

In order to design the plan's implementation, experience showed a couple of symmetries, patterns and guidelines:

- The states are in fact receptacles of information.
- For all possible states, the information is coded in binary or ASCII

format and is stored in files managed by local and networked file systems.

- All files have an address, or emplacement, or locus within the file system or the network.
- Each process can be regarded as a mapping from an input to an output.
- Any process input is typically a file or a stream of characters or numbers, as well as is its output.
- The processes need a computing facility to run upon.
- In a small low-budget network, it is advisable to run each process where the computing resort is best fit to do it, instead of concentrating all processes in the same computing resort or computing centre. The latter approach may even prove to be unreliable (for example, it is renown to be unwise to keep data and a web-server on the same computing resource).
- The whole operational system should be seen as a distributed application.

From the above guidelines, a three-fold synthesis of the basic concepts an operational system needs can be made:

1. Information (may equivalently be put as state, object, file, data).
2. Transfer (may equivalently be put as copy, download, upload, load, sending, receiving).
3. Transform (may equivalently be put as process, run, map, code).

With these basic concepts one can create a succession (or a lattice) of transfers and transformations of information leading to the final form of the information. This succession (or lattice) forms a coherent structure or building that can be schematized prior to its implementation. Its basic building block is schematized in figure 2.2.

For example, a possible lattice of the pre-operational plan in figure 2.1 is shown in figure 2.3. The lattice in figure 2.3 only describes the data process, the run process and the publish process exposed in figure 2.1. The publish process is given more detail as it is split into a data post-processing and a publishing process. Concretely put, to implement the lattice, any transfer needs to be encoded in a script in shell language; and the script must copy one or several files. On the other hand, any transformation needs to:

1. generate the program configuration file, the command line switches and command line options,

2. run the program.

Once again, a script written in a common and flexible scripting language is best suited for such task.

2.1.2 The synchronous Vs the asynchronous approach

There are two ways to tackle the implementation of any operational system:

- whether one uses a synchronous approach,

- whether one uses an asynchronous approach.

The former approach requires a watch or a timer that counts time. All the procedures - Let's define a procedure as being a transfer or a transformation, in a generic fashion - will look at the time in order to know when they should trigger. Whereas the latter approach requires a service running in the background, from above at the highest level, supervising and ordering the triggering of the procedures.

The second approach is potentially more robust than the former. Indeed, should a procedure finish later than scheduled, chances are that an error could occur in a synchronous approach; whereas, in an asynchronous approach, the background "supervisor" would wait for the procedure to finish successfully, prior to start executing the next procedure. The execution time of a procedure may vary due to several factors:

- The file transfer time inside the network is subjected to fluctuations (caused by large traffic in a limited bandwidth network). These fluctuations may vary a lot.

- The file transfer time in the internet is also subjected to these fluctuations.

- The program execution time varies a lot from a computer to another.

- The program execution time varies within a single computer (processes fight for computer resources, such as ram).

All of these factors tend, in the synchronous approach, to underestimate the effective execution time of processes, yielding the need of considering

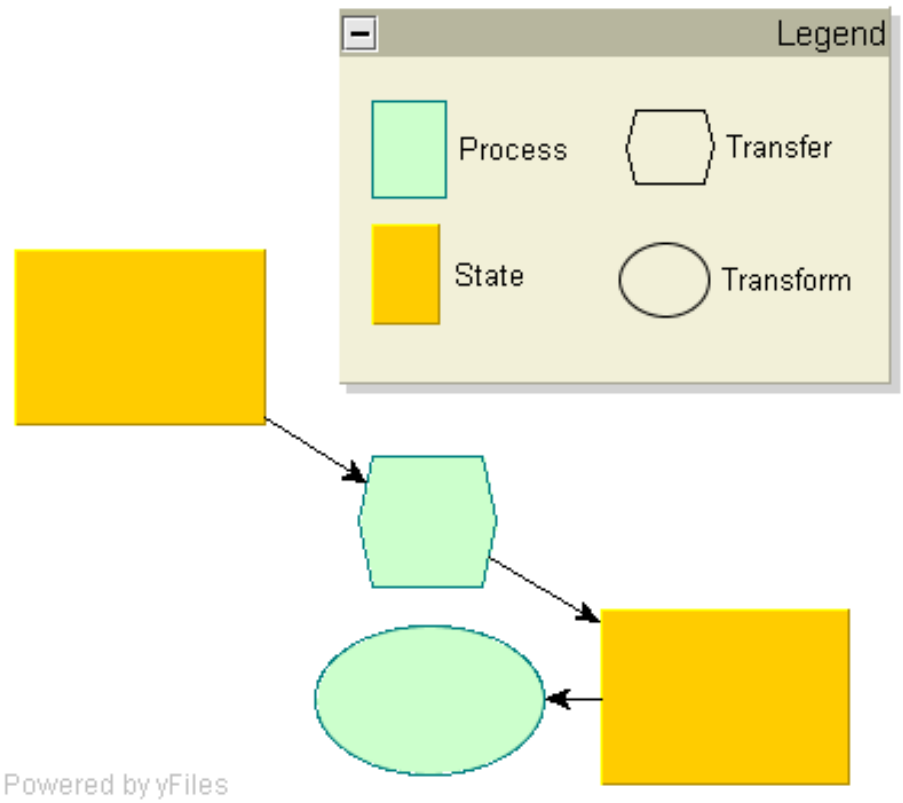

Figure 2.2: Basis of the transfer-transformation lattice. The rectangle represent the state or the object; the hexagon represent the object transfer to a new address; the ellipse represents the transformation of the object/state into a new different object/state. For example, this diagram could represent an *xml* document's transfer between computers, followed by its transformation by an *xsl* stylesheet into an *xhtml* document.

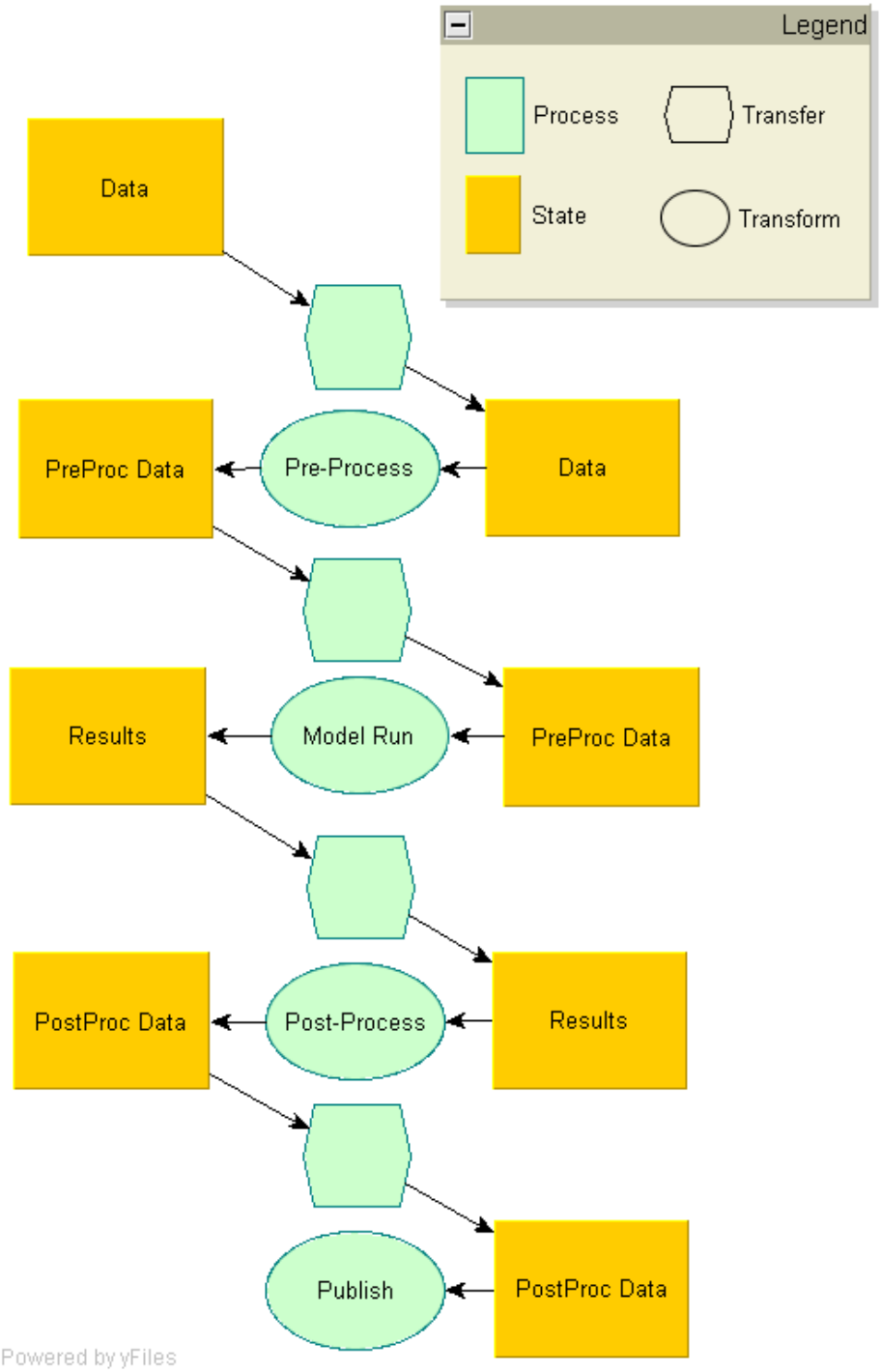

Figure 2.3: Lattice of transfers and transformations between the main states of an operational system. The lattice's base is described in figure 2.2. The baseline data (*Data*) is transferred and transformed into pre-processed data. The latter is transferred to the model's input address. The model runs and returns the output data (i.e. the model's results). The latter is transferred and transformed into post-processed data, which will eventually be transferred and published.

"slower" scenarios. Hence, the final system is under-optimized, in order to cope safely with processes delays.

Nevertheless, in spite of its downsides, the synchronous approach was the chosen method to implement the pre-operational system due to its convenience (it's far easier to configure *scheduler* in windows or *crontab* in linux, rather than programming and testing a *service* or a *daemon*, respectively). However, an effort was made during implementation so that it would rather be straightforward to adopt the asynchronous approach if need-be.

2.1.3 Logging and Error recovery

An operational system must keep track of all its action. It must be able to record its own history, for future audit purposes. Thus, all the procedures must keep some sort of time-stamped log in several files, (one per procedure for example).

An operational system must have checkpoints at regular intervals between processes. A checkpoint checks whether all objects are ready for the coming procedures and logs its verdict. If a checkpoint fails, then it must also inform the system administrator with an email (or sms) containing the checkpoint name. If a checkpoint fails then whether it stops the system and waits for manual input, whether it tries to recover from error (ex: if an ftp connection fails, the checkpoint will detect some files didn't copied successfully. Thus it can loop over the ftp connection procedure until all files are successfully transferred.)

When and where to place the checkpoints is a programmer's choice. A possible guideline would be of placing a checkpoint at the end of each transfer or transformation (figure 2.2), but this might mean overworking the programmer and the operational system. So a good deal of judgement and experience is required to define which points of the operational system are critical and error-prone.

As a general guideline based on our experience, the most common errors occur at:

- ftp and http file transfers. These are the most error prone procedures. However, these are easy errors to recover.
- Network file transfers. These are the second most error prone procedures. These are easier to recover. Both errors will delay the normal duration of the procedures.
- Programmer's bug. An operational system is complex and any change will influence all the downstream operations. It is frequent a change

made at a certain point will introduce an error which will express itself in another procedure downstream. These are generally easy to recover but require manual assistance. The delay is undeterminable.

- A model error. These are the hardest error to recover and usually require manual assistance. The delay is undeterminable.

2.2 Technology

Thus, a method for outlining an operation system was defined and chosen (the transfer-transform method), the need for a supervising entity was defined, the entity was chosen (between a clock and a background service - clock taken), and the need to keep historical records of every step of the operational system was stressed. Hence, herein comes the time to start defining which hardware and tools would best fit to implement the pre-operational model. During this selection we followed a compromise of these criteria:

- The best tool for the job, ...
- ... depends on the programmer's background and experience.

2.2.1 Hardware: Intel, Mac, AMD ?

This section is easy: usually you don't get to choose the hardware, but rather are obliged to make the best use of the available computing resources. Dealing with different kinds of hardware, will make one deal with different operating systems, and different development platforms. However, in this case, we had the opportunity to hand-pick and choose a computer resource exclusively dedicated to run the pre-operational system's model. We opted for an AMD *Opteron* 270 with 4 GB of ram DDR, coupled to a ten-fold expandable RAID board containing a 500 GB RAID 1 hard disks and a 250 GB RAID 0 operating system disk. It turned out that the intel Fortran compiler (with which we compile the MOHID water modelling system) doesn't takes much advantage of the AMD 64 bits architecture. If we could choose again, we would have gone with the intel equivalent computer. However, as the pre-operational system is a distributed application, all of the typical computer systems could be a part of the system. It's important to note however, that sticking with one type of hardware (and especially of OS) is of great convenience, although this wasn't the case in our implementation.

Furthermore, the network system which unites all the computing resources is of crucial importance, and its direct control is sometimes necessary. If you don't control the network, then probably some aspects of the pre-operational system will fail to fully work. For example, if an administrator would over-restrict the free use of the network, some important tools required by transferring procedures may be unusable (such as *ftp*, *sftp*, *scp*, *wget*, *ssh*, *smtp*, *httpd* ...). In our case, the most problematic aspect is to use ports other than the typical port 80 and port 22. Another problem is to install and configure web-pages server and other servlets, as these may only be installed in the *DMZ* (Demilitarized zone) and we don't have access to the *DMZ* configuration. We don't have access to our firewall configuration as well. All these restrictions don't ease our task of operationalizing a system. Thus, in conclusion, full access to the network administration is strongly recommended when installing an operational system. Employing a full-time network administrator, and put him in-charge of network administration tasks of the operational system is even better.

Finally, a large-bandwidth is highly recommended to accelerate all the transfer procedures between computers. Equipping the whole network with a 1 Gb LAN would increase substantially the pre-operational system overall performance.

2.2.2 Operating system: Linux, Windows?

It's simpler to have the same OS in all the network machines and resources. However, sometimes all the required tools just can't be found in the same OS. Depending on the team staff's technical expertise and background, flexibility is recommended and sacrificing the single-OS rule in favor of the best tool for the job (cross-platform) may be an asset.

In this case, we ended using common windows XP machines whenever possible (windows XP is the dominant OS in the Maretec-Hidromod culture) and installed ad-hoc linux Fedora Core OSes when a specific tool was required (*LAS*, *opendap*, running MOHID in 64 bits)

This broad-spectrum of OSes and of file systems requires some extra care when configuring the computers in the network. In our particular case, the *samba* was required to be configured in the linux-based computers so the files transfers would be seamless. Otherwise, file transfers inside the network would use the standard *UNC* file address nomenclature.

2.2.3 Development platform: Open-source vs .NET ?

The job defines which tools are best suited, and consequently, which development platform is best suited. However, the company's culture also is an important aspect to cope with. Not using the company's cultural development framework will bring a negative feedback from colleagues at different kinds of levels. In this case, the company's culture is .NET oriented, but all the available operational systems' culture is open-source and linux oriented (netcdf, opendap, LAS etc...). This was a problem, as a choice had to be made between the company's culture and the operational modelling community's culture and tools. Eventually the choice went for the latter, but at the expense of negative feedback from company's members or complete absence of it.

No true development platform was chosen. Although this may slowdown productivity at first, it allows more flexibility which is badly required to cope with the different OSes and environments. However, not using a specific development platform doesn't mean development's best practices can't be used. Here's a list of the recommended best practices to follow:

1. Use a code/document versioning system (such as cvs, subversion or other).
2. Use a bug-tracking tool (such as bug-report or bugtrac).
3. Use a code comment methodology so that code is easily readable by others and self-generated documentation is possible (such as javadoc). We propose one which is explained in detail on our wiki[1].
4. Use code semantics best-practices.

2.2.4 Program languages: scripts and shell commands

Given the nature of the procedures to develop, scripting languages and shell commands are the most natural tools to handle transferring of files and running programs that require generated input. The best tools for the job are then:

1. scripting languages allowing regular expressions, ...
2. ... whom the programmer is most familiar with.

[1] http://www.mohid.com/wiki/index.php?title=Code_Comment_Method (username:wikiguest, password:wikiguest)

Of all the scripting languages available, perl combined with local shell scripting was chosen. Because it provides flexibility ("There's More Than One Way To Do It" is its moto), expressiveness (perl's the founding language of regular expressions, also known as regexps), extensibility (over 3000 modules doing virtually anything written by the open-source community) and singularity (perl one-liners[2] allow very powerful command-line actions) perl is still widely used as of today. Its probable bigger downside is in its excessive flexibility: perl programs are difficult to grow in size and complexity as they are pretty much chaotic. However, this isn't a problem, in this case, since we don't need large single programs, but rather many little and efficient perl scripts, preferably one-liners whenever possible, which can be easily reusable and that are designed to be piped.

So here's the philosophy of use of the chosen tools:

1. Find string patterns whenever and wherever you can.
2. Use regexps abundantly.
3. Write short scripts, write many.
4. Pipe whenever and wherever you can.
5. Use perl one-liners whenever and wherever you can.
6. When performing similar actions make sure you use the same ensemble of scripts. In other words, reuse scripts abundantly.
7. Use time-stamped logging abundantly.
8. Keep track of changes using a versioning repository and a code commenting method.

In the end, you'll get a bunch of powerful scripts and customized commands you wished every OS possessed. These will prove most useful to perform file transfers and routine administrative tasks, and will make you understand why perl is also known as the "glue" language. On the other hand, perl scripts can easily generate configuration files for other tools (such as Mohid tools); thus making it eligible to handle transformation procedures as well.

[2] http://www.mohid.com/wiki/index.php?title=Perl (login:wikiguest, password:wikiguest)

2.3 Implementation

2.3.1 The outline

The input data process, as shown in the *MOHID data process* of figure 2.1, corresponds to external models data manipulation (data manipulation of large scale hydrodynamical circulation forcing, atmospherical sea surface forcing and tidal forcing). The manipulations can be discriminated in the following groups:

- Automated external models data acquisition,
- Automated conversion of external models data into data manipulable by MOHID,
- Other data manipulations.

In a general manner, data acquisition is programmed in a scripting language such as *perl* or *python*, data conversion is made by in-house existing software (*ConvertToHDF5*) and processes automation is a combination of OS scripting languages (*bash* or *batch file*) with OS' administrative management scheduling tools (*crontab* or *scheduler*).

Applying the MOHID model off western Iberia shelf process is illustrated in figure 2.1 by the processes *running analyzed* and *running prediction*. The *running analyzed* process is a hindcast mode process while the *running prediction* is a forecast mode process. Then the results are converted to netcdf with the *convert2netcdf* homemade *fortran95* tool.

The publishing process of the results, illustrated in figure 2.1, is generated using several tools:

- An *OpenDAP* server mounted on *Apache* with a *netcdf* handler, serves data over *http*, and allows efficient subset data extraction of results thanks to its *querying* system.
- A *LAS* server mounted on *Tomcat*, which allows data *browsing* and data graphical representation with scientific quality thanks to *Ferret* software.
- A centralized internet webpage which stands as a framework for the previous services (based on *xml*, *xslt* and *CMS* technology, with a possibility of exploiting *jsp* or *AJAX* technologies).

- For an efficient automated graphical representation of the results, a combined form of command line tools such as *Ferret* or *GMT*, with scripting languages such as *perl*, *python*, *php* or *cgi* is proposed to be used.

The comparison process, illustrated in figure 2.1, is the least developed process at this time. The idea would be to compare hindcasted results of the current week with forecasted results of the previous week. Of course, the rich experience of operational models' teams such as Mercator-Océan and other groups is an important aspect that is not to be overlooked. Furthermore, comparison with in-situ measurements and remote-sensing imagery is considered.

2.3.2 The blueprints

We implemented the plan defined in figure 2.1. First we made a draft tree of the main branches and procedures in figure 2.4. Basically, three branches are outlined and joined at a single node:

1. the Mercator data processing branch,
2. the MM5 data processing branch,
3. the Mohid running and publishing branch.

As the operational system is a distributed application, each procedure may be allocated to a different address on the network (or not) providing the engineer flexibility at managing its available resources. Up to seven different addresses in the network can be used. They could all be stacked at the same address, or completely different ones, depending on what's more convenient. Notice that both the Mercator and MM5 branch split into *Mercator_weekly* and *MM5_weekly procedures*, which consists in getting the data and store it; and into *Interpol_weekly* and *InterpolMM5_weekly* which consists in interpolating the data to the Mohid grids. Finally, the Mohid branch is separated into the *Run_weekly*, the *Post_run* and the *Opendap/LAS*. The *Run_weekly* consists in running the model, the *Post_run* consists in post-processing the data (such as converting the hdf5 files into netcdf format), and the *Opendap/LAS* consists in publishing the data with opendap.

The next step was to redraft each branch of figure 2.4 according to the transfer-transform method in section 2.1.1, but adding more detail to the object, which now contains three fields: the address or locus, the file extension or encoding type and its nomenclature pattern. Hence the resulting

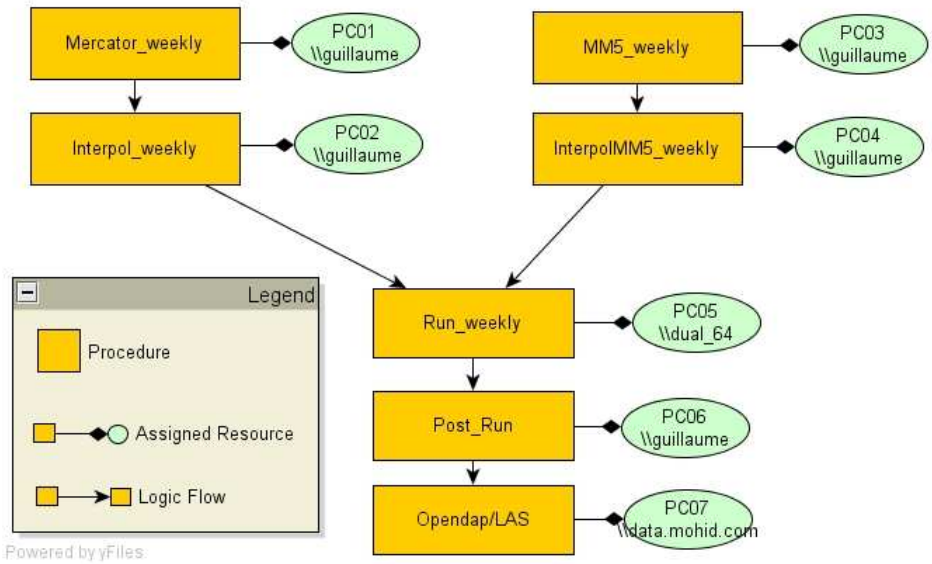

Figure 2.4: Implementing the operational plan. Operational system's procedures logical flow and assigned network resource. See legend in diagram.

Mercator branch is depicted in figure 2.5 and the MM5 and Mohid branches are in figures 2.6 and 2.7 respectively. Often, there's only need to transfer the files into a storage area where no transformation is necessary. In these situation the identity transformation is used, and is noted *Id* in the figures.

Once all the branches are sketched up (figures 2.5,2.6,2.7), one may proceed to program the branches. Note that at this stage, the engineer may try to define where to put error checks (or checkpoints), or at least to try to define a policy of how to check for errors.

2.3.3 The detailed blueprints

Hence, in a pre-programming stage, the blueprints were designed (figures 2.5,2.6,2.7) based on the main system diagram in figure 2.4, and, in a post-programming stage, we reverse-engineered the procedures and tried to present them in diagrams similar to Control Flow Graphs (CFG). Hence figure 2.8 is *Mercator_weekly*'s CFG figure 2.9 is *Interpol_weekly*'s CFG, figure 2.10 is *MM5_weekly*'s CFG, figure 2.11 is *InterpolMM5_weekly*'s CFG, figure 2.12 is *Run_weekly*'s CFG, figure 2.13 is *Post_run*'s CFG and figure 2.14 is *opendap/las*'s CFG.

It is important to note in the presented CFGs that orange rectangles are the main states, light blue triangles are the checkpoints and error checkings, and the violet rectangles are input and output files. It is also important to note that no layer was designed that illustrates the log files and their interactions with the CFGs states. Showing the log files in the CFGs would probably be a good idea in future reverse-engineering of CFGs.

2.4 Maintenance and development

2.4.1 The developer framework

As the pre-operational system is implemented and working, the developer (or another appropriately assigned human resource) is required to maintain the system. Basically the pre-operational system outputs besides data files and result files, are its historical logs and its checkpoints emails. The maintenance human resource is basically required to schedule appropriately the main procedures depicted in figure 2.4 check the emails and have a manual intervention if any error occurred.

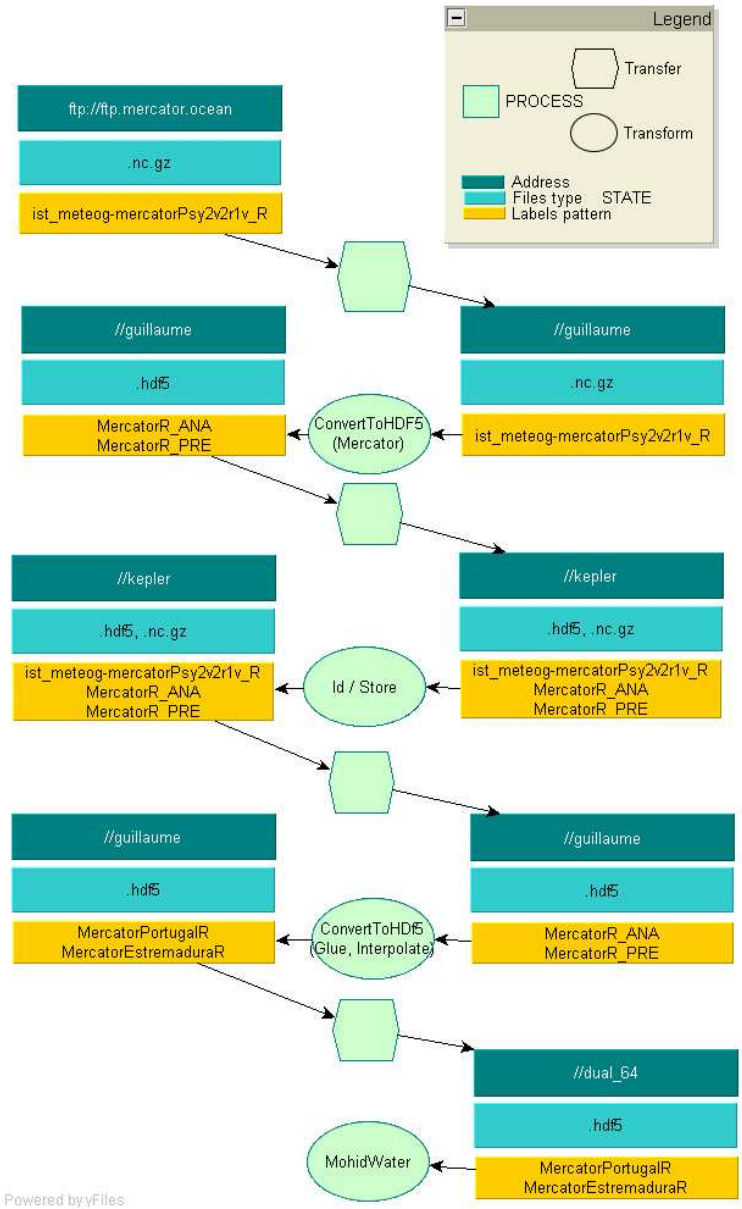

Figure 2.5: Lattice of transfers and transformations operated on the Mercator solution data. The base of the lattice is described in figure 2.2. Each object is composed of three fields: a locus (or a physical address), a file type (ex: hdf5, nc, etc...) and a nomenclature pattern (ex: all netcdf files from Mercator solution contain the following string in their name: ist_meteog-mercatorPsy2v2r1).

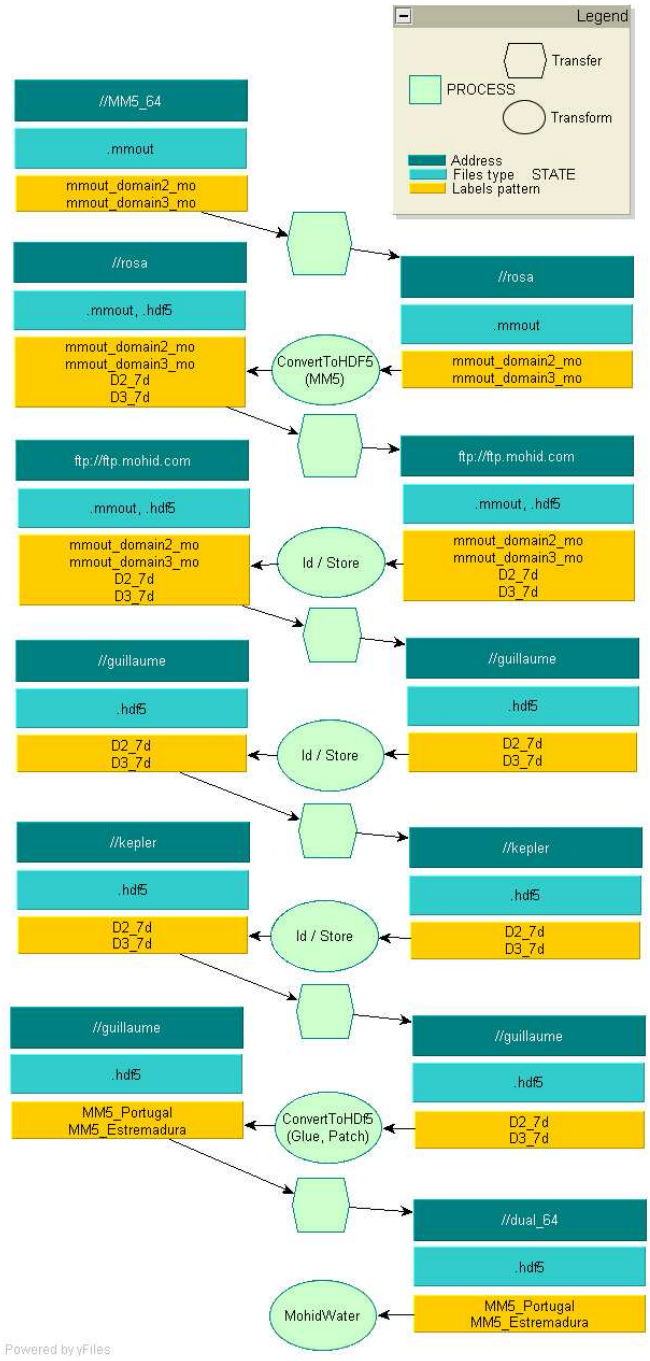

Figure 2.6: Lattice of transfers and transformations operated on the MM5 atmospheric solution. The base of the lattice is described in figure 2.2. Each object is composed of three fields: a locus (or a physical address), a file type (ex: hdf5, nc, etc...) and a nomenclature pattern (ex: all netcdf files from Mercator solution contain the following string in their name: ist_meteog-mercatorPsy2v2r1).

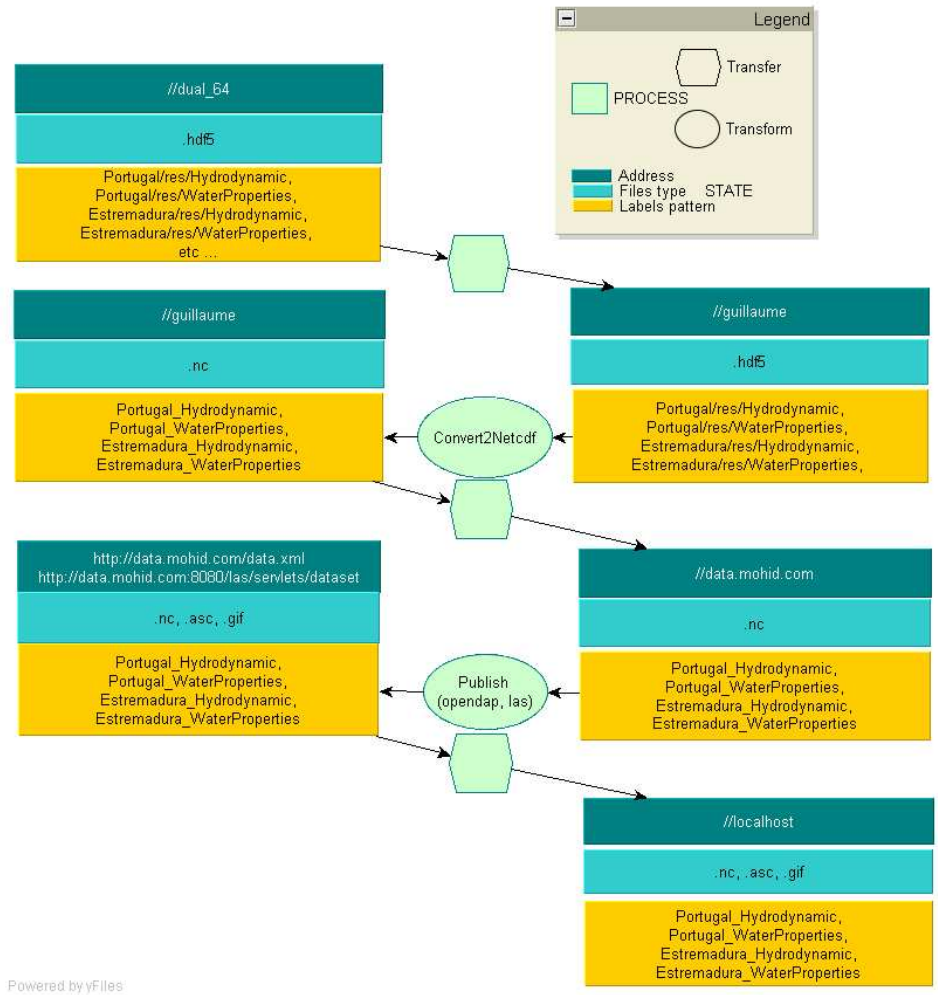

Figure 2.7: Lattice of transfers and transformations operated on the Mohid results. The base of the lattice is described in figure 2.2. Each object is composed of three fields: a locus (or a physical address), a file type (ex: hdf5, nc, etc...) and a nomenclature pattern (ex: all netcdf files from Mercator solution contain the following string in their name: ist_meteog-mercatorPsy2v2r1).

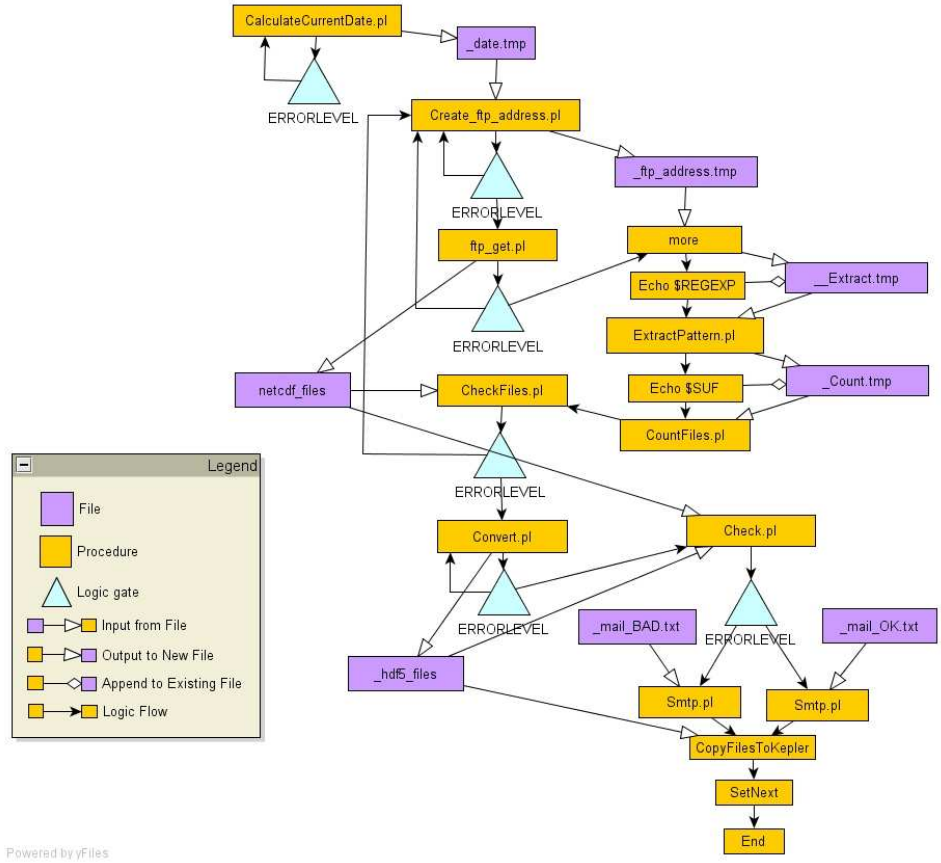

Figure 2.8: Control Flow Graph (CFG) of *Interpol_weekly* procedure. Orange boxes are procedures, routines, subroutines or programs; purple boxes are files; light blue triangles are errorlevels or checkpoints; black arrows are logic flow between procedures and errorlevels; white headed arrows mean output to files or input from files; white losangle shaped arrows mean appending output to file.

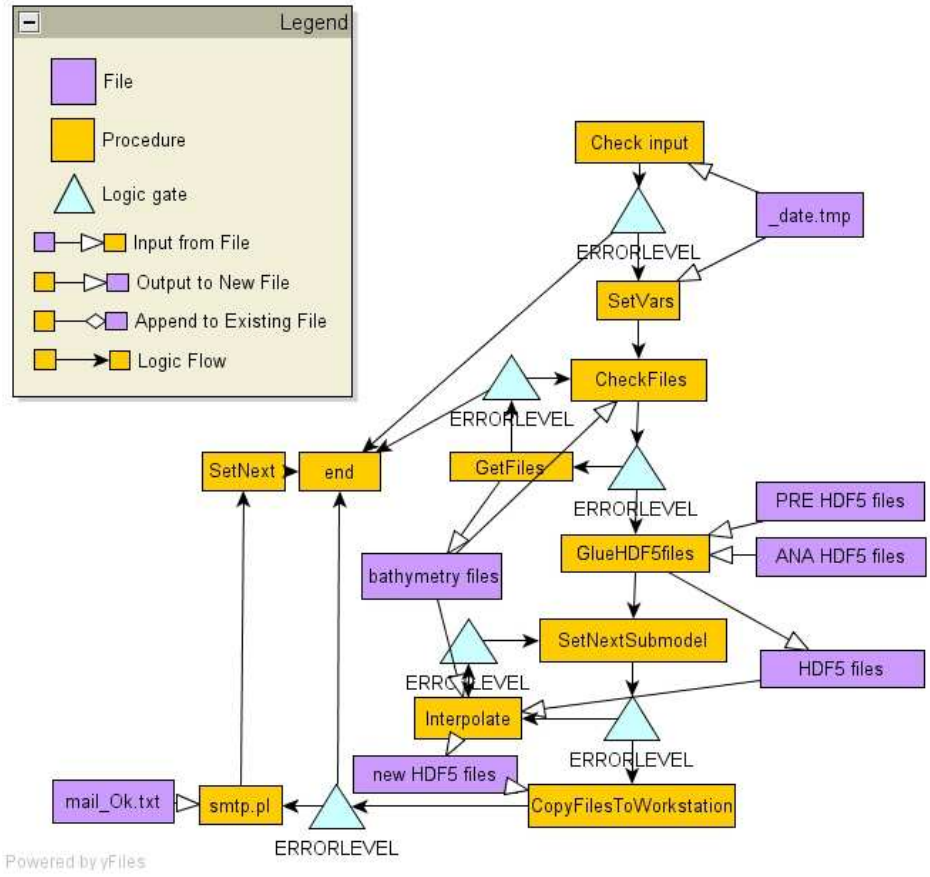

Figure 2.9: CFG of *Interpol_weekly* procedure. Orange boxes are procedures, routines, subroutines or programs; purple boxes are files; light blue triangles are errorlevels or checkpoints; black arrows are logic flow between procedures and errorlevels; white headed arrows mean output to files or input from files; white losangle shaped arrows mean appending output to file.

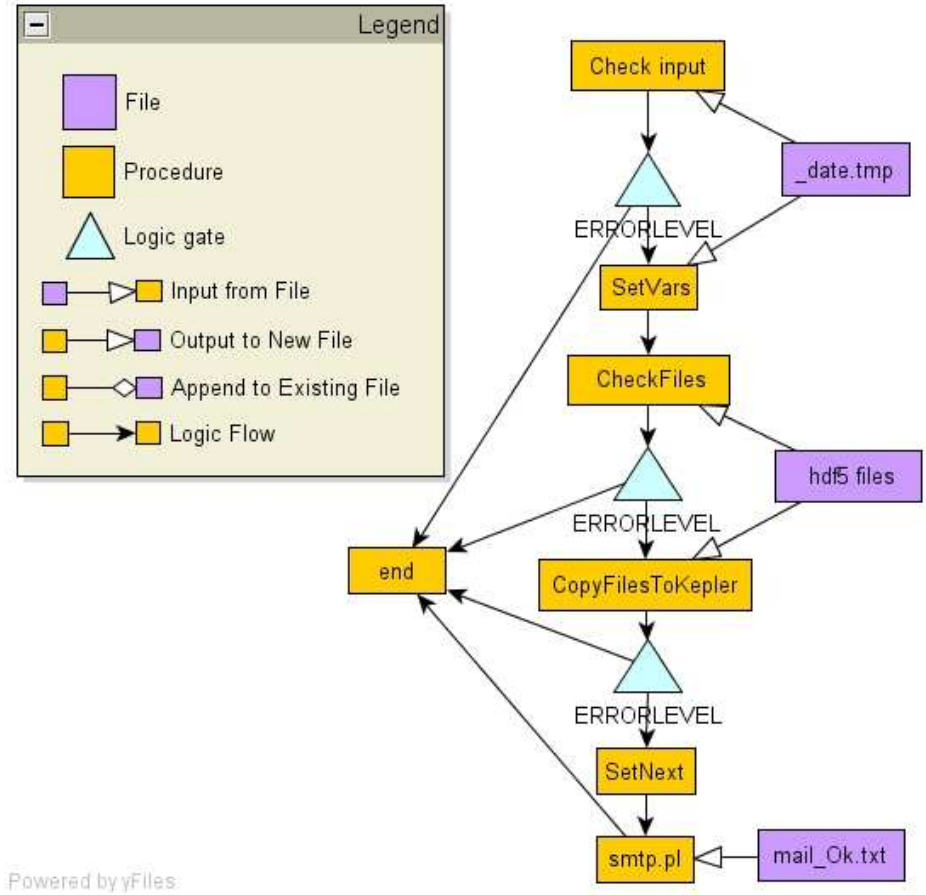

Figure 2.10: CFG of *MM5_weekly* procedure. Orange boxes are procedures, routines, subroutines or programs; purple boxes are files; light blue triangles are errorlevels or checkpoints; black arrows are logic flow between procedures and errorlevels; white headed arrows mean output to files or input from files; white losangle shaped arrows mean appending output to file.

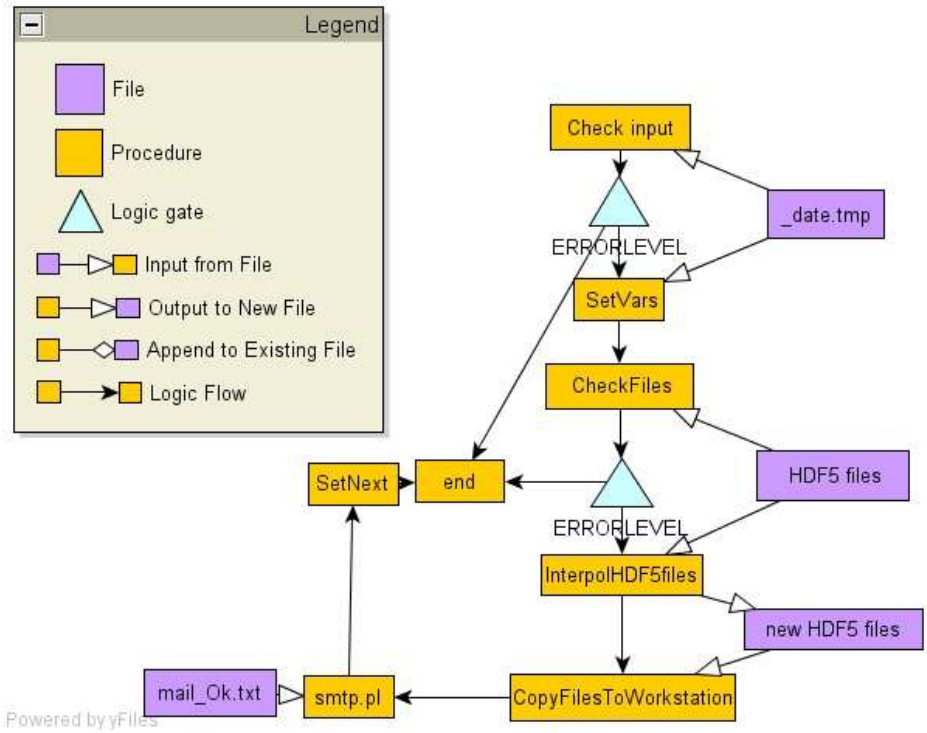

Figure 2.11: CFG of *InterpolMM5_weekly* procedure. Orange boxes are procedures, routines, subroutines or programs; purple boxes are files; light blue triangles are errorlevels or checkpoints; black arrows are logic flow between procedures and errorlevels; white headed arrows mean output to files or input from files; white losangle shaped arrows mean appending output to file.

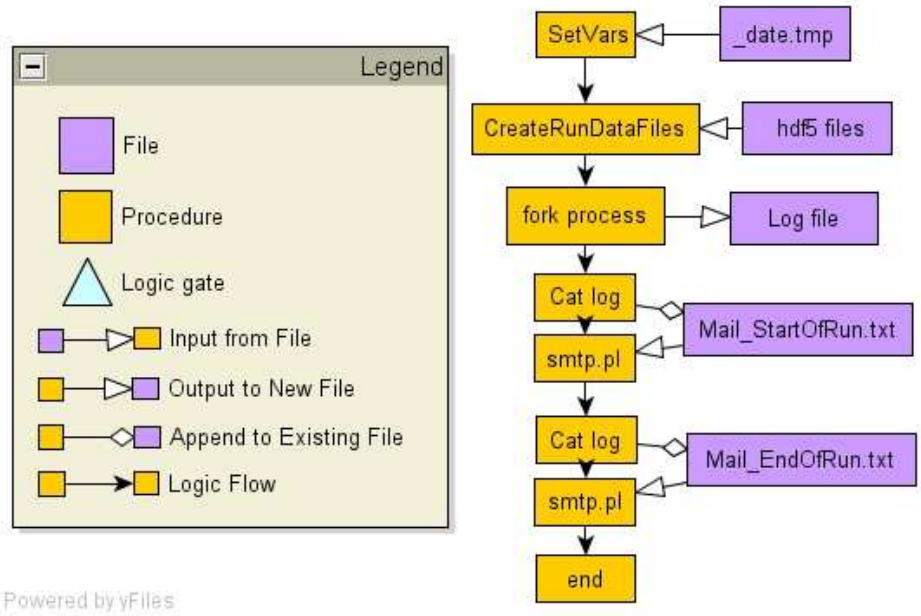

Figure 2.12: CFG of *Run_weekly* procedure. Orange boxes are procedures, routines, subroutines or programs; purple boxes are files; light blue triangles are errorlevels or checkpoints; black arrows are logic flow between procedures and errorlevels; white headed arrows mean output to files or input from files; white losangle shaped arrows mean appending output to file.

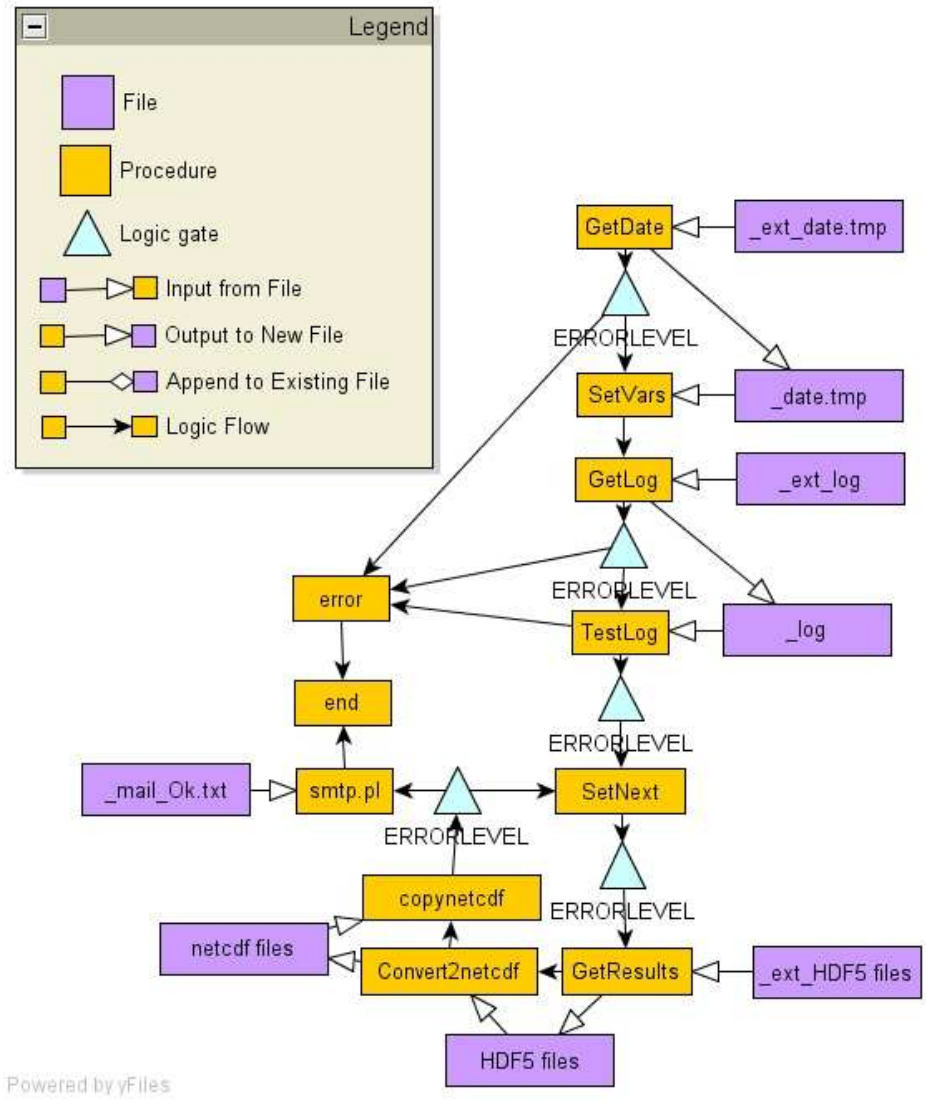

Figure 2.13: CFG of *Post_run* procedure. Orange boxes are procedures, routines, subroutines or programs; purple boxes are files; light blue triangles are errorlevels or checkpoints; black arrows are logic flow between procedures and errorlevels; white headed arrows mean output to files or input from files; white losangle shaped arrows mean appending output to file.

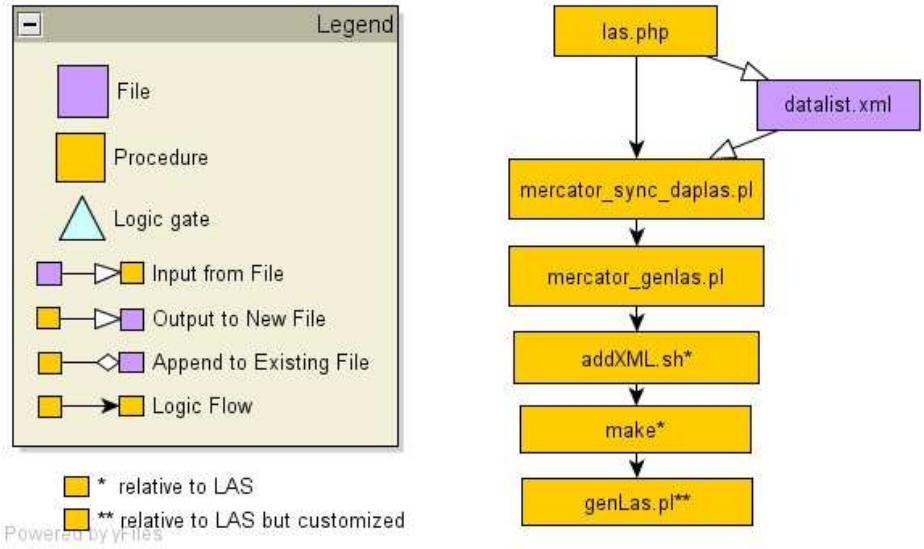

Figure 2.14: CFG of *opendap/LAS* procedure. Orange boxes are procedures, routines, subroutines or programs; purple boxes are files; light blue triangles are errorlevels or checkpoints; black arrows are logic flow between procedures and errorlevels; white headed arrows mean output to files or input from files; white losangle shaped arrows mean appending output to file.

2.4.2 The user interfaces

The implemented user interfaces are a web-page[3] built using xml/xsl technology (which is quite an advanced and experimental technology when it comes in terms of implemented web-sites) and a LAS (Live Access Server)[4]. The web-page provides a complete and automatically updated list of dataset files (the model's results) served through an opendap[5][17] server. The most used client application for quick graphical inspection of remote datasets is NcBrowse[6]. The LAS server provides a graphical user web-interface to extract data from available datasets. However, because the LAS interface is built on JSP (java server pages) and tomcat servers, it's a bit unstable and clients can and will bring down the tomcat server. This is a problem due to our overprotected network and due to the renown instability of tomcat servers and java servlets.

Figure 2.15 shows a quick snapshot of the web page datasets listing and figure 2.16 provides a quick snapshot of the Mohid LAS interface.

Thus far, although the web-page is fully operational, in the sense that netcdf datasets containing the full Mohid hydrodynamical results are readily available to download and inspect, the web-page is yet unattractive and doesn't allows quickviews or thumbnails plotted results. Some effort must be put on the web-page regarding automated graphical results generation so that it turns into a full product for the average internet user. The potential end-users may be broader than the scientific community. The scope of this outreach would beneficial for this scientific field.

The technical challenge for this lies essentially in creating automated graphical results. For instance the Mohid tools require manual users and a graphical server active. The tool that would pull the trick would have to be accessible from scripts from the command line. Tools like GMT and Ferret are possibly the best choice.

Also, as a final note, it is important to state that opendap and LAS require a linux environment.

2.5 Future work

Basically this pre-operational system is already implemented and functional. However several aspects still require to be improved or developed prior to

[3] http://data.mohid.com/index.xml
[4] http://data.mohid.com:8080/las/servlets/dataset
[5] http://www.opendap.com
[6] http://www.epic.noaa.gov/java/ncBrowse/

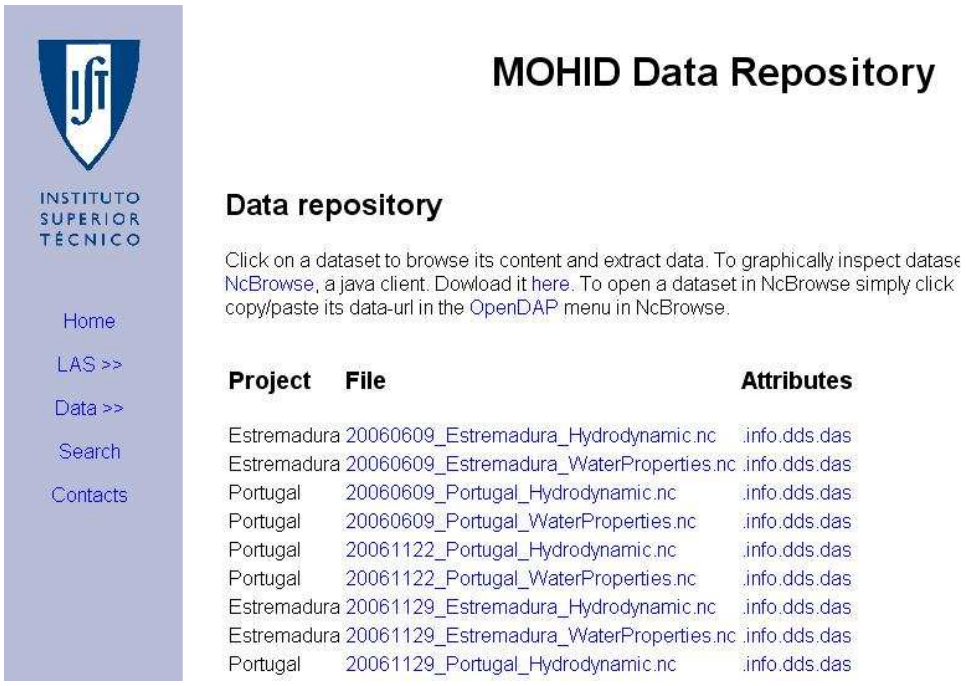

Figure 2.15: Pre-operational system web page snapshot.

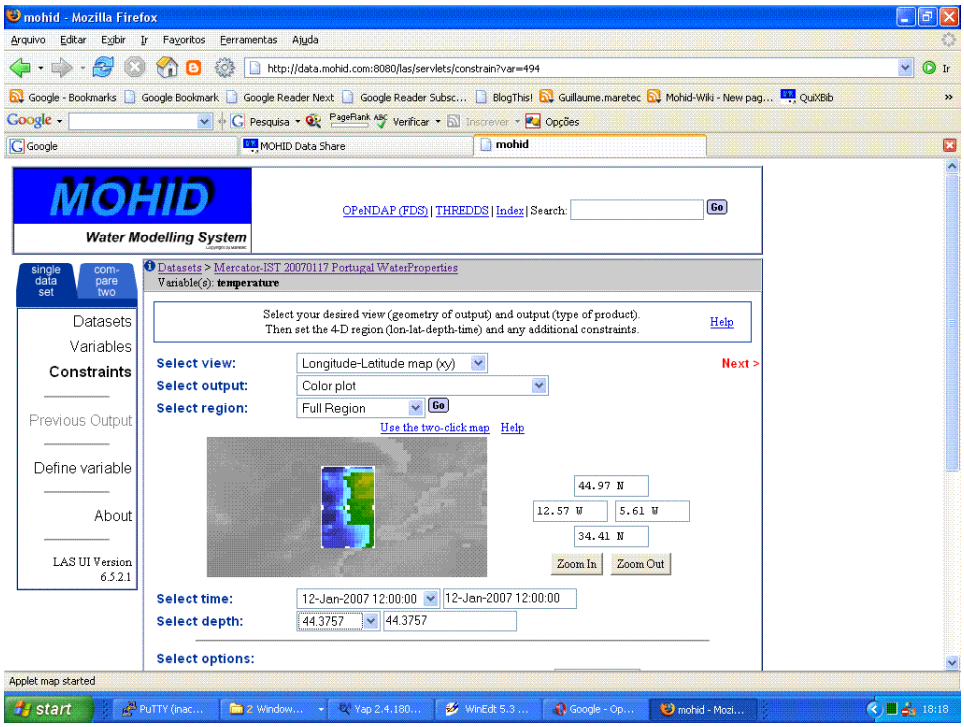

Figure 2.16: MOHID LAS (Live Access Server) snapshot.

call this system an operational system:

1. The web page needs to be more appealing.

2. Automatic graphic results must generated in the web page as thumbnails and quickviews.

3. Automatic generation of SST satellite imagery must be implemented and posted at the web-page, in order to compare with models results.

Several ways are possible for these issue. First, stepping down from the xml/xsl technology web page and going for a mainstream Content Management System (CMS) such as Joomla[7] seems an interesting and low cost solution. Second, exploring Ferret software (used by LAS to generate graphics) and develop automated scripts to generate graphics is the most probable way to go. And third, a lot of investigation needs to be done in order to obtain these satellite imageries automatically.

On the other hand, there's the parallel processing MPI limitations. Indeed, the operational system original consisted in three high-resolution models (2 km). But problems with the MPI implementation only allows one nested high-res model, thus far. Work is undertaken to overcome this issue.

[7] http://www.joomla.org/

Chapter 3

Modelling the Portuguese coastal hydrodynamical circulation

The MOHID hydrodynamical numerical model solves the Navier-Stokes equations of a rotating fluid in a β plane. The geophysical fluid is constrained to the hydrostatic and the Boussinesq approximations, as a practical result of a dimensional analysis[9]. The numerical solver uses a finite-volumes approach[33] similar to the one described by Chu[43]. MOHID solves also a seawater density non-linear state equation depending on pressure, salinity and potential temperature[35]. Finally, to calculate the turbulent vertical mixing, MOHID embeds GOTM[42][10], a $k - \epsilon$ model numerical solver. The parametrization proposed by Canuto[11] is used; as well as the wind wave induced mixing parametrization proposed by Craig[14]. The horizontal discretization is an Arakawa C grid[3]. The vertical coordinate is hybrid and generic, allowing to choose between z-level, sigma and lagrangian coordinates. The baroclinic pressure gradient term is always calculated using a z-level approach, with a linear interpolation, in order to minimize spurious pressure-gradient induced vertical velocity errors[5] [39][27]. In this application, the 2D model uses a sigma coordinate, and the tridimensional models use a lagrangian vertical coordinate with shaved-cells at the bottom[1]. The temporal numerical scheme is an alternate direction semi-implicit (ADI) method[28]. The spatial discretization numerical scheme is a total variation diminishing (TVD) scheme[21].

The modelled domains description follows in the next section. Tables 3.1 and 3.2 summarize the initial conditions and the boundary conditions.

Boundary conditions	
Surface	**modèles**
Wind stress forcing from MM5 winds through equation 3.1.	P, C
Interpolated heat fluxes from MM5 data.	*P* et *C*
Open Boundary Conditions	**models**
FRS[32] of the *M-O* solution for U, V, T and S.	*P* et *C*
Interpolation of η, U, V, T et S from *M-O*.	*P* et *C*
Barotropic mode *Blumberg* radiation[7].	*WI*
Barotropic mode *Flather* radiation[20].	*P* et *C*
Sponge layer.	*P* et *C*
Land boundary	**models**
Freshwater discharges.	*P* et *C*
Null fluxes of (U,V).	*WI*, *P* et *C*
Bottom boundary	**models**
Bottom stress forcing according to equation 3.7.	*WI*, *P* et *C*

Table 3.1: Nested models boundary conditions. The abbreviations definitions are: Zonal and meridional velocity components (U, V), potential temperature (T) and salinity (S), water level relative to a reference level (η), flow relaxation scheme (FRS), Mercator-Océan solution(M-O), western Iberia barotropic model (WI), portuguese Iberian coastal model(P), Estremadure promontory model (C).

Initial conditions	
U, V, S and T are interpolated from the *M-O* solution.	*P* et *C*
η is initialized to a reference level.	*WI*, *P* et *C*
Assimilation	
U, V, T and S *FRS* according to equation 3.5.	*P* et *C*
Spin up	
Baroclinic force and wind forcing ignition over 10 inertial periods.	*P*
FRS and *Flather* radiation ignition over 10 inertial periods.	*P*

Table 3.2: Nested models' initial conditions, assimilations and spin-up. The abbreviations definitions are: Zonal and meridional velocity components (U, V), potential temperature (T) and salinity (S), water level relative to a reference level (η), flow relaxation scheme (*FRS*), modèles de Mercator-Océan (*M-O*), western Iberia barotropic model (*WI*), portuguese Iberian coastal model(*P*), Estremadure promontory model (*C*).

3.1 Atmospheric forcing

The model is coupled with prof. J. J. Delgado Domingo's MM5 [23] atmospheric model from IST in *offline* mode. The three-level nested atmospheric model is forced with the Global Forecasting System (GFS)7 day forecast over the region bounded by 20^oO, 28^oN and 5^oO, 50^oN whose relief is exposed in figure 3.1. The nested models resolution are 81, 27 and 9 km and are composed of 25 vertical layers. It simulates winds, sensible heat, latent heat, solar radiation, precipitation, evaporation, specific humidity and cloud cover.

Wind forcing is calculated according to[37]

$$\tau_w^u = \rho_a C_a u_{10} \sqrt{u_{10}^2 + v_{10}^2} \tag{3.1}$$

where τ_w^u is the surface stress induced by wind, $\rho_a = 1.25$ kg/m^3 is air density, C_a is a drag coefficient whose range is described in Leitao2003[30], finally u_{10} and V_{10} are the horizontal components of air speed at 10 m of height above the sea surface.
The sensible heat, latent heat, solar radiation, sea surface temperature, precipitation and evaporation fluxes are also calculated.

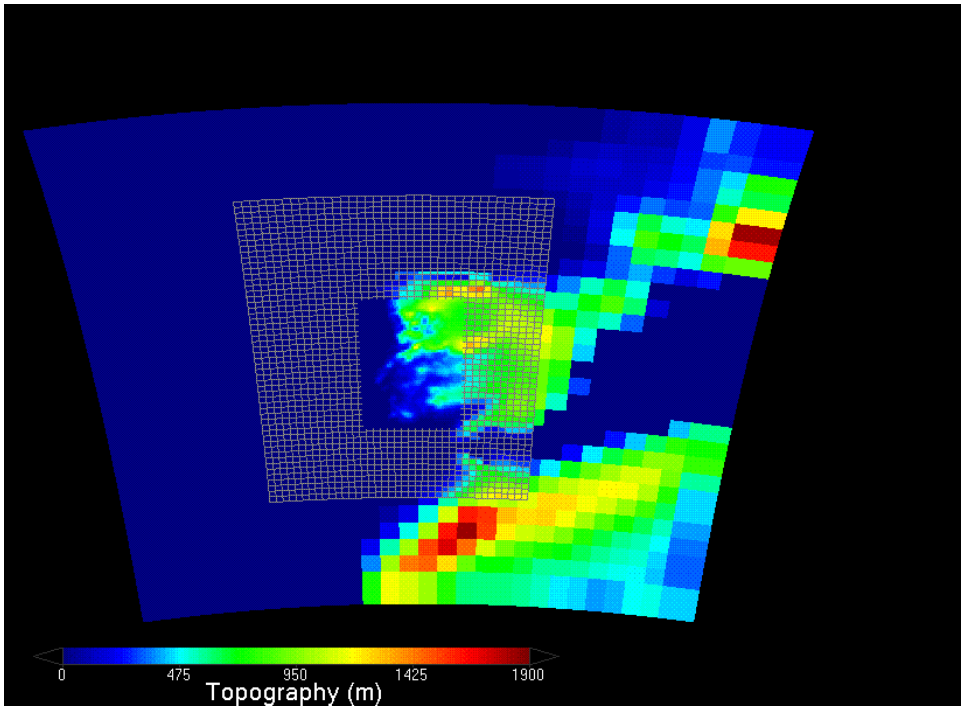

Figure 3.1: Prof. J. J. Delgado Domingos group's MM5 solution (labeled herein *MM5-IST*) nested bathymetries with 81, 27, and 9 km resolutions.

3.2 Oceanic forcing

3.2.1 The Mercator reference solution

In order to obtain coherent open boundary conditions (OBC), a good reference solution is mandatory[6]. The high resolution solution of the Northern Atlantic and the Mediterranean basin provided by Mercator-Océan, **PSY2v2r1**, is likely to be the most reliable solution available[18][4] that reproduces realistically the Northern Atlantic circulation and in particular the western Iberian coastal circulation and the Gulf of Cadiz circulation. While assimilating *in-situ* data, remote sensed sea level anomaly (SLA) and sea surface temperature (SST), as well as atmospheric forecasts fed by the European Centre for Medium Range Weather Forecast (ECMWF), the Mercator solution reproduces accurately the main characteristics of the circulation off western Iberia peninsula; namely, the Mediterranean Outflow (MO), several downstream Mediterranean veins[2][8][24] and also the formation of meddies near Cape St.Vincent and over the Estremadure bank[36][8]. However, the number of meddies formed by the model at St.Vincent Cape is inferior relatively to the observations. This is probably due to the z-level vertical coordinate choice, according to Drillet[18]. Indeed, such a choice of coordinates would underestimate the dense water sinking downstream of the Gibraltar strait because of the intense nature of the MO near the Gibraltar strait. This problem is coped with a stronger relaxation towards Reynaud's[38] climatology downstream of the Gibraltar strait[18]. Unfortunately, the bias of the results in Reynaud's climatology are propagated and an inferior temperature and salinity of about $0.75^{o}C$ and $0.15\ psu$, respectively, is obtained[18] when compared with known measurements[25]. This said, Mercator's solution is a good reference solution, most likely the best, capable of forcing a model operationally. The figure 3.2 illustrates the bathymetry of the Mercator solution extraction, whose domain ranges approximately from $24.5^{o}O$, $28^{o}N$ to $4^{o}O$, $51^{o}N$. The Mercator solution is interpolated over the MOHID meshes in two steps:

1. An interpolation using a triangulation method for each bidimensional layer.

2. A linear interpolation of each vertical column from the 43 layered Mercator columns to the 42 layered MOHID columns. The layer reduction was designed to improve the overall time performance of the models.

The Mercator solution is labeled herein *M-O*.

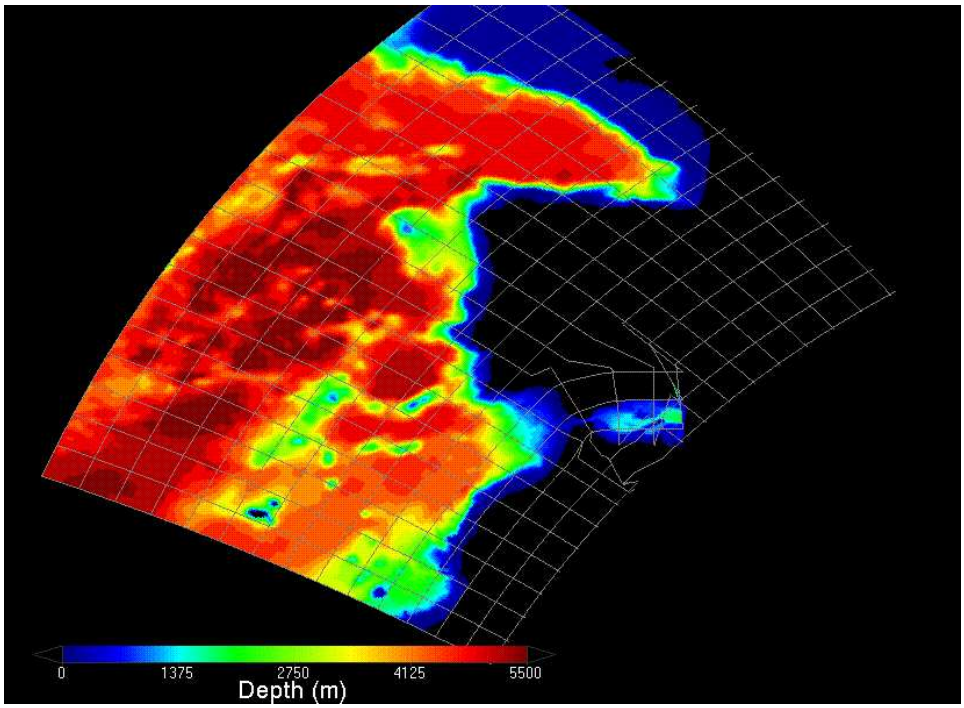

Figure 3.2: Bathymetry of the PSY2v2r1 de Mercator-Océan, dénommée *M-O*.

3.2.2 The FES2004 reference solution

The Finite Element Solutions (FES) are tidal atlases released nearly every two years, being the latest one of them the FES2004[31]. These tidal atlases result from model computation in unstructured meshes with spectral element methods(CEFMO and MOG2D-G code) applied to the nearly-linearized shallow water equations, remote-sensing data assimilation (Topex/Poseidon, ERS1 and ERS2 satellite altimetric data assimilated with the CADOR code), in-situ tidal gauges time series assimilation, and atmospheric tidal forcing (ECMWF). Being a state-of-the-art atlas, it is recommended for tidal applications[31].

3.2.3 Open boundary conditions

When it comes to open boundary conditions, two methods are frequently used: radiative methods, based on the Sommerfeld condition,

$$\frac{\partial \Phi}{\partial t} + \overrightarrow{c} \cdot \overrightarrow{n} \Phi = 0 \tag{3.2}$$

and nudging (or relaxation) methods. For an interesting review on the main OBC methods see Blayo[6]. According to his work, the Flather radiation method[20], consisting of the Sommerfeld condition combined with the continuity equation, is best for radiating the water level. However, it requires an external water level and an external barotropic flux to be known in order to be used. Indeed the Flather radiation method may be equated at the model's open boundaries in the following way:

$$(\overrightarrow{q} - \overrightarrow{q_{ref}}) \cdot \overrightarrow{n} = (\eta - \eta_{ref}) (\overrightarrow{c} \cdot \overrightarrow{n}) . \tag{3.3}$$

where $\overrightarrow{q}$ and $\overrightarrow{q_{ref}}$ are the model's and the external solution's barotropic flux, respectively; $\overrightarrow{n}$ is the external open boundary normal vector; η and η_{ref} are the model's and the external solution's water level. $\overrightarrow{c}$ is the surface gravity wave's celerity, approximated by $\sqrt{g\,H}$.

When only the external water level is known, then the Blumberg method[7], consisting of a combination between a nudging term and the Sommerfeld condition, may be used as an alternative as a radiative method for the water level. It equates for outward fluxes at the open boundary as

$$\frac{\partial \eta}{\partial t} + \overrightarrow{c} \cdot \overrightarrow{n}\, \nabla \eta = -\frac{\eta - \eta_{ref}}{T_{lag}} \tag{3.4}$$

where η is the water level, η_{ref} is the reference water level, $\overrightarrow{c}$ is the external wave celerity estimated to be $\sqrt{g\,H}$, $\overrightarrow{n}$ is open boundary external normal

vector, g is the local gravity acceleration, H is the depth and T_{lag} is the relaxation decay time. The Blumberg method relaxation decay time ranges from a shorter 200 s in deep waters to a longer 2000 s in coastal shallow waters.

For the other variables, where no accurate estimation of their celerity is available, another class of OBC method is used: the nudging method. It consists of a less brutal approach to the clamped (Dirichelet) conditions on the open boundary Γ of the domain Ω, where a relaxation decay time is introduced and an additional domain is created Ω_s, 10 cells wide, which interfaces between Ω and Γ. This approach is commonly regarded as a Flow Relaxation Scheme (FRS)[32]. The nudging term writes

$$\frac{\partial \Phi}{\partial t} = -\frac{\Phi - \Phi_{ref}}{\tau}. \tag{3.5}$$

where Φ is the relaxed variable, Φ_{ref} is the reference solution and τ relaxation time decay constant. The time decay varies from 3×10^4 s on Γ to 1×10^9 s on $\Omega \cap \Omega_s$, 10 cells to the interior. Thus the computed domain becomes $\Omega \cup \Omega_s$. Following Martinsen and Engedahl[32] recommendation, such approach is used as the main downscalling technique for S, T, u and v, respectively salinity, potential temperature, zonal velocity component, and meridional velocity component.

Additionally, in order to smooth out the nudging at Ω_s, a sponge layer, consisting of a high viscosity layer, is implemented. The viscosity terms range, inside Ω_s, from 1.8×10^4 m^2/s at Γ to 10 m^2/s on $\Omega \cap \Omega_s$. In Ω, the horizontal viscosity is considered constant at 10 m^2/s.

Finally, in order to filter out the high frequency noise generated by resonant open boundary spurious reflections, a laplacian biharmonic filter is implemented in the primitive equations. Typical values of the biharmonic filter coefficient may vary between 1×10^{10} m^4/s and 1×10^9 m^4/s.

3.2.4 Bottom forcing

The bottom stress is given by [37]

$$\tau_b^u = \rho_0 C_D u_b \sqrt{u_b^2 + v_b^2} \tag{3.6}$$

where τ_b^u is the bottom stress, u_b and v_b are the near-bottom velocity horizontal componentes, ρ_0 is the reference density, and the drag coefficient is given by[30],

$$C_D = k / \ln\left(\frac{z_D + z_0}{z_0}\right)^2 \tag{3.7}$$

where z_D is the bottom height and z_0 is the roughness length. The Von Karman constant is set to[30] $k = 0.4$. The bottom roughness length is set to $z_0 = 0.0025\ m$ for all models.

3.2.5 Lateral boundary conditions

A null mass and momentum flux is imposed at the lateral land boundary:

$$\overrightarrow{v} \cdot \overrightarrow{n} = 0$$

where $\overrightarrow{v}$ is the velocity vector and $\overrightarrow{n}$ is the normal vector at the land-water interface.

A freshwater discharge with daily values is imposed near the Tagus area for both models. The data source comes from *INAG* (*Instituto da Água*).

3.3 Barotropic model *WI*

Since the Mercator solution is rigid-lid and therefore doesn't takes into account the tide effect correctly, the idea came that a tidal reference solution should be built and linearly superposed to the Mercator reference solution in order to force coastal models with oceanic and tidal effects. Thus, a barotropic model model of western Iberia was created named *WI*, forced with the FES2004 tidal atlas solution. The atmospheric forcing would not be included as the S1 and S2 components of the FES2004 solution already take into account the tidal atmospheric forcing[31]. In figure 3.3, the model's bathymetry is illustrated. The bathymetric baseline data is taken from the ETOPO $2'$[19]. The domain has 0.06^o horizontal and $180\ s$ temporal resolution and is bounded within the interval $[-13.7^o\ -5.3^o]\ W \times [33.5^o\ 46.1^o]\ N$. The water level reference solution is computed from the FES2004 tidal harmonic components. The Blumberg radiative condition[7](eq. 3.4) is applied at the open boundaries. A biharmonic filter is implemented in the domain to filter out high-frequency noise and has a $10^9\ m^4/s$ coefficient. The barotropic force is gradually connected over 10 inertia periods.

3.4 Portuguese coast model *P*

The continental Portugal coastal model is a tridimensional baroclinic model. It may be viewed as the enhanced version relative to Coelho[13]. Composed by 42 vertical layers, it possesses a 0.06^o horizontal resolution and

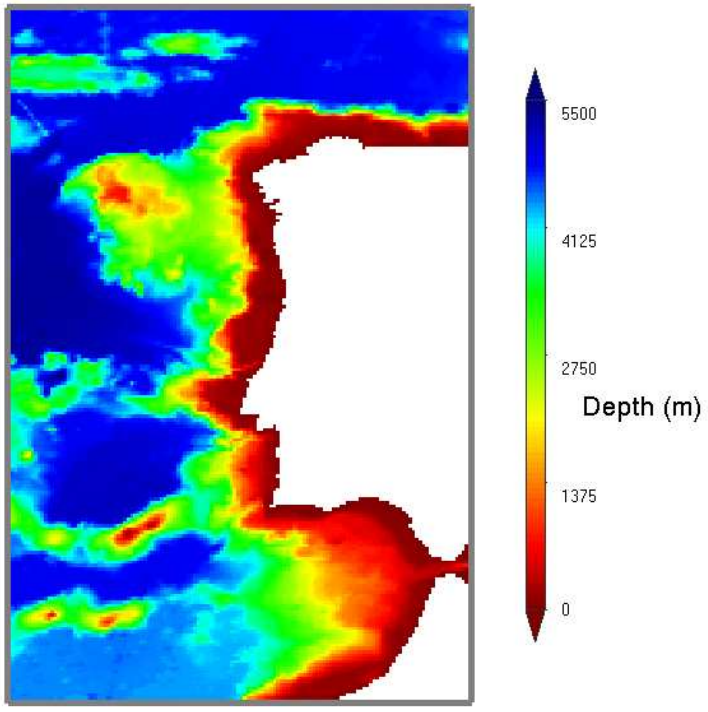

Figure 3.3: Western Iberian peninsula barotropic model bathymetry, labeled *WI*. Baseline data taken from ETOPO $2'$[19]. Bounded by $[-13.7^{o}\ -5.3^{o}]\ W\ \times [33.5^{o}\ 46.1^{o}]\ N$. 0.06^{o} spatial resolution.

a 180 s temporal resolution (Coelho only had 18 layers and 8.5 km of resolution). Bounded by $[-12.6^{o}\ -5.5^{o}]\ W\ \times\ [34.4^{o}\ 45.0^{o}]\ N$ he's submitted to the MM5 reference solution atmospheric forcing at the surface according to the description given in section 3.1, and both by the barotropic model *WI* (section 3.3) and the Mercator model reference solutions *M-O* (section 3.2.1) at the open boundaries. The wind forcing is slowly started over 10 inertia periods. The level is radiated by a Flather radiation method[20] whose barotropic flux and level reference solution are given by the linear superposition of the barotropic fluxes and water levels of *WI* and *M-O* respectively. Also, the Flather radiation method is slowly activated over 10 inertia periods. Furthermore, a FRS[32] is applied to S, T, u and v.

The biharmonic filter coefficient is set to $1 \times 10^{10}\ m^4/s$.

Turbulent horizontal viscosity is estimated roughly to 10 m^2/s inside the domain, but a sponge layer is applied at the open boundaries, ten cells wide. The sponge layer evolves gradually from the inside of the domain up to the boundary where it reaches a $1.8 \times 10^4\ m^2/s$ viscosity.

The modelled domain is labeled P and its bathymetry is shown in figure 3.4.

3.5 Estremadure bank model *C*

The Estremadure bank regional model, bounded by $[-11.2^{o}\ -8.8^{o}]\ W\ \times\ [40.3^{o}\ 37.5^{o}]\ N$, differs from P in the horizontal spatial resolution and in the temporal resolution, respectively of 0.02^{o} and 90 s. It also differs from P in the Flather radiation (eq. 3.3) where the reference level and the barotropic flux come from the P model.

This model should be able to reproduce the evolution of finer-scale physical processes. In particular those associated to the Rossby baroclinic radius of deformation who, near the western Iberia zone, should range within the $25km$ [12] in average. Stevens[40] suggests that a tenfold higher resolution than the first baroclinic Rossby radius of deformation (i.e. 2.5 km) is required in order to resolve the associated finer scale physical processes. In the western Iberia region, 0.06^{o} of horizontal resolution doesn't meets the prior requirements but 0.02^{o} does. Thus, it is expectable that finer-scale processes should appear in this model that are masked out by the rougher resolution in the P model.

This modelled domain is labelled C (as in Centre), and its bathymetry (sourced out from ETOPO $2'$[19]) is illustrated in figure 3.5.

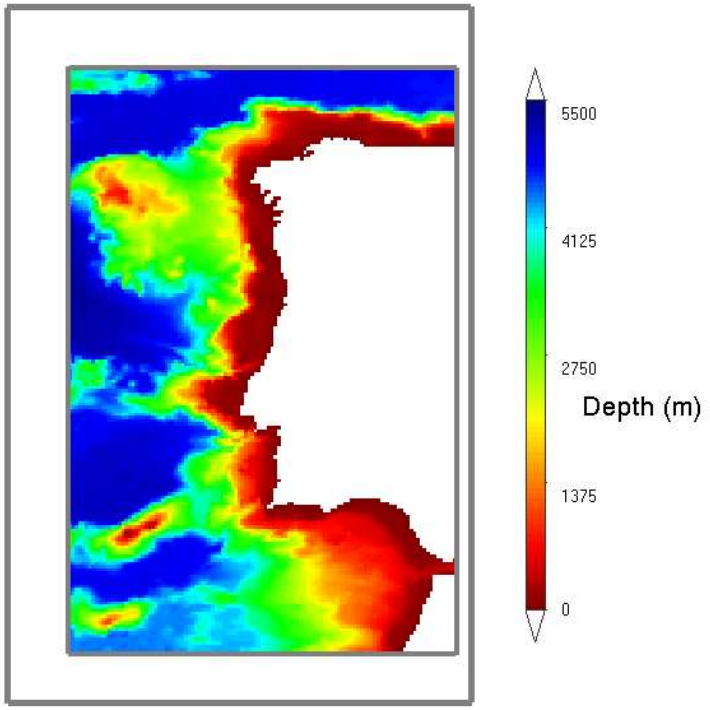

Figure 3.4: Western Iberia coast baroclinic model bathymetry. Baseline data from ETOPO $2'$[19]. The domain is labeled P. Bounded by $[-12.6^{o}\ -5.5^{o}]\ W\ \times [34.4^{o}\ 45.0^{o}]\ N$. 0.06^{o} spatial resolution.

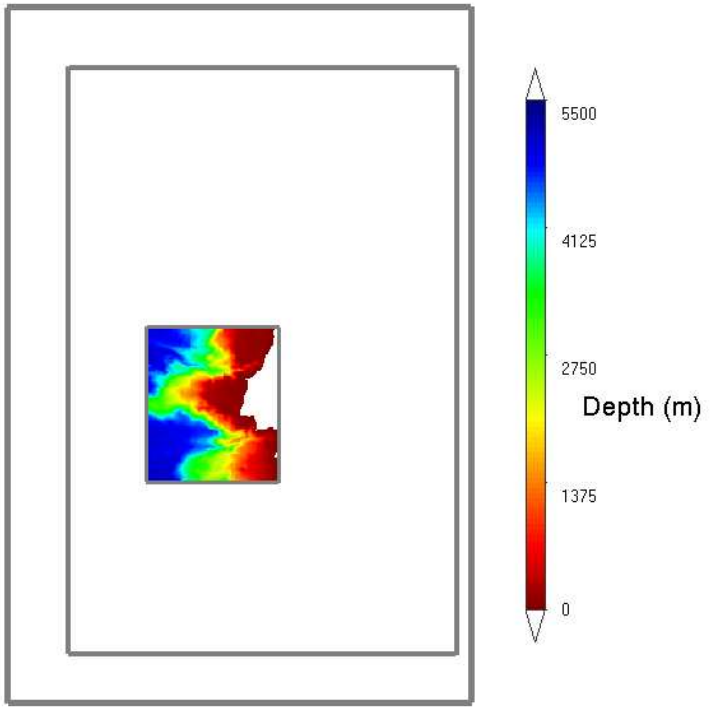

Figure 3.5: Portugal continental central regional coastal model bathymetry, labeled *C*. Baseline data taken from ETOPO $2'$[19]. Bounded by $[-11.2^{o}\ -8.8^{o}]\ W\ \times [40.3^{o}\ 37.5^{o}]\ N$. 0.02^{o} spatial resolution.

Chapter 4

Results

There are two modes of forcing:

- Analysis mode, where the oceanic forcing uses the analyzed *M-O* solution.
- Forecast mode, where the oceanic forcing uses the prediction mode *M-O* solution.

Either way, the atmospheric forcing always uses the *MM5-IST* 7 day forecasting solution. This will provide less reliable results for the analysis mode runs. This is an issue that should be addressed in a near-future.

The models are scheduled to run the past 7 days in analysis mode and the next 7 days in forecasting mode. The 14 day run requires 48 hours to finish, thus giving in the end, 5 days of ocean forecasts. The time to complete the runs can be optimized, perhaps reducing to a 40 hours run.

The first 7 days run is the model's spin-up, allowing it to slowly activate the wind-forcing, the baroclinic forces and the radiative methods while it adjusts a velocity field to the initial density field. An alternate method consisting of performing a calculus continuation by at the end of the last analysis is considered. However, performing a 7 day spin-up every week is numerically more robust than undertaking a calculus continuation, and the results are probably nearly as good. This is a key issue to be investigated further this year.

The model will have run successfully almost every week since November, 22nd 2006. On a few occasions (the January 3rd and 10th 2007), the model unstabilized. It turned out that the problem came from a bottom cell near the open boundary whose relative thickness was excessively small compared to his neighboring cells. This is a numerical problem that can occur often

when performing partial-stepping[1]. The solution was to rise the seafloor a few meters only in order to cover the problematic cell. An idea came to generate a tool that would map a field indicating the cells whose Courant number is prone to unstabilize. In this case, the problematic cell would appear instantly due to its very limited volume.

The forecasts are available on the internet[1] in *netcdf* format by an *Opendap* server[17]. They can be visually inspected with a *netcdf* client such as `ncbrowse`[2].

A comparison between the Mohid results and the Mercator solution is undertaken. The first increment of the comparison is the visual inspection. Results of the temperature and salinity fields of the Mercator solution can be visually inspected against results of the Mohid solution in figures 4.1 and 4.2 for a 2006 mid-December day. The C model results are superimposed over the P model results. We can observe a general gain in the spatial variability of fronts forming in the C domain, for all depths and all variables. This was expected due to the finer resolution of the C domain.

At the depth of the thermocline over the Estremadure bank, at the C domain at 250 m depth, internal waves interference patterns are observed in figure 4.1 due to reflections at the domain's boundaries. These interferences occur during the model's spin-up (starting about the 3rd day) and are rapidly dissipated (by the 9th day). They appear near the thermocline depth, which is where the vertical density gradient is steeper, and which is where the number of vertical layers is higher. This shows that the C domain is able to generate internal waves. This type of internal wave interference pattern doesn't appear in the P domain. This is probably due to an insufficient horizontal resolution, or simply because the domain's characteristic length and time period isn't compatible with the formation of internal waves. Near the Iberian coastal area, the characteristic length of internal waves is estimated to vary between $20 - 30$ km (close to the the first baroclinic Rossby radius of deformation). According to Stevens[41], a tenfold resolution is required in order to accurately reproduce frontogenesis and baroclinic instabilities, i.e. a $2km$ resolution in the West-Iberia coastal area. Hence, the latter argument sustains the hypothesis that the P domain has a non-permitting baroclinic instability resolution.

The latter argument favores the interest of the downscalling of a larger-scale circulation model. However, should Mercator-Océan provide a solution with a 2 km horizontal resolution, then the interest of such down-

[1] http://data.mohid.com/index.xml
[2] http://www.epic.noaa.gov/java/ncBrowse/

scalling would be probably diminished. Only downscaled models with high-resolution winds ($\sim 1\ km$) or with data-assimilation could potentially provide an added value to Mercator's solution.

The MOHID residual velocity fields at the surface and at 750 m depth are very similar to Mercator daily averaged velocity fields in figures 4.4 and 4.5, respectively. Both models reveal a quasi-geostrophic steered flow at the surface, where the flow streamlines follow contours of isobarics, and meanders around pressure regions of highs and lows. Departures from geostrophy are more evident in the Mercator solution, due to regions of isallobaric highs and lows[22] that allow zones of divergence and convergence, respectively. A nearly pure geostrophic flow is more apparent in the MOHID solution, as the regions of isallobaric highs and lows are smoothed out by the 14 day integration. The persistent regions of high and low pressures could be correlated with the topography.

Over the geostrophic steered flow at the surface, a general western-equatorward flow is patent in both solutions. A noticeable difference between the Mercator and MOHID solutions is the coastal atmospheric pressure gradient along Northern-west Iberia which shows a belt with a 20 cm low in the Mercator solution and a belt with a 20 cm high in the MOHID solution. The belts are a direct consequences of the atmospheric forcing, which is statedly different for both solutions. This implies that the southward coastal current off northern-west Iberia is more intense in the Mercator solution than in the MOHID solution. South of Cabo da Roca ($\sim 38.7^{o}\ N$), the southward coastal current decreases in intensity for the Mercator solution, whereas it intensifies in the MOHID solution. The southward coastal flow branches out to the West by Cabo da Roca (nearly detaches) in the Mercator solution. It also branches out in the MOHID solution, but the remaining coastal branch is stronger and better defined in the MOHID solution. In the South-western boundary, Azores current branches flow into the domain and recirculate out of the domain in both solutions. In the MOHID solution, the northmost branch of the Azores current meanders and then merges with the coastal equatorward flow by Cape São Vicente. The northmost Azores current branch is less patent in the Mercator solution due to the high departures from geostrophy allowed by the small one-day time average. The coastal poleward current flows out of the domain off the Moroccan coast in both models, after it branches out South of the Algarve coast. The latter ramification, better defined in the 14 day averaged MOHID solution, will flow past the Strait of Gibraltar, into the Mediterranean basin.

At 750 m depth, the westward pattern of the flow is still predominant, but the southward trend tends to be replaced with a more zonal flow in both

solutions. Convergence and divergence zones departing from geostrophy are quite flagrant south of $\sim 38.7^{o}N$ in the Mercator solution. The coastal slope current reverts its direction at depth, flowing northward and is better defined and more intense in the MOHID solution. The Azores current main branch flows souther at depth than at the surface. It branches out in the Gulf of Cadiz and recirculates north, merging with the Mediterranean Water undercurrents, and flows out of the domain, through the southern boundary, off the Moroccan coast. The coastal slope current South of Algarve is produced by Mediterranean Waters entraining into the Atlantic Waters, it is fed with the Azores current branch that recirculates in the Gulf of Cadiz, then it turns northward by the Cape São Vicente, always following the slope. Past Cape São Vicente, the slope current branches out and feeds the Azores current main branch. This coastal slope current is a feature of the renown Iberian Poleward Current (IPC) and is well described throughout the litterature[41].

The work of Drillet[18] validates the capability of the Mercator solution of accurately reproducing the meddies life-cycle (since their genesis near Cape São Vicente or over the Estremadure bank to their dissolution in Atlantic waters) as well as the characteristical Mediterranean veins of the area. Figure 4.6 is a series of meridional vertical cross sections of salinity of the Gulf of Cadiz spaced of $0.5^{o}E$ each in the interval $[-8^{o}E\ -6.5^{o}E]$. The cross sections show the formation of deep Mediterranean Water flowing past the Gibraltar Strait into the Atlantic, forming the Mediterranean salt tongue[8]. The overflow of denser Mediterranean waters entrains at the Gibraltar Strait under the less dense North Atlantic Central Water (NACW) and downslopes[15] along the continental slope on the northern margin, south of Algarve as a density-driven current. As it flows westwards, at about $8^{o}\ W$, it reaches neutral buoyancy and detaches from the bottom near 700 m depth and continues as a boundary undercurrent, then it descends down to 1000 m depth[8] near $8.5^{o}\ W$ (fig.4.6) where it seems to attain hydrostatic equilibrium. The Mediterranean salt tongue turns northward past Cape São Vicente and probably continues flowing northward to as far as Porcupine bank ($50^{o}\ N$)[24]. Figure 4.8 is a series of zonal vertical cross sections of salinity of western Iberia spaced at $1^{o}N$ intervals in the range $[36.5^{o}N\ 43.5^{o}N]$. Figures 4.6 and 4.8 evidence the main Mediterranean vein by the anomalous salinity maximum. The depth of salinity maxima varies between the 800 m and 1200 m depth. The number of salinity maxima varies from one to two, sometimes three. It is interesting to see how the boundary drive main Mediterranean veins follow the poleward undercurrent by inspecting the number of salinity maxima, each maximum corresponding

to one MW vein. One interesting experiment would be to test the sensitivity of the neutral buoyant Mediterranean water veins depth to different seawater density state equations. The MOHID model uses UNESCO seawater density equation of state[35] to calculate density from potential temperature, salinity and pressure; but more recent equations may be used such as those of Jackett[26] or McDougall[34]. Figure 4.7 shows vertical meridional salinity cross sections from western Iberia south coast to the north coast. It shows clearly the meridional extension of the Mediterranean tongue and its main vein, at about 1000 m depth.

Figure 4.3 compares results from the Mercator solution interpolated over the P domain and the results from the P and C models and a NOAA satellite sea surface temperature (SST) image. All results are for the same day. Mercator results are daily average, MOHID results are instantaneous at 12h00 and the satellite image was taken during the mid-afternoon. Though it's hard to analyze the results, the MOHID finer resolution model has a higher spatial variability and is likely to resemble more the satellite SST.

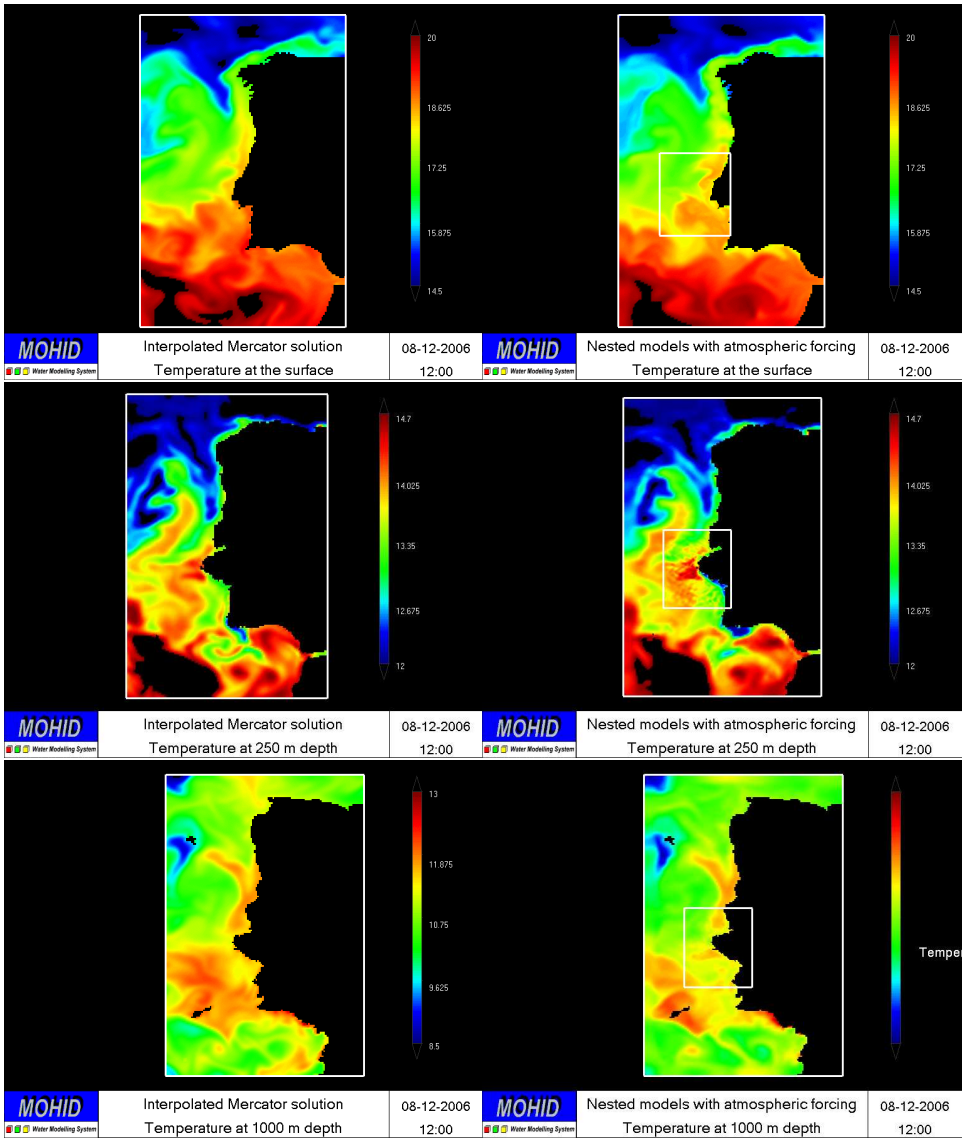

Figure 4.1: To the left, the interpolated temperature fields of the *M-O* solution. To the right, the superposition of the temperature fields of the C model over the P model. The temperature scale's interval is $[14.5 20.0]\,^{o}C$ at the surface (top), $[12.0 14.7]\,^{o}C$ at 250 m (middle) and $[8.5 13.0]\,^{o}C$ at 1000 m (bottom). The graphical tool is Mohid GIS.

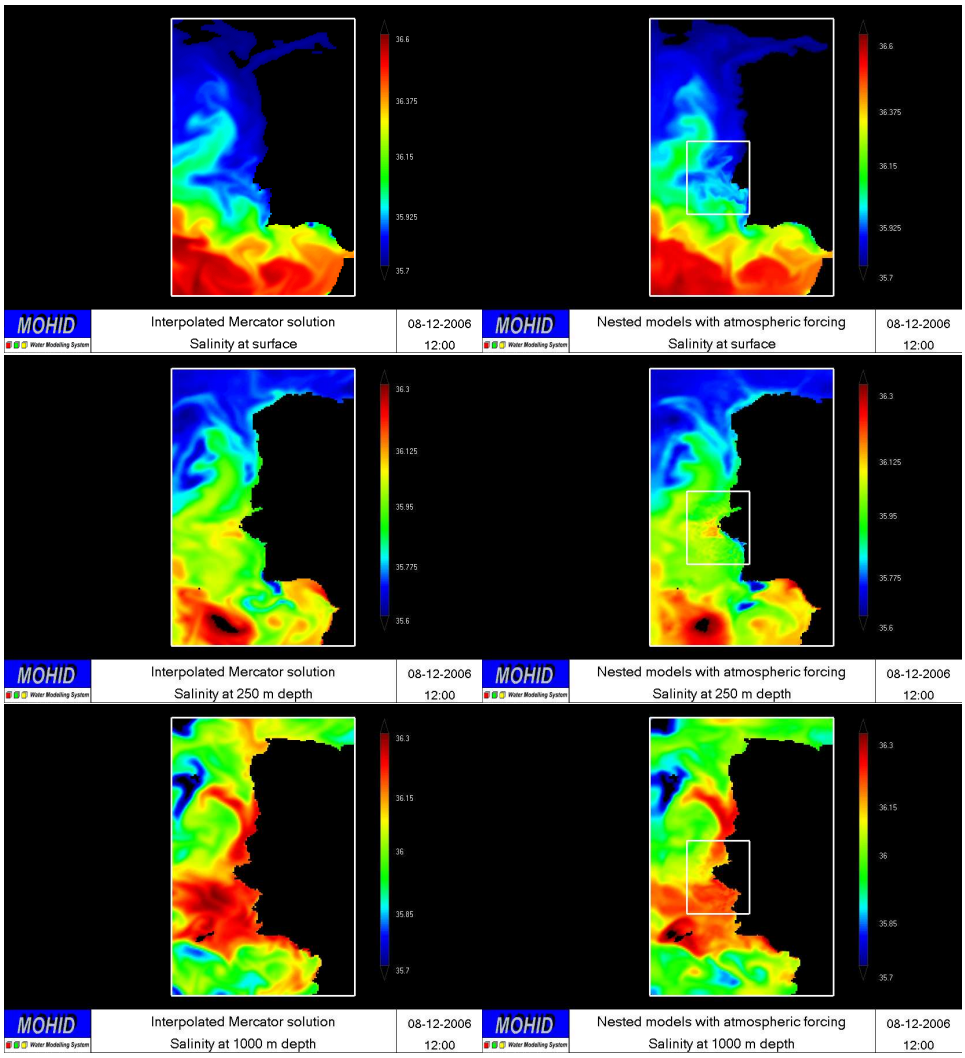

Figure 4.2: To the left, the interpolated salinity fields of the *M-O* solution. To the right, the superposition of the salinity fields of the *C* model over the *P* model. The salinity scale's interval is [35.736.6] at the surface (top), [35.636.3] at 250 m (middle) and [35.736.3] at 1000 m (bottom). The graphical tool is Mohid GIS.

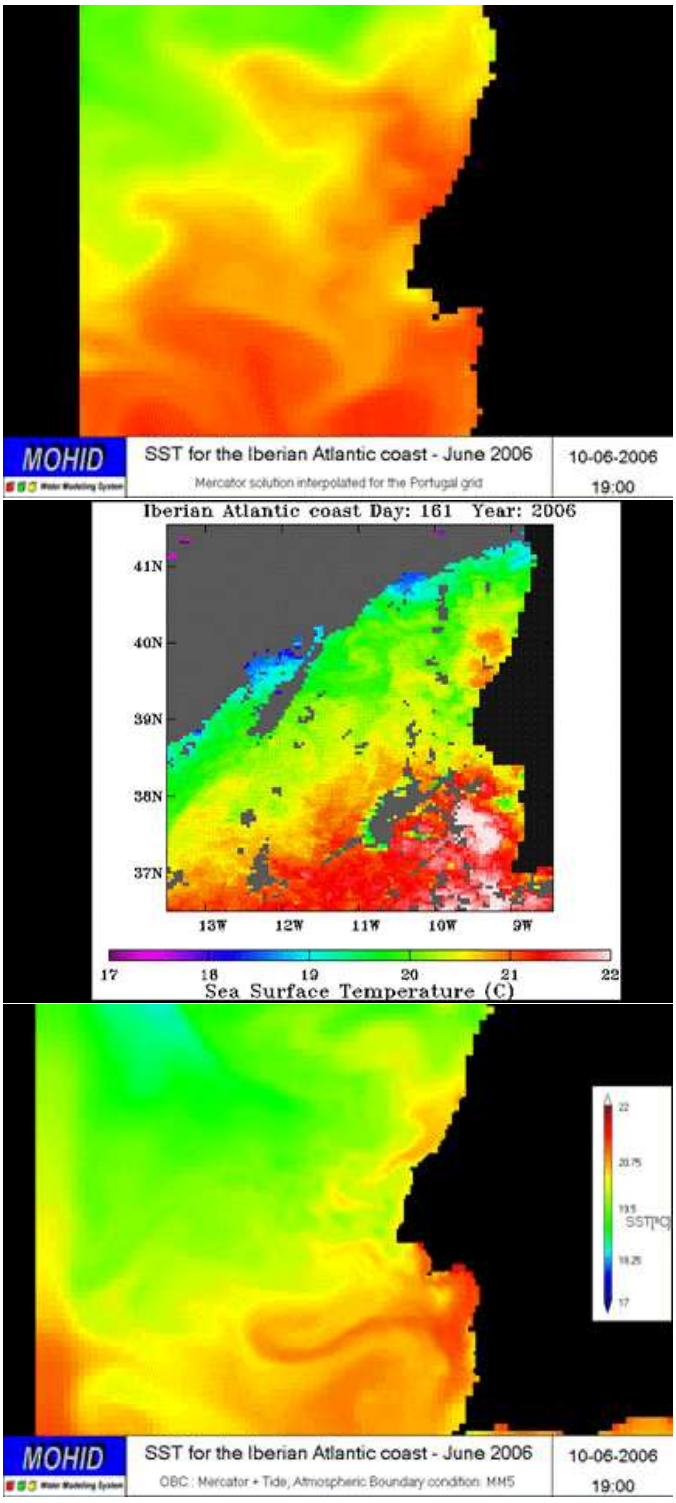

Figure 4.3: The Mercator solution sea surface temperature daily average on the September 9th 2006, at the top. A NOAA sea surface temperature satellite image taken during the same day, at the middle. At the bottom, the Mohid instantaneous solution taken the same day at 19h00 hours. The temperature scale is set to $[17^oC\ 22^oC]$. However, the color palettes differs between the satellite images and the model's fields. The graphical tool used is *Mohid GIS*.

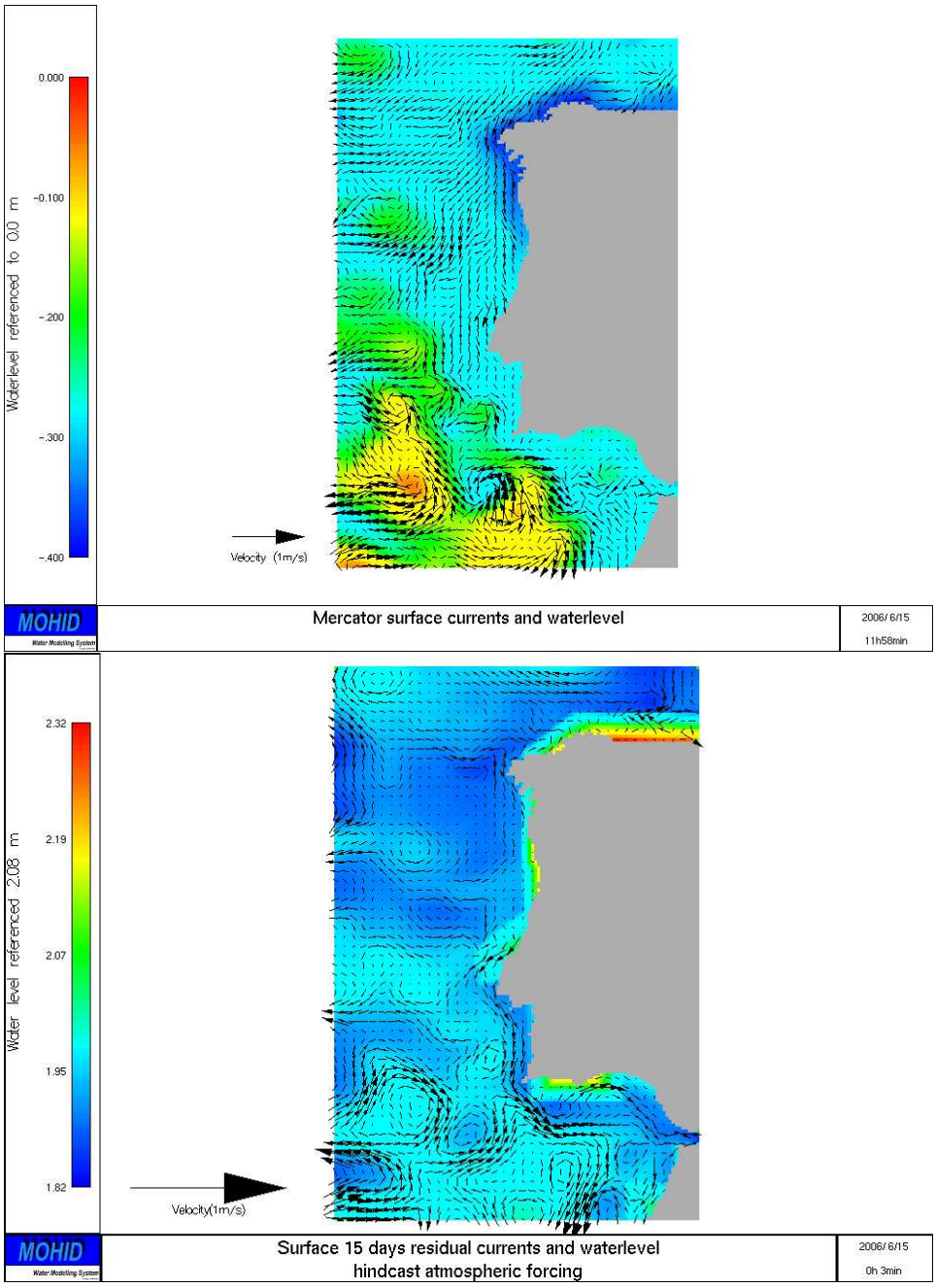

Figure 4.4: On top, the horizontal velocity instantaneous vectorial field and the water level field of the Mercator solution interpolated over the P domain near the surface. At the bottom, the 14 day average solution for the same variables of the MOHID P dom. The mean level amplitude variation is around 0.4 m to the left and 0.5 m to the right. The graphical tool used is *Mohid PostProcessor*.

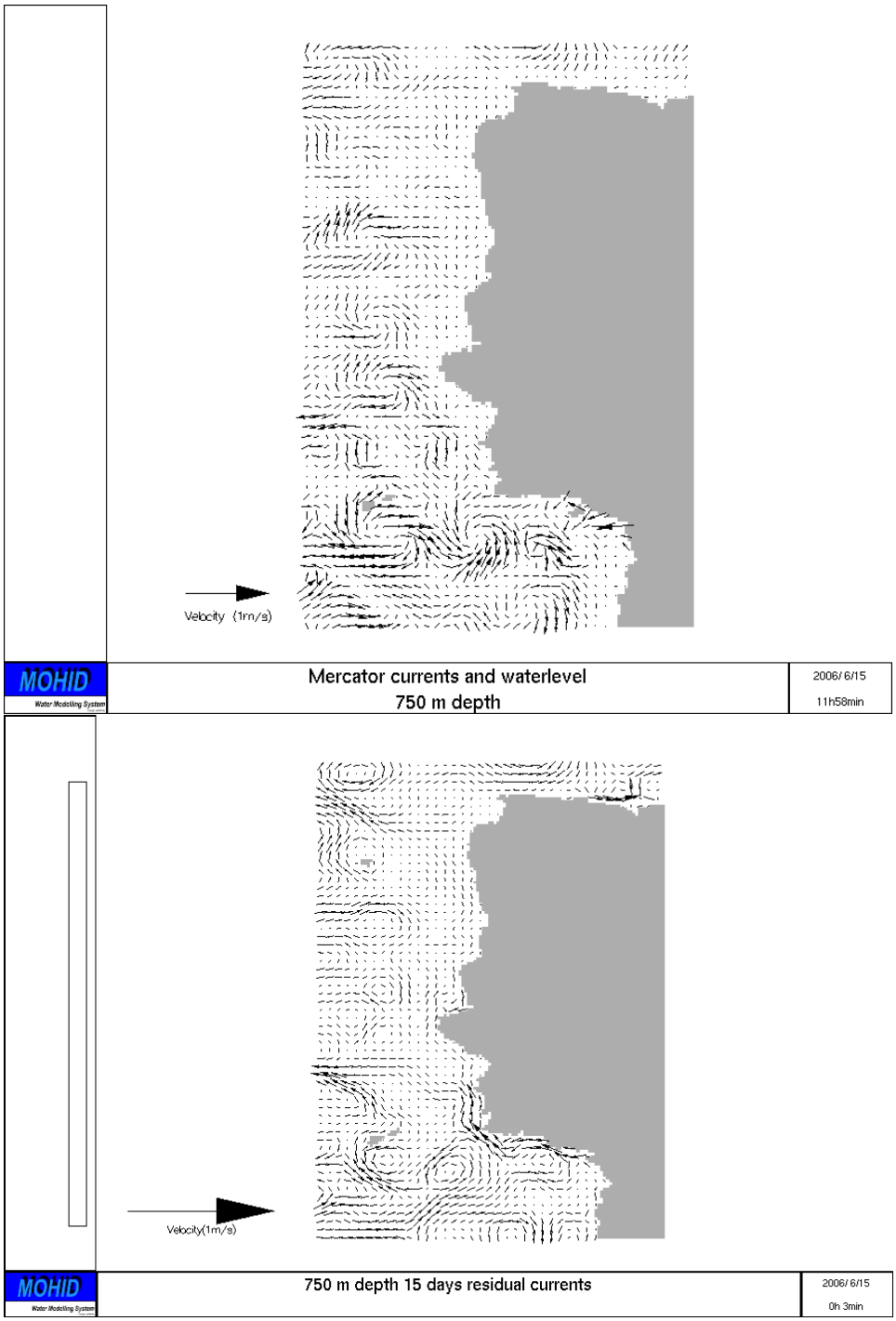

Figure 4.5: On top, the horizontal velocity instantaneous vectorial field and the water level field of the Mercator solution interpolated over the P domain at 750 m of depth. At the bottom, the 14 day average solution for the same variables of the MOHID P dom. The mean level amplitude variation is around 0.4 m to the left and 0.5 m to the right. The graphical tool used is *Mohid PostProcessor*.

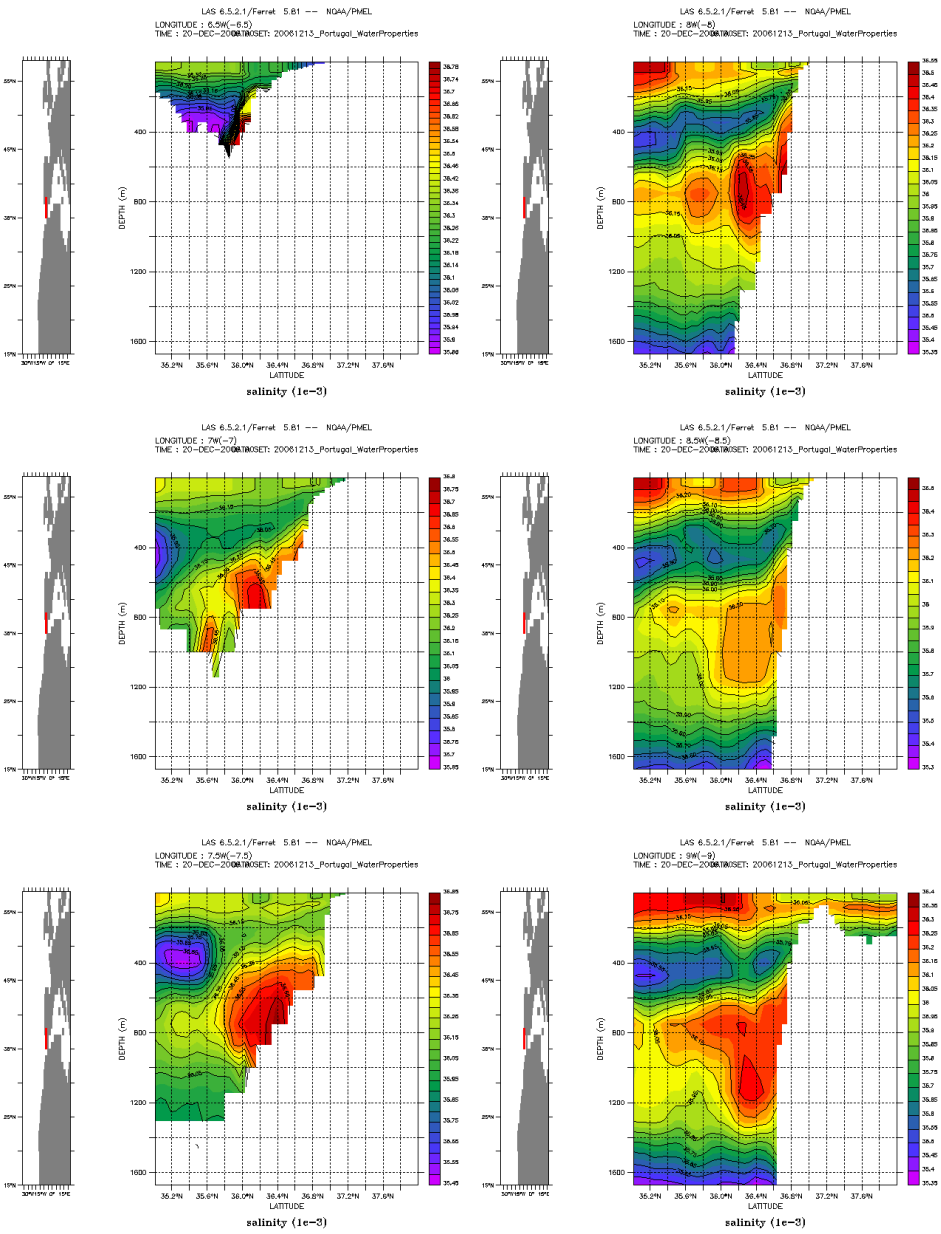

Figure 4.6: Two-column reading from top to bottom, then from left to right. Set of YZ salinity vertical cross sections dated 20/12/2006 over $[-8^oE - 6.5^oE]$ of the P model. Salinity contour scales and spacing follow respectively, according to the template [minimum:step:maximum]: $[35.86 : 0.04 : 36.82]$, $[35.65 : 0.1 : 36.80]$, $[35.45 : 0.1 : 36.85]$ $[35.35 : 0.1 : 36.55]$, $[35.30 : 0.1 : 36.60]$, $[35.35 : 0.1 : 36.40]$

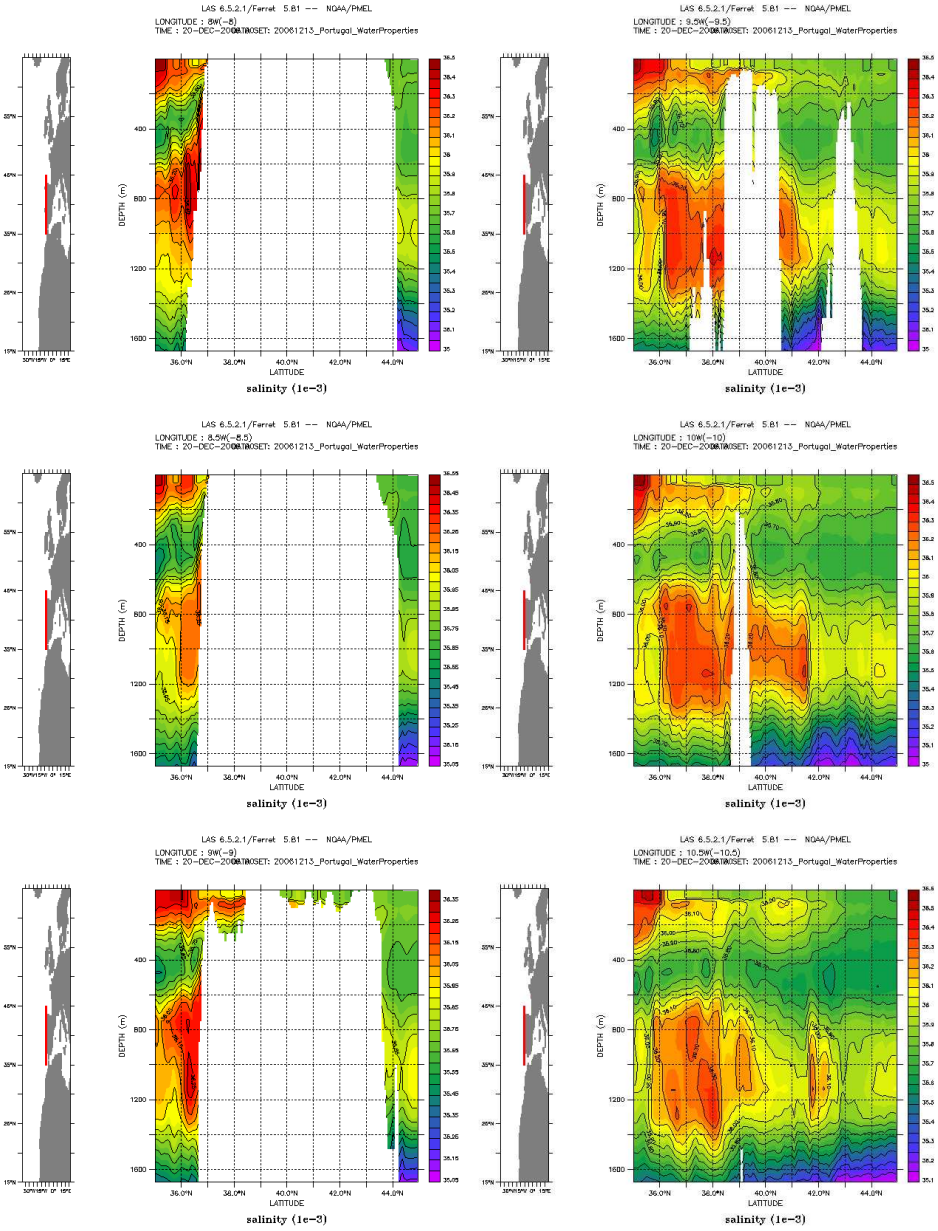

Figure 4.7: Two-column reading from top to bottom, then from left to right. Set of YZ salinity vertical cross sections dated 20/12/2006 over $[-10.5^{o}E\ -8^{o}E]$ of the P model. Contours of salinity are spaced of 0.1. Salinity contour scales follow respectively according to the template [minimum:maximum]: [35.036.5], [35.0536.55], [35.0536.40], [35.036.5], [35.036.5], [35.136.6]

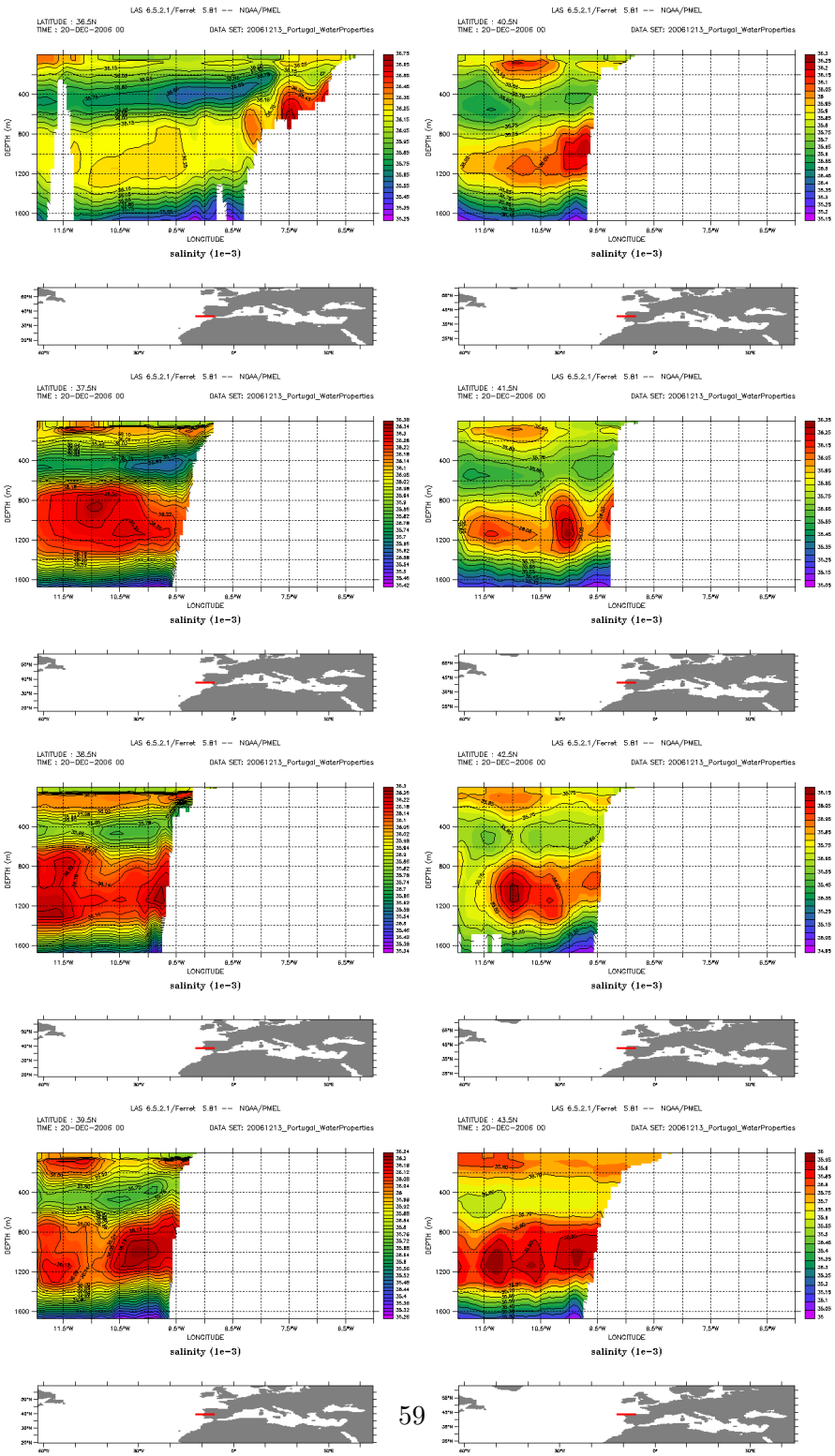

Figure 4.8: Two-column reading from top to bottom, then from left to right. Set of XZ salinity vertical cross sections dated 20/12/2006 over $[36.5^{o}N\ 43.5^{o}N]$ of the P model. Salinity contour scales and spacing follow respectively according to the template [minimum:step:maximum]: $[35.25 : 0.04 : 36.75]$, $[35.42 : 0.04 : 36.38]$, $[35.34 : 0.04 : 36.30]$, $[35.26 : 0.04 : 36.24]$, $[35.15 : 0.1 : 36.30]$, $[35.05 : 0.1 : 36.35]$, $[34.95 : 0.1 : 36.25]$, $[35.0 : 0.1 : 36.0]$

Chapter 5

Conclusions

The objectives set up for this first year's quarter were:

1. To have a gain in the spatial and temporal variability of the structures formed by the density gradients (instabilities, fronts).

2. The prediction and analysis mode MOHID's solutions should have a tendency to converge.

3. To have the system the as much automatized as possible.

The first and last objectives were correctly obtained. Henceforth, starts the work to quantify the gain obtained in variability by the finer resolution C model, as well as the comparison of the results with the Mercator solution.

Thus, our goal is to run the model over several months in analysis mode, in order to see when do the MOHID and Mercator solutions start to diverge. Also, comparison with available CTD data and tidal stations will be attempted. These results will be the subject of a poster for the EGU annual congress[1], and of a talk for the CILAMCE Iberian congress[2]. The abstracts were already accepted.

[1] http://meetings.copernicus.org/egu2007/
[2] http://cmne2007.inegi.up.pt/index_en.asp

Appendix A

Technical specifications

A.1 Workstation technical specifications

For running the operational model described in section 2.2.1, the following workstation was acquired:

- AMD Opteron 270 dual core, dual processor, 64 bit architecture at 2 GHz clocking,
- 4 GB DDR2 ram,
- RAID 0 250 GB disk capacity for OS and software,
- RAID 1 500 GB disk capacity for data (extensible to 2 TB),
- a redundant power source and an UPS.

A Fedora Core 5 operating system was installed.

A.2 MOHID benchmark

MOHID was always run under Intel chipsets. AMD and the 64 bit architecture are new technology at MARETEC. Although the new AMD Opteron chipsets are acclaimed in the international news as being equal or superior to Intel's, benchmarking MOHID revealed that:

- The cpu frequency is the most relevant aspect for performance: 3 GHz systems are estimated to run 50% faster than 2 GHz systems, within the same chipset and architecture.

- In a 64 bits architecture, floating-point calculations with 64 bit precision are as fast as 32 bit precision; while there's a 90% performance loss in 32 bits architecture.

- Ad-hoc openmp implemented directives show a 20% gain with two cores. The gain is roughly the same using all four cores.

- MPI directives have not yet been tested in the new workstation. An efficient MPI debugging practice needs to be investigated or developed.

The new AMD workstation offers top performance at MARETEC for 64 bit precision calculation. However, in 32 bits precision, a single core is 25% slower than an Intel P4 3.4 GHz one. Compiling with other fortran compilers such as Absoft's or Portland Groups's may yield a better performance than Intel's.

The linux OS allows up to four users to launch their applications in single-core mode safely and without any loss in performance.

For future hardware acquisitions, Intel chipsets are still the best for running MOHID. Also, maximum cpu clocking ($>$ 3.4 GHz) is advised for substancial improvement in peak performance.

Given the current multi-core platforms trend, investing in an efficient parallelization methodology is crucial for MOHID to follow Moore's Law.

Bibliography

[1] Adcroft, A., Hill, C., and Marshall, J. Representation of Topography by Shaved Cells in a Height Coordinate Ocean Model. *Monthly Weather Review 125*, 9 (1997), 2293–2315.

[2] Ambar, I. Physical, chemical and sedimentological aspects of the mediterranean outflow off iberia. 4137–4177.

[3] Arakawa, A. Computational design for long-term numerical integration of the equations of uid motion: Two-dimensional incompressible ow. Part I. *Journal of Computational Physics 1*, 1 (1966), 119–143.

[4] Bahurel, P., De Mey, P., De Prada, T., Dombrowsky, E., Josse, P., Le Provost, C., Le Traon, P. Y., Piacentini, A., and Siefridt, L. MERCATOR, forecasting global ocean. *AVISO Altimetry Newsletter 8* (2001), 14–16.

[5] Beckmann, A., and Haidvogel, D. B. Numerical simulation of flow around a tall isolated seamount. part i: Problem formulation and model accuracy. *Journal of Physical Oceanography 23*, 8 (1993), 1736–1753.

[6] Blayo, E., and Debreu, L. Revisiting open boundary conditions from the point of view of characteristic variables. *Ocean Modelling 9* (2005), 231–252.

[7] Blumberg, A. F., and Kantha, L. H. Open Boundary Condition for Circulation Models. *Journal of Hydraulic Engineering 111*, 2 (1985), 237–255.

[8] Bower, A. S., Serra, N., and Ambar, I. Structure of the mediterranean undercurrent and mediterranean water spreading around the southwestern iberian peninsula. *Journal of Geophysical Research 107*, C10 (2002).

[9] Bryan, K. A numerical method for the study of the world ocean. *J. Comput. Phys 4* (1969), 347–376.

[10] Burchard, H., and Others. *Applied Turbulence Modelling in Marine Waters*. Springer, 2002.

[11] Canuto, V. M., Howard, A., Cheng, Y., and Dubovikov, M. S. Ocean Turbulence. Part I: One-Point Closure Model Momentum and Heat Vertical Diffusivities. *Journal of Physical Oceanography 31*, 6 (2001), 1413–1426.

[12] Chelton, D. B., Deszoeke, R. A., Schlax, M. G., Naggar, E. K., and Siwertz, N. Geographical Variability of the First Baroclinic Rossby Radius of Deformation. *Journal of Physical Oceanography 28*, 3 (1998), 433–460.

[13] Coelho, H., Neves, R., White, M., Leitão, P., and Santos, A. A model for ocean circulation on the iberian coast. *Journal of Marine Systems 32*, 1 (2002), 153–179.

[14] Craig, P. D., and Banner, M. L. Modeling Wave-Enhanced Turbulence in the Ocean Surface Layer. *Journal of Physical Oceanography 24*, 12 (1994), 2546–2559.

[15] Deleersnijder, E. Upwelling and upsloping in three-dimensional marine models. *Applied Mathematical Modelling 13* (1989), 462–467.

[16] Delgado Domingos, J. J., and Trancoso, A. R. Meteo - ist, http://meteo.ist.utl.pt/. Internet service, 2005.

[17] Doty, B. E., Wielgosz, J., Gallagher, J., and Holloway, D. GrADS and DODS/OPENDAP. *Proceedings of the 17th International Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology, American Meteorological Society Albuquerque, NM 385* (2001).

[18] Drillet, Y., Badie, B. R., Siefridt, L., and Le Provost, C. Meddies in the Mercator North Atlantic and Mediterranean Sea eddy-resolving model. *Journal of Geophysical Research 110*, C3 (2005).

[19] Etopo, N. Bathymetry. *Product Information Catalogue, see also http://www. ngdc. noaa. gov/mgg/global/seltopo. html* (1988).

[20] Flather, R. A. A tidal model of the northwest European continental shelf. *Mem. Soc. R. Sci. Liege 10*, 6 (1976), 141–164.

[21] Fletcher, C. A. J., and Srinivas, K. *Computational Techniques for Fluid Dynamics 1.* Springer, 1991.

[22] Gill, A. E. *Atmosphere-ocean dynamics.* Academic Press New York, 1982.

[23] Grell, G. A., Dudhia, J., and Stauffer, D. R. A description of the fifth-generation Penn State/NCAR Mesoscale Model (MM5). NCAR Tech. *Note TN-398+ STR 122* (1995).

[24] Iorga, M. C., and Lozier, M. S. Signatures of the Mediterranean outflow from a North Atlantic climatology 1. Salinity and density fields. *Journal of Geophysical Research 104*, C11 (1999), 25985–26010.

[25] Iorga, M. C., and Lozier, M. S. Signatures of the Mediterranean outflow from a North Atlantic climatology 2. Diagnostic velocity fields. *Journal of Geophysical Research 104*, C11 (1999), 26011–26030.

[26] Jackett, D. R., and Mcdougall, T. J. Minimal adjustment of hydrographic profiles to achieve static stability. *Journal of Atmospheric and Oceanic Technology 12*, 2 (April January, JanuarySeptemberSeptemberMay 1995), 381–389.

[27] Kliem, N., and Pietrzak, J. D. On the pressure gradient error in sigma coordinate ocean models: A comparison with a laboratory experiment. *Journal of Geophysical Research 104*, C12 (1999), 29781–29800.

[28] Leendertse, J. J. *Aspects of a Computational Model for Long-period Water-wave Propagation.* Rand Corporation for the United States Air Force Project Rand, 1967.

[29] Leitão, J. C., Leitão, P., Braunchweig, F., Fernandes, R., Neves, R., and Montero, P. Emergency activities support by an operational forecast system 'e the prestige accident. *4th Seminar of the Marine Environment, Rio de Janeiro (2003)* (2003).

[30] Leitão, P. C. *Integration of Scales and Processes in the marine Environment Modelling.* PhD thesis, Technical Superior Institute, Lisbon, 2003.

[31] Lyard, F., Lefevre, F., Letellier, T., and Francis, O. Modelling the global ocean tides: modern insights from fes2004. *Ocean Dynamics 56*, 5-6 (December 2006), 394–415.

[32] Martinsen, E. A., and Engedahl, H. Implementation and testing of a lateral boundary scheme as an open boundary condition in a barotropic ocean model. *Coastal engineering 11*, 5-6 (1987), 603–627.

[33] Martins, H., Santos, A., Coelho, Neves, R., and Rosa, T. Numerical simulation of internal tides. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 214*, 6 (2000), 867–872.

[34] Mcdougall, T. J., Jackett, D. R., Wright, D. G., and Feistel, R. Accurate and computationally efficient algorithms for potential temperature and density of seawater. *Journal of Atmospheric and Oceanic Technology 20*, 5 (May January, February00March 2003), 730–741.

[35] Millero, F. J., and Poisson, A. International one-atmosphere equation of state of seawater. *Deep-Sea Res 28* (1981), 625–629.

[36] Pichevin, T., and Nof, D. The eddy canon. *Deep-Sea Res 43*, 9 (1996), 1475–1507.

[37] Pietrzak, J., Jakobson, J. B., Burchard, H., Vested, H. J., and Petersen, O. A three-dimensional hydrostatic model for coastal and ocean modelling using a generalised topography following coordinate system. *Ocean Modelling 4*, 2 (2002), 173–205.

[38] Reynaud, T., Legrand, P., Mercier, H., and Barnier, B. A new analysis of hydrographic data in the Atlantic and its application to an inverse modelling study. *International WOCE Newsletter 32* (1998), 29–31.

[39] Shchepetkin, A. F. A method for computing horizontal pressure-gradient force in an oceanic model with a nonaligned vertical coordinate. *J. Geophys. Res 108(C3)* (2003), 3090–3090.

[40] Stevens, D. P. On open boundary conditions for three dimensional primitive equation ocean circulation models. *Geophysical and astrophysical fluid dynamics 51*, 1-4 (1990), 103–133.

[41] Stevens, I., Hamann, M., Johnson, J. A., and Fiúza, A. F. G. Comparisons between a fine resolution model and observations in the iberian shelf-slope region. *Journal of Marine Systems 26*, 1 (2000), 53–74.

[42] Umlauf, L., and Burchard, H. Second-order turbulence closure models for geophysical boundary layers. A review of recent work. *Continental Shelf Research 25* (2005), 725–827.

[43] *Finite volume ocean circulation model* (2002), vol. 3, IEEE.