Distributed collaborative environment for software applications

S. Dormido-Canto[1], J. Vega[2], J. Chacón[1], E. Fabregas[1], G. Farias[3]

[1] Dpto. de Informática y Automática - UNED; [2] Laboratorio Nacional de FUSIÓN – CIEMAT; [3] Pontificia Universidad Católica de Valparaiso, Chile

ABSTRACT

This paper shows the implementation of a robust and generic collaborative system that allows the execution of applications in heterogeneous and distributed environments. In particular, it is related to the application and dissemination of advanced methodologies in relation to machine learning methods in the field of nuclear fusion. The collaborative environment will allow authorized users to launch applications in an easy and totally transparent way. The expression "totally transparent" means that the applications will be hosted by any computational resource that belongs to the collaborative environment and the user only has to provide his input data. The distributed collaborative environment (including high performance computing) will be accessible from web resources. The generated software repository will be an open space that includes all the implemented methods. An illustrative application based on the mitigation of stray-light in the Thomson Scattering diagnostic of the TJ-II stellarator is introduced.

INTRODUCTION / MOTIVATION

The distributed collaborative environment can be used in different real problems. In this context, the field of Nuclear Fusion has been chosen due to both the massive character of the databases generated in nuclear fusion experiments and the complexity of the data analysis.

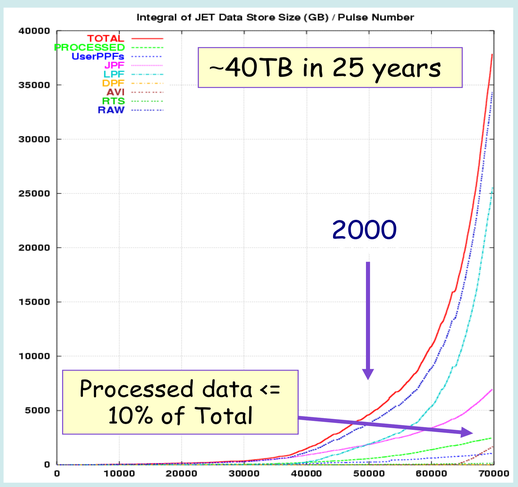


FIGURE 1: The amount of processed data in JET is about 10% of all data

Data mining techniques are essential methods to analyse fusion databases not only due to their massive character but also for the complexity of the data interpretation. In particular, software tools for intelligent searches in the databases], automatic events detection hidden knowledge discovery or data-driven models to understand the plasma nature and its control are of crucial importance. In this sense, it is desirable and practical to have a distributed collaborative environment that allows sharing of data, methods, techniques and software in order to supply these resources among researchers working on related topics. At present, the size of the JET database is about 80 Tbytes and the database evolution shows that the size is doubled every two years. JET statistics show that only a 10% of the data are processed (**Figure 1**).

ARCHITECTURE AND USER INTERFACE

The overall architecture is depicted in **Figure 2**. It is a distributed system where there are different collaborative entities, which may be phisically located in different places. Each entity is composed of computing servers which performs the computations (PCs, HPC clusters, etc), file servers that allow to store and access data, firewalls, etc.

An access point is a special kind of entity, with an application server which provides users with a web interface where they can manage their jobs. A job is a predefined processing task which have been automatized to be executed on demand and to solve an specific problem.

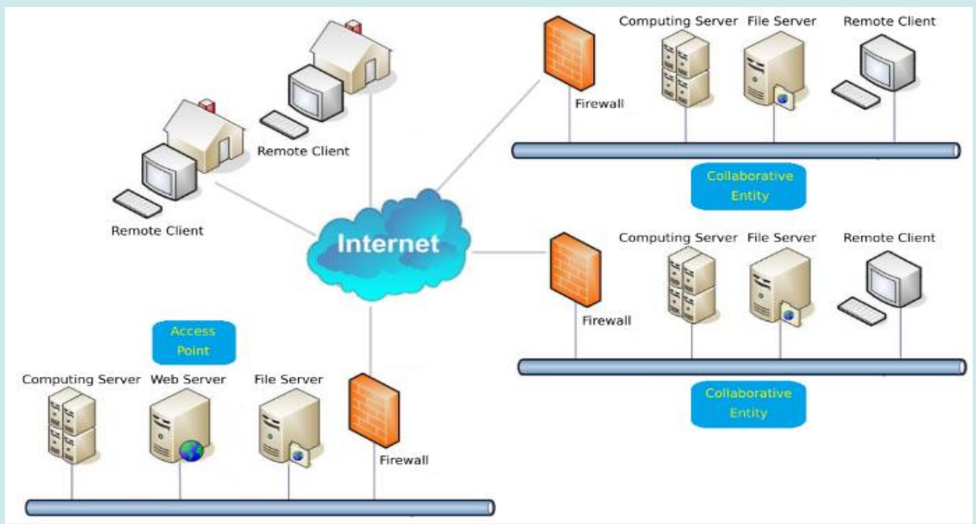


FIGURE 2: Architecture of the system

The application server is composed of three important and differentiated components: the *web server*, that provides the web interface, the *authentication agent*, that control users' access, and the *job manager*, which submits the jobs to the computing servers.

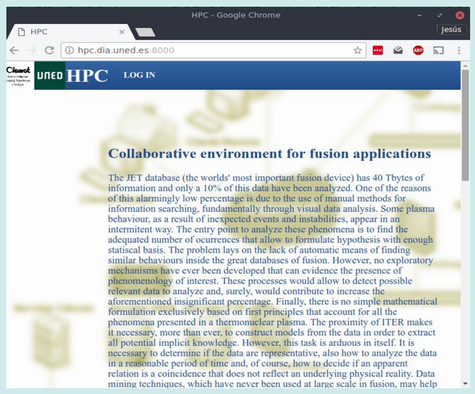


FIGURE 3: Home page view (http://hpc.dia.uned.es)

A job has a name, and a short description, which should help to easily identify its main purpose. Optionally, there can be one or more input and output files, and finally there is a document with the complete specification of the task, where it should be clearly stated what is the purpose of the task, what kind of data is expected, which are the results generated, and any other important information that users of the task should know.

The user is thus completely unaware of other tasks, but also of where the allowed ones are actually being launched. There is no need to know implementation details such as the method used to access the computing server, or the commands to start a new task.

Through the web interface, users can: 1) Launch jobs, 2) providing the input data.View the execution state of previously launched job and, 3) Obtain the results generated by the finished job

As an example, let us look at the job 'Image processing in Thomson Scattering' (**Figure 4**). This job expect to receive an input file 'image.txt', containing the image data to be processed, and return a file 'results.zip' which contains several images. To launch this job, user must click at the play button corresponding that job, and then the interface will ask for the required input file. Once the data has been provided, the job manager sends the job to its corresponding computing resource. From that moment on, the job status can be viewed in the list of submitted jobs. All these actions can be accesed through web services.

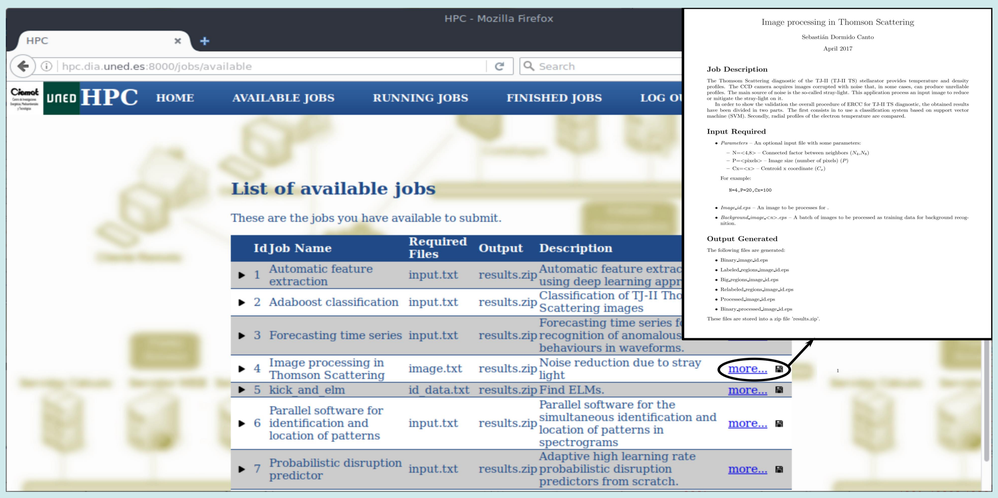


FIGURE4: Jobs view

AN ILLUSTRATIVE APPLICATION

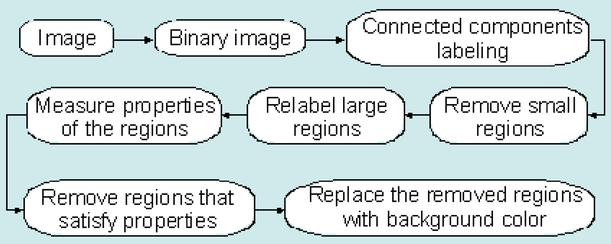


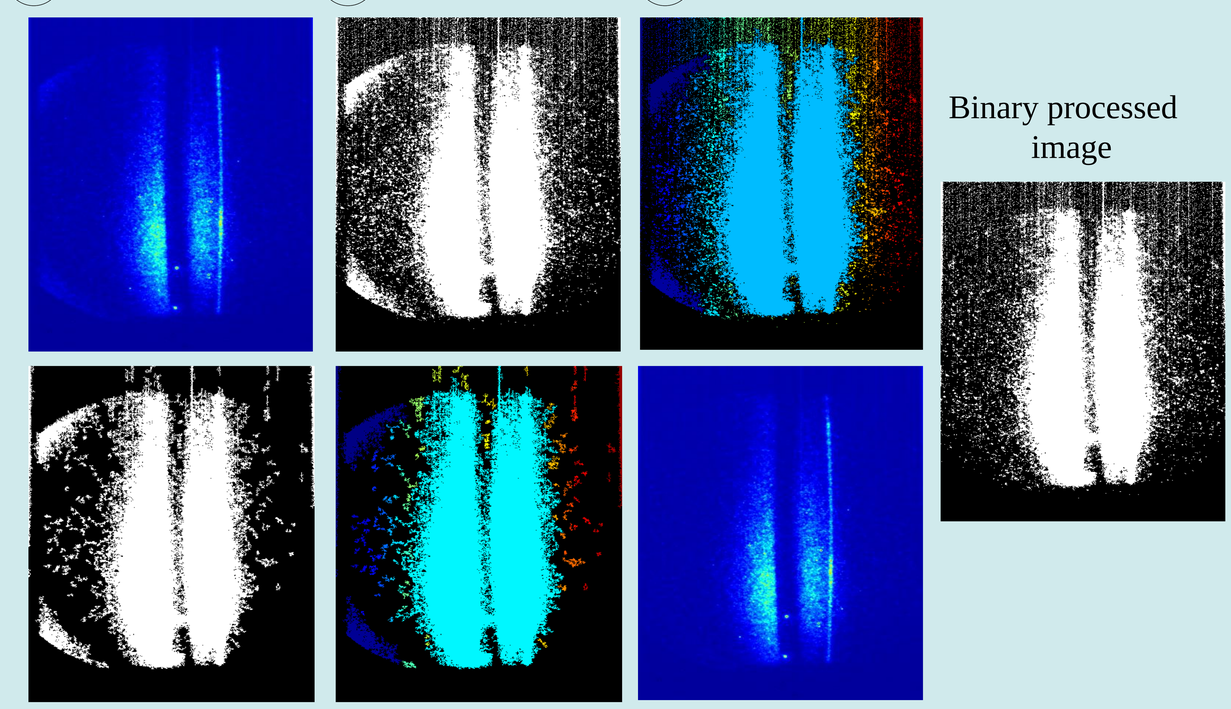
FIGURE 5: The flowchart for ERCC

The Thomson Scattering diagnostic of the TJ-II (TJ-II TS) stellarator provides temperature and density profiles. The CCD camera acquires images corrupted with noise that, in some cases, can produce unreliable profiles. The main source of noise is the so-called stray-light. In this section we describe an application in order to reduce or mitigate the stray-light on the images.

In our case, we define an approach based on extraction regions with connected components (ERCC).

Specifically the procedure has been implemented is summarized in the flowchart of the **Figure 5**.

EXAMPLE CORRESPONDING TO DISCHARGE WITH NBI HEATING:



CONCLUSIONS

The utility of having a collaborative environment for software applications has been demonstrated in order to share data, methods, techniques and software. The environment allows the scalability required by the applications to be executed

In particular, ERCC approach has proven to be useful removing stray light in the TJ-II TS diagnostic without eliminating significant information.

ACKNOWLEDMENTS

This work was supported by the Spanish Ministry of Economy and Competitiveness under the Projects Nos. ENE2015-64914-C3-1-R and ENE2015-64914-C3-2-R by the UNED project GID2016-6-1, and by the Chilean Ministry of Education under the Project FONDECYT 1161584. .