

# DESARROLLO DE UN ENTORNO COLABORATIVO PARA APLICACIONES DE FUSIÓN

Jesús Chacón<sup>1</sup>, Ernesto Fabregas<sup>1</sup>, Sebastián Dormido-Canto<sup>1</sup>, Jesús Vega<sup>2</sup>, Jose Sánchez<sup>1</sup>

<sup>1</sup> Universidad Nacional de Educación a Distancia.

<sup>2</sup> Laboratorio Nacional de Fusión. Ciemat

## Resumen

*El entorno colaborativo desarrollado permite realizar un envío de tareas a ser procesadas en clusters de supercomputación, sin necesidad de que los usuarios de las mismas deban conocer la infraestructura subyacente o estar familiarizados con las órdenes de más bajo nivel necesarias para interactuar con estos sistemas.*

**Palabras clave:** Entorno colaborativo, repositorio de aplicaciones, fusión

## 1. INTRODUCCION

La base de datos de JET (el dispositivo de fusión más importante del mundo) tiene 40 Tbytes de información y solamente se han analizado un 10 % de sus datos. Una de las razones de este alarmante bajo porcentaje es atribuible a las búsquedas de información mediante procedimientos manuales, fundamentalmente a través del análisis visual de datos. Algunos comportamientos del plasma, como resultado de eventos inesperados e inestabilidades, aparecen de manera intermitente. El punto de entrada para analizar estos fenómenos es encontrar el número adecuado de ocurrencias que permitan formular hipótesis con suficiente base estadística. El problema estriba en que no existen medios automáticos para buscar comportamientos similares dentro de las grandes bases de datos de fusión. Por otro lado, tampoco se han desarrollado nunca mecanismos exploratorios que puedan evidenciar la presencia de fenomenología de interés. Estos procesos permitirían detectar posibles datos relevantes a analizar y, desde luego, contribuirían a elevar el nimio porcentaje expuesto anteriormente. Por último, no existe una formulación matemática sencilla, que basada exclusivamente en primeros principios, dé cuenta de toda la fenomenología presente en un plasma termonuclear. La proximidad de ITER hace necesario, hoy más que nunca, construir modelos a partir de los datos para poder extraer todo el potencial conocimiento implícito. Sin embargo, esta tarea es ardua en sí misma. Es necesario determinar si los datos son representativos, también cómo analizar los datos en un periodo

de tiempo razonable y, desde luego, cómo decidir si una relación aparente es una ocurrencia casual que no refleja una realidad física subyacente. Las técnicas de minería de datos, que nunca se han utilizado a gran escala en fusión, pueden ayudar a estos fines y sobre todo a aumentar el paupérrimo 10 % de datos analizados en JET. Por consiguiente, se plantea la necesidad para los entornos de fusión, de desarrollar de mecanismos inteligentes de acceso a datos, generalizar el uso de técnicas de reconocimiento y exploración de datos, así como extracción de conocimiento oculto. Para la investigación, desarrollo y aplicación de estas técnicas se utilizarán las bases de datos del TJ-II y JET. El mayor énfasis de aplicabilidad se pondrá en JET, como dispositivo más próximo a ITER en sus condiciones de operación.

El objetivo de este trabajo está relacionado con la creación de un entorno colaborativo seguro sobre una infraestructura computacional distribuida (incluyendo supercomputación). En primer lugar, se trata de rentabilizar el esfuerzo efectuado en el desarrollo de aplicaciones genéricas de aprendizaje automático. El software genérico que se desarrolle en el marco de este proyecto (y el que se estime necesario en relación con el aprendizaje automático) se pondrá a disposición de una amplia comunidad científica, tanto del entorno de fusión como externos. Habrá una infraestructura distribuida de ordenadores (incluso de supercomputación) dedicados a la ejecución de software de aprendizaje automático. El acceso a cualquier recurso computacional se producirá siempre a través de un portal web, que implementará mecanismos de seguridad para identificación y autorización de los usuarios. Además, para aprovechar la ubicuidad que ofrecen los servidores web, también se podrá utilizar la infraestructura mencionada para el acceso a aplicaciones específicas de usuario que se ejecuten en el entorno descrito.

Para realizar la gestión de usuarios y recursos se tuvieron en cuenta diferentes alternativas: servicios Web genéricos [4], [5] o entornos dedicados como Emersion [3]. Lógicamente el entorno, al ser distribuido, ha de estar securizado por un sistema distribuido de identificación y autorización. Ini-

cialmente, se consideró el uso del sistema PAPI [1]. Este sistema es el que securiza el acceso web a los sistemas de adquisición de datos y control de diagnósticos del TJ-II [5] así como el acceso al entorno colaborativo europeo Federación EFDA [2]. No obstante, finalmente se optó por desarrollar un sistema propio de autenticación basado en sistemas de clave pública.

La organización del resto del artículo es la siguiente. En la sección 2 se describen brevemente los plasmas de fusión. En la sección 3 se presenta el entorno colaborativo en desarrollo. Finalmente, en la sección 4 se presentan algunas conclusiones.

## 2. PLASMAS DE FUSION

Los plasmas de fusión son sistemas físicos regidos por complejas interacciones altamente no lineales y que necesitan diagnosticarse de la mejor manera posible. La base de datos de cualquier dispositivo actual se compone de millones de señales. Por ejemplo, la base de datos del estelarator español TJ-II gestiona más de tres millones de señales por las más de 17000 descargas producidas. La mayoría de estas señales son de evolución temporal, con un número de muestras entre 10000 y varios millones. En el tokamak JET (con una duración de descarga de 40 s), se han realizado más de 70000 disparos y el número de señales que se adquieren en cada descarga puede ser superior a 9000. En la actualidad, el tamaño de la base de datos de JET es de 40 Tbytes y su tendencia de crecimiento refleja que la cantidad de datos adquiridos se duplica cada dos años. Puesto que las técnicas de medida van generando día a día un número creciente de señales, es fácil imaginar lo que supondrá la base de datos de ITER. El número de señales que se esperan recoger en cada disparo de ITER está entre 500.000 y 1.000.000, por lo que la tasa esperada de datos es de Pbytes/año. Sin embargo, cualquier colección de datos carece de valor sin unos mecanismos eficientes para extraer información y conocimiento de los mismos. Bases de datos como las de JET no pueden ser analizadas en su globalidad ni tan siquiera por los mejores algoritmos (en términos de velocidad y requerimientos de memoria). De hecho, datos de JET evidencian que solamente un 10% de los datos adquiridos se procesan (Figura 1) [1]. Esta circunstancia pone de manifiesto que en la actualidad, procesar toda la información disponible es un problema de mucha más envergadura que la falta de ella. La baja tasa de datos analizados obedece a unos mecanismos impropios de acceso a la información y que se fundamentan en búsquedas manuales. Nunca han existido en fusión sistemas automatizados para poder desterrar tales medios manuales, ni tampoco

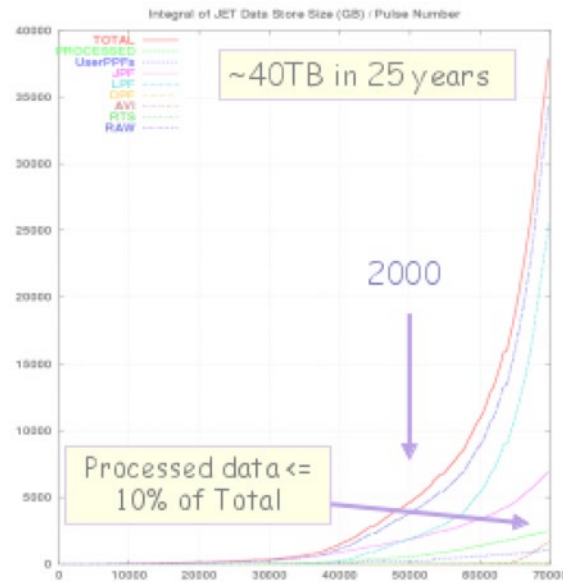


Figura 1: (cortesía de J. W. Farthing). La cantidad de datos procesados en JET apenas alcanza el 10% del total.

para detectar evidencias interesantes de analizar, ni, por supuesto, para facilitar la extracción de conocimiento oculto. Para remediar esta situación, las técnicas de análisis moderno en el entorno deben orientarse hacia la consecución de tres objetivos primordiales:

1. Desarrollo de mecanismos inteligentes de acceso a datos
2. Generalización del uso de técnicas de reconocimiento y exploración de datos
3. Extracción de conocimiento

Un plasma en condiciones termonucleares es demasiado complejo para ser descrito matemáticamente y de forma completa en función de primeros principios. En ausencia de modelos de primeros principios, la gran cantidad de datos de los que se dispone (formas de onda e imágenes) pueden utilizarse para obtener modelos en base a las dependencias que aparecen entre ellos, es decir, modelos construidos a partir de las medidas. La disponibilidad en fusión de conjuntos de datos tan enormes, tan complejos y tan ricos en significado se ha de aprovechar como una ventaja estratégica. La habilidad para extraer cualquier conocimiento útil que esté escondido en las bases de datos será el caballo de batalla de estos métodos, cuyo formalismo deberá basarse en técnicas de reconocimiento de patrones. Los modelos obtenidos, además de explicar el comportamiento de los plasmas, deben ayudar a poder predecir situaciones más o menos catastróficas (como por ejemplo las interrupciones

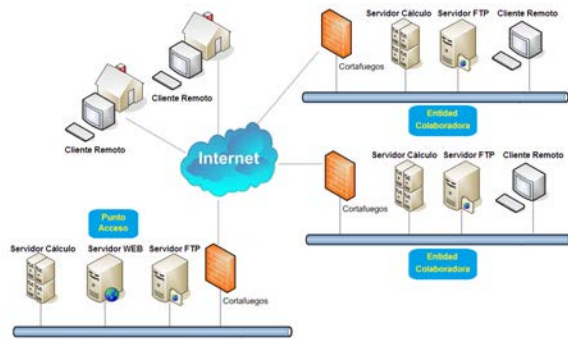


Figura 2: Entorno colaborativo distribuido

en Tokamaks) y poder desarrollar mecanismos de control (en tiempo real) que eviten daños estructurales irreversibles en los dispositivos. Las técnicas antes enumeradas se van a aplicar tanto a las bases de datos del TJ-II como a las del JET. Debido a que la duración de la descarga del TJ-II está limitada a menos de 500 ms, todos los desarrollos sobre la base de datos de este dispositivo se aplicarán en diferido. Sin embargo, en el caso de JET las descargas son de 40 s y se podrán analizar algunas aplicaciones para su uso en tiempo real, aplicaciones tales como detectar precursores de disrupciones, identificar regímenes L/H, clasificar ELMs, localizar barreras internas de transporte, descubrir la presencia de inestabilidades y, en general, ser capaces de determinar o predecir eventos específicos para poder obrar en consecuencia. Aunque los dominios de aplicabilidad de este proyecto van a ser los entornos experimentales de TJ-II y JET, sin embargo, los resultados serán esenciales de cara a ITER. ITER va a ser un dispositivo de pulso largo (1000 s) que va a generar millones de señales con muy alta dimensionalidad. Las grandes masas de datos que se adquieran en ITER solamente podrán ser analizadas de manera eficiente con métodos avanzados como los aquí propuestos. Además, será obligado el uso de técnicas de análisis en tiempo real que permitan evaluar la evolución de la descarga y poder corregir derivas no deseables. A este respecto, la información proporcionada por los sistemas técnicos (sistemas de control de máquina) será insuficiente. Los sistemas experimentales (diagnósticos) también deberán proporcionar datos valiosos para el sistema de control del dispositivo. Esto implicará construir sistemas de reconocimiento de patrones a partir de fuentes de datos distribuidas y heterogéneas.

### 3. ENTORNO COLABORATIVO DISTRIBUIDO

Dentro del proyecto en que se enmarca este trabajo, se van a desarrollar algoritmos generales de

aprendizaje automático con métodos avanzados de análisis de datos. La complejidad de su desarrollo y la más que evidente utilidad para múltiples aplicaciones, hace razonable la creación de un repositorio de software con todo lo generado. Este almacén de software va a ser un espacio abierto al que se podrán ir añadiendo nuevos métodos genéricos. La explotación del entorno también será abierta y quedará a disposición de la comunidad científica. Pero es importante mencionar los dos tipos de usuarios que se prevén. Por un lado, habrá usuarios que podrán acceder de forma limitada a los recursos de cálculo. Los accesos tendrán como objeto valorar alguna de las aplicaciones con un conjunto reducido de datos para algún propósito específico. Por otro lado, habrá usuarios que podrán acceder a todas las capacidades del entorno. Las aplicaciones estarán distribuidas en múltiples servidores (incluso de supercomputación), de tal manera que en todo momento será sencilla la incorporación de nuevo software desde cualquier equipo. En el desarrollo del proyecto se dispondrán 3 equipos de supercomputación (2 del CIEMAT y 1 de la UNED) y múltiples servidores individuales.

Se pretende así obtener un entorno colaborativo lo más robusto y genérico posible que permita el acceso y ejecución de aplicaciones en recursos computacionales heterogéneos de una manera automática y transparente para el usuario. En este punto cabe destacar que en el entorno colaborativo se podrán incluir también aplicaciones concretas de usuario para su ejecución remota desde cualquier localización.

#### 3.1. Arquitectura

La arquitectura propuesta (Figura 2) se basa en la definición de un servidor de aplicaciones, cuya función principal es ofrecer una interfaz a través de servicios web (Figura 3) que permita lanzar diferentes tareas que pueden estar localizadas físicamente en lugares diferentes.

Esencialmente, el problema se trata de la distribución de tareas (en inglés, *workload*) entre diferentes nodos de computación. Se trata de un *sistema de gestión de recursos distribuido (DRMS)* que debe cumplir los siguientes requisitos:

- Debe permitir distribuir trabajos entre una serie de nodos de computación heterogéneos (Los nodos pueden tener un SO tipo Unix o Windows, Mac, etc.).
- Pueden ser cluster o simples PCs.
- La instalación y configuración requeridas en los nodo de computación debería ser mínima,



Figura 3: Portada del entorno colaborativo.

para facilitar la incorporación de nuevos servidores de tareas.

Para la definición del entorno colaborativo, se consideran inicialmente tres diferentes roles: *administrador*, *desarrollador de tarea* y *usuario*.

La función de cada uno de ellos es:

- **Administrador** Se encarga de mantener el correcto funcionamiento del sistema, supervisar la creación de nuevos usuarios y gestionar los permisos de acceso al sistema, etc.
- **Desarrollador de tarea** Su función es añadir tareas al sistema para hacerlas disponibles a los usuarios. A su vez, son necesarias varias subtareas como desarrollar el código de la tarea, documentar el uso, etc.
- **Usuario final** Es el que ejecuta las tareas disponibles en el sistema, aporta datos de entrada y recoge resultados.

El escenario de uso considerado es el siguiente: el usuario accede a un portal web (Figura 3). A continuación, tras realizar correctamente la autenticación que le habilita para usar el sistema, éste le muestra una lista de acciones que dependerán de los permisos asignados al usuario (recordemos que había diferentes roles y niveles de acceso). Para un usuario normal, el sistema debe permitir:

- Lanzar una tarea, enviando los datos a procesar según la especificación de la tarea (que debe estar recogida en un documento asociado).
- Ver el estado de ejecución de sus procesos lanzados con anterioridad (qué tareas hay en ejecución, finalizadas, con error, etc.).
- Recuperar los datos generados como resultado de sus trabajos, en caso de que se hayan

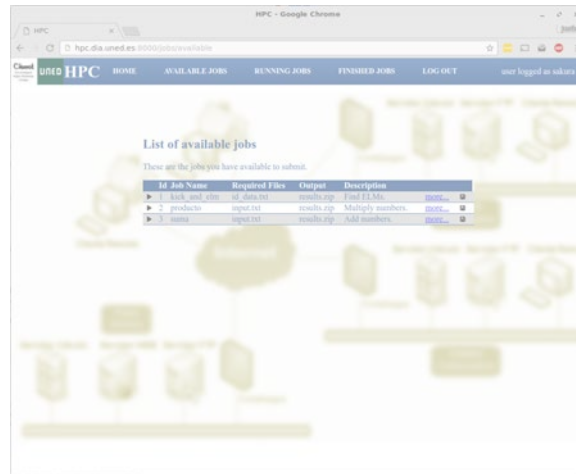


Figura 4: Lista de las tareas disponibles para ejecutar.

finalizado correctamente, o recuperación de archivos de *registro* para poder depurar los errores.

A continuación se describen más detalladamente cada una de estas acciones.

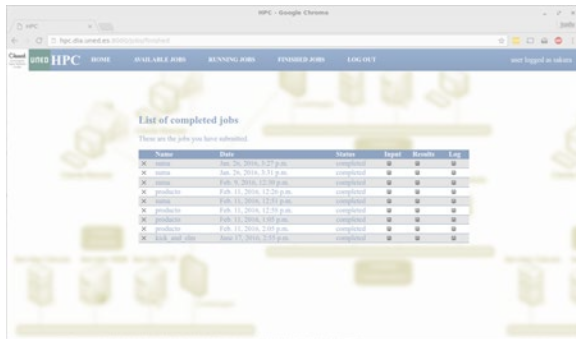
### 3.1.1. Ejecución de tareas

Una interfaz web lista los trabajos disponibles (ver Figura 4). Es importante señalar que el usuario solo tiene acceso a un subconjunto de aplicaciones a las que se le ha concedido permiso, sin que tenga conocimiento de la existencia de otras tareas o incluso de las máquinas en las que se ejecutan. Todo este proceso es transparente al usuario, tanto por motivos de seguridad como de facilidad de uso. De esta manera, no es preciso conocer detalles de bajo nivel como la forma de acceso o las órdenes necesarios para lanzar una tarea en las máquinas de origen.

Para cada tarea de las que se presentan en la lista, el usuario tiene disponible tanto una descripción breve de la aplicación, como un documento de descripción detallada en el que se precisa cuál es el propósito, cuáles son los datos de entrada requeridos, qué resultados se van a generar, y cualquier otra información considerada de interés por el creador de la tarea.

Una vez que el usuario elige lanzar un trabajo de la lista, y selecciona los datos a enviar que estarán guardados en un archivo de texto o cualquiera que sea el formato requerido, el trabajo se encola para su ejecución en la máquina que alberga el trabajo en cuestión. Los datos se envían a la máquina y se planifica su ejecución.





The screenshot shows a web browser window with the URL 'http://hpc.unicam.ac.uk'. The page title is 'HPC - Google Chrome'. The main content area is titled 'List of completed jobs' and contains a table with columns: Name, Date, Status, Input, Results, and Log. The table lists several jobs, all with a status of 'completed'.

Name	Date	Status	Input	Results	Log
...	...	completed	...	...	...
...	...	completed	...	...	...
...	...	completed	...	...	...
...	...	completed	...	...	...
...	...	completed	...	...	...
...	...	completed	...	...	...
...	...	completed	...	...	...
...	...	completed	...	...	...
...	...	completed	...	...	...
...	...	completed	...	...	...

Figura 5: Consulta del estado de las tareas enviadas.

### 3.1.2. Consulta de estado

Dependiendo de la envergadura de la tarea y de la carga de procesamiento del nodo en el que se ejecuta, es posible que el tiempo de ejecución sea considerable. Por este motivo, el sistema se puede considerar asíncrono, ya que el usuario inicia la ejecución de una tarea, olvidándose de ella hasta que sea notificado por el entorno de algún evento (como la conclusión de la tarea o errores de ejecución).

No obstante, en cualquier momento se puede consultar el estado de las tareas enviadas con anterioridad. Una interfaz web muestra la lista de procesos lanzados por el usuario (ver Figura 5), con el estado de ejecución (en espera, ejecutándose, terminado, error, etc.). El usuario selecciona un trabajo de entre los disponibles, y se muestra información detallada.

### 3.1.3. Recuperación de resultados

Obviamente, el usuario debe poder acceder al resultado generado por sus trabajos, que se tendrá que almacenar de forma local (al menos temporalmente) ya sea en el servidor o en el propio nodo de computación. Dentro del perfil asociado al usuario, existe un apartado dedicado a las tareas ya concluidas en el cual se proporciona, para cada una de las tareas, un enlace a través del cual se pueden descargar tanto los datos de entrada, como el resultado generado y el registro (redirección de la salida estándar de la aplicación a un archivo de texto) que permite verificar el funcionamiento de la aplicación.

## 3.2. APLICACIONES

Actualmente se han introducido tres aplicaciones en el sistema. En principio, se contemplan dos tipos de ejecución según el origen de los datos a procesar: datos enviados por el usuario y datos almacenados en el servidor. Normalmente, para pe-

## Kicks and ELMs

Ernesto Fábregas

January 2016

### Job Description

This job searches across difference sets of experimental data to find correlated peaks (kicks and ELMs). Receive an input file (e.g. *id\_data.txt*) with the id number corresponding to the data sequence to be analyzed.

### Input Required

Since the data series are already stored in the server, this job expects to receive the *id* number corresponding to the particular data to be analyzed.

E.g:

d80264

### Output Generated

For each valid *id* number, the following files are generated:

- ELMs\_TBEO\_*id*.txt
- ELMs\_TC30\_*id*.txt
- KICKs\_*id*.txt
- fig1.eps
- fig2.eps

These files are stored into a zip file 'results.zip'.

Figura 6: Documentación asociada a una tarea, en la cual se describe el propósito, los datos requeridos y la salida que genera.

queños volúmenes de datos el usuario puede aportarlos directamente desde su ordenador o estación de trabajo. Sin embargo, para grandes volúmenes de datos es más adecuado almacenar estos en el servidor. De esta forma, sería necesario aportar simplemente un rango de datos o un identificador que permita al servidor saber qué subconjunto de los datos debe procesar.

Una de las tareas incorporadas al repositorio, desarrollada en Matlab, permite realizar el análisis de diferentes series temporales de datos obtenidos en experimentos de fusión, y que tienen como objetivo la búsqueda de *kicks* y ELMs. Como se menciona en el apartado anterior, la descripción breve de tarea se muestra dentro de la lista de tareas disponibles de los usuarios autorizados, y la descripción en detalle aparece en el documento correspondiente (Figura 6). Como resultado de la ejecución, se obtienen varios archivos de datos y gráficas (Figura 7).

En relación a las otras dos tareas disponibles, estas son dos ejemplos simples, pero completos, que se han añadido como plantilla para poder desarrollar con mayor facilidad futuras aplicaciones.

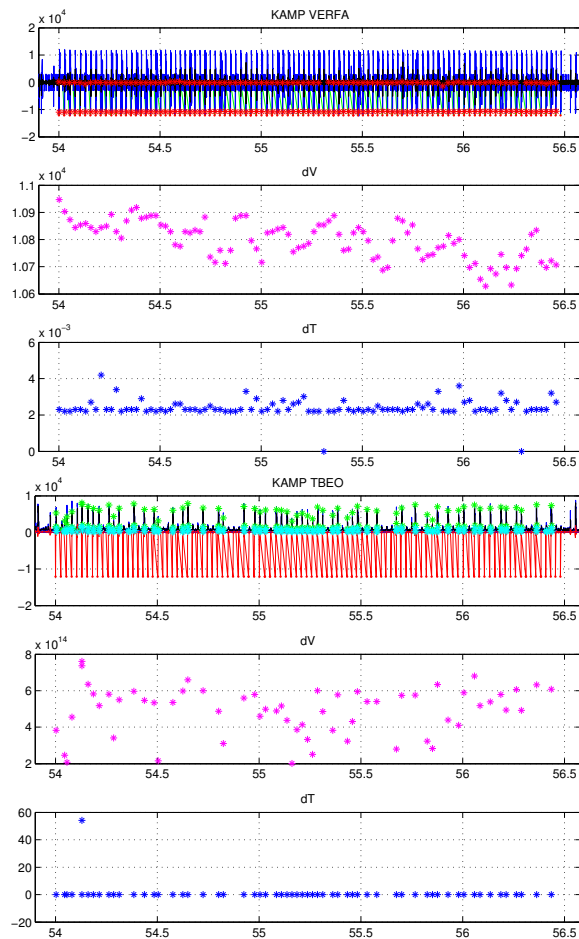


Figura 7: Resultados obtenidos en la ejecución de una tarea.

## 4. CONCLUSIONES

Se ha desarrollado un entorno colaborativo web que permite acceder a un repositorio de aplicaciones de procesamiento de datos obtenidos de experimentos de fusión. De esta manera, los usuarios no tienen necesidad de conocer la infraestructura subyacente o estar familiarizados con las órdenes de más bajo nivel necesarias para interactuar con estos sistemas.

La idea es tener un repositorio que contenga una amplia gama de aplicaciones reutilizables, fomentando de esa manera la colaboración entre diferentes grupos de investigación, que pueden beneficiarse por un lado de la compartición de conocimiento y esfuerzo de desarrollo y, por otro lado, de recursos computacionales. Aunque el sistema se encuentra todavía en fase inicial de desarrollo, es ya funcional y las primeras experiencias han sido satisfactorias, por lo que futuros esfuerzos se dedicarán al desarrollo de nuevas aplicaciones que puedan ser incorporadas al repositorio.

## Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Economía y Competitividad, dentro de los proyectos ENE2015-64914-C3-2-R, ENE2015-64914-C3-1-R, ENE2012-38970-C04-03 y DPI2014-55932-C2-2-R.

## Referencias

- [1] R. Castro, D. R. López, and J. Vega. An authentication and authorization infrastructure: the PAPI system. *Fusion Engineering and Design*, 81:2057–2061, 2006.
- [2] R. Castro, J. Vega, A. Portas, D. López, S. Balme, J. Theis, P. Lebourg, H. Fernandes, A. Neto, A. Duarte, F. Oliveira, F. Reis, K. Purahoo, K. Thomsen, W. Schiller, and J. Kadlecik. PAPI based federation as a test-bed for a common security infrastructure in EFDA sites. *Fusion Engineering and Design*, 83(2–3):486 – 490, 2008.
- [3] R. Dormido, H. Vargas, N. Duro, J. Sanchez, S. Dormido-Canto, G. Farias, F. Esquembre, and S. Dormido. Development of a web-based control laboratory for automation technicians: The three-tank system. *Education, IEEE Transactions on*, 51(1):35–44, 2008.
- [4] G. Farias, R. De Keyser, S. Dormido, and F. Esquembre. Developing networked control labs: A matlab and easy java simulations approach. *Industrial Electronics, IEEE Transactions on*, 57(10):3266–3275, 2010.
- [5] J. Vega, E. Sánchez, A. Portas, A. Pereira, A. Mollinedo, J. Muñoz, M. Ruiz, E. Barrera, S. López, D. Machón, R. Castro, and D. López. Overview of the TJ-II remote participation system. *Fusion Engineering and Design*, 81(15–17):2045 – 2050, 2006.