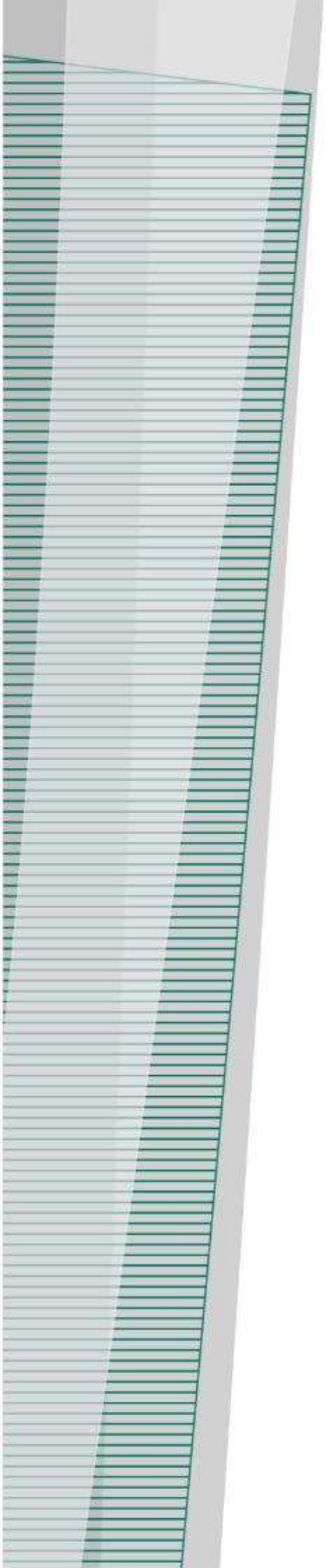

LABORATÓRIO NACIONAL DE ENGENHARIA CIVIL

Infraestrutura Nacional de **Computação Distribuída**

# Relatório de resultado – Piloto Experimentação Cloud Computing

(E6 – Relatório dos testes relativos às alíneas (a) e (b) do nr. 1 da cláusula 3º do protocolo:

# EVALUATION OF CIVIL ENGINEERING USE CASES IN A PUBLIC CLOUD COMPUTING FRAMEWORK. Assessment on CPU computational resources)

EXT/2016
15-November-2016

FCT Fundação para a Ciência e a Tecnologia
MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E ENSINO SUPERIOR

FCCN Computação Científica Nacional

Authors:
Anabela Oliveira
João Rogeiro
Alberto Azevedo
José Barateiro
João Rico
António Inês

Revised by:
André Fortunato

# OUTLINE

# LIST OF FIGURES

# LIST OF TABLES

## EXECUTIVE SUMMARY

In this report, we present a comparison of model performance indicators for several operational coastal forecast systems and structural engineering predictions and evaluations executed in local workstations, in HPC cluster nodes and in a pilot cloud, aiming at contributing to the best choice for the National Infrastructure for Scientific Computing.
Results show that the scalability and flexibility of cloud computing resources makes them an attractive alternative for the implementation of multiple forecast systems using serial, non-MPI models, as well as for sensor data acquisition and processing applications.
For MPI-based models, tests using cloud virtual machines with resources equal to or lower than the smaller physical bases performed well relative to the other resources. However, as the cloud resources under testing did not reach the optimal number of processors for the present use cases, the HPC cluster remained the best option, as it fits better the requirements for the typical dimensions of computational grids for multi-scale (port to ocean) analysis.
Federated cloud resources allowed a better performance for small pool sizes, allowing the combination of the processing power of several hosts. However, the performance does scale very badly if the choice relies in any combination that uses many processes (by using many hosts or many processes within each host), even if we use resources with some hardware assistance.
Further testing is still necessary to explore this possibility in detail, taking into account the need to assure an adequate quality of service (QoS), especially to meet forecasting deadlines and real-time streaming bandwidth.
We conclude that an evolution from the current cluster setup to a cloud-based architecture will satisfy most of our simulation requirements while offering a more flexible and affordable computing environment.

Keywords: cloud, grid, parallel computing, forecast systems, numerical models, optimal performance

# 1 INTRODUCTION

In the scope of the Portuguese National Distributed Computing Infrastructure (INCD, www.incd.pt), FCT/FCCN has established a protocol with LIP and LNEC to jointly promote an experimental pilot of cloud computing. LNEC's tasks in this pilot are the following:

- Test and evaluate the performance of civil engineering applications in the pilot cloud infrastructure developed by LIP and FCT/FCCN;
- Compare the tests performed in the previous task with commercial cloud infrastructures;
- Test the feasibility of using GPUs, also in a cloud computing approach, to accelerate remote sensing detection algorithms and numerical modeling programs through the implementation of heterogeneous computing (CPU+GPU).

This report presents the outcomes of the first task. LNEC has implemented, tested and compared the performance of several coastal forecasting workflow applications and several structural engineering applications in the pilot cloud computing infrastructure and other HPC infrastructures.

Besides this Introduction, the report is organized as follows. In Chapter 2, the use cases are described and characterized for computational assessment. Chapter 3 presents a short description of the Infrastructures used in the performance analysis for CPU, memory and other variables. Chapter 4 presents the evaluation for the two sets of tests using the Pilot cloud infrastructure and the other HPC resources available at LNEC, including the evaluation procedure. The report closes with the conclusions from the several comparative analyses.

# 2 USE CASES DESCRIPTION

## 2.1 OVERVIEW OF THE USE CASES AREAS

### Water Information Forecast Framework for coastal applications

The ability to simulate and forecast the dynamics of estuarine and coastal zones is essential to assess the social, ecological and economic impacts of human interventions as well as climate variability and changes, and to support the sustainable management of these regions. Nowcast-forecast information systems are part of this effort (Oliveira et al., 2014), by integrating remote monitoring networks (Gomes et al., 2013), distributed or parallel computing resources (due to high-demanding computation processes), either for short forecasting or resulting data analysis, and web-based information systems to support timely management decisions. In particular, coastal nowcast-forecast systems (Jesus et al., 2012) are now operational tools for the management of harbors, marine resources and emergency operations, which provide precise and timely information on waves and currents conditions.

The nowcast-forecast information system used herein - Water Information Forecast Framework (WIFF) - is based on the deployment of a generic forecasting platform, adaptable to any geographical location, that was customizable by LNEC for coastal applications (Jesus et al., 2012; 2013). The system integrates a set of parallel computing numerical models that run in a high-performance environment. The WIFF platform is capable of coupling waves, tides, storm surges, river flows, precipitation and winds,

providing forecasts of water levels, currents, water temperatures, salinity and waves for a target area. It was recently extended for water quality, including fecal contamination in sewer systems and estuarine water bodies (Rodrigues et al., 2013) and oil spills in coastal regions (Boer et al., 2014).

### Fecal contamination early-warning systems for coastal management

Coastal and estuarine waters' quality for recreational and bathing uses is a requirement expressed in several national laws and European frameworks (e.g. Bathing Water Framework, BWF). This concern is particularly relevant due to the several uses that occur in these areas, some of which may affect their quality. For instance, domestic effluents or pluvial discharges may lead to fecal contamination events which constitute a potential hazard for public health. Thus, understanding and anticipating these events is essential in the assessment and prevention of social, ecological and economic impacts of both human interventions and climate change in these areas, and, ultimately, to their sustainable management.

Coastal and estuarine nowcast-forecast systems are important tools to support sustainable management, as models predicting capacities allow the anticipation of potential risk situations and the prevention of their consequences. Regarding water quality, nowcast-forecast systems are now emerging (e.g. http://bathingwater.dhigroup.com/earlywarningsystem.html, http://www.waterman.hku.hk/beach/member), but there are still only a few nowcast-forecast systems addressing fecal contamination issues.

Hydrodynamics and fecal contamination are simulated herein using ECO-SELFE. ECO-SELFE is a 3D community numerical model which solves the three-dimensional shallow-waters equations for circulation (SELFE, Zhang and Baptista, 2008) and the three-dimensional transport equation, including advection-dispersion and decay due to mortality and settling, for fecal contamination (Rodrigues et al., 2011), among many other processes (e.g. Rodrigues et al., 2009).

SELFE solves the three-dimensional shallow–water equations and calculates the free-surface elevation, the three-dimensional fields of velocities, salinities and water temperatures. It is coupled to the fecal contamination module through the transport equation, taking advantage of the user-defined transport module available within SELFE.

The fecal bacteria model allows the simulation of two fecal bacteria (fecal coliforms, *Escherichia coli* - *E. coli* and enterococcus), considering the die-off and sedimentation processes as sink terms. Two options were available for the definition of the die-off rate: a user-defined constant coefficient and the formulation proposed by Canteras et al. (1995), which presents a direct relation between the die-off rate and the environmental conditions (irradiation, salinity and temperature). The model was further extended to include an additional formulation to simulate the die-off rate of fecal coliforms dependent on water temperature (Servais et al., 2007,; Brauwere et al., 2011):

$$k_d = k_{20} \frac{\exp\left(-\frac{(T-25)^2}{400}\right)}{\exp\left(\frac{25}{400}\right)} \quad (1)$$

where $k_D$ is the die-off rate ($s^{-1}$), $k_{20}$ is the die-off rate at 20 ºC ($1.25x10^{-5}$ $s^{-1}$) and $T$ is the water temperature (ºC). Numerically, the coupled model is based on a finite-element/finite-volume numerical method. The domain is discretized horizontally with unstructured triangular grids, and vertically with hybrid coordinates (S-coordinates and Z-coordinates), allowing for high flexibility in both directions. The model is fully parallelized using Message Passing Interface (MPI) which improves the computational times (Rodrigues et al., 2009).

## Oil spill risk systems for coastal management

Natural and man-made disasters worldwide, like oil spills, may induce damage of catastrophic proportions, affecting both human lives and property (Dietrich 2012). Even if the accidents and their consequences are unpredictable, many prevention and mitigation actions can be taken to reduce the severity of the disaster effects. These problematics have motivated the creation of specific advanced tools to assist emergency agents in such environmental disasters, taking into account the various forcing mechanisms and relevant processes of oil spills in estuarine and marine environments.

Building coastal nowcast-forecast systems that detail all processes of oil spill evolution and pathways can improve their importance as a real effective asset for coastal managers, oil companies and consulting firms. Indeed, they can significantly improve local contingency plans and contribute to effective emergency management and early warnings issue to protect assets at risk in the event of accidents.

The forecast of oil spills within the WIFF platform is done with VOILS (Azevedo et al., 2014). VOILS is a 2D/3D Eulerian-Lagrangean transport and weathering oil spill model. All the major oil weathering processes are implemented, such as: advection, spreading, evaporation, emulsification, dispersion, dissolution and shoreline retention (including oil reposition). VOILS also takes advantage of unstructured horizontal grids to efficiently represent complex coastal boundaries, enhancing the capabilities of the modeling system for cross-scale applications that can range from ocean to coastal zones, estuaries and ports. Besides regular outputs, VOILS also produces several indicators for risk management. Examples are exposure time, time for first oil arrival, oil density and viscosity evolution, evaporation and emulsification rates, oil mass and shoreline retention areas.

The structure of the VOILS system was vaguely inspired in MOSM (Multiphase Oil Spill Model - Tkalich, 2006). The domain is divided in two layers: surface and water column. The surface slick thickness is determined by solving the layer-averaged Navier-Stokes equations and the 3D advection-diffusion equation is used to simulate oil dynamics in the water column. The 3D baroclinic shallow-water equations are solved first with SELFE, thereby providing velocities, water levels and the vertical eddy diffusivities required to solve the transport models. Then, the surface transport equation is solved by using the 2D transport

model. The surface processes advection, spreading, evaporation, water-in-oil emulsification and shoreline retention are solved in the VELA model with the oil weathering model incorporated. Finally, in 3D mode runs, the transport equations for the water column are solved with the tracers transport module incorporated in SELFE.

The Surface slick Model (modified VELA model) consists in the coupling of the VELA transport model (Oliveira and Fortunato, 2002) with an oil weathering module. The oil weathering module includes the most relevant processes that occur during an oil spill event (e.g. spreading, evaporation, water-in-oil emulsification, shoreline retention, dispersion, dissolution and sedimentation)

This model uses an Eulerian-Lagrangian scheme to solve:

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x_i}(hu_i) - \frac{\partial}{\partial x_i}\left(D\frac{\partial h}{\partial x_i}\right) = R_h \qquad (2)$$

Here, h is the slick thickness; $u_i$ are the horizontal velocity components (obtained from the circulation model); $R_h$ are the physical-chemical kinetic terms; D is the slick spreading-diffusion function; t is the time; $x_i$ are the horizontal Cartesian coordinates.

The transport equation (2) is decoupled in four equations: the advection and the horizontal velocity gradients equations are solved using a Lagrangian approach (Backward Method of Characteristics); the sources and sinks, in which the oil weathering algorithms are implemented, are determined at the foot of the characteristic lines; finally the diffusion equation is solved using a finite volume method in a time-space framework.

The water column model is based on model SELFE. SELFE (http://www.stccmop.org/CORIE/modeling/selfe; Zhang and Baptista, 2008) is an open-source unstructured-grid model for 3D baroclinic circulation. This model solves the 3D shallow-water equations with the hydrostatic and Boussinesq approximations, and transport equations for salt and heat. The numerical scheme of the model is based on a finite-element/finite-volume scheme. The advection in the momentum and transport equations is treated with Eulerian-Lagrangian methods. The backtracking can be solved using either an Euler scheme or a fifth-order embedded Runge–Kutta scheme. The advection of the transport equation can be treated with an Eulerian-Lagragian method (ELM), a finite-volume upwind scheme or a TVD (Total Variation Diminishing) scheme.

The water column oil model is coupled with the hydrodynamic model through a user-defined transport module which allows the user to solve the transport equation along the water column, for any number of tracers besides temperature and salinity. This module can be roughly described as an adaptation of the temperature and salinity transport algorithms to a generic tracer. In SELFE, the tracer's advection can be solved through an upwind or a TVD schemes. The oil weathering component includes the processes of emulsification, dispersion and dissolution. The VOILS system used herein is based on SELFE version v1.5k2.

## Dam safety control/Numerical simulation of the structural behavior of concrete dams

Dams are constructed for specific purposes, mainly for water supply, power production, flood control, navigation, irrigation and sedimentation control. As large structures, they are exposed to high risks, due to the high impact that can be produced from structural failures, which might drive to catastrophic consequences, including: loss of life (reduction of loss of life is the top priority of emergency planning); environmental damage; property damage (e.g., dam flood plain); damage of infrastructure; loss of power plant and electricity production; socio-economic impact; political impact, etc. As a consequence, the structural safety is a key factor during the entire life of a dam, from its initial design to the final abandonment and demolition.

The safety of the system formed by the concrete dam, the foundation, the reservoir and the downstream area of the dam should be evaluated on its structural, hydraulic, operational and environmental components. The interpretation of the observed structural behavior is usually based on the establishment of correlations between the loads, the structural properties (including material properties) and the structural responses. These responses, expressed in terms of displacements, strains, drained flow, etc., are compared with predicted values generated by behavior models.

The surveillance activities of concrete dams have a preventive purpose, to make timely decisions to prevent or minimize the consequence of an abnormal behavior. These activities include the necessary actions to understand the real condition of the dam, and in cases of abnormal behavior, to adopt appropriate measures, including the conditioning of the exploitation, the construction activities for conservation and rehabilitation, or in an extreme case, to warn civil protection authorities in order to protect people and property.

Dam safety control is performed through three main activities:

- Monitoring: includes the sensor gathering, but also visual inspections and analyses to the physical and chemical properties of materials;
- Analysis and Interpretation: to create information from the data captured in the monitoring activity. This analysis and interpretation intends to improve the overall knowledge about the dam behavior and optimize the models that can be used to characterize the structural behavior system, which is a complex system composed by the dam, its reservoir, and foundations;
- Safety Assessment and Decision Making: to evaluate the overall safety and support decision making, based on hypothesis tests, models exploitation (in time and space), as well as specialized alerts and notifications that can be taken into account in order to take adequate measures to correct any anomaly or reduce its impact in time.

Indeed, the analysis of data measured by the monitoring systems and their comparison with statistical, physical and mathematical models is critical for safety control assessment.

The analysis of the behavior of large structures, which is of paramount importance for their design and maintenance, demands either experimental or numerical modelling of those structures. As a result of the fast-paced evolution of computational systems' capacity and availability, numerical models are currently the main tool in structural behavior research.

Structural behavior analysis of concrete dams at LNEC relies on numerical models based on the finite element method (FEM). A tridimensional FEM model is applied to the time-dependent thermal, hygrometric and mechanical behavior of such dams.

With such purpose, a viscoelastic model has been developed, making it possible to analyze the structural behavior of concrete dams or other concrete structures subject to swelling, by considering the different load histories during their lifetime (Coelho et al., 2014). The discretization of load histories allows us to consider:

- the action of the dead weight of concrete;
- the actions corresponding to variations in the reservoir water level (hydrostatic pressures and uplifts);
- the thermal actions corresponding to seasonal air and water temperatures, by taking into account the influence of solar radiation; and
- the free swelling of concrete.

In order to model the concrete time behavior, considering the concrete maturing process, the incremental constitutive relation corresponding to a Kelvin chain is used. By assuming the hypothesis of the material being isotropic and assuming that the Poisson ratio is constant over time, the constitutive relation for 3D equilibrium is:

$$\varepsilon(t) - \varepsilon^{0}(t) = \int_{t_0}^{t} J(t, t_0) C d\, \sigma(t_0) \tag{3}$$

where,

$$C = \begin{bmatrix} 1 & -\upsilon & -\upsilon & & \\ -\upsilon & 1 & -\upsilon & & \\ -\upsilon & -\upsilon & 1 & & \\ & & & \Xi & \end{bmatrix}_{6x6}, \quad \Xi_{3x3} = \begin{bmatrix} 2(1+\upsilon) & & \\ & 2(1+\upsilon) & \\ & & 2(1+\upsilon) \end{bmatrix}, \quad \upsilon \text{ Poisson' s ratio} \tag{4}$$

The time period of the analysis is discretized in time intervals $\Delta t_r = t_r - t_{r-1}$. Applying the previous equation to the extremes of interval $\Delta t_r$ and assuming the time intervals to be small, so as to have the elastic modules $E(t)$ and $E^i(t)$ approximately constant and equal to their values in the mean instant of the interval $\Delta t_r$, i.e. $E(t) = E_{(r-\frac{1}{2})}$ and $E^i(t) = E^i{}_{(r-\frac{1}{2})}$, it follows that

$$\Delta\varepsilon_r - \Delta\varepsilon_r^0 = \frac{1}{E_r^*} C \Delta\sigma_r + \Delta\varepsilon_r^* \tag{5}$$

or, as the constitutive viscoelastic equation in incremental form,

$$\Delta\sigma = E_r^* C^{-1}(\Delta\varepsilon_r + \Delta\varepsilon_r^0 + \Delta\varepsilon_r^*) \quad (6)$$

From the relation between deformations and displacements, it is possible to obtain, by time derivation, the kinematic relation as:

$$\dot{\varepsilon} = B\dot{u}_e \quad (7)$$

where $\dot{\varepsilon}$ is the vector of deformation velocities and $\dot{u}_e$ us the vector of velocities in nodal points of each finite element.
On the basis of the constitutive viscoelastic incremental relation, the vector of stress derivatives is given by:

$$\dot{\sigma} = E^*(t)C^{-1}(\dot{\varepsilon} \quad + \dot{\varepsilon}^0 \ + \dot{\varepsilon}^* \ ) \quad (8)$$

Using the principle of virtual works, the following equilibrium equation is obtained for each finite element:

$$K^e(t)\sigma_e = (\dot{f}^a)_e + \dot{(\dot{f}^0)}_e + (\dot{f}^*)_e \quad (9)$$

where $K^e(t)$ is the elementary stiffness matrix, given by:

$$K^e(t) = \int_V B^T E^*(t) C^{-1} B dV \quad (10)$$

The vectors $(\dot{f}^a)_e$, $(\dot{f}^0)_e$ and $(\dot{f}^*)_e$ correspond, in each finite element, to the time derivatives of nodal forces equivalent to applied loads, to prescribed strains and to the effect of the load history, respectively.
For small time intervals $\Delta t_r$ the time derivatives can be approached by incremental relations $\dot{\sigma} \cong \dot{\frac{\Delta\sigma_r}{\Delta t_r}}$ and $\dot{f} \cong \dot{\frac{\Delta f_r}{\Delta t_r}}$. Therefore the equilibrium equation can be rewritten for the entire structure, in the incremental form:

$$K(t) = \Delta f^a + \Delta f^0 + \Delta f^* \quad (11)$$

This model serves as the basis for the implementation of the numerical solution that will be considered. The core of the computation is solving this equation, which is done by establishing a separation between instantaneous and delayed responses (the *phase* of the computation), and considering that the increments due to loads, $\Delta f^a$ and $\Delta f^0$, occurring at an interval $\Delta t_r = t_r - t_{r-1}$ are applied at instant $t_{r-1}$ and remain constant in the interval.

Subsequently, the system is solved for the increment in displacements $\Delta u_r^{(inst)}$:

$$\Delta K(E_{r-1/2})\Delta u_r^{(inst)} = \Delta f^a + \Delta f^0 \qquad (12)$$

Having obtained $\Delta u_r^{(inst)}$, both the stress increments $\Delta \sigma_S^{(inst)}$ and the corresponding strain increments $\Delta \varepsilon_S^{(inst)}$ are computed. Then, the nodal forces due to swelling actions are computed again until the variations in the stress field are sufficiently small so as to fulfil the convergence conditions and by ensuring the equilibrium and compatibility conditions.
The analysis of the delayed behavior begins with the computation of the global stiffness matrix $\Delta K$ for $E = E_r^*$. Then, the vector of nodal forces equivalent to the effect of the load history $\Delta f_r^*(\Delta \varepsilon^*)$ is calculated and subsequently the equation system is solved:

$$\Delta K(E_r^*)\Delta u_r^{(del)} = \Delta f^* \qquad (13)$$

Thus $\Delta u_r^{(del)}$ and the total increment in displacements corresponding to the interval $\Delta t$ are obtained. This enables computing the stress and strain increments corresponding to the viscoelastic behaviour of concrete.

## Sensor data acquisition and processing/Multilinear regression

According to the Portuguese Dam Safety Legislation (RSB, 2007), LNEC is responsible for keeping an electronic archive of data concerning the dam safety and maintaining an updated knowledge about the behavior of dams. The behavior of dams is continuously monitored by multiple sensors (e.g., plumblines) installed in strategic points of the dam structure. Typically, a concrete dam is monitored by hundreds to few thousands of sensors. Raw data, usually known as readings, are manually collected by human operators or automatically collected (automatic monitoring systems), and transformed into engineering quantities by specific algorithms. The dam safety monitoring information includes, essentially, sensor properties, readings and engineering quantities. All these data must be sent and centralized at LNEC in order to produce a mandatory electronic archive.
Data gathering applications and a Web Information System are responsible to support the data acquisition and transformation processes. It is known that the variations of hydrostatic pressure and temperature are the main actions that must be considered when analyzing the physical quantities generated by the monitoring systems. As a consequence, multiple linear regressions (MLR) can be used to determine their relationship with the expected response (physical quantity).
In large dams, the expected response is approximated by the following effects: (i) elastic effect of the hydrostatic pressures; (ii) elastic effect of temperature, depending on thermal conditions; and (iii) time effect (considered irreversible). This is an example of a statistical

and scientific calculation workflow that is commonly implemented using the R Project for Statistical Computing[1].

## 2.2 TECHNICAL DESCRIPTION OF THE USE CASES

### Water Information Forecast Framework for coastal applications

WIFF runs on Linux operating system, and its core is composed by a set of scripts, scheduled to run periodically, which prepare and launch the simulations for each forecast model (Figure 1). The scripts interact with a database to retrieve input data to force the models, including river flows, water temperature and atmospheric data.

WIFF's main process begins by setting up the environment of the current simulation, which involves creating the folder structure that will contain all information required to run the simulations, including the preparation scripts configurations. Simulation requirements include the state of the previous run (and/or other forecast models simulation results in the ensemble mode), forecasts from regional circulation models and atmospheric models, and data from field sensors. Most of this information is retrieved from databases and used on-the-fly during the simulations.

After running all simulations the output results are processed, using visualization tools such as VisTrails or the matplotlib library, to generate model forecast products and data/model comparisons. The end-user can visualize all the products and interact with the given results, using the WIFF user interface (UI). This UI is basically a customized deploy of Drupal, a PHP-based Content Management System (CMS), that is used to access model metadata, status and products. Map server (Geoserver) with Web Map Services (WMS) support has been added to WIFF, allowing geospatial placement of products, as well as model output query capabilities. With the goal of a more usable UI, a Web Geographic Information System (WebGIS) was developed in Flex, using the OpenScales library to handle geospatial information. This WebGIS is built in a modular and generic way, allowing not only the visualization of WIFF data, but also spatial data from other sources.

WIFF products are reused in the WebGIS through georeferencing methods and their deployment in the Geoserver is automatic. This deployment is done via a set of scripts, which convert TIFF products into GeoTIFF, and then deploy these data into Geoserver. WMS layers are served by Geoserver and displayed in the WebGIS, projected on top of a Bing Map Layer. This UI allows the user to intuitively manipulate existing products, offering, among others, zooming capabilities and a slider which allows easy navigation through model simulation results in a given interval, the capacity to probe on model forecast results and the ability to set simulations on-demand.

[1] http://www.r-project.org

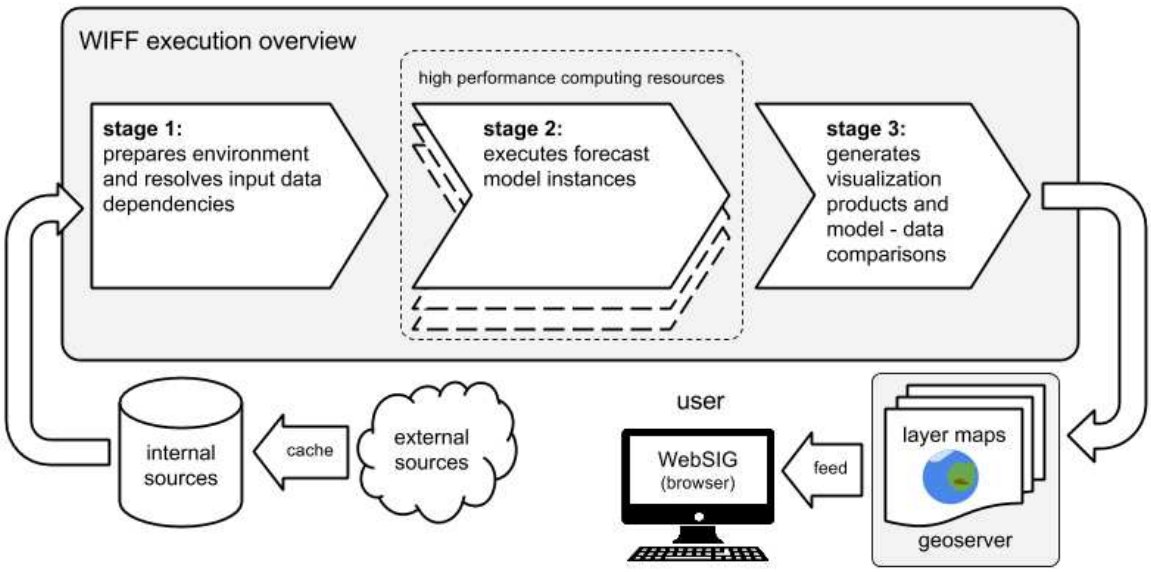


**Figure 1 - Workflow for the Water Information Forecast Framework (adapted from Rogeiro et al., 2014)**

## Fecal contamination early-warning system in the Tagus estuary

The Tagus estuary hydrodynamics and fecal contamination are simulated using ECO-SELFE 3.1d, considering the simulation of two fecal bacteria tracers: *Escherichia coli* (*E. coli*) and fecal coliforms.

The model setup derived from the application of Costa et al. (2012). A detailed description of the Tagus estuary hydrodynamics and fecal contamination model setup, calibration and validation can be found in Rodrigues et al. (2013). The Tagus estuary model was further extended from this application to account for the relevant urban discharges in the waterfront Algés-Alcântara-Terreiro do Paço and the horizontal grid was improved for a detailed representation of this area. The domain is discretized with a horizontal grid with about 54000 elements and 29000 nodes (Figure 2) and a vertical grid with 20 SZ levels (15 S levels and 5 Z levels).

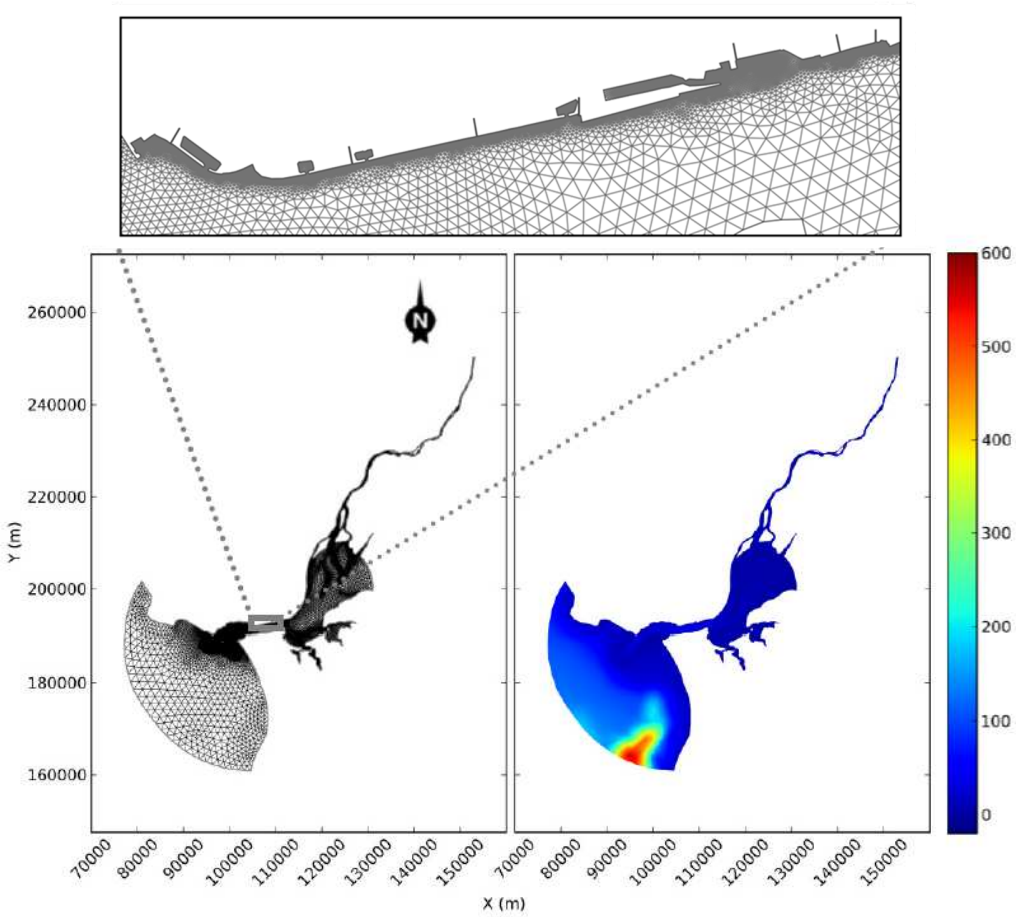


**Figure 2 – Horizontal grid and bathymetry (in meters relative to Mean Sea Level, MSL), with detailed view of the Alcântara outfall and nearby outfalls.**

The forecast simulations entailed the complex workflow described in Figure 3.

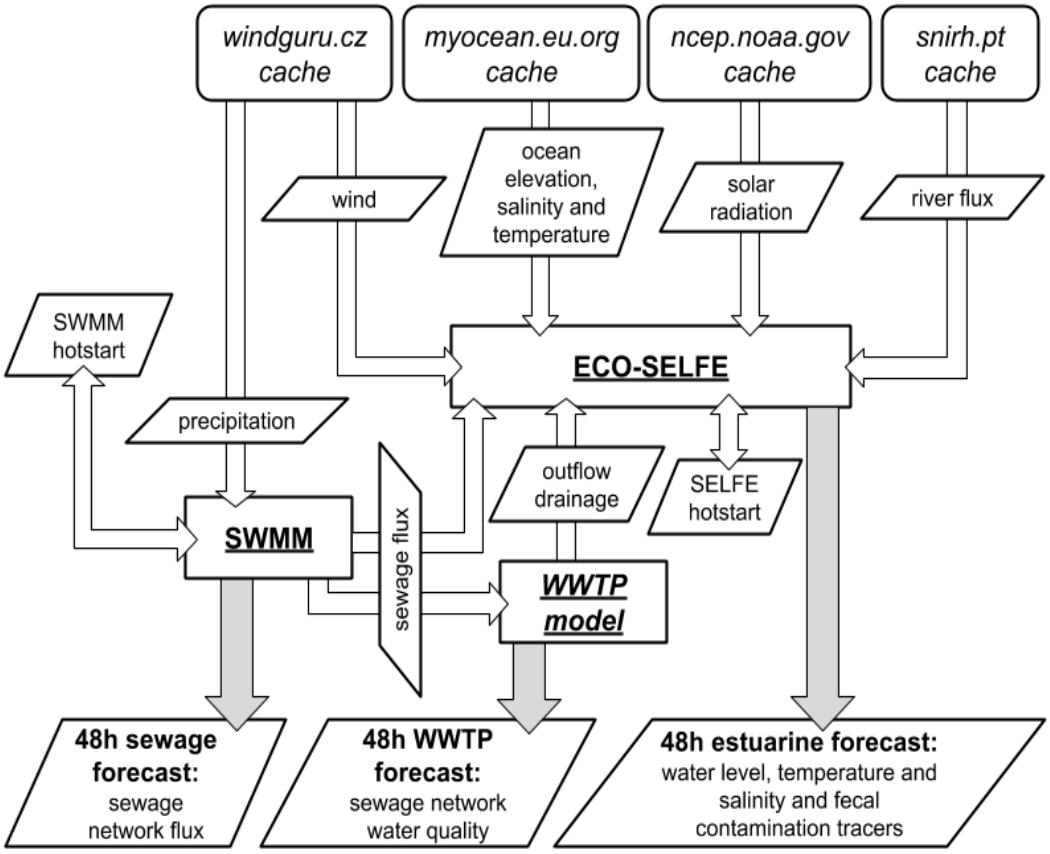


**Figure 3 - Workflow for numerical modeling components for fecal contamination model: steps 1 and 2 of the WIFF workflow (adapted from Rogeiro et al., 2014)**

## Oil spill risk management system

The complete workflow of an oil spill forecast simulation for a specific site of the Portuguese coast was implemented in the Aveiro lagoon. This workflow is composed by four main blocks, the first two being the object of the present analysis:

1) The first block of the workflow corresponds to the determination of the hydrodynamic forecast for the coastal area with SELFE model. The model receives as inputs the meteo-oceanographic conditions and the tide and river inputs that are imposed in the open boundaries of the simulation grid (Figure 4). SELFE model provides the forecasts for the next two following days.

The hydrodynamic simulations will be executed with SELFE model (Zhang and Baptista, 2008 – available at http://www.stccmop.org/CORIE/modeling/selfe/) in 2D mode. The implementation of the SELFE model to the Aveiro lagoon is based on the work of Rodrigues et al. (2011).

2) The second block of this workflow serves to calculate the transport and transformation of 6 pre-defined oil spills in the study region. The model VOILS (Azevedo et

al., 2014) receives the outputs of the hydrodynamic model, from Block 1, and calculates the transport of the oil at the water surface, and all the oil weathering processes that act upon the oil during a spill event.

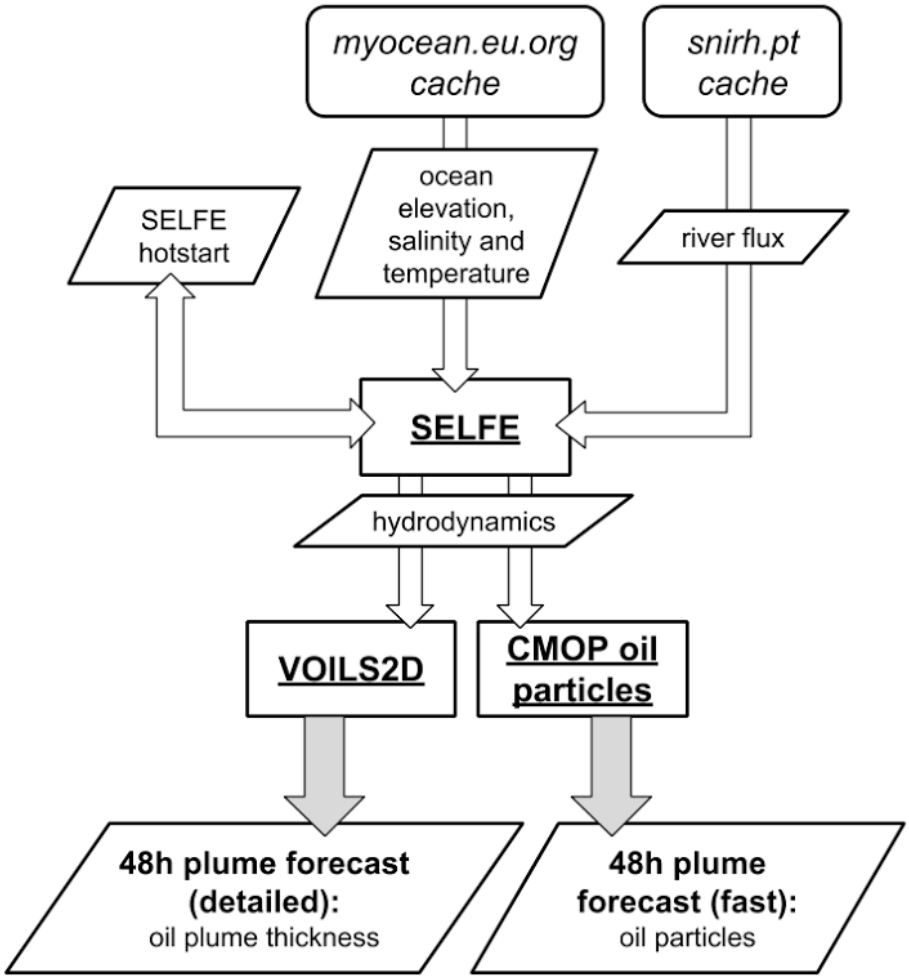


**Figure 4 – Workflow for numerical modeling components: steps 1 and 2 of the WIFF workflow (adapted from Rogeiro et al., 2014)**

3) The third Block of the workflow consists in the creation and georeferencing of the output images generated by the hydrodynamic and oil spill models. This task is performed with scripts developed at LNEC and also GDAL programs. Each hydrodynamic simulation provides approximately 96 maps, in which are included the velocity, sea surface elevation, salinity and temperature horizontal fields. Regarding the oil spill model, each simulation produces a trajectory map for every time step and an exposure time map for each run. The tiling procedure must be performed for every map produced by the numerical simulations.

4) The last block is composed by the webservices of the forecast platform (WIFF), which allow the final user to access the georeferenced products.

## Dam safety control/Numerical simulation of the structural behavior of concrete dams

A subset of the scientific applications being used at LNEC to support structural safety assessment as described in section 2.1 was selected to be included in the pilot cloud test portfolio. An overview of their characteristics is shown in Table 1.

**Table 1- Test scenarios – structures.**

| Application | Language | OS | Details |
|---|---|---|---|
| Finite Element Method (FEM) analysis | C, bash | Linux | Parallel program (MPI) |
| Statistical analysis | R | Any | |

The first application (for Finite Element Method (FEM) analysis) addresses the problem of numerically simulating the structural behavior of concrete dams over time, considering the impact of hydrostatic pressure, temperature variations and concrete swelling.
The second application is used to support the statistical interpretation of sensor data.
In order to run the test scenario's applications in the cloud environment, it is important to thoroughly detail how they work. This section's purpose is to specify, for each test scenario:

i. Requirements: which input data and runtime conditions are needed;
ii. Workflow: how the simulation is processed and how it is executed;

Results: once the workflow has been carried out, what outputs are expected and how do we assess their quality.
(i) Requirements
The inputs that must be provided to these models and many of the deployments details are common to all three numerical simulation applications.
The following inputs must be provided:

i. Simulation mesh
    a. Number, type and physical properties of elements
    b. Number and physical coordinates of nodes
    c. Element-node correspondence table
ii. Simulation details
    a. Timespan and discretization of time steps
    b. Actions to be considered (e.g. hydrostatic pressure, concrete swelling)
    c. Action-specific measurements (e.g. water height, temperature)

In addition, the following run-time conditions are required for each application:

- TF2:
    - standard gcc compiler
    - MPI and PETSc libraries
    - can be run in 1 or more processors

TF2 consists of a main body written in C along with a few pre- and post-processing bash scripts.

(ii) Workflow

The main simulation process, which is repeated for every time step, is described in Figure 5.

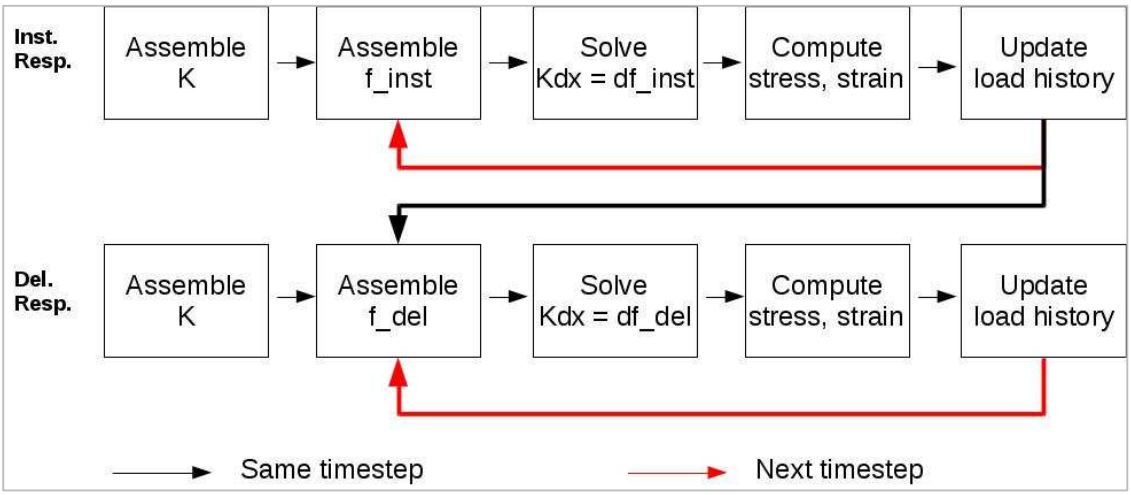


**Figure 5 – Numerical simulation workflow.**

Once the application is run, the simulation will be computed according to the supplied input parameters without any interaction with the user. Therefore, once set in motion, the program will go through the workflow described in Figure 5 without any further modifications; complete results will be available once all time steps have been iterated through.

(iii) Results

The application generates the following results:

- for each mesh element
  - the symmetric 3x3 stress tensor
  - the symmetric 3x3 strain tensor
- for each mesh node, a tridimensional displacement vector

These outputs are created at the end of both phases and for every simulation time step.

## Sensor data acquisition and processing/Multilinear regression

(i) Requirements

- Sensor data generated based on a set of extraction parameters

(ii) Workflow

Figure 6 details a multiple linear regression (MLR) process used in dam safety. For demonstration purposes, this process was isolated from the generic information system. Overall, the process includes 5 steps:

- Extract data: based on a set of extraction parameters, this process generates the sensor data that will be used in the MLR model;

- Generate regression: based on a set of regression parameters (e.g., equation to estimate the hydrostatic effect), this process generates the regression controls that configure the parameters for the MLR model;
- Execute regression: this process executes the regression parameterized in the regression control, using the dataset generated in the extract data process;
- Generate aggregation: since a dam has a large number of sensors and a regression is used for each physical quantity associated with each sensor, we might need to run hundreds or thousands of regressions. Thus, the process is capable to aggregate all MLR executions into one aggregated report. This step generates the controls that define how this data is aggregated;
- Produce report: collects all the results produced by the several executions of MLR models and compiles them into a single report.

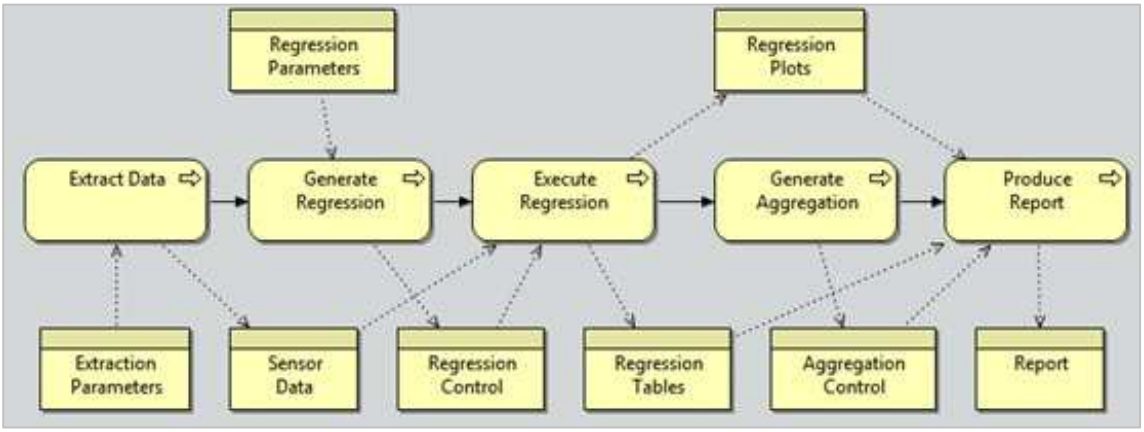


**Figure 6 – Multiple linear regression process in dam safety**

(iii) Results

This process outputs the following results:

- Regression plots
- Regression tables

# 3 DESCRIPTION OF THE INFRASTRUCTURES USED FOR THE PERFORMANCE ANALYSIS

## 3.1 BASELINE SYSTEMS FOR COMPARISON

The several workflows under testing in the cloud are already running in different environments, from local resources to HPC cluster nodes. Below is a description of the main characteristics of these computing environments.

### Local workstations

The Hydraulics and Environment Department of LNEC holds an IT datacenter where several workstations and servers are used for daily forecast runs and model products processing. In this study two distinct resources were used: Hydra and Luna. An Intel laptop was also used as a baseline for structural simulation and MLR analysis. Details are available in Table 2.

### Medusa cluster/INGRID/INCD

MEDUSA is one of the clusters integrated in the Portuguese Science Foundation Digital Research Infrastructures Roadmap and the Portuguese Grid network. The computational resources of this cluster include: the original Medusa and the two recent nodes: Medusa_it and Medusa_Intel. Details are available in Table 2.

## 3.2 PILOT CLOUD COMPUTING RESOURCES

The present pilot cloud provides an infrastructure as a service (IaaS) solution using OpenStack [30] software. Two distinct cloud deployments, named Aurora and Nimbus, were used in conjunction with KVM hypervisors. Details are available in Table 2.

**Table 2- Computational Resources Description**

| Resource Name | Resource Type | Processing | Memory | Operating System |
|---|---|---|---|---|
| luna | workstation | 6 virtual cores on<br>2 × Xeon X5650<br>2 × 6c @ 2.66Ghz | 5 GB | Scientific Linux 5 |
| hydra | workstation | 8 virtual cores on<br>2 × Xeon E5645<br>2 × 6c @ 2.4Ghz | 16 GB | Scientific Linux 6 |
| aurora | cloud VM | 8 virtual cores on<br>2 × Xeon E5440<br>2 × 4c @ 2.83Ghz | 8 GB | CentOS 7.1 |
| medusa | cluster node | 268 cores on<br>67 × 2 × Opteron 280<br>67 × 2 × 2c @ 2.4 Ghz | 67 × 4 GB<br>(268 GB) | Scientific Linux 6 |
| medusa_it | cluster node | 32 cores on<br>4 × Opteron 6220<br>4 × 8c @ 3Ghz | 256 GB | Scientific Linux 6 |
| Intel laptop | PC | 2 cores on Intel Core2 Duo P8600@ 2.40 GHz | 4Gb | Fedora 20 |
| medusa_intel | cluster node | 12 cores on<br>2 × Xeon E5-2620<br>2 × 6c @ 2 Ghz | 32 GB | Scientific Linux 6 |
| nimbusA | cloud VM | 24 virtual cores on<br>2 × Xeon E5-2680 v3<br>2 × 12c @ 2.50GHz | 98 GB | CentOS 7.1 |
| nimbusB | cloud VM | 24 virtual cores on<br>2 × Xeon E5-2650L v3<br>2 × 12c @ 1.80GHz | 98 GB | CentOS 7.1 |

Aurora is based on OpenStack Havana running on blades with 8 Intel CPU cores and 24 GB of RAM. The blades are connected via gigabit Ethernet. A virtual machine (VM) configuration flavor with 8 vCPUs and 8 GB of memory was used. The communication between virtual machines was implemented via the OpenStack Neutron component using tenant networks based on virtual switches and GRE encapsulation over Gigabit Ethernet.

**Table 3- Aurora Pilot cloud's different instances flavors**

| Name | m.tiny | m.small | m.medium | m.large | m.xlarge |
|---|---|---|---|---|---|
| # of cores | 1 | 1 | 2 | 4 | 8 |
| RAM | 512 MB | 2 GB | 4 GB | 8 GB | 16 GB |
| Capacity (GB) | 5 | 20 | 40 | 80 | 160 |

Nimbus is based on OpenStack Kilo running on servers with 24 Intel CPU cores. These servers come in two types. The Nimbus A type has 512 GB and Intel Xeon E5-2680v3 CPUs, while the Nimbus B type has 198 GB and Intel E5-2650L v3 CPUs. The servers were interconnected via 10 Gigabit Ethernet using Intel X710 network cards. A virtual machine flavor with 24 vCPUs and 98 GB of memory was used to execute on both A and B types. The communication between VMs was again established via OpenStack Neutron virtual networks using virtual switches and GRE encapsulation. In addition a second network interface provided by Neutron through Single Root I/O Virtualization (SRIOV) was added to each VM. The SRIOV virtual function interfaces enabled direct access from the VMs to the 10 Gigabit Ethernet network bypassing the virtual switches and avoiding encapsulation. SRIOV is a standard specification from the Peripheral Component Interconnect Special Interest Group (PCI-SIG) that allows a PCI Express device to appear to be multiple separate PCI Express devices. These PCI devices are denominated virtual functions and can be provided to VMs via PCI pass-through. OpenStack Neutron is capable of orchestrating SRIOV virtual functions in combination with VLAN tenant networks. This setup was chosen to evaluate the performance benefits of SRIOV in a cloud multi-tenant virtualized environment.

# 4 EVALUATION ANALYSIS

## 4.1 DESCRIPTION OF EVALUATION PROCEDURE

For all cases the execution time is the metric used to evaluate the benchmarking performance. Additionally, the MPI tests allow producing more comparative metrics since there's another variable present, the resource size. Instead of comparing just execution times in different resources, it's also possible to further analyze how performance scales within different configurations.

To help with the scaling analysis, besides execution time, several other metrics such as global performance, process performance, performance gains and performance scaling, were also defined. While global performance and process performance are simple derivation products of the execution time and independent values, performance gains and performance scaling are intra-comparative metrics. These last two metrics allow comparing each one of the first two metrics over different configuration sizes. Table 4 details the extra metrics used in the comparative analysis.

**Table 4- Benchmark metrics**

| | |
|---|---|
| Global Performance (gp) | steps / time |
| Process Performance (pp) | steps / time / procs |
| Performance Gains | gpB / gpA |
| Performance Scalling | ppB / ppA |

## 4.2 EVALUATION FOR THE WATER INFORMATION FORECAST FRAMEWORK

### Fecal contamination early-warning systems for coastal management

In these tests ECO-SELFE was run in the pilot cloud, local workstations and the two most recent Medusa nodes (Medusa_it and Medusa_Intel).

The execution times for the fecal contamination use case indicate that running ECO-SELFE in the cloud resources does not affect its performance, at least not in a noticeable way (Figure 7). These preliminary results, using cloud virtual machines with resources equal to or lower than the physical bases, lead to comparable results between the cloud and the workstations, making the cloud a potentially attractive alternative under these conditions.

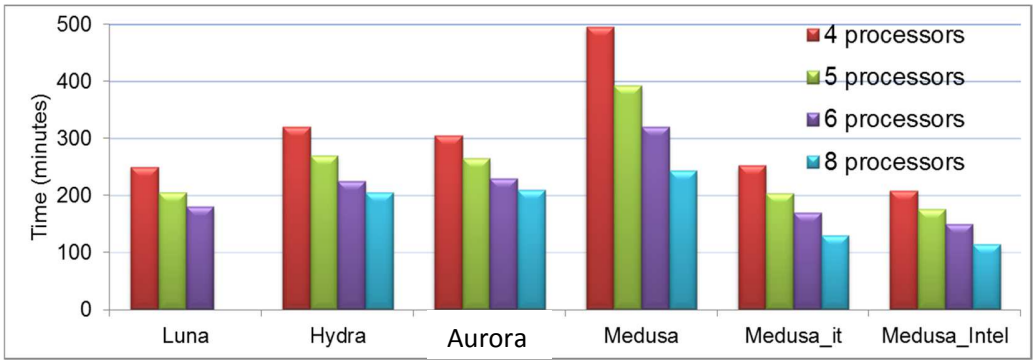


**Figure 7 – CPU time (minutes) for the several solutions with increasing processors number**

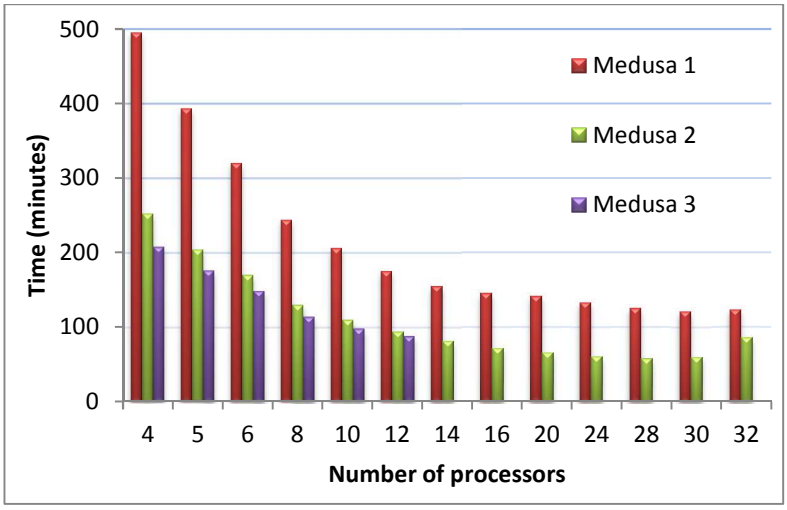


**Figure 8 – Optimal number of processors analysis for the three Medusa nodes**

However, the optimal number of processors for this application is larger than 8 units. The optimal analysis was then conducted using the grid cluster nodes medusa, medusa_it and medusa_intel, and nimbus cloud resources. For this particular case, the optimal performance is achieved close to 16 to 20 cores, depending on the resources (Figure 8). This value depends on the number of horizontal grid nodes and vertical layers in the model deployment, as analyzed in Costa et al., (2009), as well as on the computational resources size. As performance degrades when the resources are reaching the maximum available value, execution time could potentially be improved if more processing resources were available.

The global and process performance metrics help to understand how the increase in processing resources impacts performance and efficiency of this model. Using these metrics, the small gains associated with a larger number of processors become clearer and indicate that the optimal is far from the resources available in the pilot cloud (Figure 9).

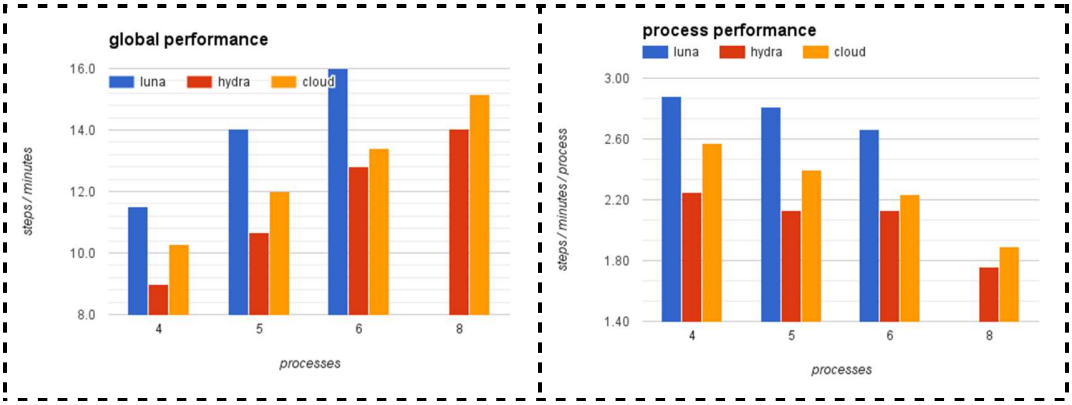


**Figure 9 – Global and process performance for ECO-SELFE simulations in the workstations and pilot cloud**

Indeed, the analysis of performance in the grid clusters shows that the best performance is achieved with 28 processors (Figure 10).

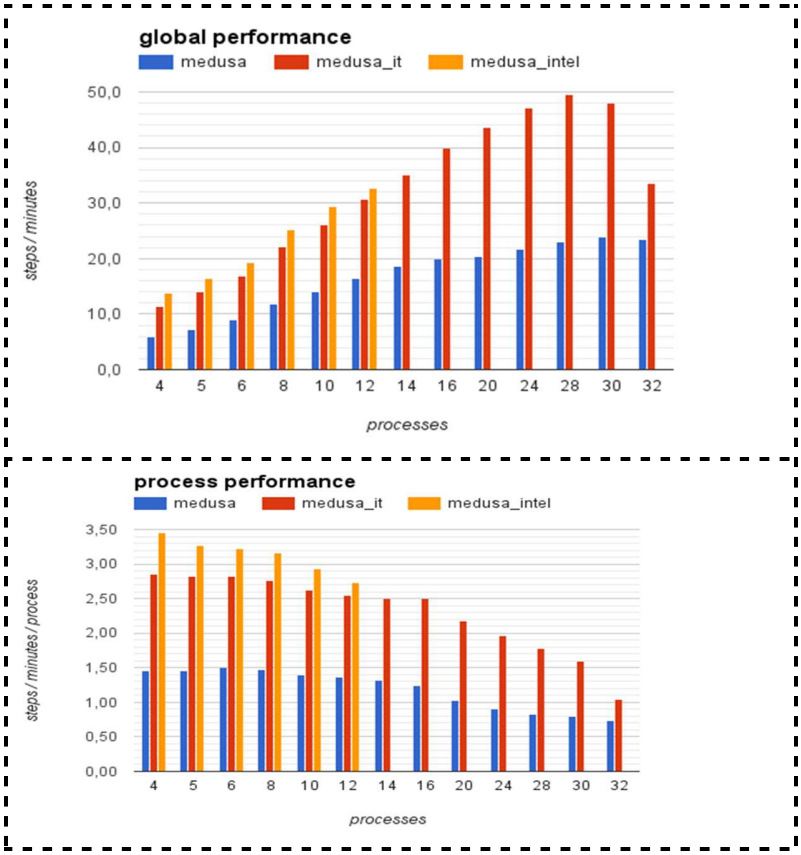


**Figure 10 – Global and process performance for ECO-SELFE simulations in the grid clusters**

Unlike performance metrics, which are used to assess performance and efficiency of each configuration individually, scaling metrics allow to compare the same aspects between any two configurations. They can be very useful at rating the relation between the evolution in performance and resource usage. For example, how much performance is gained by doubling the resources or what is the efficiency of adding 50% more resources. These metrics allow verifying quickly whether adding more resources is worthwhile.

Our results show that performance degrades much quicker in the workstations and cloud environments than in the grid clusters (Figure 11 and Figure 12).

In summary, for MPI-based models, results using cloud virtual machines with resources equal to or lower than the physical bases lead to comparable results between the cloud and the workstations. However, an HPC cluster remains the best option, mostly due to its specialized infrastructure (which offers optimized communication between memory and CPUs) and also to a larger number of cores available, which fit better the optimal configurations for the typical dimensions of computational grids for multi-scale (port to ocean) analysis.

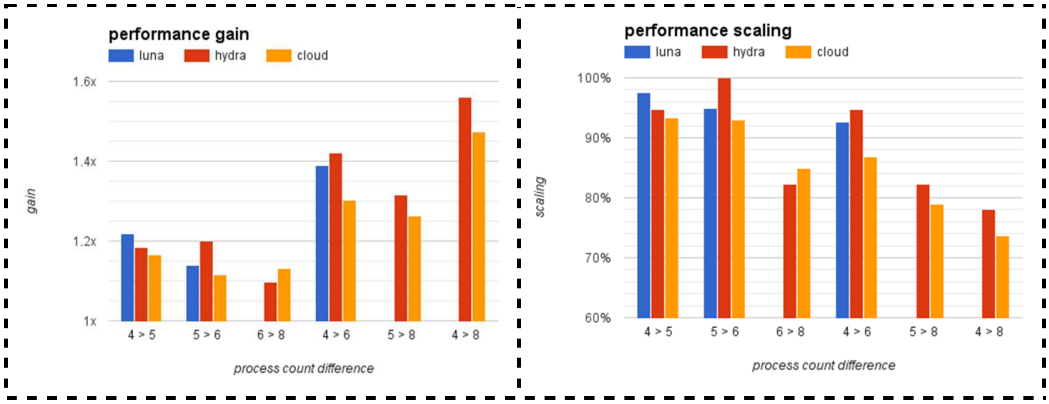


**Figure 11 – Performance gain and scaling for ECO-SELFE simulations**

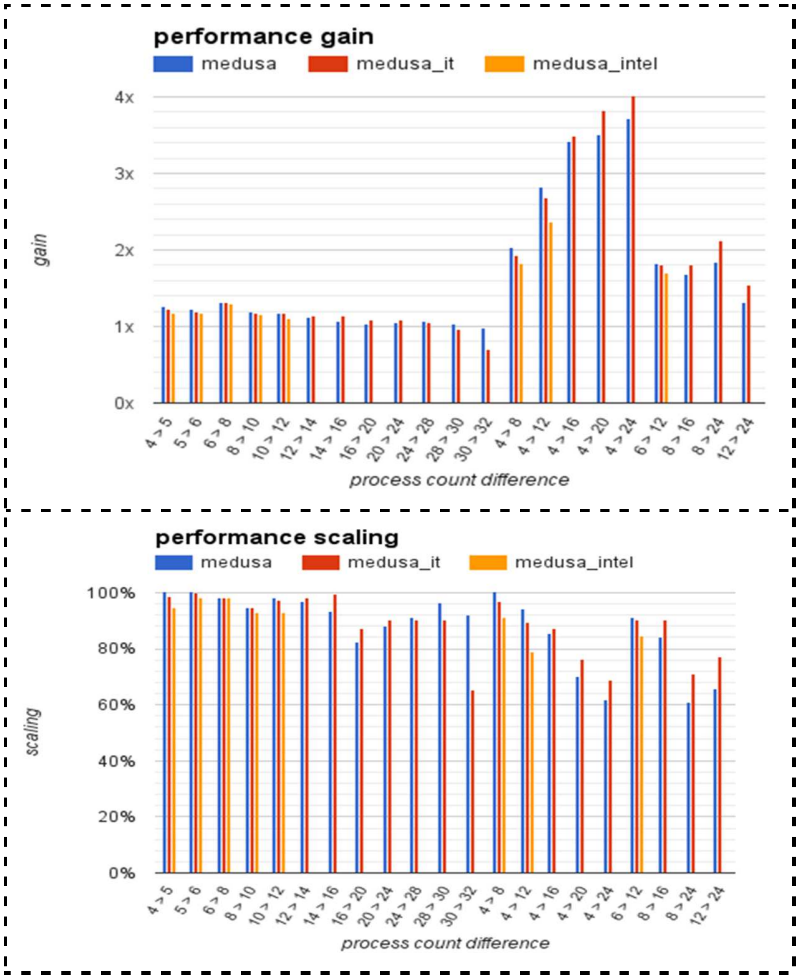


**Figure 12 - Performance gain and scaling for ECO-SELFE simulations in the MEDUSA cluster nodes**

As the optimal number of processors falls beyond the resources available at the Aurora cloud resources, the analysis above was extended to the nimbus cloud. Resources were used herein to search for the optimal number and compare it with the cluster medusa_it, to further verify the adequacy of cloud resources for this type of deployments. Furthermore,

we extend our analysis to investigate the usage of cloud federation (using MPI), with the nimbus cloud. Although the optimal number of processors for this application falls close to the limit of the physical machine behind nimbus, it is important to assess the importance of the overhead brought by cloud federation. Indeed, applications with larger computational grids would require the availability of a number of processors above the typical physical dimension of the hardware, rendering this performance assessment quite relevant for real-time operational simulations.

Federation results (Figure 10) show that the model scales very badly in such situations, even using virtual hardware assistance (SRIOV) for network communication instead of regular encapsulation. Only with smaller pools is it able to offer better results than single hosts with less resources, which in many cases can be the only way to achieve quicker executions. These results are caused by the way the model exchanges information between processes, sending huge amounts of tiny messages, which results in a high overhead to deal with that amount of messages. It is possible that using communication methods employing RDMA the overhead is considerably reduced, allowing for much better performance scaling with big resource pools.

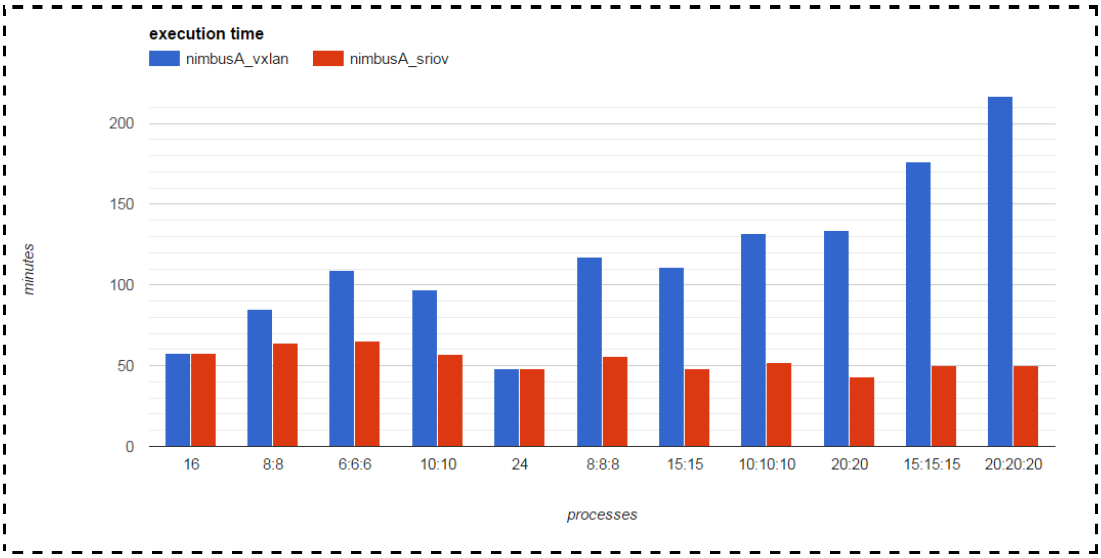


**Figure 13 - Execution time for cloud federation tests (X:X:X denote 3 hosts with X processes each).**

## Oil spill risk systems for coastal management

### Parallel runs of the hydrodynamic model: performance and scalability analysis

For the Aveiro lagoon setting and computational grid, a new MPI deployment of SELFE was run in the pilot cloud, local workstations and the two most recent MEDUSA nodes (medusa_it and medusa_intel), just for the hydrodynamic simulation.

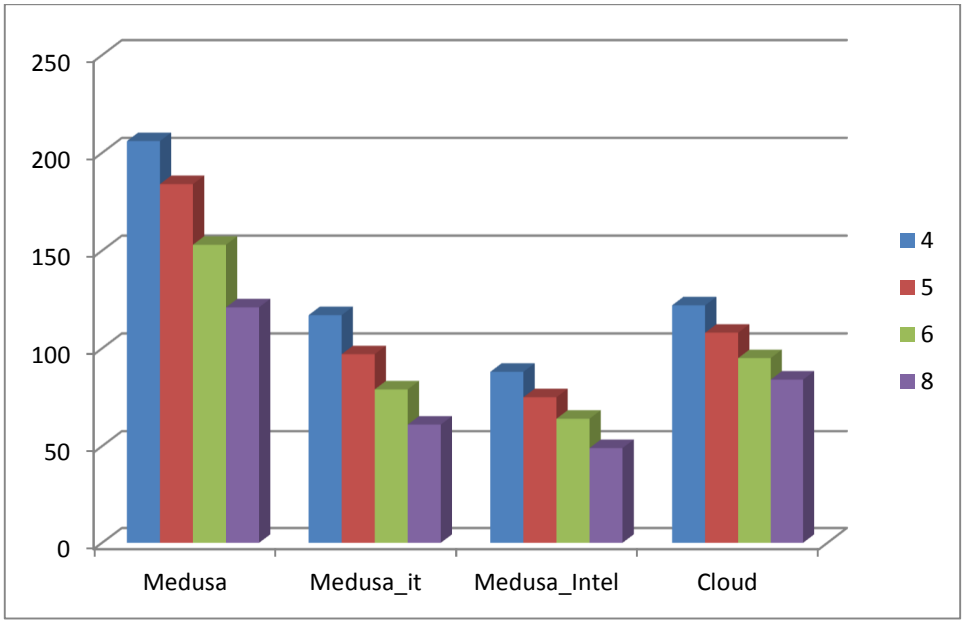


**Figure 14 – CPU time (minutes) for the several solutions with increasing processors number**

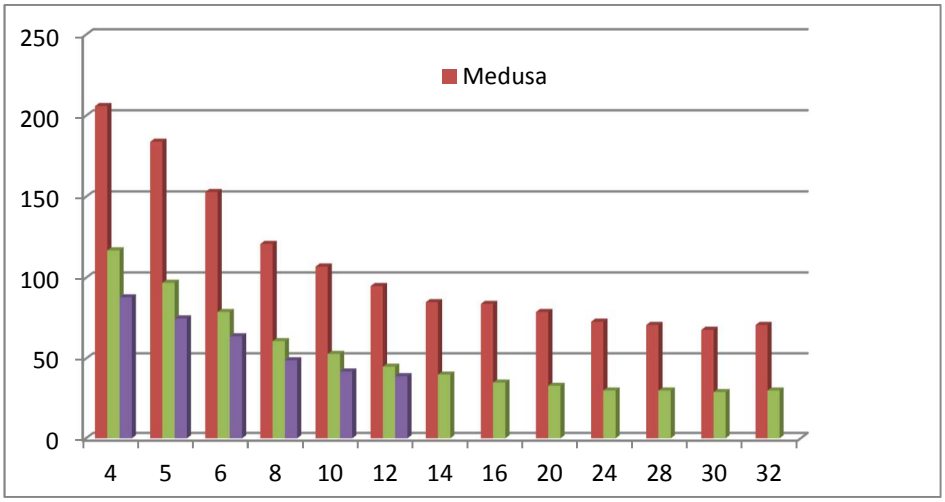


**Figure 15 – Optimal number of processors analysis for the three Medusa nodes (CPU time in minutes)**

The execution times for the hydrodynamic only simulations further show that running SELFE in the cloud resources does not affect its performance (Figure 14). Using cloud virtual machines with resources equal to or lower than the physical bases lead to comparable results between the cloud and the workstations.

For this application, the optimal number of processors for this application is also larger than 8 units. The optimal analysis was then conducted using the grid cluster nodes medusa, medusa_it and medusa_intel, leading to an optimal performance of 24 processors (Figure 15).

## Serial runs of the oil model: performance analysis

The serial model VOILS was run in the pilot cloud, local workstations and the two most recent MEDUSA nodes (medusa_it and medusa_intel), for six oil spill scenarios (Figure 16), dully repeated 5 times.

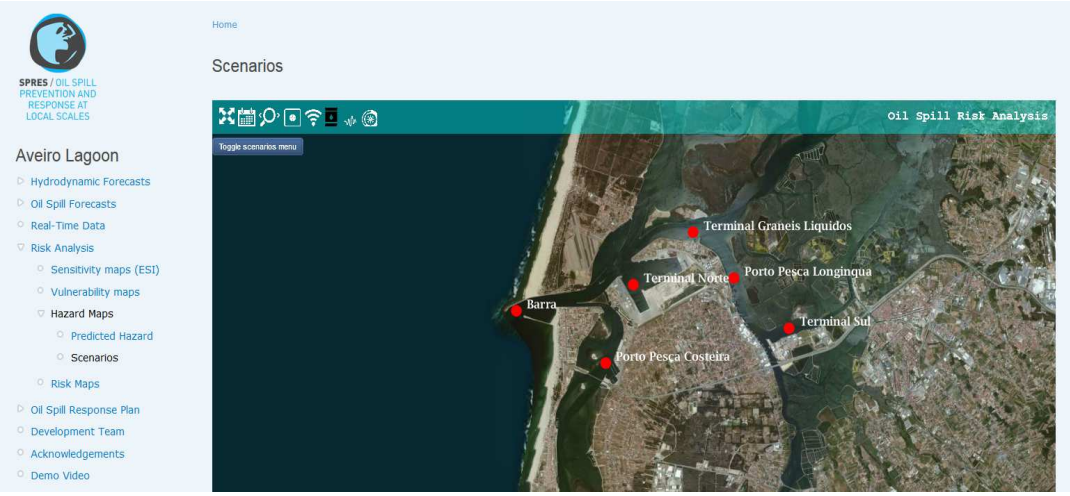


**Figure 16 – Initial oil spill locations for the six scenarios**

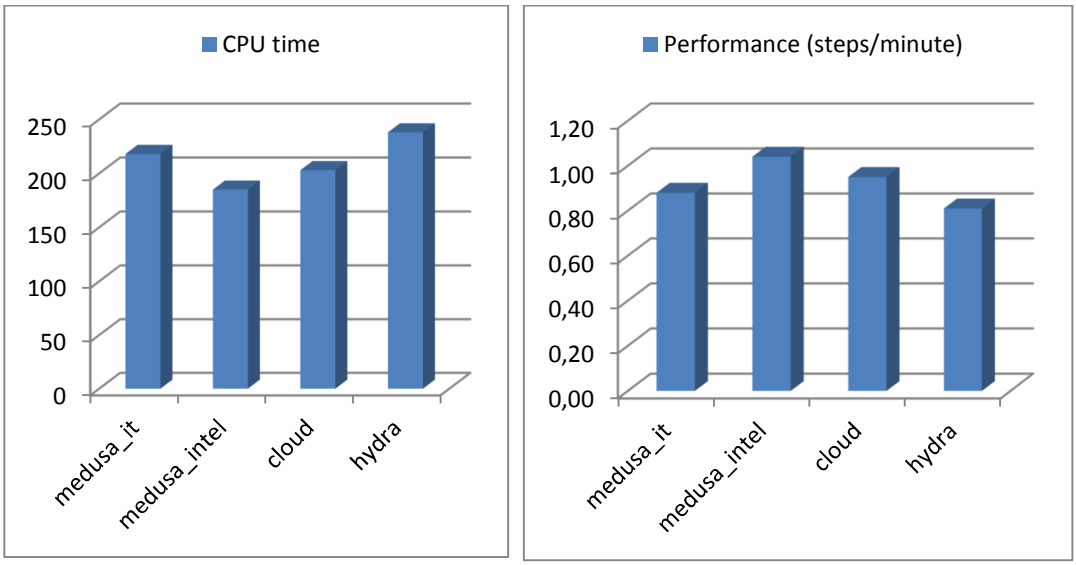


**Figure 17 – a) CPU time (minutes) and b) performance for the several solutions**

Results show that, taking into account the type of resources underneath each of the computational options, the pilot cloud leads to an execution time adequate to the purpose of oil spill forecasts (Figure 17). While execution time is smaller for the newer grid cluster nodes, the cloud is faster than the old cluster node and comparable to the workstations.

For serial, non-MPI models, cloud computing resources offer thus a solid alternative for the operational tasks. Computational performances in the present tests compares well with both workstations and HPC grid cluster nodes, taking into account the hardware

differences. The scalability and flexibility of cloud resources allows the implementation of multiple forecast systems (for instance for all estuaries in the Portuguese coast) established in the coastal managers resources and resorting to the cloud for computational power.

### 4.3 DAM SAFETY CONTROL/NUMERICAL SIMULATION OF THE STRUCTURAL BEHAVIOR OF CONCRETE DAMS

For this test, the parallel program tf2, written in C (described in section 2.2.2) was run 3 times for each configuration, using as input data the mesh of the Baixo-Sabor dam. Two different configurations were run: the first comparing a local dual-core computer with the pilot cloud for a 12 time steps run, and the second comparing MEDUSA with the pilot cloud for a 260 time steps run (corresponding to 10 years of data).

Table 5 and Table 6 show the run times for the different flavors and the local environment baseline. The tests were run using the m1.small, m1.medium, m1.large and m1.xlarge flavors. These were chosen because they possess, respectively, 1, 2, 4 and 8 cores available. The operating system in the instances used in this test was centOS 7. The results in figure 17 were normalized so that both single core runtimes equal 1. Figure 18 shows the global and process performance.

In every case, the criteria of functional completeness and functional correctness[2] were completely met. The programs run in the same manner as in the local environment and the results produced were also the same. The relative time behavior was very similar, and the absolute times were 13% and 8% faster in the pilot cloud for the single and dual core case respectively.

**Table 5 – Run times and average run time for different flavors.**

| | Intel laptop | Intel laptop | m1.small | m1.medium | m1.large | m1.xlarge |
|---|---|---|---|---|---|---|
| # cores: | 1 | 2 | 1 | 2 | 4 | 8 |
| Average run time (s) | 270 | 180 | 234 | 165 | 126 | 126 |
| Std. dev. (%) | 0.9 | 2.6 | 0.5 | 2.4 | 0.8 | 0.9 |

[2] Functional completeness: degree to which the set of functions covers all the specified tasks and user objectives. Functional correctness: degree to which a product or system provides the correct results with the needed degree of precision. (ISO/IEC 25010 (2011)).

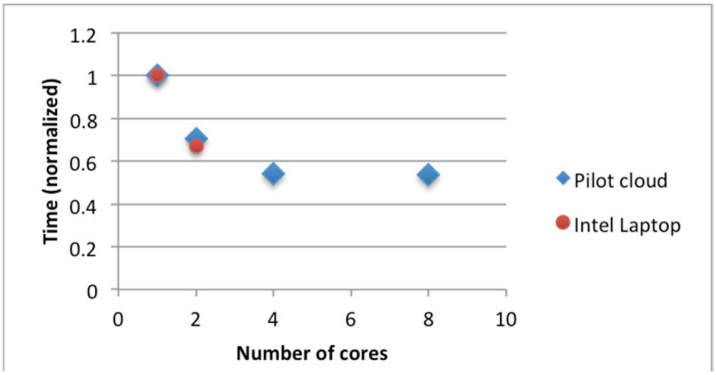


**Figure 18 – Average run time (s) for increasing number of cores**

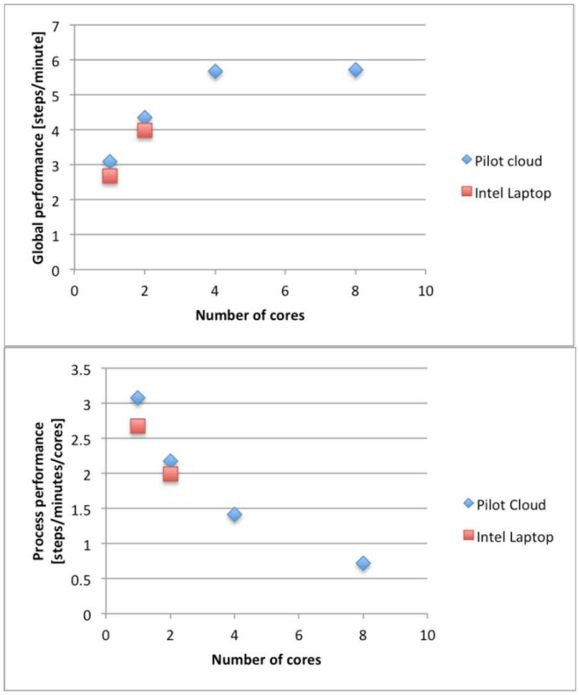


**Figure 19 – Global and process performance for the 12 timesteps Baixo-Sabor dam.**

The results for the 10 years simulation are shown in Table 6 and Figures 19 and 20.

**Table 6 – Run times and average run time for different numbers of cores for the 10 years run.**

| | # of proc. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|
| MEDUSA | avg. (s) | 3683.3 | 2405.7 | 2378.0 | 2124.3 | 2195.0 | 2026.0 | 2082.7 | 2057.3 |
| | std. dev. (%) | 0.5 | 0.1 | 0.2 | 1.3 | 0.1 | 0.0 | 0.7 | 0.6 |
| Pilot cloud | avg. (s) | 1961.7 | 1256.3 | 1208.0 | 1121.0 | 1117.7 | 1050.7 | 1106.3 | 1089.0 |
| | std. dev. (%) | 0.4 | 1.7 | 1.6 | 1.7 | 2.2 | 0.2 | 0.6 | 2.8 |

For MPI-models up to the number of cores on a single machine (8 in the case of m.xlarge), the pilot cloud showed almost the same relative time behavior for different number of cores, as shown in figure 18, while at the same time being almost 50% faster than the local cluster for the same number of cores (due to better hardware).

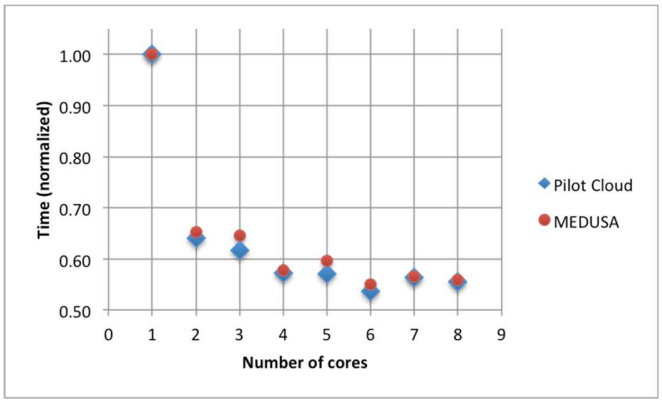


**Figure 19 – Normalized run times for increasing number of cores**

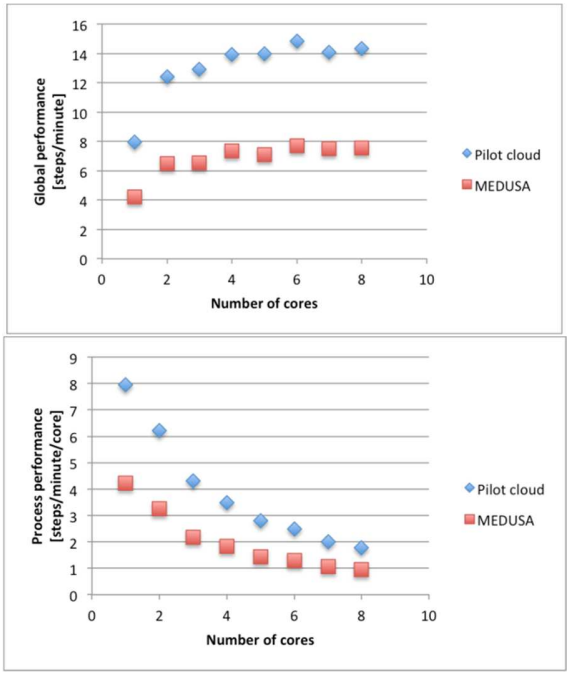


**Figure 20 – Global and process performance for the 10 years Baixo Sabor run for increasing number of cores**

### 4.4 SENSOR DATA ACQUISITION AND PROCESSING/MULTILINEAR REGRESSION

The results achieved by testing the R application for multilinear regression are show in Table 7. This is a sequential application and, as expected, the time behavior does not improve by running in a machine with more cores (in fact the performance slightly degrades). The criteria of functional completeness and functional correctness were met, with both configurations producing the same results. It should be noted that there was an additional step required to run the application from its original Windows environment, namely the re-encoding of the data (using GNU's *recode* program).

The cloud provides an advantageous alternative, because of its ease of use and flexibility while maintaining or improving functionality, correctness and performance. In the case where the memory and capacity are sufficient to efficiently run a sequential program, the smaller instance (with least amount of cores) should be chosen since the performance will be the same (if not slightly better) and more resources are left available for the rest of the users of the cloud.

**Table 7 – Run times and average run time for different flavors.**

| | Local | m1.small | m1.medium | m1.large | m1.xlarge |
|---|---|---|---|---|---|
| # cores: | 1 | 1 | 2 | 4 | 8 |
| Average run time (s) | 214 | 178 | 183.3 | 184 | 186 |
| Std. dev. (%) | 2.8 | 1.0 | 0.8 | 0.9 | 0.9 |

# 5 CONCLUSIONS

We presented herein a comparison of model performance indicators for several operational coastal forecast simulations deployments and structural engineering tests executed in local workstations, in the MEDUSA cluster nodes and in a pilot cloud, aiming at establishing the best choice for resources in the National Infrastructure for Scientific Computing in Portugal.

For serial, non-MPI models, cloud computing resources compare well to grid and local workstations for the operational tasks. The scalability and flexibility of cloud resources makes them an attractive alternative for the implementation of multiple forecast systems (for instance for all estuaries in the Portuguese coast) and sensor data acquisition and processing applications.

For MPI-based models, tests using cloud virtual machines with resources lower or equal to the smaller physical bases performed well relative to the other resources. However, the smaller cloud resources under testing (Aurora cloud) did not allow searching for the optimal number of processors for the present use cases. As a consequence, the HPC cluster remains as the best option, as it fits better the requirements for the typical dimensions of

computational grids for multi-scale (port to ocean) analysis. The access to further cloud resources sites via federation allowed the exploitation of these models in a larger scale (through the nimbus cloud). Results showed that using any combination that uses many processes (by using many hosts or many processes within each host) the performance scales very badly, even if we use resources with some hardware assistance. But for smaller pool sizes, like 3 or 4 hosts with 8 to 10 processes each, the results can be very satisfying, allowing combining the processing power of several hosts.

Further testing is still necessary to explore this possibility in detail. These future studies should handle the challenges related with the need to assure an adequate quality of service (QoS), especially to meet forecasting deadlines and real-time streaming bandwidth. The tests should also include data communication technologies with latencies and performance beyond standard Ethernet technology.

We conclude that an evolution from the current cluster setup to a cloud-based architecture may satisfy most of our simulation requirements while offering a more flexible and affordable computing environment.

The work presented herein was disseminated in a number of publications and public scientific events, listed in the Annex.

# Acknowledgements

This work was funded by FCT/FCCN through the project “Pilot cloud”. The authors would like to thank Marta Rodrigues for the support in the ECO_SELFE simulations.

# References


Azevedo, A., Oliveira, A., Fortunato, A.B., Zhang, J., Baptista, A.M., (2014). A cross-scale numerical modeling system for management support of oil spills accidents, Marine Pollution Bulletin Volume: 80 Issue: 1-2 Pages: 132-147.

Brauwere A., Brye B., Servais P., Passerat J., Deleersnijder E. (2011). Modelling Escherichia coli concentrations in the tidal Scheldt river and estuary. Water Research, 45, 2724-2738.

Canteras J.C., Juanes J.A., Pérez L., Koev K.N. (1995). Modelling the coliforms inactivation rates in the Cantabrian Sea (Bay of Biscay) from in situ and laboratory determinations of T90. Water Science and Technology, 32, 37-44.

Coelho, J., Silva, A., and Gomes, J. P. (2014). Parallelization of a Tridimensional Finite Element Program for Structural Behaviour Analysis. In *Euro-Par 2014: Parallel Processing Workshops* (pp. 36-47). Springer International Publishing.

Costa, M., A. Oliveira, M. Rodrigues, A. Azevedo, 2009. Application of Parallel, High-Performance Computing in Coastal Environmental Modeling: Circulation and Ecological Dynamics in the Portuguese Coast, Proceedings of the 3rd IBERGRID, 375-386.

den Boer, S., Azevedo, A., Vaz, L., Costa, R., Fortunato, A.B., Oliveira, A., Tomás, L.M., Dias, J.M., Rodrigues, M., 2014. Development of an oil spill hazard scenarios database for risk assessment. In: Green, A.N. and Cooper, J.A.G. (eds.), Proceedings 13th International Coastal Symposium (Durban, South Africa), Journal of Coastal Research, Special Issue No. 66, pp. 539-544, ISSN 0749-0208.

Dietrich, J.C., C.J.Trahan, M.T.Howard, J.G.Fleming, R.J.Weaver, S.Tanaka, L.Yu, R.A. LuettichJr., C.N.Dawson, J.J.Westerink, G.Wells, A.Lu, K.Vega, A.Kubach, K.M.Dresback, R.L. Kolar, C.Kaiser, R.R.Twilley. 2012. Surface trajectories of oil transport along the Northern Coastline of the Gulf of Mexico", Continental Shelf Research, 41, 17-47.

Gomes, J.L., Jesus, G., Rodrigues, M., Rogeiro, J., Azevedo, A. and Oliveira, A. 2013. Managing a Coastal Sensors Network in a Nowcast-forecast Information System, Proceedings of the Sixth Int. Workshop on Next Generation of Wireless and Mobile Networks (NGWMN-2013), 6 pages.

Jesus, G., Gomes, J., Oliveira, A., den Boer, S. and Azevedo, A. 2013. From a nowcast-forecast information system to an oil spill risk assessment and response tool, Geomundus 2013, Spain, 6 pages.

Jesus G., Gomes J., Ribeiro N.A., Oliveira A. 2012. Custom deployment of a Nowcast-forecast information system in coastal regions, Geomundus 2012.

Oliveira, A., A.B. Fortunato, 2002. Towards an oscillation-free, mass conservative, Eulerian–Lagrangian transport model, J. Comput. Phys., 183 (1) (2002), pp. 142–164

Oliveira, A., Jesus, G., Gomes, J.L., Rogeiro, J., Azevedo, A., Rodrigues, M., Fortunato, A.B., Dias, J.M., Tomas, L.M., Oliveira, E.R. Alves, F.L., den Boer, S., 2014. An interactive WebGIS observatory platform for enhanced support of coastal management. In: Green,

A.N. and Cooper, J.A.G. (eds.), Proceedings 13th International Coastal Symposium (Durban, South Africa), Journal of Coastal Research, Special Issue No. 66, pp. 507-512, ISSN 0749-0208.

Rodrigues M., Oliveira A., Queiroga H., Fortunato A.B., Zhang Y.J. (2009). Three-dimensional modeling of the lower trophic levels in the Ria de Aveiro (Portugal). Ecological Modelling, 220 (9-10), 1274-1290.

Rodrigues M., Oliveira A., Guerreiro M., Fortunato A.B, Menaia J., David L.M., Cravo A. (2011). Modeling fecal contamination in the Aljezur coastal stream (Portugal). Ocean Dynamics, 61(6), 841-856.

Rodrigues M., Costa J., Jesus G., Fortunato A.B., Rogeiro J., Gomes J., Oliveira A., David L.M. (2013). Application of an estuarine and coastal nowcast-forecast information system to the Tagus estuary. Proceedings of the 6th SCACR – International Short Course/Conference on Applied Coastal Research, 10 pp.

Rogeiro, J., João L. Gomes, Gonçalo Jesus, Anabela Oliveira, Marta Rodrigues, Alberto Azevedo, André B. Fortunato. 2014. WIFF: Water Information Forecast Framework, abstract, 13th International Workshop on Multiscale (Un)-structured mesh numerical ocean modeling, 2pp.

RSB (2007) - Regulamento de Segurança de Barragens, Decreto-Lei nº 344/2007 de 15 de outubro de 2007.

Servais P., Garcia-Armisen T., George I., Billen, G. (2007). Fecal bacteria in the rivers of the Seine drainage network (France): sources, fate and modelling. Science of the Total Environment, 375, 152–167.

Tkalich, P. 2006. A CFD solution of oil spill problems, Environ. Model. Softw., 21 (2006), pp. 271–282.

Zhang Y., Baptista A.M. (2008). SELFE: a semi-implicit Eulerian– Lagrangian finite-element model for cross-scale ocean circulation. Ocean Modeling, 21(3–4), 71–96

# Annex: Dissemination

## PUBLIC PRESENTATIONS IN SCIENTIFIC EVENTS

- **MEC2015 3ª Conferência sobre morfodinâmica estuarina e costeira**
  Previsão em tempo real da hidrodinâmica e contaminação fecal em ambiente cloud e hpc - Marta Rodrigues, João Rogeiro, André Fortunato e Anabela Oliveira
- **EGI 2015:**
  Application of cloud computing to a tridimensional finite element parallel program for structural analysis of concrete dams - João Rico, João Coelho, J. Piteira Gomes, António Silva
  Running multi-scale, high resolution coastal models on HPC grid and cloud resources: an operational comparison - João Rogeiro; Alberto Azevedo, Marta Rodrigues, Anabela Oliveira, João Paulo Martins, Mário David, João Pina, Jorge Gomes
- **CPGZC2015 VIII Congresso sobre Planeamento e Gestão das Zonas Costeiras dos Países de Expressão Portuguesa:**
  Sistema Multi-escala de Previsão em Tempo Real da Dinâmica Estuarina E Costeira: Desafios para a Operacionalização em Ambiente Cloud e HPC - Anabela Oliveira; João Rogeiro; Alberto Azevedo; André B. Fortunato; Ricardo Tavares Da Costa; Marta Rodrigues; Kai Li; João Paulo Martins; Mário David; João Pina; Jorge Gomes
- **CC2015 The Fifteenth International Conference on Civil, Structural and Environmental Engineering Computing, Prague, Czech Republic 1-4 September 2015**
  Running High Resolution Coastal Forecasts: Moving from Grid to Cloud Resources - J. Rogeiro, A. Azevedo, M. Rodrigues, A. Oliveira

## PUBLICATIONS

- Marta Rodrigues, João Rogeiro, André Fortunato e Anabela Oliveira, (2015). Previsão em tempo real da hidrodinâmica e contaminação fecal em ambiente cloud e HPC, MEC 2015, Matias et al., (eds), 2pp.
- J. Rogeiro, A. Azevedo, M. Rodrigues, A. Oliveira, "Running High Resolution Coastal Forecasts: Moving from Grid to Cloud Resources", in J. Kruis, Y. Tsompanakis, B.H.V.Topping, (Editors), "Proceedings of the Fifteenth International Conference on Civil, Structural and Environmental Engineering Computing", Civil-Comp Press, Stirlingshire, UK, Paper 218, 2015. doi:10.4203/ccp.108.218 (http://dx.doi.org/10.4203/ccp.108.218)
- Anabela Oliveira; João Rogeiro; Alberto Azevedo; André B. Fortunato; Ricardo Tavares Da Costa; Marta Rodrigues; Kai Li; João Paulo Martins; Mário David; João Pina; Jorge Gomes (2015). Sistema Multi-escala de Previsão em Tempo Real da Dinâmica Estuarina E Costeira: Desafios para a Operacionalização em Ambiente Cloud e HPC. VIII Congresso sobre Planeamento e Gestão das Zonas Costeiras dos Países de Expressão Portuguesa, APRH (eds), 15 pp.