MULTICELLULAR BOX DOCKS: AN USER FRIENDLY INTERNET BASED SOFTWARE FOR DESIGN IN CIVIL ENGINEERING

José Pereda*, Antonio Foces*, José A. Garrido* & Benjamín Suárez†

* Departamento de Resistencia de Materiales de la Universidad de Valladolid,
C/ Paseo del Cauce S/N, 47011 Valladolid, Spain.
e-mail: {pereda, foces, garrido} @eis.uva.es

† International Center for Numerical Methods in Engineering (CIMNE),
Edificio C-1, Campus Norte UPC,
C/ Gran Capitán S/N, 08034 Barcelona, Spain
e-mail: benjamin.suarez@upc.es

Key words: Multicellular Concrete Box Dock, Internet, Servlets, Finite Element Method

Abstract: *A software package developed for the complete design, analysis and reinforce of multicellular concrete box docks is presented. Box docks are structures extensively used for building vertical harbor quays or breakwaters. At the online version this process is achieved by using the Internet web. The user's computer (the client) has the pre-processing (geometry definition and boundary conditions) and post-processing (visualization of results) light-weight routines. A graphic user interface is required to guide the user in a friendly way through the different steps of the whole process using several wizards implemented to this end. On the other side, the heavy-weight calculation routines are hosted on a suitable remote server, with no need of any user interface. The communication and data transfer between the server and the client will be routed by servlets (Java programs) hosted in the server. They will control the tasks of performing the calculations by the finite element method, testing the calculus status, and sending the available results at user's request. In turn, GiD® is used for meshing and rendering post-processing results purposes and CALSEF® is employed for solving ones. Communications, external programs calling and data transferring are transparency processes to the user, although he controls their state at any time, being able to make the opportune decisions.*

1 INTRODUCTION

In engineering problems, the analysis step uses to be the most computing resources and time consuming, usually requiring a very high performance computer. With the so called “net-computing” the user avoids spending a big budget in the continuous acquisition of big calculation machines and very expensive specific programs for any problem he has to solve.

Suitable hardware and software equipment may form a powerful server so users from any part in the world may access to different programs with heavy-weight calculation routines, therefore having an important reduction of cost.

Box docks are structures extensively used for building vertical harbor quays or breakwaters and nowadays continuous extensions and improvements are made in many coastal zones. Box docks are a fast and practical alternative to build docks. Placed on rock fill, distances of one or two kilometres are covered in terms of weeks, even in high depths. They are precast repetitive elements and therefore the interest to make profitable its production. Here is where the program being presented could serve to design the internal and external dimensions, to verify the box dock behaviour against predicted loads and to calculate the reinforcement requirements.

2 MULTICELLULAR BOX DOCKS: DESIGN AND ANALYSIS

2.1 Preprocessing

With the main geometry variables introduced by the user, the whole box dock is defined. Loads are applied over the box in every different load situation it will be exposed.

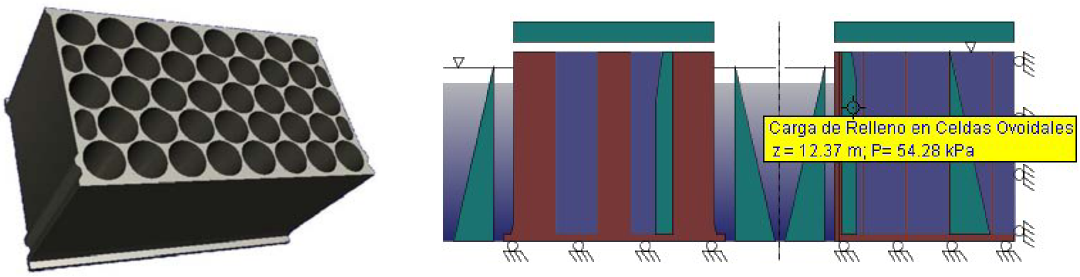


Figure 1. Preprocess: geometry and loads definition.

2.2 Analysis step

The user sets the basic options for the process to be executed through an interactive wizard, so a coarse or fine mesh is employed to solve one or more loads combinations (Figure 2).

Alter the wizard is finished, a calculus request is sent to the server with the data related to the box dock and the different selected options.

For every single request being launched by the program, the servlet *chkOnLine* (Figure 3-a) checks the server is online. For the first time, the servlet *Registro* (Figure 3-b) registers new users in the server database. Afterwards, for any user request the servlets check the user is registered and has any calculus licence left. These requests may ask for a particular calculus of a load combination, or for the reinforcement redefinition of some calculated load situation; they may check the calculus status, may ask for the process cancellation or, finally, may ask for the results download (Figure 3-c).

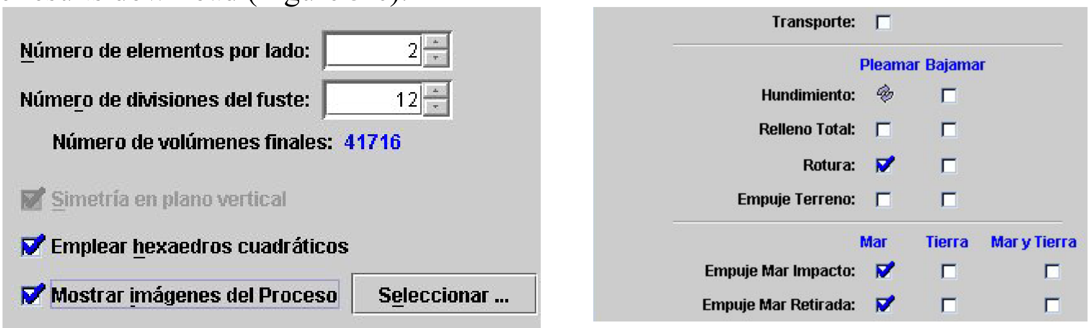
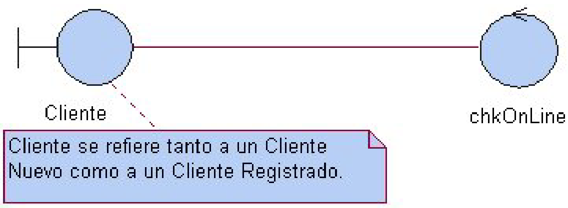
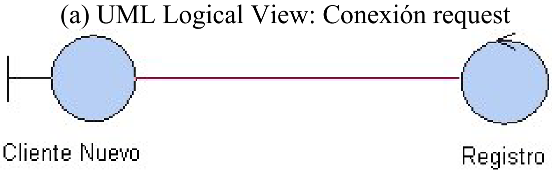


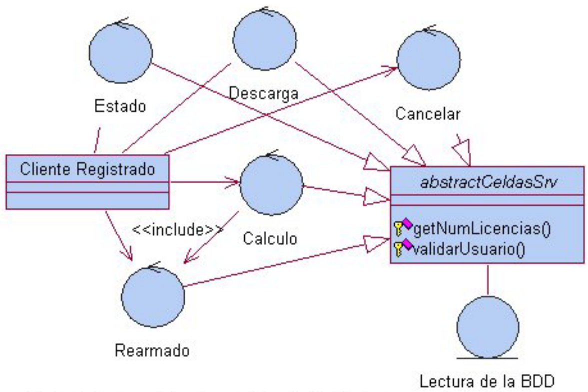
Figure 2. Options selection and loads combinations



(a) UML Logical View: Conexión request



(b) UML Logical View: Registration request



(c) UML Logical View: Different process request

Figure 3. UML Logical View of the Server Servlets

The calculus algorithm begins with the generation of the geometry based in the user data. This geometry is read with GiD©, in which a structured mesh is performed. The solver CALSEF© gives displacements and stresses for any node in the mesh, which are used again in GiD© to generate postprocess images, on user request (Figure 4).

Automatically, at the same time geometry is being developed, critical points ("*throats*") are set at places of shorter distance between adjacent cells, to perform the integration of the stresses in the postprocessing. The resulting bending moment and axial force are used to calculate the required longitudinal reinforcement for the inside walls (circumferential and vertical reinforcement) and for the outside ones (horizontal and vertical), Figure 5.

2.3 Postprocessing

On being notified the process has ended, the user may download the results to analyze and visualize the final reinforcement (Figure 5). He may go back to preprocessing or analysis if it's necessary to change any variable or he may finish the design plotting the final drawings.

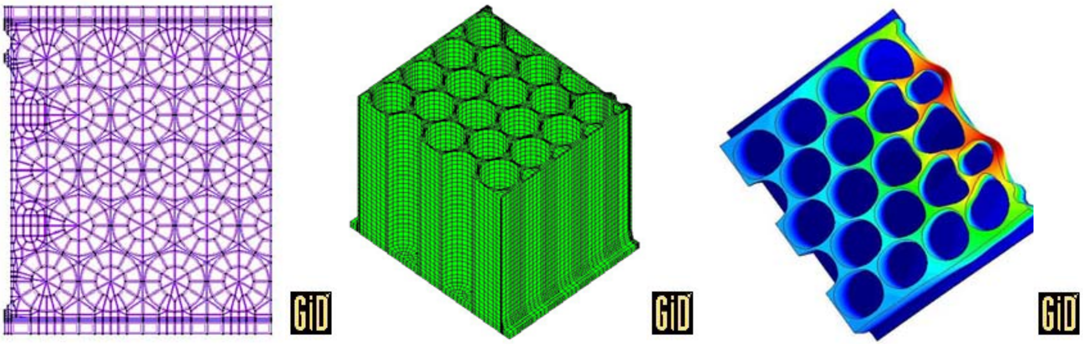
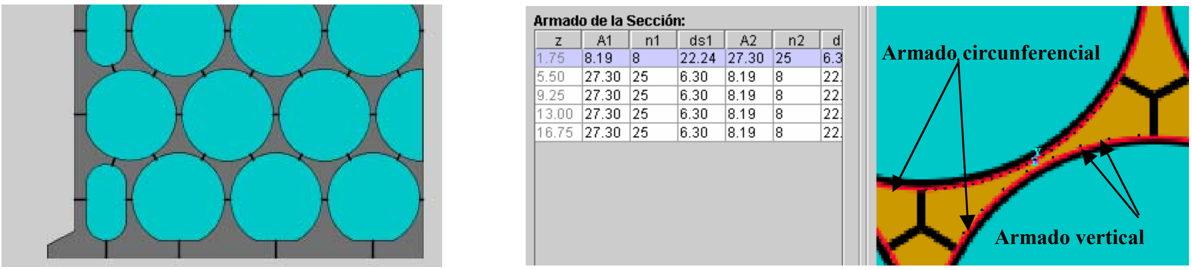


Figure 4. Geometry generation, mesh and results



z	A1	n1	ds1	A2	n2	d
1.75	8.19	8	22.24	27.30	25	6.3
5.50	27.30	25	6.30	8.19	8	22.
9.25	27.30	25	6.30	8.19	8	22.
13.00	27.30	25	6.30	8.19	8	22.
16.75	27.30	25	6.30	8.19	8	22.

Figure 5. "*Throats*" definition and reinforcement distribution

3 CONCLUSIONS

The online version of a program for multicellular reinforced concrete box docks analysis and design has been presented. It guides a non-expert user in finite elements nor in net-computing through the complete design of a box dock, by defining the geometry and several load conditions, requesting the analysis step to be performed in a well-conditioned server, by terms of the finite element method. Communications, external programs calling and data transferring are transparency processes to the user, although he controls their state at any time.

4 BIBLIOGRAPHY

[1] E. Comerma, "Diseño automatizado de cajones multicelulares flotantes para obras porturarias", *Tesina de Especialidad*, UPC (1999)

[2] B. Suárez, A. Foces, J. Pereda, M. Cacho, "Treating the geometry of a floating caisson to obtain a structured mesh in GiD", *1st Conference on Advances and Applications of GiD*. CIMNE Nº-210, (Feb 2002)

[3] ROM, Puertos del Estado, "*Recomendaciones para Obras Marítimas – ROM 0.2-90, ROM 0.5-94*", MOPTMA (1990)

[4] J. Hunter, W. Crawford, *Java Servlet Programming*, 2nd Edition, ISBN 0-596-00040-5, O'Reilly & Associates, Inc (2001).

[5] C. Larm, "*Applying UML and Patterns*". Prentice Hall PTR, (1997)