

WEBPAPI: PLATAFORMA WEB PARA LA MEDICIÓN DE EVENTOS EN EL PROCESADOR

*Andrés Jesús Mesones, José Manuel Palomares, Manuel A. Ortiz, Ezequiel Herruzo,
José Ignacio Benavides*

Escuela Politécnica Superior. Universidad de Córdoba. el1bebej@uco.es.

RESUMEN

El trabajo que se presenta muestra un sistema web realizado por el área de Arquitectura y Tecnología de Computadores de la Universidad de Córdoba. Este sistema (WebPAPI) es utilizado como herramienta docente en las prácticas de algunas de las asignaturas que imparte el área y está basada en una arquitectura propia cliente-servidor en internet. También puede ser utilizada por usuarios “no alumnos” para evaluar los programas que utilizan. El sistema permite a los usuarios que lo utilicen poder analizar sus programas desde el punto de vista de utilización de las distintas unidades funcionales del procesador. Para ello se hace uso de las librerías PAPI (Performance API) disponibles de forma gratuita en internet. Nuestro sistema facilita el uso de estas librerías y permite que se puedan analizar los contadores hardware del procesador para uno o varios programas, con distintas opciones de compilación; también permite que estos análisis puedan realizarse en distintos procesadores de diferentes máquinas de una forma sencilla.

1. INTRODUCCIÓN

En este artículo vamos a dar una visión general de la Plataforma Web para la Medición de Eventos en el Procesador. Para ello, en primer lugar se explica qué es y para que sirven las librerías PAPI, después se habla del sistema implementado; en los siguientes puntos veremos primero una descripción general del sistema, indicando de qué partes se compone y su funcionamiento; seguido a esto se presentan las partes destinadas al usuario, al administrador del sistema, y la parte que contiene la funcionalidad principal del sistema. Para finalizar veremos un ejemplo de uso del sistema.

PAPI (Performance Application Programming Interface) son unas librerías de programación [5] que nos permiten realizar pruebas de rendimiento de hardware con código creado por nosotros mismos; usando estas librerías podemos crear benchmarks, medir los FLOPS, los MIPS, etc. Estas librerías constan de 40 funciones de bajo nivel y 6 funciones de alto nivel. Con todas estas funciones se nos proporciona acceso a los contadores hardware de nuestro procesador y con ellos se pueden medir ocurrencias de diferentes eventos para obtener las medidas de rendimiento que deseamos [1], [2], [3], [4]. PAPI nos da soporte para contadores hardware en Windows NT (y sucesores) y Linux, incluyendo este último soporte para varias arquitecturas de procesador.

Tratamos de desarrollar un software que ayude al análisis de rendimiento de procesadores, con la idea de proveer un servicio web mediante el que realizar medidas de rendimiento en diferentes arquitecturas de procesador, desde la misma herramienta y sin necesidad de que el usuario posea dichas máquinas; de forma que el usuario sólo necesitará un ordenador con acceso a internet y un navegador web.

En la siguiente sección se describe el sistema de una forma general. La sección 3 describe la parte del usuario, el subsistema de administrador se indica en la sección 4. En la sección 5 se indica el funcionamiento general del sistema. Un ejemplo de uso se muestra en la

sección 6 y finalmente las conclusiones y referencias bibliográficas se presentan en las secciones 7 y 8 respectivamente.

2. DESCRIPCIÓN GENERAL DEL SISTEMA

El sistema consta de dos subsistemas; el primero es el servidor web que atiende a los alumnos/usuarios, y el segundo es el que realiza las mediciones y atiende las peticiones del servidor anterior.

El primer subsistema es un servicio web (en adelante llamado subsistema servidor), encargado de proporcionar un interfaz para la recolección de tests, organizar las mediciones y mostrar resultados; controla el acceso a todo el sistema por los usuarios, y mantiene una base de datos con información acerca de las máquinas disponibles, los eventos existentes, las mediciones realizadas, y más información, se describirá más adelante al ver los apartados de usuario y administrador. Para esta parte se ha utilizado el servidor web Apache y el gestor de bases de datos MySQL [6]; el sistema web está implementado en PHP [7], por su cómoda integración en el servidor Apache, y por las facilidades que este lenguaje proporciona para el acceso a bases de datos con el gestor MySQL.

El segundo subsistema (en adelante llamado subsistema máquina) es un servidor implementado en Java [8], para facilitar la portabilidad, y se encarga de la recepción de código fuente perteneciente a la medición, la compilación del mismo, y el envío de resultados al servidor central; al dar los resultados de la medición informa sobre si ha habido algún error en la compilación de la medición. Este servidor requiere tener instalado en la máquina donde se ejecute la librería PAPI y el compilador GNU del lenguaje C (gcc).

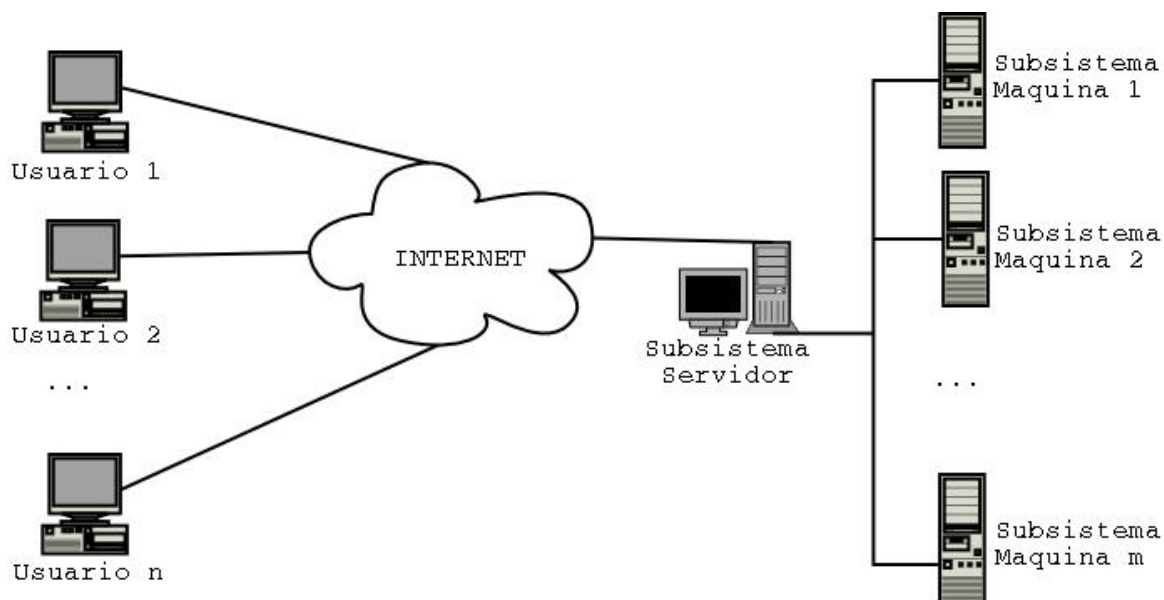


Figura 1. Esquema General del Sistema.

En todo el sistema web se utiliza la misma estructuración de contenidos, exceptuando la pantalla de bienvenida; a saber:

- la parte central se divide en tres, a la izquierda se encuentra la ayuda correspondiente a cada pantalla, en el centro se visualizan los datos, los formularios, en definitiva el contenido real de dicha pantalla, y a la derecha el menú, que dependiendo del tipo de usuario contendrá unos ítems u otros
- abajo un pie de página con el nombre del autor

En la figura 1 se observa como se distribuye todo el sistema; por un lado en el subsistema servidor se aloja el servicio web y la base de datos, y en cada una de los subsistemas máquina está corriendo el servidor de mediciones descrito en el segundo párrafo de este apartado.

Funcionamiento: los clientes se conectan a través de internet al servidor central, utilizando para ello la interfaz que proporciona el sistema web ya descrito, una vez conectados podrán crear tests, realizar mediciones, etc., y es este servidor central el que se encarga de todas las transacciones con las máquinas para realizar mediciones, recibir los resultados y almacenar los datos necesarios en la base de datos.

3. APLICACIÓN: VISTA DEL USUARIO/ALUMNO

En este apartado se describe el funcionamiento general de la parte del sistema web para el usuario, a excepción del manejo de tests y realización de pruebas, que se detallará en un apartado destinado a ello.

En primer lugar, cuando el usuario accede al sistema con su navegador web visualiza la siguiente pantalla (figura 2).

aquí.' To the right of the login fields is a 'Bienvenido' message: 'Bienvenido a la plataforma web para la medición de eventos en el procesador. Si deseas realizar mediciones ingresa tu login y password para acceder al sistema completo. Si no posees un usuario solicítalo en el registro. A disposición tienes ayuda y estadísticas públicas como usuario anónimo. Esperamos que te resulte de utilidad.' The footer of the page reads 'Arquitectura de Computadores.- Universidad de Córdoba'."/>

.. PLATAFORMA WEB PARA MEDICION DE EVENTOS DEL PROCESADOR ..

09-02-2004 | 13:02:55

LOGIN

login:

pepe

password:

Entrar

Registro

Si aún no tienes una cuenta de usuario, puedes solicitar una [aquí](#).

..Bienvenido

Bienvenido a la plataforma web para la medición de eventos en el procesador.
Si deseas realizar mediciones ingresa tu login y password para acceder al sistema completo. Si no posees un usuario solicítalo en el registro.
A disposición tienes ayuda y estadísticas públicas como usuario anónimo.
Esperamos que te resulte de utilidad.

Arquitectura de Computadores.- Universidad de Córdoba

Figura 2. Bienvenida al sistema.

En esta pantalla es donde el usuario deberá introducir su login y su password del sistema, para entrar en la parte destinada al usuario; una vez introducidos dichos datos y validados el usuario accede a la pantalla que se muestra en la figura 3, es un listado de los tests que el usuario ya ha introducido y que permanecen guardados para poder realizar futuras mediciones con ese código fuente. (El uso de esta pantalla se detalla en apartados posteriores)

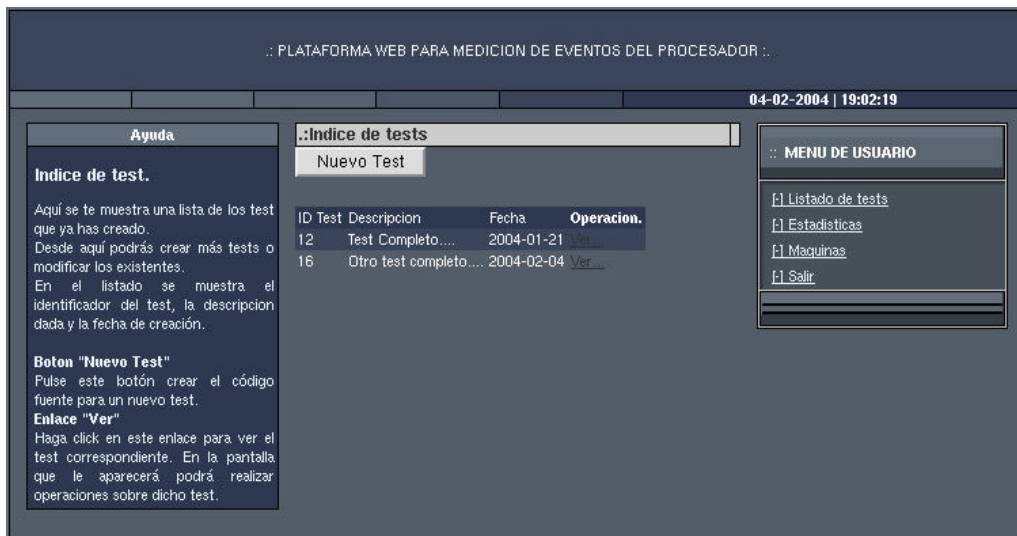


Figura 3. Listado de tests.

En el menú de usuario, podemos ver las diferentes opciones que tenemos, y que explicaremos a continuación, a saber: Listado de tests, Estadísticas, Máquinas, y Salir. El listado de tests es la misma pantalla que se acaba de explicar. La opción Salir es la que permite al usuario finalizar la sesión en el sistema.



Figura 4. Estadísticas del usuario.

La siguiente pantalla a tratar es la correspondiente a la opción Estadísticas (figura 4); en esta pantalla se muestran al usuario estadísticas acerca de las mediciones que ya ha realizado:

- **Eventos:** es un conteo de cuantas veces se ha utilizado cada evento en las mediciones realizadas por el usuario que se llevaron a cabo con éxito. Se muestran sólo los eventos que se han utilizado al menos una vez, para el resto el conteo sería igual a 0.
- **Maquinas:** es un conteo de cuantas veces se ha utilizado cada máquina en las mediciones realizadas por el usuario que se llevaron a cabo con éxito. Se muestran sólo las máquinas que el usuario ha utilizado al menos una vez, para el resto de máquinas el resultado del conteo es 0.
- **Usuario éxito mediciones:** es un conteo de cuantas mediciones del total de las realizadas por el usuario se llevaron a cabo con éxito y cuantas fracasaron.



Figura 5. Información de máquinas para el usuario.

La última pantalla que vemos en la parte del usuario es una lista de fichas sobre las máquinas disponibles para realizar mediciones; al usuario se le muestra para cada máquina el nombre, el sistema operativo, y un campo descriptivo donde el administrador debe insertar información acerca del hardware que tiene esa máquina.

4. APLICACIÓN: VISTA DEL ADMINISTRADOR

Aquí se describe la parte destinada a la administración del sistema; para ello el administrador debe ingresar en el sistema pero accediendo al archivo “admin/index.php”, donde tendrá que introducir su login y su password. Una vez validado el login y el password el administrador podrá acceder a las opciones que detallamos en los siguientes párrafos.



Figura 6. Estadísticas para el administrador.

Estadísticas Generales:

- **Eventos:** Muestra el número total de veces que se ha usado cada evento sobre el total de mediciones que se llevaron a cabo con éxito en el sistema.
- **Maquinas:** Muestra el número total de veces que se ha usado cada máquina sobre el total de mediciones que se llevaron a cabo con éxito en el sistema.
- **Éxito en mediciones:** Muestra los totales de mediciones que tuvieron éxito y las que fracasaron.
- **Conexiones:** Cuenta las conexiones realizadas ordenadas por franjas horarias.

Estadísticas por usuarios: son las mismas que las generales pero agrupadas por usuarios, en esta opción también se pueden ver sólo las estadísticas de un usuario, o ver los datos de todas las conexiones realizadas por el mismo.

Gestión de usuarios: desde esta pantalla el administrador podrá crear, modificar o eliminar usuarios del sistema, así como gestionar las solicitudes de registro como usuario en el sistema. Al crear o modificar un nuevo usuario, el administrador también puede crear más administradores del sistema.



Figura 8. Gestión de maquinas.



Figura 7. Gestión de usuarios.

Gestión de máquinas: desde esta pantalla el administrador puede crear, modificar o eliminar máquinas del sistema; dichas máquinas son en las que se pueden realizar mediciones; antes de añadir la máquina el administrador verifica si está funcionando el servidor de mediciones en la misma.

5. FUNCIONAMIENTO (USO DEL LISTADO DE TEST)

En este apartado vamos a describir el uso principal del sistema, el manejo de tests y la realización de mediciones.

Para ello partimos del listado de tests (figura 3); en esta pantalla tenemos un listado de los tests que el usuario ya introdujo en el sistema. Desde aquí podemos crear un nuevo test o pasar a ver uno de los ya introducidos; la pantalla para crear un nuevo test es muy similar a la

de ver test y a la de modificar; explicaremos en detalle la pantalla donde se visualiza y se explicarán las diferencias a encontrar en la correspondiente a la creación y a la modificación de tests.

En la pantalla que aparece al pulsar “ver test” (figura 9) se nos muestra el identificador del test, la fecha de inserción, y los siguientes campos:

- Descripción: es una descripción del test donde el usuario puede anotar cosas relacionadas con el test.
- Código Previo: es el trozo de código fuente previo al código sobre el que se medirán los eventos; aquí se incluyen declaraciones de tipos, la apertura de la función main, la declaración de variables y cualquier operación que se realice antes de la parte donde se medirán los eventos (p.ej. reserva de memoria dinámica).
- Código Test: es el trozo de código donde se medirán los eventos hardware.
- Código Posterior: es el código fuente restante; aquí se debe llegar al menos hasta el cierre de la función main; es importante no cerrar dicha función en el trozo anterior, sino en éste, para el correcto funcionamiento.

Estos mismos campos aparecen en las pantallas correspondientes a Modificar Test y a Nuevo Test, pero en estas últimas, las cajas de texto no son de sólo lectura; en Nuevo Test dichas cajas están vacías, y en Modificar Test contienen la información del test a modificar.

Desde esta pantalla de visualizado de test podremos eliminar el test, modificarlo, utilizarlo o visualizar resultados de las mediciones ya realizadas con este test. Pasemos a ver la pantalla de visualizado de resultados.

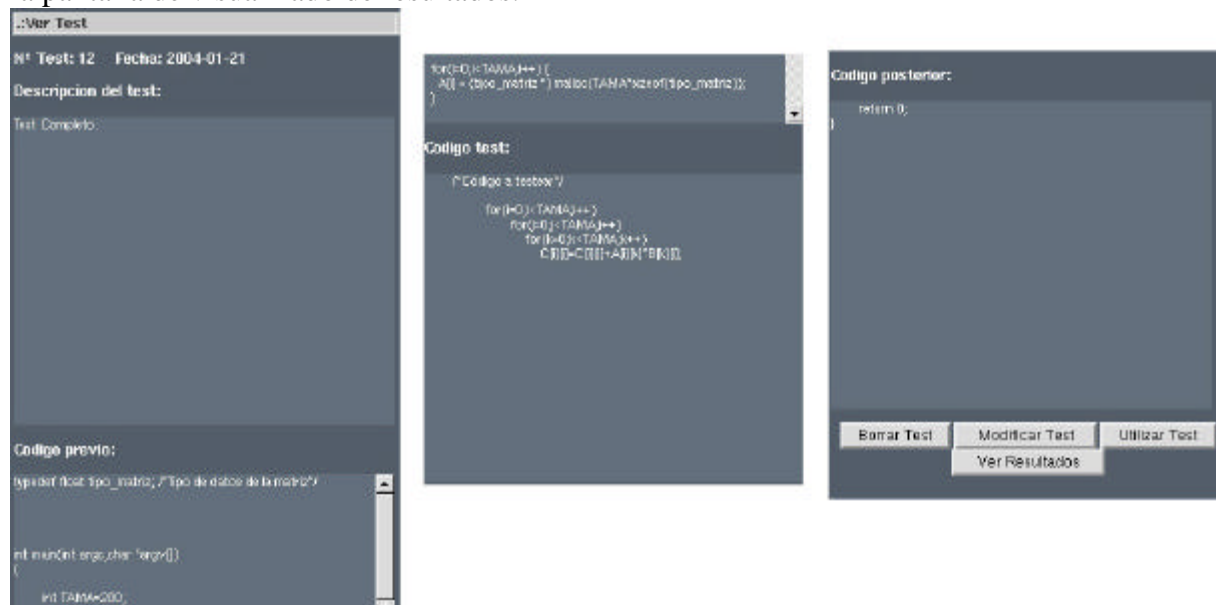


Figura 9. Ver Test. (figura partida, de izquierda a derecha en la figura es de arriba abajo en la pantalla).

Si pulsamos en Ver Resultados se nos muestra la pantalla (figura 10) de visualización de resultados, en la cual se ven los datos acerca de todas las mediciones realizadas con este test; dichos datos son: identificador de la medición, un campo que indica si ha habido éxito al medir, la fecha de realización, el nombre del evento que se midió, el nombre de la máquina donde se midió y la cifra numérica correspondiente al resultado obtenido.

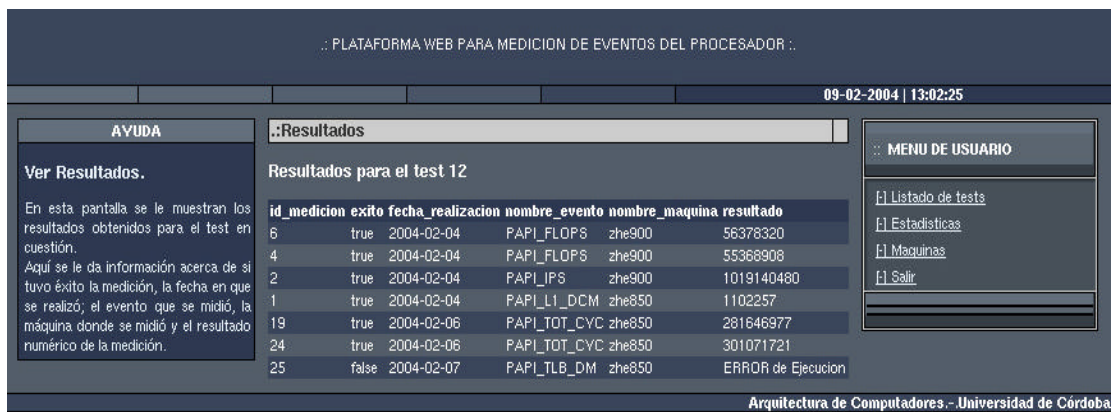


Figura 10. Resultados.

Lo último que nos queda por ver es cómo se lleva a cabo una medición de eventos utilizando uno de los tests que ya tenemos en el sistema; para ello, en la pantalla Ver Test (figura 9) pulsamos el botón “Utilizar Test”, y seguimos los tres siguientes pasos (figura 11):

-Paso 1; seleccionamos la máquina sobre la que se realizará la medición y el grupo de eventos donde se encuentra el evento que deseamos medir, y pulsamos siguiente.

-Paso 2; seleccionamos el evento que deseamos medir, pulsamos siguiente y esperamos al resultado, el usuario no deberá tocar el botón actualizar de su navegador ni cerrarlo para una correcta recepción del resultado.

-Paso 3; visualizamos el resultado, en este punto se nos muestran los datos acerca de la medición, y el resultado obtenido para la misma; estos datos ya se han guardado en la base de datos y podremos volver a verlos en el listado de resultados (figura 10).



Figura 11. Pasos para realizar medición.

6. EJEMPLO DE USO

En este último apartado vamos a ver un ejemplo de uso del sistema, utilizando un test basado en el siguiente código fuente:

EJEMPLO DE CODIGO FUENTE A UTILIZAR EN TEST

```
#include<stdio.h>
typedef float tipo_matriz; /*Tipo de datos de la matriz*/
int main(int argc,char *argv[])
{
    int TAMA=200; /*Tamaño de la matriz*/
    tipo_matriz **A,**B; /*Matrices*/
    int i=0,j=0,k=0; /*Indices*/
    tipo_matriz **C; /*Matriz destino*/
    float semilla; /*Semilla para obtener numeros aleatorios*/

    /*Reservamos memoria para las matrices*/
    A=(tipo_matriz **)malloc(TAMA*sizeof(tipo_matriz *));
    for(i=0;i<TAMA;i++) {
```

```

    A[i] = (tipo_matriz *) malloc(TAMA*sizeof(tipo_matriz));
}

B=(tipo_matriz **)malloc(TAMA*sizeof(tipo_matriz *));
for(i=0;i<TAMA;i++) {
    B[i] = (tipo_matriz *) malloc(TAMA*sizeof(tipo_matriz));
}

C=(tipo_matriz **)malloc(TAMA*sizeof(tipo_matriz *));
for(i=0;i<TAMA;i++) {
    C[i] = (tipo_matriz *) malloc(TAMA*sizeof(tipo_matriz));
}

/*Obtenemos la semilla, que es a su vez aleatoria*/
semilla = srand(getpid());
/*Cargamos las matrices con números aleatorios*/
for(i=0;i<TAMA;i++) {
    for(j=0;j<TAMA;j++) {
        A[i][j] = (int) rand(semilla)%100;
    }
}

for(i=0;i<TAMA;i++) {
    for(j=0;j<TAMA;j++) {
        B[i][j] = (int) rand(semilla)%100;
    }
}
/*Realizamos las operaciones*/
    for(i=0;i<TAMA;i++)
        for(j=0;j<TAMA;j++)
            for(k=0;k<TAMA;k++)
                C[i][j]=C[i][j]+A[i][k]*B[k][j];
return 0;
}

```

Leyenda de los colores:

- Negro: código que no debemos insertar.
 - Azul: código previo al test.
 - Rojo: código a testear. (Código test)
 - Verde: código posterior.
-

Tabla 1. Código fuente para el ejemplo de uso.

Las divisiones necesarias se muestran en el código fuente, mediante la coloración del texto. Pasemos a ver el uso del sistema; emplearemos para este ejemplo un usuario que no ha introducido aún ningún test en el sistema.

En la pantalla de bienvenida introducimos el login y el password para este usuario y pasamos al listado de test, que en este caso está vacío. (En las figuras para el ejemplo mostraremos solamente el área de trabajo).

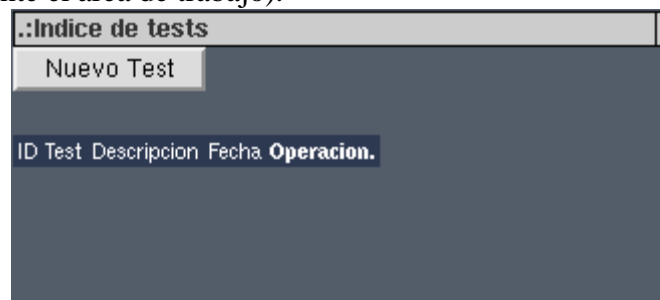


Figura 12. Ejemplo de uso: Listado de test

Comenzaremos introduciendo un nuevo test, para lo que en el listado de tests pulsamos el botón “Nuevo Test”, con lo que iremos a la pantalla correspondiente, aparece un formulario vacío, donde pegaremos los trozos de código según se ha explicado previamente, en los diferentes campos que aparecen en esta pantalla, quedando los campos como sigue (la figura está partida en 3 trozos):

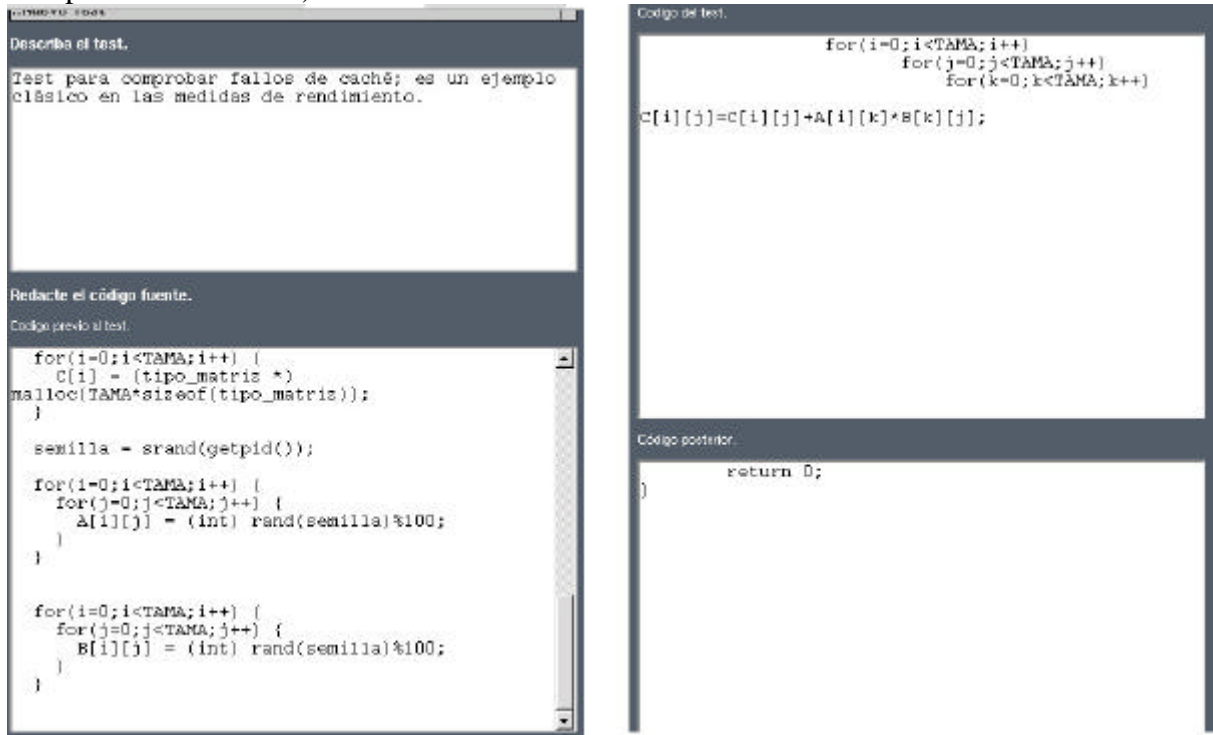


Figura 13. Ejemplo de uso: Nuevo Test

Una vez rellenados los campos pulsamos el botón Enviar para guardar el test en el sistema; aparecerá una pantalla confirmando si se ha guardado el test.

Ahora iremos de nuevo al listado de test, donde ahora debe aparecer el test que ya hemos guardado en el sistema; pulsamos en el enlace Ver de dicho test para visualizar los datos correspondientes a dicho test y poder acceder a las posibles acciones a realizar con el mismo.

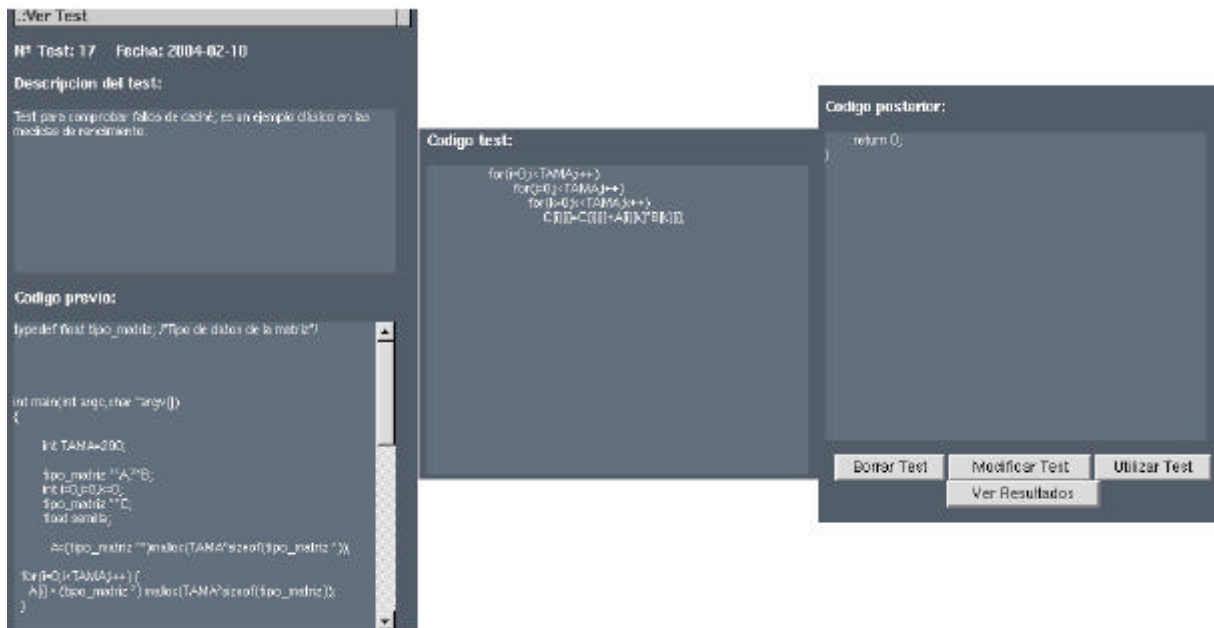


Figura 14. Ejemplo de uso: Ver Test

En esta pantalla pulsaremos el botón Utilizar Test para poder efectuar una medición utilizando dicho test.



Figura 15. Ejemplo de uso: Pasos al realizar la medición

Pasamos a una pantalla en la que debemos elegir la máquina, en este caso "zhe900", y el grupo de eventos donde se encuentra el evento a medir; escogemos el grupo "Operaciones en punto flotante"; pulsamos siguiente y aparece una pantalla que muestra los datos anteriormente seleccionados y nos da a elegir los eventos del grupo elegido; seleccionamos "PAPI_FLOPS" para medir el número de operaciones en punto flotante por segundo; pulsamos siguiente y

esperamos, en esta espera no debemos actualizar la página si queremos recibir el resultado correctamente. La última pantalla que aparece nos muestra el resultado para esta medición, además de los datos que la conforman.

7. CONCLUSIONES

Se ha diseñado y realizado un sistema que puede ser utilizado por distintos tipos de usuarios y para distintos fines. Puede ser utilizado por alumnos para realizar las prácticas de asignaturas en las que se analice el rendimiento de procesadores, se estudie el uso de las unidades funcionales, o cómo afectan las distintas opciones de optimización del compilador en el código máquina generado y cómo éste utiliza el hardware del procesador.

También puede ser utilizado por los alumnos para realizar trabajos técnicos sobre materias de las asignaturas, como el análisis de uso de unidades funcionales en la ejecución de benchmarks en distintos procesadores. Este tipo de trabajo es el que puede ser realizado utilizando este software por usuarios no alumnos (profesores, becarios, ...).

El sistema incorpora una batería de estadísticas de empleo clasificada por usuarios, eventos medidos y máquinas utilizadas. El uso de estas estadísticas permite al administrador incorporar nuevas máquinas para hacer mediciones, ampliar el espacio del servidor destinado al sistema, cambiar el sistema servidor por otro más potente, etc.

El empleo de este sistema para trabajos técnicos ha sido muy satisfactorio por la información adicional que usuario puede incorporar (sobre todo estadísticas) a los trabajos realizados. Próximamente se realizará el uso del sistema en las prácticas de algunas asignaturas donde se espera pueda ser muy beneficioso para los alumnos por varios motivos, 1º la sencillez de manejo del sistema; 2º el análisis de programas reales sobre máquinas físicas; 3º la disponibilidad del sistema a través de internet.

8. BIBLIOGRAFÍA

- [1] D.F.Bacon, S.L. Graham and O.J. Sharp, *Compiler Transformations for High Performance Computing*, **ACM Computing Surveys**, Vol. 26, No. 4, December 1994, pp.345-420.
- [2] John L. Hennessy and David A. Patterson, *Arquitectura de Computadores: Un Enfoque Cuantitativo*, **McGraw-Hill**, 1993. ISBN: 84-7615-912-9
- [3] David A. Patterson and John L. Hennessy, *Estructura y Diseño de Computadores*, **Editorial Reverté**, 1999. ISBN: 84-291-2619-8
- [4] J.R. Allen and K. Kennedy, *Optimizing Compilers for Modern Architectures: A Dependence-based Approach*, **Morgan Kaufmann Publishers**, October 2001. ISBN: 1-55860-286-0
- [5] <http://icl.cs.utk.edu/projects/papi/>. Website oficial de PAPI.
- [6] <http://www.mysql.com>. Website oficial de MySQL.
- [7] <http://www.php.net>. Website oficial de PHP y documentación de la librería de funciones.
- [8] <http://java.sun.com>. Website oficial de JAVA y documentación del lenguaje.