

Universidad de Valladolid  
E.T.S Ingeniería Informática  
Grado en Ingeniería Informática [Tecnologías de la información]

**Curso 2017/2018**  
**Evaluación de Sistemas Informaticos**  
**Práctica de Laboratorio 1 (2ª parte)**  
**Evaluación del rendimiento de un servidor web.**

Grupo de Laboratorio 07  
Esteban Pellejero, Sergio  
González Bravo, Miguel  
González Pérez, Daniel  
Rabadán Martín, Borja

# Índice

<b>1. Introducción</b>	<b>3</b>
<b>2. Sistema en evaluación</b>	<b>3</b>
<b>3. Planificación y diseño de las pruebas</b>	<b>4</b>
3.1. Condiciones del experimento . . . . .	4
3.2. Escenarios . . . . .	4
3.3. Métricas utilizadas . . . . .	6
<b>4. Preparación del entorno de pruebas - JMeter</b>	<b>6</b>
4.1. Instalación y funcionamiento básico de JMeter . . . . .	6
4.2. Configuración de JMeter . . . . .	7
<b>5. Ejecución de las pruebas</b>	<b>9</b>
<b>6. Análisis exploratorio de datos</b>	<b>10</b>
6.1. Escenario 1 . . . . .	10
6.1.1. Prueba 1 - 1 hilo y 150 steps . . . . .	10
6.1.2. Prueba 2 - 150 hilos y 1 setps . . . . .	12
6.1.3. Prueba 3 - 150 hilos y 1 steps . . . . .	14
6.1.4. Prueba 4 - 150 hilos y sin fin . . . . .	15
6.2. Escenario 2 . . . . .	17
6.2.1. Prueba con 10 hilos . . . . .	17
6.2.2. Prueba con 20/40/70hilos . . . . .	19
6.2.3. Prueba 5 - 130 hilos . . . . .	19
6.2.4. Prueba 6 - 240 hilos . . . . .	21
6.2.5. Prueba 7 - 440 hilos . . . . .	24
<b>7. Conclusiones</b>	<b>25</b>
<b>8. Bibliografía</b>	<b>27</b>
<b>Appendices</b>	<b>28</b>
<b>ANEXO ANEXO I.</b>	<b>28</b>
<b>ANEXO ANEXO II.</b>	<b>28</b>
<b>ANEXO ANEXO III.</b>	<b>31</b>

## 1. Introducción

En esta segunda parte de la práctica se va a evaluar la productividad y los tiempos de respuesta de una aplicación en ejecución en un servidor web con el software Apache instalado. El estudio se limitará a evaluar las variables descritas para los diferentes tipos de sesiones de los usuarios (escenarios) bajo diferentes intensidades de carga.

## 2. Sistema en evaluación

El sistema en evaluación es el mismo que se ha descrito en la primera parte de la práctica, por lo menos a nivel hardware, así que no se va a entrar en detalle.

A nivel software existen algunos cambios, ya que se va a ejecutar la aplicación WebCalendar en nuestro servidor Apache, y para ello vamos a hablar primero sobre el funcionamiento y configuración de esta aplicación, cómo funciona y el software en el que se basa (PHP).

- **PHP:** La información relativa a php se puede obtener de una de las páginas de prueba de la primera parte de esta práctica, gracias a que tiene la función `phpinfo()`. Se dispone de la versión 7.0.22 para esta práctica.
- **WebCalendar:** Es software libre, desarrollado en PHP con el objetivo de proporcionar una herramienta colaborativa para el manejo común de un calendario. Esta aplicación ofrece la posibilidad de crear un calendario multiusuario, donde estos pueden crear eventos, añadir usuarios a esos eventos, crear grupos en los calendarios, etc.

WebCalendar necesita PHP para funcionar, en concreto una versión mayor que la 4.1.0. En nuestro caso se dispone de la versión 7, por lo que no debería haber ningún problema.

Los principales datos de configuración de WebCalendar se pueden ver en la **Figura 1**

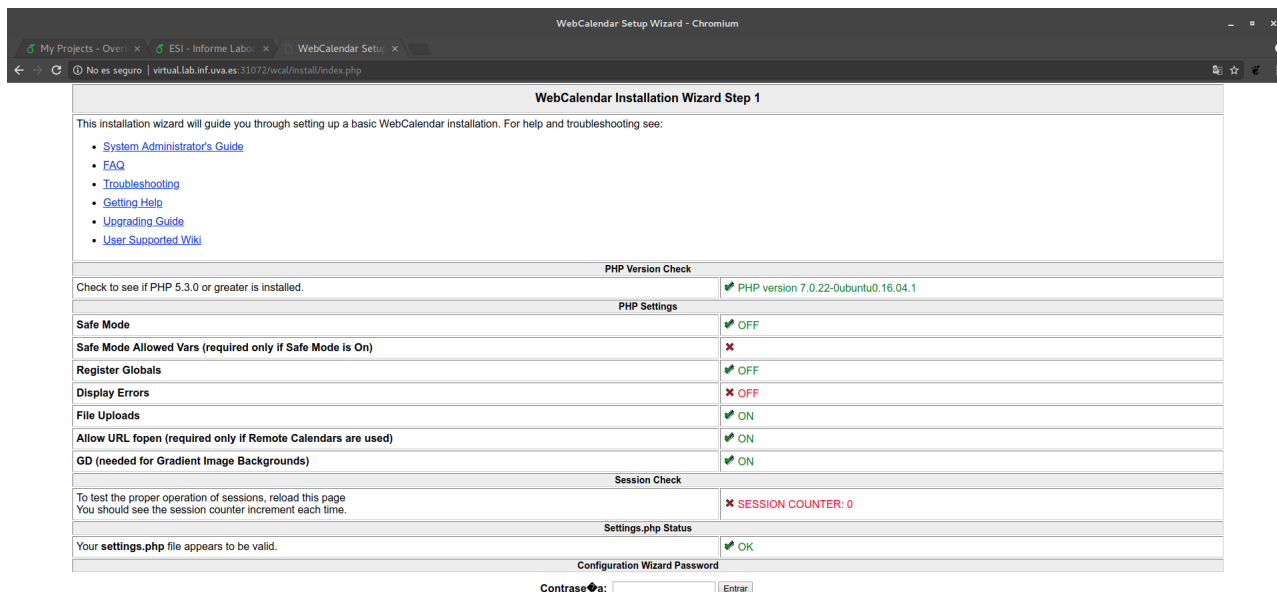


Figura 1: Configuración básica de WebCalendar

### 3. Planificación y diseño de las pruebas

La carga de trabajo de un sistema web se caracteriza en términos de sesiones de usuario. Una sesión es un conjunto de peticiones de usuario-respuesta del sistema, encadenadas y relacionadas, con el fin de obtener un servicio del sistema. Por su parte, WebCalendar presenta la funcionalidad básica y típica de una aplicación de agenda: identificación de usuario, consulta, inserción de un nuevo apunte, edición de una entrada existente y borrado.

En nuestro modelo de pruebas se va a tener en cuenta solo dos tipos de sesiones, una ligera y otra más cargada, para ver cómo se comporta el sistema en situaciones normales y en situaciones de estrés y congestión. Ambos escenarios se verán sometidos a diferentes intensidades de carga. Por último, se detallarán las conclusiones de los datos obtenidos.

#### 3.1. Condiciones del experimento

A nivel de red y de máquina las condiciones son las mismas que las expuestas en la primera parte de la práctica. Por lo que para no alargar el documento no vamos a repetir aquí las condiciones.

#### 3.2. Escenarios

A continuación se expone una descripción formal de los escenarios que simularán las sesiones de los usuarios.

<b>Acción</b>	<b>Descripción</b>
Título	Consulta de apunte
Resumen	El sistema muestra un evento para una fecha seleccionada junto con los detalles del evento
Precondición	Ninguna
Postcondición	El sistema muestra la información del evento
Secuencia Principal	1 - El actor usuario selecciona una semana en la pesataña de semanas 2 - El sistema muestra la semana elegida 3 - El usuario selecciona un evento 4 - El sistema muestra los detalles del evento 5 - El usuario introduce en el campo buscar un evento 6 - El sistema muestra todas las fechas en las que tiene lugar esa actividad 7 - El usuario selecciona la pestaña Mi calendario >> home 8 - El sistema muestra la página principal de la aplicación
Secuencia Alternativa	Ninguna
Secuencia Excepcional	Ninguna

Figura 2: Caso de Uso: *Consulta de apunte*

<b>Acción</b>	<b>Descripción</b>
Título	Modificar apunte
Resumen	El sistema muestra un evento para una fecha seleccionada junto con los detalles del evento y además guarda la información modificada por el usuario.
Precondición	Ninguna
Postcondición	El sistema almacena la información del evento modificada
Secuencia Principal	1 - El actor usuario selecciona identificarse (login) 2 - El sistema pide el usuario y la contraseña 3 - El usuario introduce el usuario, la contraseña y selecciona entrar 4 - El sistema muestra el calendario del usuario 5 - El usuario selecciona una semana 6 - El sistema muestra los eventos para la semana elegida 7 - El usuario selecciona un evento 8 - El sistema muestra la información del evento seleccionado 9 - El usuario selecciona editar el evento 10 - El sistema muestra la pantalla de edición 11 - El usuario edita el evento y confirma su edición 12 - El sistema muestra la semana con el evento modificado 13 - El usuario selecciona logout para salir de su sesión 14 - El sistema muestra la página de logout 15 - El usuario selecciona acceder al calendario público 16 - El sistema vuelve a la página principal
Secuencia Alternativa	Ninguna
Secuencia Excepcional	Ninguna

Figura 3: Caso de Uso: *Modificar apunte*

### 3.3. Métricas utilizadas

## 4. Preparación del entorno de pruebas - jmeter

### 4.1. Instalación y funcionamiento básico de jmeter

Para ara medir el rendimiento de la aplicación en nuestro servidor web, vamos a utilizar una herramienta incorporada en el paquete de Apache, JMeter (versión 3.3).

La instalación de JMeter es muy sencilla, simplemente hay que descargar el software de la página [http://jmeter.apache.org/download\\_jmeter.cgi](http://jmeter.apache.org/download_jmeter.cgi), moverlo al directorio elegido, descomprimirlo y ejecutarlo.

Una vez instalado, a continuación se expone una breve explicación de su funcionamiento. Inicialmente debe definirse un plan de pruebas y dentro de ese plan de pruebas un grupo de hilos. Este grupo de hilos simulará la cantidad de usuarios que a su vez simulan los casos de uso descritos anteriormente. Se pueden añadir también algunos elementos de configuración de internet o de formato de las peticiones. Después se añade un controlador y dentro de él los muestreadores, que contienen la realización de los casos de uso descritos anteriormente. (**Figura 2 y Figura 3**). Todo este procedimiento se puede guardar en una sesión, para poder lanzar distintas pruebas sobre el mismo. Por último, una vez grabada la sesión, añadimos los receptores (listeners), que son los van a capturar la información propia de las pruebas, para su posterior análisis. Ahora se puede configurar el número de hilos para cada prueba. La estructura básica de funcionamiento de JMeter se puede ver en la **Figura 4**

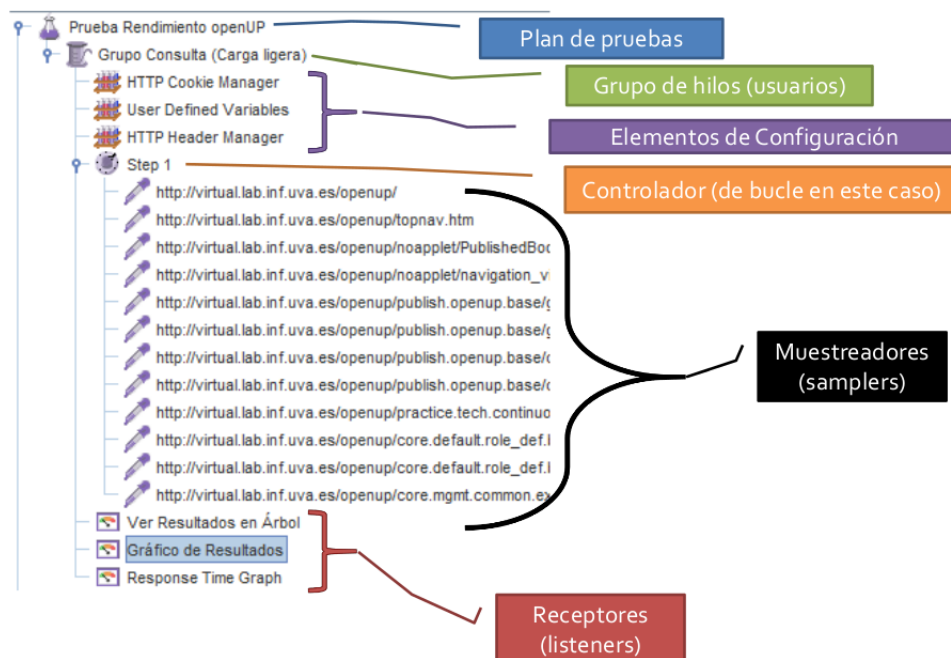


Figura 4: Funcionamiento básico de Jmeter

## 4.2. Configuración de jmeter

La configuración de JMeter es la misma que se nos proporciona en las transparencias de laboratorio, por lo que se expone un breve resumen:

- **Añadir un grupo de Hilos al plan de pruebas:** previamente creado el plan de pruebas añadimos un grupo de hilos. Ahí configuramos el número de hilos (cantidad de usuarios), periodo de subida y la cantidad de veces que se ejecuta el test.
- **Añadimos los valores de las peticiones http:** en nuestro caso nuestro servidor es `virtual.lab.inf.uva.es` y el puerto 31072.
- **Añadimos un muestreador:** en este caso para peticiones de tipo HTTP.
- **Añadir receptores:** añadimos las vistas de los diferentes resultados que queremos obtener. En nuestro caso hemos añadido: ver resultados en árbol, generar resumen de resultados, gráfico de resultados, response-time graph, informe agregado, reporte resumen y dos plugins para la monitorización de los recursos del servidor.
- **Grabación de sesiones:** la grabación de las sesiones es un proceso delicado. Primero tenemos que configurar JMeter para indicarle la correcta grabación de las sesiones, para ello primero tenemos que añadir un controlador de grabación al grupo de hilos y un servidor proxy HTTP al banco de trabajo. En el controlador de grabación tenemos que especificar qué tipos de páginas nos interesan y cuales no, para hacer un primer filtro.

El servidor proxy es para que toda la información que nosotros ejecutemos en el navegador, pase primero por el servidor proxy levantado por JMeter, y luego salga a la red. Gracias al paso intermedio por JMeter, este puede registrar todas las interacciones que hacemos en la página web y las respuestas recibidas.

La configuración del servidor proxy hay que hacerla tanto en JMeter como en el navegador Firefox, que será el utilizado para simular las sesiones. El servidor se ejecutará en el puerto 7000.

Una vez que tenemos todo configurado, podemos arrancar las grabaciones y empezar a usar Firefox para realizar los casos de uso descritos anteriormente (**Figura 2** y **Figura 3**). Hecho esto, tendremos que tener en el árbol de plan de pruebas una estructura similar a la de la **Figura 5** tras grabar la primera sesión, que representa el primer escenario y la **Figura 6**, que representa el segundo escenario.

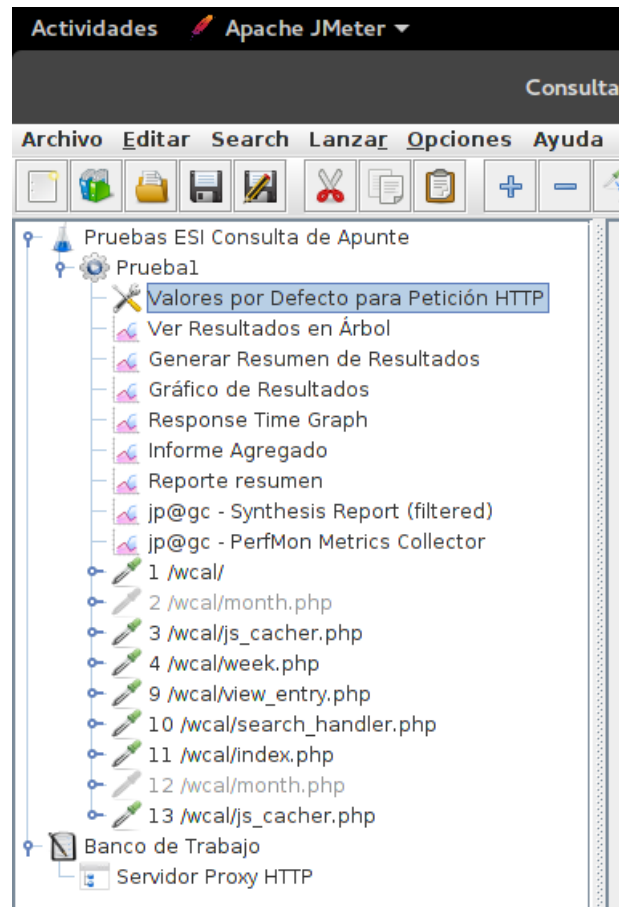


Figura 5: Resultado de JMeter tras grabar el primer escenario



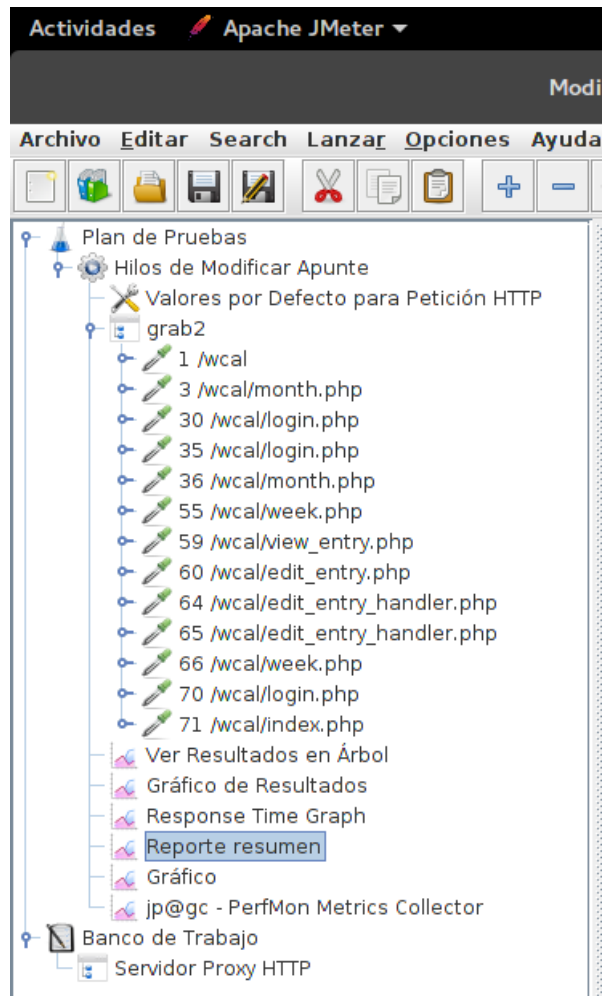


Figura 6: Resultado de JMeter tras grabar el segundo escenario

## 5. Ejecución de las pruebas

Una vez que se dispone de los escenarios grabados, hay que ejecutar las pruebas. Estas pruebas se pueden lanzar de dos formas, en modo gráfico y en modo no gráfico. La diferencia principal es que las pruebas grandes es preferible lanzarlas en modo no gráfico, recoger los resultados y luego analizar las salidas en modo gráfico. De esta manera se reduce la carga del sistema y evitamos fallos a la hora de lanzar cargas pesadas.

Para cada una de las pruebas hay que configurar el grupo de hilos con los datos pertinentes. Los datos pedidos del grupo de hilos para el escenario de consulta de apunte se encuentran en la **Figura 7**. Los datos para el escenario de modificar apunte se encuentran en la **Figura 8**.

<b>Prueba</b>	1	2	3	4
<b>No. Hilos</b>	1	150	150	150
<b>Contador Bucle</b>	150	1	1	"Sin Fin"
<b>Periodo Subida</b>	10	10	1	5
<b>Duración</b>	-	-	-	45s

Figura 7: Tabla con las intensidades para el escenario 1

<b>Prueba</b>	1	2	3	4	5	6	7
<b>No. Hilos</b>	10	20	40	70	130	240	440
<b>Contador Bucle</b>	"Sin Fin"	"Sin Fin"	"Sin Fin"	"Sin Fin"	"Sin Fin"	"Sin Fin"	"Sin Fin"
<b>Periodo Subida</b>	-	-	-	-	-	-	-
<b>Duración</b>	600+	600+	600+	600+	600+	600+	600+

Figura 8: Tabla con las intensidades para el escenario 2

Una vez se han ejecutado todas las pruebas al servidor, podemos pasar a su posterior análisis.

## 6. Análisis exploratorio de datos

El objetivo de estas pruebas de carga es analizar los tiempos de respuesta y la productividad del servidor para servir las páginas web requeridas, para posteriormente visualizar el rendimiento del servidor para detectar posibles cuellos de botella del sistema y puntos de saturación. Es por ello que vamos a dividir el análisis exploratorio en 4 partes.

En la primera vamos a hablar de los resultados obtenidos en el escenario 1. Para ello proporcionaremos las propias gráficas obtenidas del propio JMeter. La segunda parte irá dedicada al estudio del rendimiento del servidor web en el caso del primer escenario.

En la tercera parte se hablará de los resultados obtenidos en el escenario 2 y en la cuarta se hablará del rendimiento proporcionado por el servidor web durante las pruebas del escenario 2.

### 6.1. Escenario 1

Recordamos que las pruebas del escenario 1 son muy variadas. En concreto realizamos 4 cargas a nuestro servidor, simulando el escenario 1, que es una consulta de apunte en el calendario. Esta operación no es muy costosa ya que las operaciones que realiza un usuario contra el sistema son pocas, de corta duración y no inyectan información en el servidor, esto último es importante.

Como las 4 pruebas son diferentes y buscan explicar fenómenos más o menos diferentes, vamos a hacer una pequeña diferencia para explicarlas todas.

#### 6.1.1. Prueba 1 - 1 hilo y 150 steps

Las condiciones de la primera prueba las podemos ver en la **Figura 7**. En este caso tenemos 1 hilo y contador de bucle de 150. Esto quiere decir que vamos a tener un solo usuario en el sistema,

pero que repite el escenario 150 veces.

Los resultados obtenidos a nivel de **productividad** para esta fase se pueden visualizar en la siguiente tabla:

Etiqueta	# Muestr...	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimie...	Kb/sec	Sent KB/s...
1 /wcal/	150	33	29	34	52	176	24	178	0,00%	7,2/min	0,11	0,04
3 /wcal/js_cacher.p...	150	29	27	30	41	172	23	176	0,00%	7,2/min	0,28	0,04
4 /wcal/week.php	150	3752	3733	3924	4001	4156	3505	4187	0,00%	7,1/min	1,16	0,05
9 /wcal/view_entry....	150	60	59	67	73	103	52	108	0,00%	7,2/min	0,32	0,05
10 /wcal/search h...	150	76	71	96	109	173	60	200	0,00%	7,2/min	0,54	0,06
11 /wcal/index.php	150	4425	4412	4613	4651	4786	4147	4913	0,00%	7,1/min	3,25	0,10
13 /wcal/js_cacher...	150	59	48	78	99	208	39	212	0,00%	7,2/min	0,28	0,04
Total	1050	1205	63	4344	4464	4628	23	4913	0,00%	49,7/min	5,90	0,40

Figura 9: Resultados productividad escenario 1 prueba 1

La productividad obtenida es de  $49.7 \simeq 50$  peticiones/minuto, que indica que el servidor sirve unas 0.9 peticiones/segundo. Como es un cliente repitiendo 150 veces el mismo escenario, en total (por el número de páginas visitadas) tenemos 1050 accesos a páginas.

Los resultados obtenidos para el **tiempo de respuesta** son:

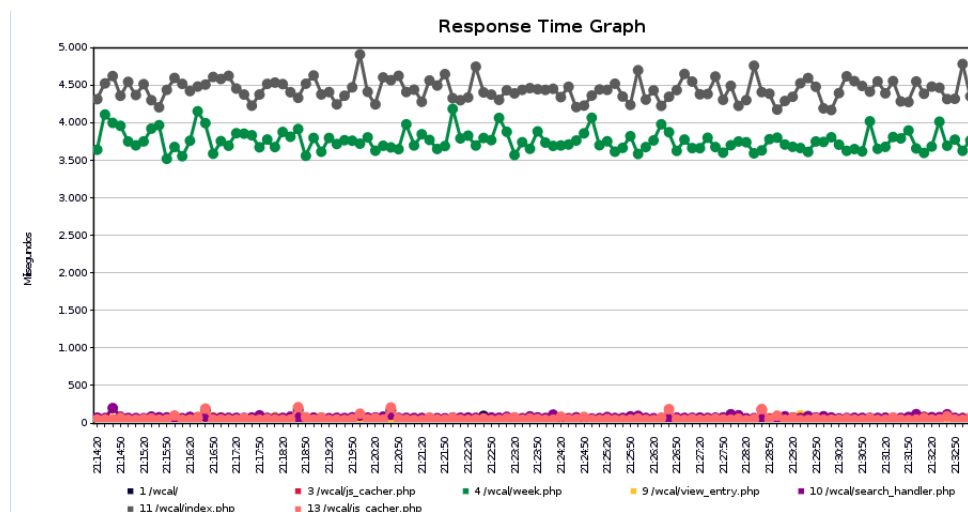


Figura 10: Resultados tiempo respuesta escenario 1 prueba 1

Como podemos observar, los tiempos de respuesta en el que el servidor tarda en servir una página al cliente son muy elevados (3.5 - 5 segundos) para dos páginas diferentes (líneas verde y gris). Es normal que estas páginas sean las que más tardan ya que son las que más carga tienen. Las páginas en cuestión son /wcal/index.php, que tiene todo el contenido principal del calendario público; y la página /wcal/week.php que contiene también toda la información relacionada con los eventos para una semana dada. El resto son páginas con información menor, y por lo eso el tiempo de respuesta es tan bajo (casi 0). Aún así, podemos ver que aunque el tiempo de respuesta de la página /wcal/week.php es elevado, sigue siendo sustancialmente menor que el de /wcal/index.php. Esto es lógico debido a que el contenido generado por una semana, es bastante menor que el que genera index, que inicialmente muestra todo el mes actual.

Los resultados a nivel de **rendimiento del servidor** para este escenario son:

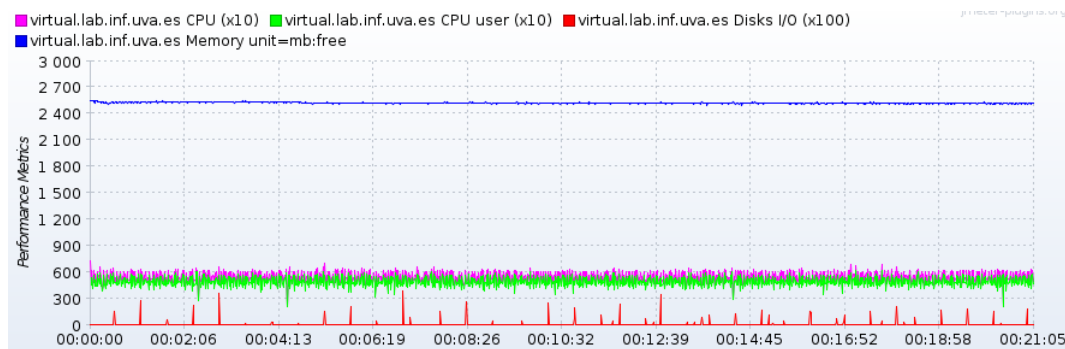


Figura 11: Resultados rendimiento servidor escenario 1 prueba 1

Los resultados de rendimiento obtenidos muestran una mayor demanda de memoria (línea azul) que de cualquier otro recurso, con diferencia. Los dispositivos de E/S (línea roja) casi no tienen participación, salvo picos momentáneos, que seguramente se deban a momentos de paginación, en los que al estar el uso de memoria tan elevado, se empieza a almacenar información en el disco hasta que se descongestiona un poco la memoria. Las conclusiones finales de este documento aportarán datos más interesantes a nivel de rendimiento, ya que se comparará dicho rendimiento en base a las diferentes pruebas realizadas.

### 6.1.2. Prueba 2 - 150 hilos y 1 setps

En este tipo de prueba consiste en cargar al sistema con 150 usuarios que se conectan de una manera concurrente. El único respiro que le damos al servidor es que no se conectan los 150 usuarios al mismo tiempo, sino que haya una diferencia de 10 segundos entre cada cliente (periodo de subida).

Los resultados obtenidos a nivel de **productividad** para esta fase se pueden visualizar en la siguiente tabla:

Etiqueta	# Muest...	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimie...	Kb/sec	Sent KB/s...
1 /wcal/	450	76522	100508	130800	130941	131158	24	131201	44,44%	13,4/min	0,33	0,04
3 /wcal/js_cacher.p...	450	29170	38518	67607	67711	68199	23	68326	0,00%	13,1/min	0,51	0,08
4 /wcal/week.php	450	219334	328312	344417	350416	390817	3505	392484	0,00%	11,3/min	1,79	0,08
9 /wcal/view_entry....	450	598	313	867	1046	3531	52	22880	0,00%	11,3/min	0,50	0,08
10 /wcal/search_h...	450	2263	2745	4498	4901	5383	22	5695	0,00%	11,3/min	0,84	0,10
11 /wcal/index.php	450	227122	346421	357035	359938	366154	46	371513	0,00%	9,9/min	4,41	0,14
13 /wcal/js_cacher....	450	67	52	109	162	201	29	318	0,00%	9,9/min	0,38	0,06
Total	3150	79297	3667	340791	352227	361784	22	392484	6,35%	1,1/sec	8,13	0,52

Figura 12: Resultados productividad escenario 1 prueba 2

Como podemos observar la productividad es un poco mayor en este caso, se sirve 1.1 peticiones/segundo, lo cual es una medida mayor que en la prueba 1. En este caso, podemos ver que en las primeras peticiones a /wcal/ se han producido errores, en concreto un 44.44% de errores, lo cual evidentemente disminuye los resultados obtenidos para la productividad, ya que no se sirven tantas peticiones/segundo como se desearía.

Los resultados obtenidos para el **tiempo de respuesta** son:

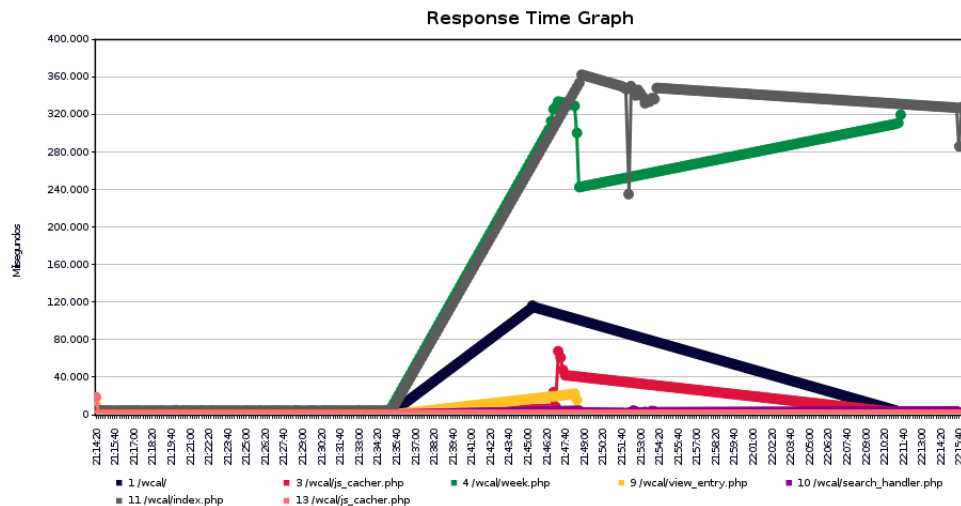


Figura 13: Resultados tiempo respuesta escenario 1 prueba 2

Aunque la caracterización del rendimiento del servidor va en la siguiente gráfica y se explica a continuación, existe un fenómeno que merece la pena destacar. Podemos observar cómo el tiempo de respuesta se dispara cuando el sistema entra en hiperpaginación. Esto quiere decir que en ese momento el sistema tiene que almacenar la información en el disco porque la memoria principal se ha agotado. Como el tiempo de respuesta del disco es de ordenes mucho mayores que el de la memoria principal, en la gráfica se observa como se dispara el tiempo de respuesta. El orden de tiempo medio de servicio de una petición se eleva hasta los 400.000 ms (es decir 400 segundos). Casi todas las páginas disminuyen su tiempo de respuesta después del momento de hiperpaginación, aunque sigue siendo muy elevado para las de /wcal/index.php y /wcal/week.php.

Los resultados a nivel de **rendimiento del servidor** para este escenario son:

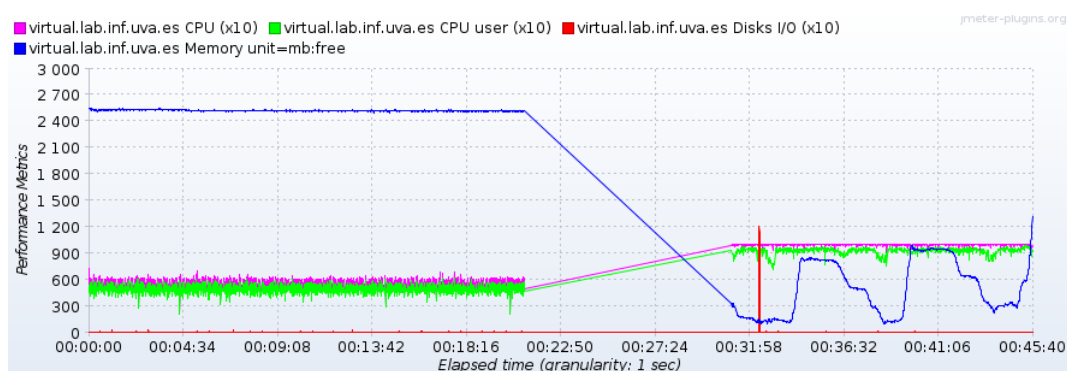


Figura 14: Resultados rendimiento servidor escenario 1 prueba 2

Los resultados a nivel de rendimiento de esta prueba son, cuanto menos, inesperados. Nuestros pensamientos iniciales eran que al lanzar 150 hilos de manera concurrente al sistema, se dispararía el uso de la CPU, pero los resultados obtenidos indican lo contrario. Esto se puede deber a que no es suficiente nivel de concurrencia como para que se note una sobrecarga en la CPU (a diferencia, por ejemplo, que en la primera práctica de la asignatura, que había veces que el nivel de concurrencia era del orden de miles de clientes concurrentes al sistema).

De todas maneras, se puede ver claramente el fenómeno de hiperpaginación del sistema operativo. La memoria se ha saturado y en ese momento la función que hace referencia a la memoria cae en picado, porque deja de usarse. Sin embargo, a la vez el uso del disco se dispara, junto con el uso del procesador. Esto se debe, como hemos explicado anteriormente, a que al haberse ocupado toda la memoria se han empezado a cargar las páginas en el disco. A partir de ese momento el uso de la memoria ha ido a momentos en los que se llenaba, se descargaba, y así continuamente.

### 6.1.3. Prueba 3 - 150 hilos y 1 steps

La diferencia de este caso y el anterior radica en el periodo de subida. En la prueba anterior se simulaba la conexión de 150 usuarios con un periodo de diferencia de conexión entre ellos de 10 segundos (que se puede considerar como un periodo normal, estable para el servidor). En este caso se conectan con una diferencia de 1 segundo. Habrá que ver cómo esta situación de estrés afecta al sistema.

Los resultados obtenidos a nivel de **productividad** para esta fase se pueden visualizar en la siguiente tabla:

Etiqueta	# Muest...	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimie...	Kb/sec	Sent KB/s...
1 /wcal/	600	58148	5094	130686	130890	131142	24	131201	33,33%	10,5/min	0,23	0,04
3 /wcal/js_cacher.p...	600	22383	4020	58662	67647	68155	23	68326	0,00%	10,5/min	0,41	0,06
4 /wcal/week.php	600	242388	322535	342083	348013	390358	3505	392484	0,00%	9,6/min	1,53	0,07
9 /wcal/view_entry....	600	591	410	917	1029	2767	52	22880	0,00%	9,6/min	0,43	0,07
10 /wcal/search_h...	600	2502	3020	4437	4763	5377	22	5695	0,00%	9,6/min	0,71	0,09
11 /wcal/index.php	600	252248	343891	356188	358631	364852	42	371513	0,00%	8,8/min	3,91	0,13
13 /wcal/js_cacher....	600	68	53	119	168	208	24	318	0,00%	8,8/min	0,34	0,05
Total	4200	82618	3239	340469	351086	360228	22	392484	4,76%	1,0/sec	7,23	0,47

Figura 15: Resultados productividad escenario 1 prueba 3

La tabla obtenida para mostrar la productividad no difiere mucho de la tabla anterior, vemos que la productividad se mantiene bastante constante, unas 1.0 peticiones/segundo. Vemos que sigue habiendo un gran porcentaje de fallos en las primeras peticiones a la página /wcal/, lo que decrementa la productividad de nuestro sistema. En las conclusiones finales del documento se desarrollarán más estas características.

Los resultados obtenidos para el **tiempo de respuesta** son:

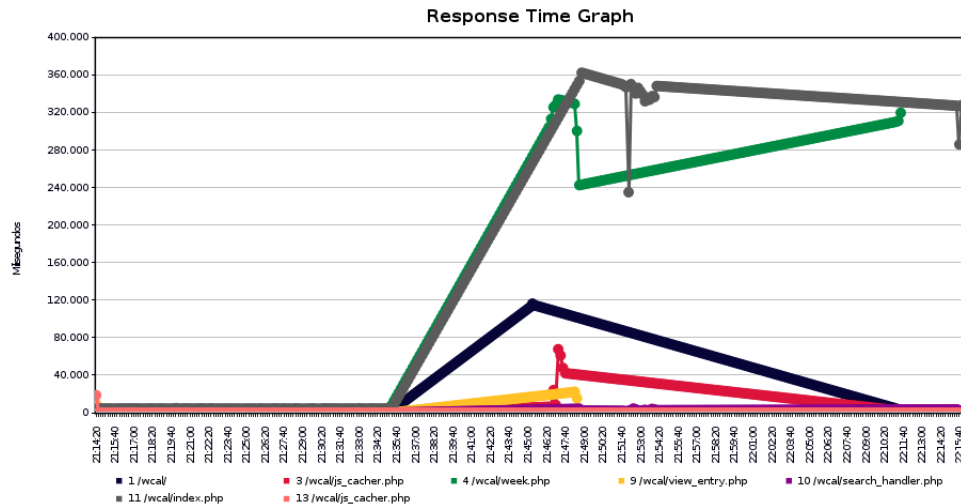


Figura 16: Resultados tiempo respuesta escenario 1 prueba 3

Los tiempos de respuesta son los mismos que en el caso anterior y de hecho se produce el mismo fenómeno de hiperpaginación en el sistema. Por lo tanto la gráfica de rendimiento tampoco difiere mucho.

Los resultados a nivel de **rendimiento del servidor** para este escenario son:

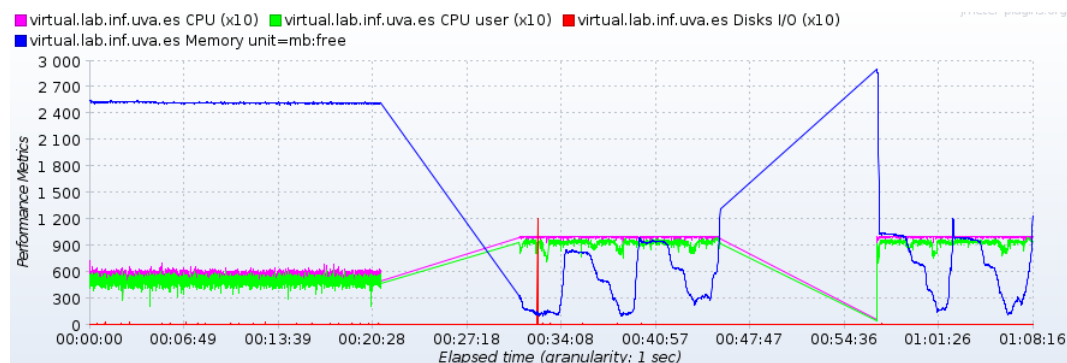


Figura 17: Resultados rendimiento servidor escenario 1 prueba 3

Como pasaba en la prueba anterior, la gráfica del comportamiento del servidor se ve que se produce hiperpaginación, se dispara el disco, la memoria cae y se empieza a usar la CPU. Algo curioso es que cuando el sistema se recupera de la hiperpaginación, se vuelve a utilizar la memoria y cae en picado el uso de la CPU. Este comportamiento del sistema es un poco errático y a primera vista no tenemos una explicación formal de porque se produce ese descenso en el uso de la CPU.

#### 6.1.4. Prueba 4 - 150 hilos y sin fin

Finalmente, en esta prueba lo que se intenta es probar el sistema con una sobrecarga bastante elevada. Esto se debe a que lanzamos 150 clientes como en las anteriores, pero se lanzan una cantidad infinita de veces. El único respiro que tiene el servidor es que los clientes se conectan al sistema con una diferencia temporal de 1 seg. Evidentemente como esta prueba va a sobrecargar

mucho al servidor, tiene que ser corta. Por ello se realiza únicamente durante 45 segundos. Por lo menos la parte intensa, luego veremos que la prueba en general dura mucho más de lo esperado.

Los resultados obtenidos a nivel de **productividad** para esta fase se pueden visualizar en la siguiente tabla:

Etiqueta	# Muest...	Media	Mediana	90% Line	95% Line	99% Line	Mín	Máx	% Error	Rendimie...	Kb/sec	Sent KB/s...
1 /wcal/	750	46782	3093	130385	130843	131142	21	131201	26,67%	38,5/hour	0,01	0,00
3 /wcal/js_cacher.p...	750	18090	2145	51449	67630	68155	9	68326	0,00%	38,5/hour	0,02	0,00
4 /wcal/week.php	750	253878	312450	340618	345819	390358	3505	392484	0,00%	38,4/hour	0,10	0,00
9 /wcal/view_entry....	600	591	410	917	1029	2767	52	22880	0,00%	9,6/min	0,43	0,07
10 /wcal/search_h...	600	2502	3020	4437	4763	5377	22	5695	0,00%	9,6/min	0,71	0,09
11 /wcal/index.php	600	252248	343891	356188	358631	364852	42	371513	0,00%	8,8/min	3,91	0,13
13 /wcal/js_cacher....	600	68	53	119	168	208	24	318	0,00%	8,8/min	0,34	0,05
Total	4650	84367	3066	337876	349933	359938	9	392484	4,30%	4,0/min	0,45	0,03

Figura 18: Resultados productividad escenario 1 prueba 4

En total vemos que se ha hecho un total de 4650 peticiones al sistema. En este caso la productividad del sistema a disminuido de manera abismal. La premisa que dábamos antes de que el sistema iba a estar muy cargado la vemos aquí. La productividad de las primeras páginas es increíblemente alta (38.5 peticiones/hora). Estos datos iniciales son lógico ya que el sistema, al estar atendiendo tantas peticiones tiende a saturar los recursos y hay momentos en los que no dispone de los suficientes como para seguir sirviendo las peticiones.

Vemos que la productividad total del sistema baja hasta las 4 peticiones/min. Antes hablábamos de que mas o menos se completaba una petición por segundo, mientras que ahora son 4peticione-s/min ( $\approx 1$  petición cada 25 segundos).

Los resultados obtenidos para el **tiempo de respuesta** son:

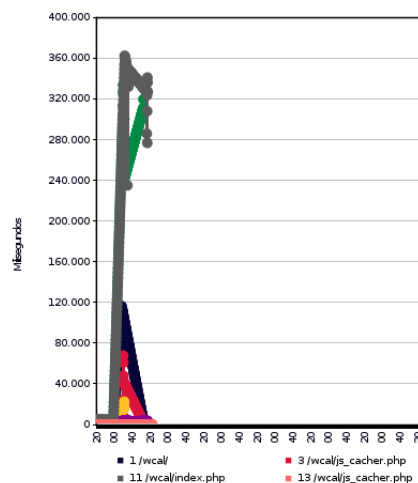


Figura 19: Resultados tiempo respuesta escenario 1 prueba 4

Como el caso de prueba está pensado para saturar al sistema, podemos observar que los tiempos de respuesta para todas las páginas se disparan. Esto se debe a la carga del sistema, sobre todo en momentos iniciales, que hace al servidor incapaz de servir las respuestas en un tiempo adecuado (falta de recursos, sobrecarga de memoria, cpu, etc).



Los resultados a nivel de **rendimiento del servidor** para este escenario son:

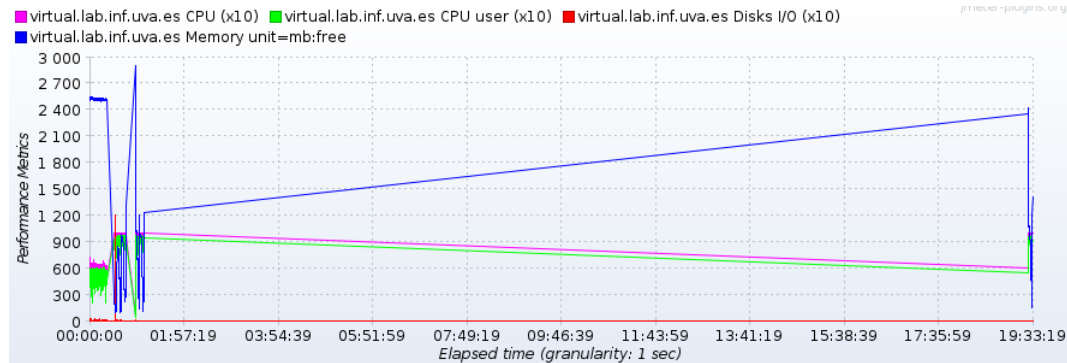


Figura 20: Resultados rendimiento servidor escenario 1 prueba 4

Cuando un sistema se encuentra en sobrecarga, los resultados del mismo son bastante erráticos. Se agota un recurso, otro se queda ocioso, cuando un recurso se recupera el otro se dispara porque necesitaba el que estaba ocupado, etc. Este tipo de comportamientos los podemos ver en la gráfica de rendimiento del servidor.

Vemos que en momentos iniciales se dispara la memoria y a cpu, ambos van a picos. Conforme se va desarrollando la prueba, vemos una carga muy elevada tanto para la memoria como para la CPU.

## 6.2. Escenario 2

Recordamos que el escenario 2 se describe en la **Figura 3**. Para las pruebas de este escenarios vamos a seguir un enfoque diferente. Van a ser pruebas de capacidad. En estas pruebas vamos a ir aumentando de manera incremental la carga del sistema, lo que nos permitirá ver hasta dónde es capaz de llegar el sistema sin que se degrade su rendimiento de manera considerable. A continuación se muestran los datos recogidos de las diferentes pruebas.

### 6.2.1. Prueba con 10 hilos

Los resultados obtenidos a nivel de **productividad** para esta fase se pueden visualizar en la siguiente tabla:

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Está...	% Error	Rendimie...	Kb/sec	Sent K...	Media ...
1 /wcal	10	13827	6588	16643	3105,79	0,00%	22,8/min	10,59	0,38	28557,0
4 /success.txt	10	260993	260262	261783	602,61	100,00%	2,1/min	0,07	0,00	2072,0
3 /wcal/month.php	10	8659	4902	12357	3004,26	0,00%	18,9/min	8,32	0,11	27019,0
5 /success.txt	10	261059	260358	262073	526,04	100,00%	2,1/min	0,07	0,00	2072,0
11 /wcal/js_cacher.php	10	58	45	105	17,55	0,00%	22,5/min	0,87	0,13	2372,0
10 /wcal/js_cacher.php	10	21	19	27	2,26	0,00%	22,5/min	1,53	0,13	4169,0
14 /wcal/css_cacher.php	10	31	24	61	10,70	0,00%	22,5/min	1,59	0,14	4340,0
30 /wcal/login.php	10	41	26	59	11,66	0,00%	22,5/min	0,62	0,15	1691,0
31 /wcal/css_cacher.php	10	29	22	63	11,45	0,00%	22,5/min	1,59	0,13	4340,0
35 /wcal/login.php	10	17705	4695	27141	8753,29	0,00%	11,2/min	5,29	0,27	29086,0
36 /wcal/month.php	10	19867	4698	25675	6993,89	0,00%	8,1/min	3,56	0,05	27019,0
38 /wcal/js_cacher.php	10	70	44	183	39,81	0,00%	8,6/min	0,33	0,05	2372,0
37 /wcal/css_cacher.php	10	34	26	56	8,87	0,00%	8,6/min	0,61	0,05	4340,0
55 /wcal/week.php	10	17938	7856	23370	4523,85	0,00%	7,2/min	1,08	0,05	9276,0
59 /wcal/view_entry.php	4	120	77	210	53,43	0,00%	4,8/min	0,21	0,04	2733,8
60 /wcal/edit_entry.php	4	48	41	61	7,79	0,00%	4,8/min	0,63	0,04	7991,0
62 /wcal/js_cacher.php	4	27	24	30	2,17	0,00%	4,8/min	0,39	0,03	4943,0
61 /wcal/js_cacher.php	4	26	23	30	2,59	0,00%	4,8/min	0,19	0,03	2372,0
64 /wcal/edit_entry_handler....	4	5080	2245	6345	1652,40	0,00%	4,3/min	0,22	0,09	3105,0
65 /wcal/edit_entry_handler....	4	22794	19979	25977	2143,33	0,00%	3,3/min	1,47	0,09	27639,0
66 /wcal/week.php	2	20737	19629	21845	1108,00	0,00%	2,7/min	0,41	0,02	9276,0
67 /success.txt	1	260840	260840	260840	0,00	100,00%	13,8/hour	0,01	0,00	2072,0
Total	167	38431	19	262073	84830,53	12,57%	11,7/min	1,94	0,09	10197,7

Figura 21: Resultados productividad escenario 2 prueba 10

la productividad con estos niveles de carga son aceptables: Se sirve una petición cada 5 segundos. Es razonable teniendo en cuenta que el periodo de subida es de 14 segundos (es decir se conecta un cliente cada 14 segundos hasta llegar a 10). No es una carga muy elevada y el sistema responde bien.

Los resultados obtenidos para el **tiempo de respuesta** son:

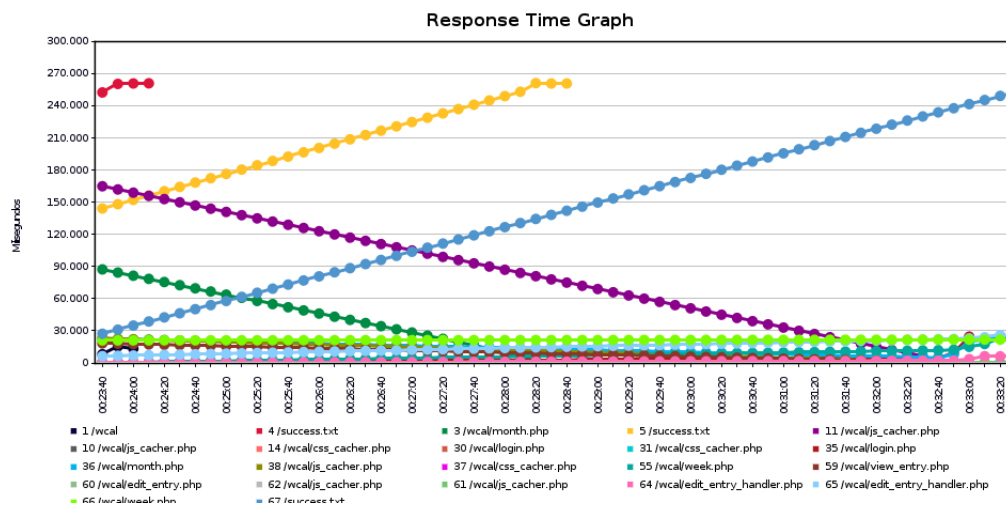


Figura 22: Resultados tiempo respuesta escenario 2 prueba 10

En los tiempos de respuesta tenemos que obviar los resultados obtenidos para las páginas que pone cacher. Se nos ha olvidado quitarlos al hacer las grabaciones. En líneas generales es lo esperado, las páginas con más carga como las que muestran los eventos del mes o la semana son las que más tiempo de respuesta tienen.

Los resultados a nivel de **rendimiento del servidor** para este escenario son:

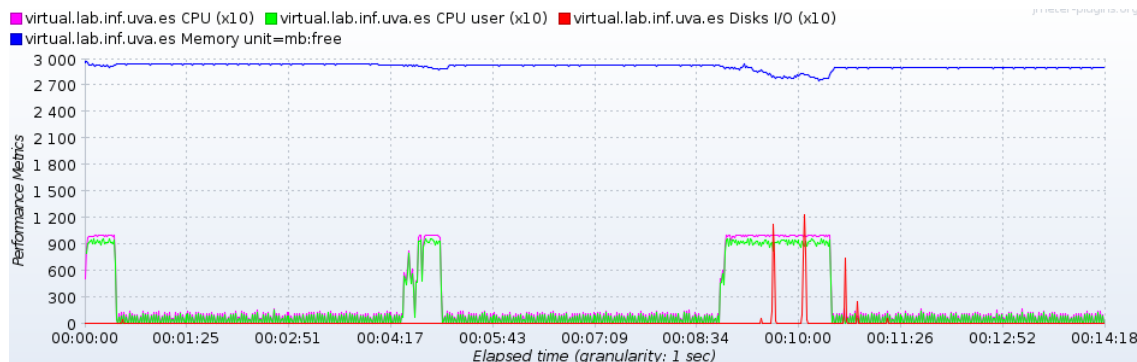


Figura 23: Resultados rendimiento servidor escenario 2 prueba 10

A estos niveles de carga, podemos ver un comportamiento lógico y esperado en una situación normal. La memoria con un uso alto y la cpu con algún pico de precesamiento.

### 6.2.2. Prueba con 20/40/70hilos

Si somos objetivos las pruebas realizadas con 20, 40 y 70 hilos no aportan una gran información a los resultados obtenidos con 10 hilos. Sí, es verdad que la carga del sistema es un poco mayor, pero la productividad se mantiene estable entorno a las 14.5 peticiones por minuto en los 2 primeros casos. Hay que destacar que la productividad para el caso con 70 hilos empieza a bajar de manera abismal. Veremos si en las siguientes pruebas sigue descendiendo, y de ser así sabremos cual es nuestro punto de máxima productividad (de manera aproximada, en un intervalo relativamente pequeño).

Los resultados obtenidos en los tiempos de respuesta son muy similares a los obtenidos en la prueba con 10 hilos. Las páginas que más tardan siguen siendo las que muestran los eventos de las semanas y de los meses.

Cierto es que en lo relativo al rendimiento del servidor hay ligeros cambios: Mayor carga, más picos de uso de CPU, pero en verdad nada que merezca la pena destacar.

Aunque se haya comentado de manera breve los resultados obtenidos, los datos concretos correspondientes a los test (productividad, tiempo de respuesta y rendimiento para cada una de las cargas: 20/40/70 hilos) se encuentran disponibles en el **ANEXO II**.

### 6.2.3. Prueba 5 - 130 hilos

Ahora que hemos aumentado el orden de magnitud en 1 (10 hilos → 130 hilos), los resultados son mucho más representativos. Cabe destacar que como ahora empiezan a ser pruebas más pesadas, estas han sido lanzadas desde la terminal para evitar la sobrecarga impuesta por el entorno gráfico, que es alta.

Las diferentes pruebas paran cuando se pasa el tiempo especificado a la plataforma (600 segundos + periodo de subida), en nuestro caso el periodo de subida se encuentra entre los 12 y los 14 segundos (se iba variando un poco para ver como se comportaba, pero no encontramos mayor diferencia de comportamiento con variaciones tan pequeñas).

Al hacer el análisis mediante la terminal se ha optado por ver los resultados en el formato HTML que proporciona JMeter. A continuación se exponen los resultados:

Requests	Executions			Response Times (ms)							Network (KB/sec)	
Label	#Samples	KO	Error %	Average	Min	Max	90th pct	95th pct	99th pct	Throughput	Received	Sent
Total	780	0	0.00%	144147.40	26	351746	332406.10	345047.05	349750.49	0.86	9.33	0.00
1 /wcal	130	0	0.00%	825.96	32	2760	2169.70	2454.90	2731.48	10.76	6.50	0.00
3 /wcal/month.php	130	0	0.00%	320957.16	220128	351746	347800.00	349846.70	351656.10	0.36	9.58	0.00
30 /wcal/login.php	130	0	0.00%	749.59	26	1549	1246.00	1375.15	1531.33	0.94	1.55	0.00
35 /wcal/login.php	130	0	0.00%	7430.67	155	13076	11694.80	12115.05	12975.25	0.89	1.00	0.00
36 /wcal/month.php	130	0	0.00%	294175.47	235170	335244	330147.50	330833.50	335129.92	0.27	7.22	0.00
55 /wcal/week.php	130	0	0.00%	240745.55	211571	294669	281324.90	285396.75	293887.80	0.29	2.62	0.00

Figura 24: Resultados productividad escenario 2 con 130 hilos

A nivel de **productividad**, el resultado del experimento ha sido bastante inesperado. La productividad ha mejorado con gran diferencia. Antes hablábamos de productividades de entorno al 12.7 peticiones por segundo ahora hablamos de valores cercanos a las 50. Antes de comentar estos valores se ha de destacar que la prueba con estos hilos ha finalizado a los 700 segundos aproximadamente (lo que tarda en crear y cerrar los hilos sobrecarga un poco el tiempo). Pero durante este tiempo a JMeter no le ha dado tiempo a completar todo el escenario. Como se puede ver en la **Figura 24**, solo ha realizado el acceso a 6 de las páginas que formaban el escenario (cierto es que a las 6 han accedido los 130, pero no lo han completado).

Como solo han accedido a 6, nos fijamos en que las páginas de login tienen una productividad tremenda, el resto de páginas no se diferencian tanto de las productividades anteriores (entorno a los 15-16 peticiones/seg). Las páginas login de WebCalendar son formularios muy sencillos y no tardan nada en servirse, es por ello que la productividad aumenta. Podemos estimar que la productividad total, si se hubiera completado la prueba con todas las páginas, hubiera sido algo mayor que en los casos anteriores, aunque seguiría siendo similar.

En lo respectivo a los **tiempos de respuesta**, seguimos sin ver grandes cambios. Las páginas que mayores tiempos de respuesta tienen son las páginas /wcal/month.php y /wcal/week.php, que son las que más información tienen que cargar y mostrar al usuario.

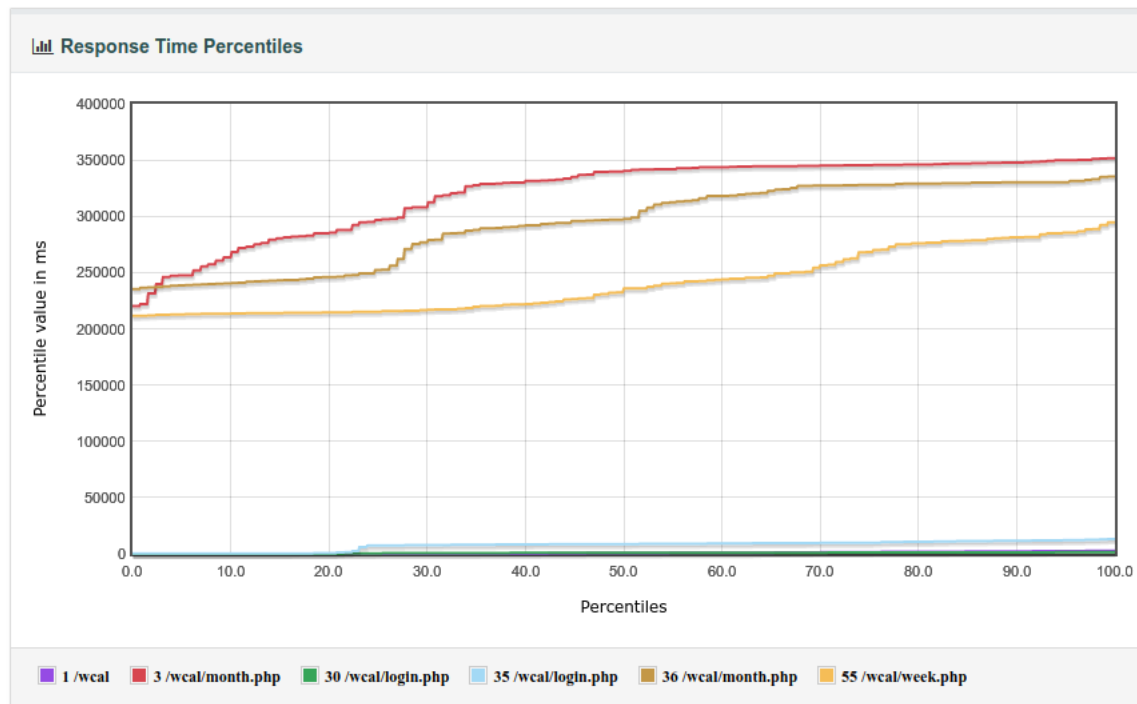


Figura 25: Resultados tiempo respuesta escenario 2 con 130 hilos

El análisis del **rendimiento** se siguen realizando con el plugin perfmon de jmeter. El resultado es el siguiente:

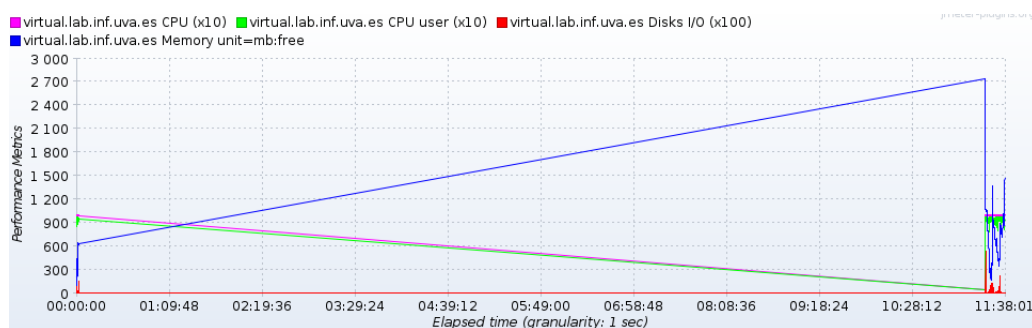


Figura 26: Resultados rendimiento escenario 2 con 130 hilos

Podemos observar que conforme va avanzando el tiempo, la cantidad concurrente de clientes en el sistema va aumentando. Además, como las páginas php son páginas dinámicas cuyo contenido se ejecuta en el servidor, el uso de la memoria del sistema va a aumentando. También es extraño el descenso del uso de CPU del sistema. Una de las teorías para explicar esto puede ser que las operaciones de memoria son mucho más lentas que las de CPU y como tiene que hacer muchas llamadas a la memoria, el procesador está ocioso gran parte del tiempo.

#### 6.2.4. Prueba 6 - 240 hilos

Como veremos en las gráficas, las pruebas ejecutadas con 240 hilos ya tienen un impacto bastante alto en el rendimiento de nuestro servidor.

las gráficas para la **productividad** son:

Requests		Executions		Response Times (ms)							Network (KB/sec)	
Label	#Samples	KO	Error %	Average	Min	Max	90th pct	95th pct	99th pct	Throughput	Received	Sent
Total	878	0	0.00%	211071.78	29	835170	713182.20	714920.05	715983.72	1.04	10.50	0.00
1 /wcal	240	0	0.00%	263967.85	29	716215	715427.00	715814.30	716156.61	0.33	0.20	0.00
3 /wcal/month.php	152	0	0.00%	394203.04	284518	835170	420861.00	421585.90	834099.93	0.18	4.76	0.00
30 /wcal/login.php	150	0	0.00%	936.53	29	2113	1512.40	1761.55	1998.76	1.02	1.68	0.00
35 /wcal/login.php	150	0	0.00%	8835.23	165	15737	14577.40	14970.30	15566.66	0.97	1.09	0.00
36 /wcal/month.php	150	0	0.00%	348559.63	271198	395861	389929.10	391546.35	395812.04	0.28	7.30	0.00
55 /wcal/week.php	36	0	0.00%	230560.31	162898	263368	258042.30	260379.40	263368.00	0.13	1.17	0.00

Figura 27: Resultados productividad escenario 2 con 240 hilos

Como podemos comprobar gracias a este escenario, los resultados de productividad anteriores son resultados extremos que no debemos de tener demasiado en cuenta. Con esta carga de hilos volvemos a las productividades que veníamos arrastrando de 1 petición por segundo. Estos datos sí que cuadran más en el marco en el que estamos trabajando. Aquí sí que podemos apreciar la sobrecarga debido a tantos clientes concurrentes ya que algunas páginas tienen productividades de 7 petición en 1 minuto, que no llega a una petición cada 10 segundos aproximadamente. Podemos encontrar una tabla de JMeter un poco más representativa en el ANEXO II.

En lo respectivo a los **tiempos de respuesta**, obtenemos esta gráfica:

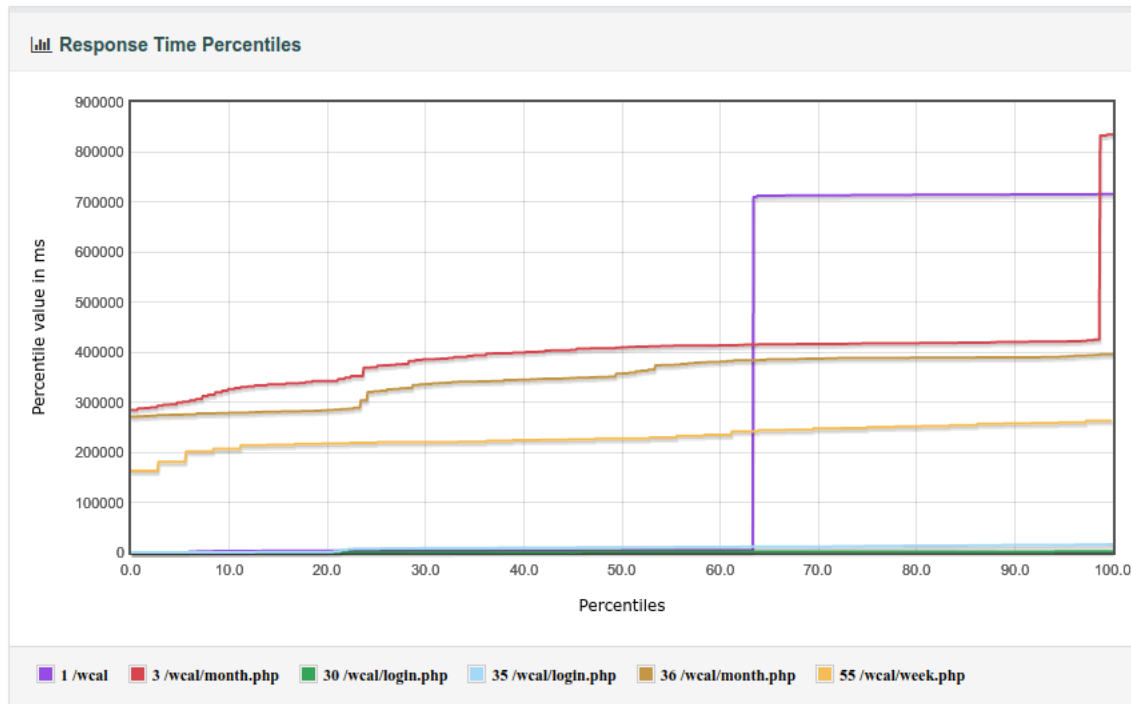


Figura 28: Resultados tiempo respuesta escenario 2 con 240 hilos

Cabe destacar que a mitad de las pruebas los tiempos de respuesta para la página principal de la aplicación webcalendar se hacen muy altos, seguramente debido a la sobrecarga de clientes concurrentes.

El **rendimiento** del servidor web es:

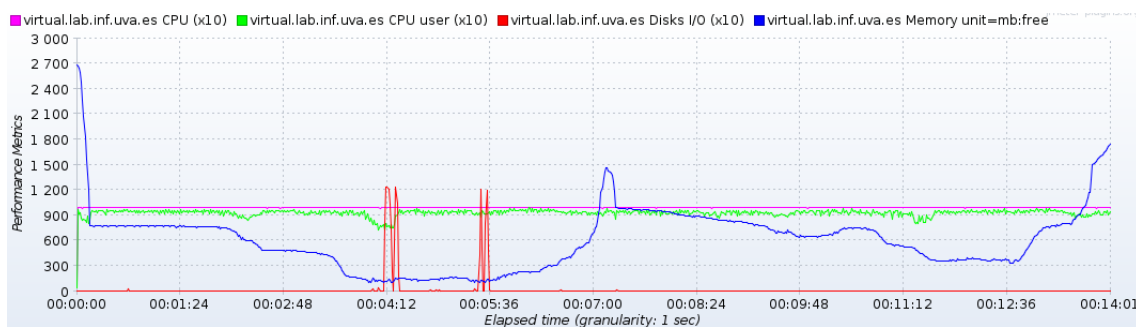


Figura 29: Resultados rendimiento escenario 2 con 240 hilos

Como podemos observar, con tantos clientes concurrentes el rendimiento de nuestro servidor web ya se ve bastante deteriorado. En esta prueba vemos cómo el uso del procesador es bastante alto durante casi toda la prueba, la gráfica de la memoria empieza a definir patrones extraños, se llena, se vacía, empieza a tirar de disco (hiperpaginación), se vuelve a disparar la memoria ... Podemos decir que nuestro servidor empieza a tocar el límite de su vida óptima.

### 6.2.5. Prueba 7 - 440 hilos

La prueba realizada con una carga de 440 hilos es bastante curiosa, porque falla por completo. Por ejemplo en la **Figura 30** vemos que el reporte resumen que nos proporciona JMeter nos informa de que la mayoría de las pruebas han fallado. Tenemos cerca del 100% de fallos en el lanzamiento de las pruebas.

Esto se puede deber a varias cosas, que el servidor SQL se desborda y no admite tantas peticiones, que la memoria se llena y se empiezan a matar procesos, que la CPU se sobrecarga y tampoco puede manejar tanto volumen de datos ...

Hemos intentado ejecutar la prueba a diferentes horas del día (9, 12, 16, 22 y 5) en días diferentes pensando que habría coincidido con algún momento de carga puntual en el sistema, pero los resultados obtenidos son los mismos. Esto nos hace pensar que los fallos se pueden ocasionar por dos motivos:

- **Software:** alguno de los programas implicados en servir las peticiones no está bien configurado. Es decir quizá alguno de los parámetros del servidor web, si no se cambia, limite el número de peticiones simultáneas. El servidor SQL también puede que se desborde y basta con leves ajustes de configuración.
- **Hardware:** puede que los recursos de la máquina sea insuficientes para manejar ese volumen de transacciones.

Personalmente nosotros creemos que los problemas son derivados del software más que del hardware, porque la gráfica que nos muestra el perfmon (**Figura 31**) nos indica que el uso de la cpu va cayendo, debido a los fallos y sin embargo la memoria va aumentando, esto se puede deber a que se intentan cargar las páginas y se intenta servir a los hilos, en un primer momento se traen las peticiones a la memoria pero luego por algún fallo del resto del sistema se cancela ese proceso, por eso se va llenando la memoria.

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
1 /wcal	14008	19770	0	748795	103350,80	94,70%	2,5/min	0,07	0,00	1623,6
3 /wcal/mont...	13603	17840	0	961908	76594,54	96,80%	2,5/min	0,10	0,00	2488,3
30 /wcal/logi...	13547	5121	0	299029	31936,49	97,42%	2,4/min	0,07	0,00	1680,9
35 /wcal/logi...	13493	3924	0	307207	26854,81	96,68%	2,4/min	0,07	0,00	1652,3
36 /wcal/mon...	13278	10351	0	461869	59186,30	97,12%	2,4/min	0,09	0,00	2388,3
55 /wcal/wee...	12950	2029	0	348880	24275,21	99,30%	2,3/min	0,06	0,00	1709,9
59 /wcal/view...	12859	1	0	5216	56,89	99,91%	846,5/sec	1370,01	0,00	1657,2
60 /wcal/edit...	12859	2	0	4815	61,67	99,85%	815,0/sec	1325,75	0,00	1665,6
64 /wcal/edit...	12833	0	0	155	5,23	100,00%	1285,6/sec	2079,90	0,00	1655,7
65 /wcal/edit...	12833	3	0	6020	109,01	99,91%	802,2/sec	1296,94	0,00	1655,6
66 /wcal/wee...	12811	0	0	147	4,68	100,00%	1283,9/sec	2077,40	0,00	1656,8
70 /wcal/logi...	12811	1	0	4690	59,76	99,91%	874,1/sec	1413,45	0,00	1655,9
71 /wcal/inde...	12811	2	0	4764	55,18	99,85%	852,4/sec	1377,98	0,00	1655,5
Total	170696	4720	0	961908	42969,03	98,52%	30,7/min	0,89	0,00	1783,3

Figura 30: Resultados reporte resumen escenario 2 con 440 hilos



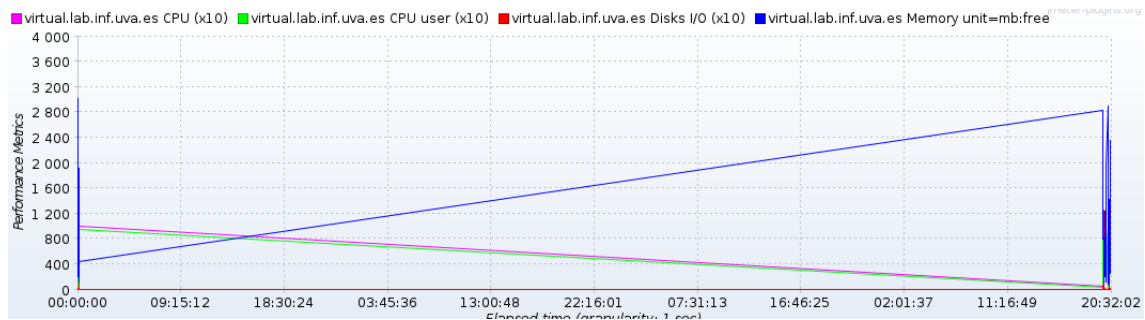


Figura 31: Resultados rendimiento escenario 2 con 440 hilos

En resumen, lo que debería de haber pasado con la prueba de los 440 hilos es como con la prueba de 230 pero potenciado, es decir, el uso de recursos del servidor se dispara y se intenta a toda costa ejecutar las peticiones pedidas al servidor. Como no podemos obtener resultados fehacientes de esta prueba, solo podemos extraer pequeñas hipótesis.

## 7. Conclusiones

### Productividad:

Dados los datos obtenidos, hemos observado que la productividad no se ve afectada de manera notable al aumentar el número de peticiones que crea un solo cliente o el número de clientes que crean las peticiones por separado. Sin embargo, cuando aumentamos tanto el número de clientes como la cantidad de peticiones que crea cada uno, el tiempo de respuesta incrementa drásticamente.

También hemos notado que aumentar el tiempo de respiro entre peticiones no es tan notable como se podría esperar, siendo mucho menos relevante que la cantidad de clientes y peticiones.

Lo mencionado anteriormente podemos resumirlo en la siguiente fórmula:

$$\text{Carga} = \text{Peticiones} * \text{Clientes}$$

Como se puede observar, si aumenta el número de clientes, hay un incremento directo de la carga. Lo mismo ocurre con las peticiones. Sin embargo, si aumentamos ambas variables el incremento pasa a ser exponencial, de manera que la carga de nuestro sistema se dispara.

Dicho esto, nuestro sistema podría aguantar a un grupo pequeño de personas accediendo a él o a un grupo de tamaño considerable que cree pocas peticiones, pero bajo ningún concepto serviría para un uso real o comercial.

### Sistema:

Como ya se ha comentado en otros apartados, el sistema llega al punto en el que se llena la memoria y necesita hacer hiperpaginación. Es en este momento en el que el uso de CPU aumenta y el rendimiento del sistema empeora drásticamente (Un cuello de botella)

Dicho esto, tenemos 2 opciones para mejorar nuestro sistema.

1) Mejorar la memoria. Esto implicaría que el punto de hiperpaginación (Nuestro cuello de botella) se alcanzase más tarde.

2) Mejorar la CPU. De esta manera, una vez llegado el punto de hiperpaginación, el sistema sería capaz de responder más rápidamente a las peticiones.

Dado que ambas opciones son buenas, siendo más fácil y barato conseguir un aumento de memoria, optaríamos por esta primera opción inicialmente, y una mejora de la CPU en el futuro.

## **8. Bibliografía**

- [1] J.M. Marqués. Transparencias de la asignatura Evaluación de Sistemas Informáticos.
- [2] Jmeter. <https://jmeter.apache.org/usermanual/index.html> .
- [3] Molero. Evaluación y Modelado del rendimiento de los Sistemas Informáticos.

## ANEXO I.

Configuraciones, al no ser demasiadas, se han nombrado en las secciones correspondientes del texto cuando hablábamos del sistema en estudio. También es posible que se haya hecho referencias a la práctica anterior, ahí es donde de verdad se describían en profundidad las características y configuraciones.

## ANEXO II.

### ■ Escenario 2, 20 hilos:

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Está...	% Error	Rendimie...	Kb/sec	Sent K...	Media ...
1 /wcal	20	35650	15184	41794	6730,70	0,00%	23,8/min	11,07	0,40	28557,0
4 /success.txt	20	260923	260243	262127	541,13	100,00%	4,1/min	0,14	0,00	2072,0
3 /wcal/month.php	20	19381	4809	29382	10433,35	0,00%	18,7/min	8,22	0,11	27019,0
5 /success.txt	20	261295	260515	261968	381,73	100,00%	3,7/min	0,13	0,00	2072,0
11 /wcal/js_cacher.php	20	72	40	111	21,69	0,00%	20,2/min	0,78	0,11	2372,0
10 /wcal/js_cacher.php	20	23	17	52	8,13	0,00%	20,2/min	1,37	0,11	4169,0
14 /wcal/css_cacher.php	20	37	24	76	14,28	0,00%	20,2/min	1,42	0,12	4340,0
30 /wcal/login.php	20	48	27	96	19,80	0,00%	20,1/min	0,55	0,13	1691,0
31 /wcal/css_cacher.php	20	40	22	88	20,43	0,00%	20,1/min	1,42	0,12	4340,0
35 /wcal/login.php	20	29136	5221	40338	13298,99	0,00%	12,0/min	5,69	0,29	29086,0
36 /wcal/month.php	8	21583	4738	38017	11349,87	0,00%	5,1/min	2,23	0,03	27019,0
38 /wcal/js_cacher.php	5	61	47	91	16,33	0,00%	6,4/min	0,25	0,04	2372,0
37 /wcal/css_cacher.php	5	33	23	53	10,42	0,00%	6,4/min	0,45	0,04	4340,0
55 /wcal/week.php	5	23048	4489	38116	12658,78	0,00%	3,5/min	0,53	0,02	9276,0
59 /wcal/view_entry.php	2	100	66	135	34,50	0,00%	6,5/min	0,29	0,05	2734,0
60 /wcal/edit_entry.php	2	43	41	46	2,50	0,00%	6,5/min	0,84	0,05	7991,0
62 /wcal/js_cacher.php	2	27	26	29	1,50	0,00%	6,5/min	0,52	0,04	4943,0
61 /wcal/js_cacher.php	2	29	28	31	1,50	0,00%	6,5/min	0,25	0,04	2372,0
64 /wcal/edit_entry_handler....	2	2698	1060	4337	1638,50	0,00%	5,2/min	0,27	0,11	3105,0
65 /wcal/edit_entry_handler....	2	17196	10357	24036	6839,50	0,00%	2,6/min	1,18	0,08	27639,0
66 /wcal/week.php	2	25153	14996	35310	10157,00	0,00%	1,7/min	0,26	0,01	9276,0
67 /success.txt	1	261811	261811	261811	0,00	100,00%	13,8/hour	0,01	0,00	2072,0
Total	238	53667	17	262127	95642,74	17,23%	16,9/min	2,92	0,13	10624,6

Figura 32: Resultados productividad escenario 2 con 20 hilos

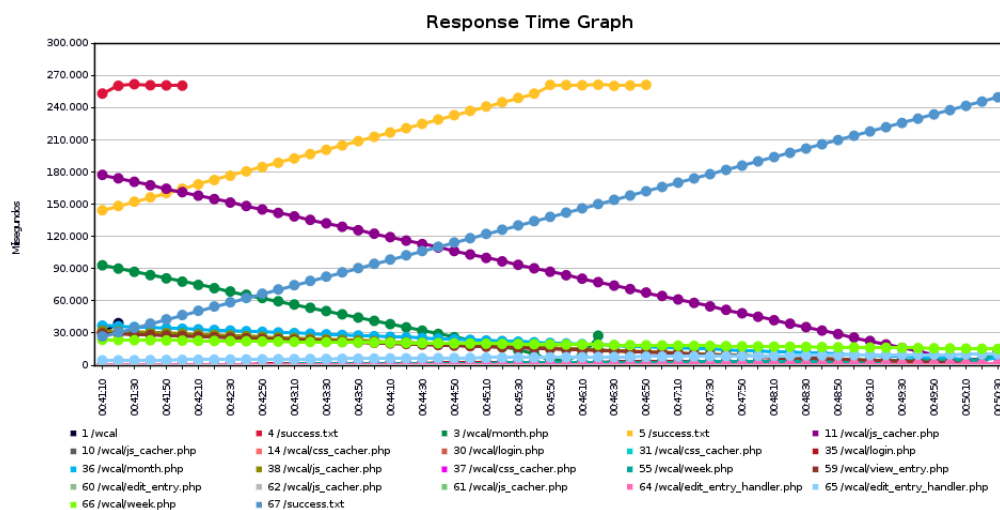


Figura 33: Resultados tiempo respuesta escenario 2 con 20 hilos

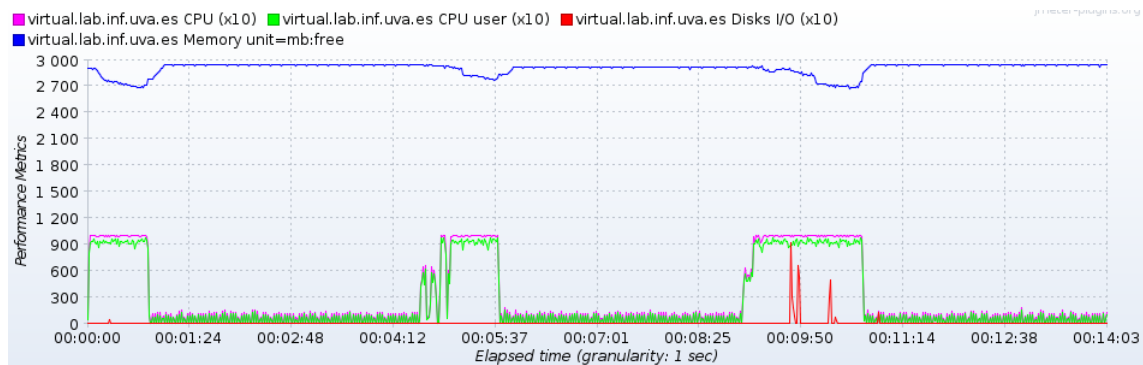


Figura 34: Resultados rendimiento servidor escenario 2 con 20 hilos

### ■ Escenario 2, 40 hilos:

Etiqueta	# Muestras	Media	Min	Máx	Desv. Está...	% Error	Rendimie...	Kb/sec	Sent K...	Media ...
1 /wcal	40	84525	48124	92420	9535,33	0,00%	23,5/min	10,95	0,40	28557,0
4 /success.txt	40	261336	260257	262270	638,84	100,00%	7,6/min	0,26	0,00	2072,0
3 /wcal/month.php	40	47976	4453	67912	23305,92	0,00%	19,8/min	8,71	0,11	27019,0
5 /success.txt	40	261372	260284	262422	639,22	100,00%	6,4/min	0,21	0,00	2072,0
11 /wcal/js_cacher.php	7	75	43	138	32,58	0,00%	11,4/min	0,44	0,06	2372,0
10 /wcal/js_cacher.php	7	20	17	23	2,03	0,00%	11,4/min	0,77	0,06	4169,0
14 /wcal/css_cacher.php	7	75	23	323	101,52	0,00%	11,4/min	0,80	0,07	4340,0
30 /wcal/login.php	7	33	27	51	7,46	0,00%	11,4/min	0,31	0,08	1691,0
31 /wcal/css_cacher.php	7	26	23	31	2,71	0,00%	11,4/min	0,80	0,07	4340,0
35 /wcal/login.php	7	9959	4852	14843	3801,95	0,00%	8,4/min	3,96	0,20	29086,0
36 /wcal/month.php	4	9180	4943	13376	3535,32	0,00%	5,4/min	2,37	0,04	27019,0
38 /wcal/js_cacher.php	3	49	37	60	9,46	0,00%	6,4/min	0,25	0,04	2372,0
37 /wcal/css_cacher.php	3	31	29	34	2,05	0,00%	6,4/min	0,45	0,04	4340,0
55 /wcal/week.php	3	8707	4346	11864	3185,35	0,00%	4,5/min	0,68	0,03	9276,0
59 /wcal/view_entry.php	2	96	96	97	0,50	0,00%	6,1/min	0,27	0,05	2734,0
60 /wcal/edit_entry.php	2	35	35	36	0,50	0,00%	6,1/min	0,80	0,05	7991,0
62 /wcal/js_cacher.php	2	28	27	29	1,00	0,00%	6,1/min	0,49	0,04	4943,0
61 /wcal/js_cacher.php	2	41	28	54	13,00	0,00%	6,1/min	0,24	0,04	2372,0
64 /wcal/edit_entry_handler....	2	2128	1076	3181	1052,50	0,00%	5,3/min	0,27	0,11	3105,0
65 /wcal/edit_entry_handler....	2	11100	8698	13503	2402,50	0,00%	3,4/min	1,54	0,10	27639,0
66 /wcal/week.php	1	10072	10072	10072	0,00	0,00%	6,0/min	0,90	0,04	9276,0
67 /success.txt	1	260419	260419	260419	0,00	100,00%	13,8/hour	0,01	0,00	2072,0
Total	229	116332	17	262422	111328,80	35,37%	15,8/min	3,34	0,10	12994,5

Figura 35: Resultados productividad escenario 2 con 40 hilos

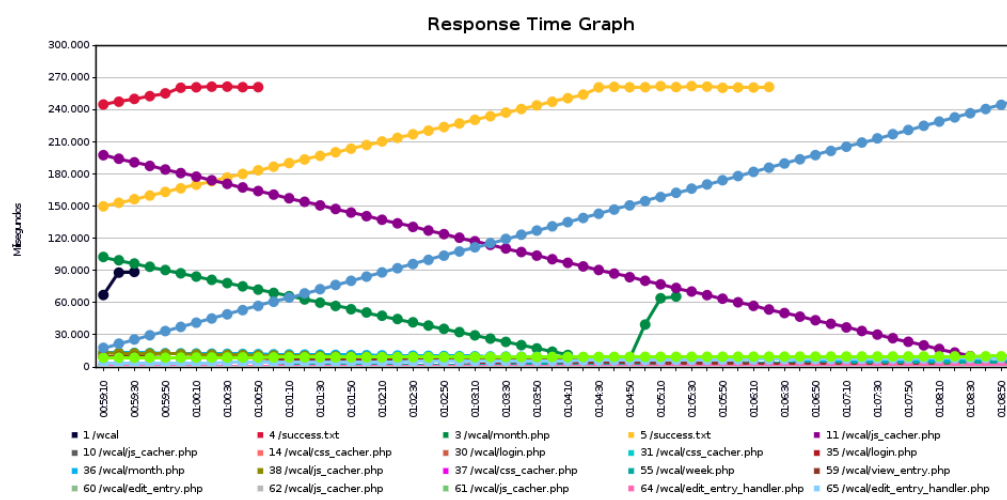


Figura 36: Resultados tiempo respuesta escenario 2 con 40 hilos

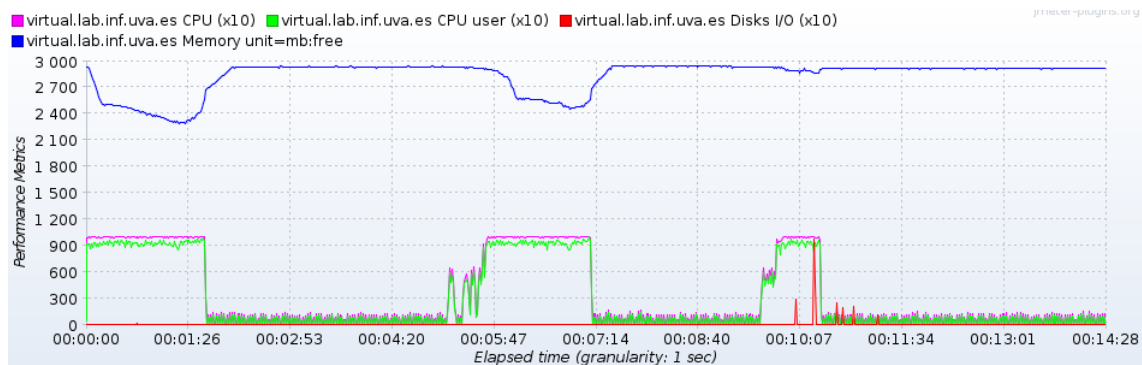


Figura 37: Resultados rendimiento servidor escenario 2 con 40 hilos

### ■ Escenario 2, 70 hilos:

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Está...	% Error	Rendimie...	Kb/sec	Sent K...	Media ...
1 /wcal	141	24157	16	92420	38329,67	0,00%	7,4/hour	0,02	0,00	8544,7
4 /success.txt	40	261336	260257	262270	638,84	100,00%	7,6/min	0,26	0,00	2072,0
3 /wcal/month.php	141	119235	4453	193941	57338,35	0,00%	7,4/hour	0,05	0,00	27019,7
5 /success.txt	40	261372	260284	262422	639,22	100,00%	6,4/min	0,21	0,00	2072,0
11 /wcal/js_cacher.php	7	75	43	138	32,58	0,00%	11,4/min	0,44	0,06	2372,0
10 /wcal/js_cacher.php	7	20	17	23	2,03	0,00%	11,4/min	0,77	0,06	4169,0
14 /wcal/css_cacher.php	7	75	23	323	101,52	0,00%	11,4/min	0,80	0,07	4340,0
30 /wcal/login.php	77	181	26	460	126,27	0,00%	4,1/hour	0,00	0,00	1691,0
31 /wcal/css_cacher.php	7	26	23	31	2,71	0,00%	11,4/min	0,80	0,07	4340,0
35 /wcal/login.php	77	2622	146	14843	2880,05	0,00%	4,1/hour	0,00	0,00	3687,8
36 /wcal/month.php	74	166196	4943	185291	37897,66	0,00%	3,9/hour	0,03	0,00	27019,9
38 /wcal/js_cacher.php	3	49	37	60	9,46	0,00%	6,4/min	0,25	0,04	2372,0
37 /wcal/css_cacher.php	3	31	29	34	2,05	0,00%	6,4/min	0,45	0,04	4340,0
55 /wcal/week.php	73	149319	4346	165449	29978,80	0,00%	3,9/hour	0,01	0,00	9150,4
59 /wcal/view_entry.php	72	199	37	1177	207,27	0,00%	3,8/hour	0,00	0,00	2709,6
60 /wcal/edit_entry.php	72	78	33	561	84,42	0,00%	3,8/hour	0,01	0,00	7988,1
62 /wcal/js_cacher.php	2	28	27	29	1,00	0,00%	6,1/min	0,49	0,04	4943,0
61 /wcal/js_cacher.php	2	41	28	54	13,00	0,00%	6,1/min	0,24	0,04	2372,0
64 /wcal/edit_entry_handler....	72	41564	1076	50415	7004,07	0,00%	3,8/hour	0,00	0,00	3105,0
65 /wcal/edit_entry_handler....	72	985	93	13503	1925,25	0,00%	3,8/hour	0,00	0,00	1594,1
66 /wcal/week.php	71	138002	119	160828	31934,50	0,00%	3,7/hour	0,01	0,00	9145,9
67 /success.txt	1	260419	260419	260419	0,00	100,00%	13,8/hour	0,01	0,00	2072,0
70 /wcal/login.php	32	37	26	102	14,29	0,00%	15,0/min	0,34	0,00	1392,0
71 /wcal/index.php	31	41	27	100	18,84	0,00%	14,7/min	0,22	0,00	919,0
Total	1124	69117	16	262422	86024,72	7,21%	58,7/hour	0,14	0,00	9107,0

Figura 38: Resultados productividad escenario 2 con 70 hilos

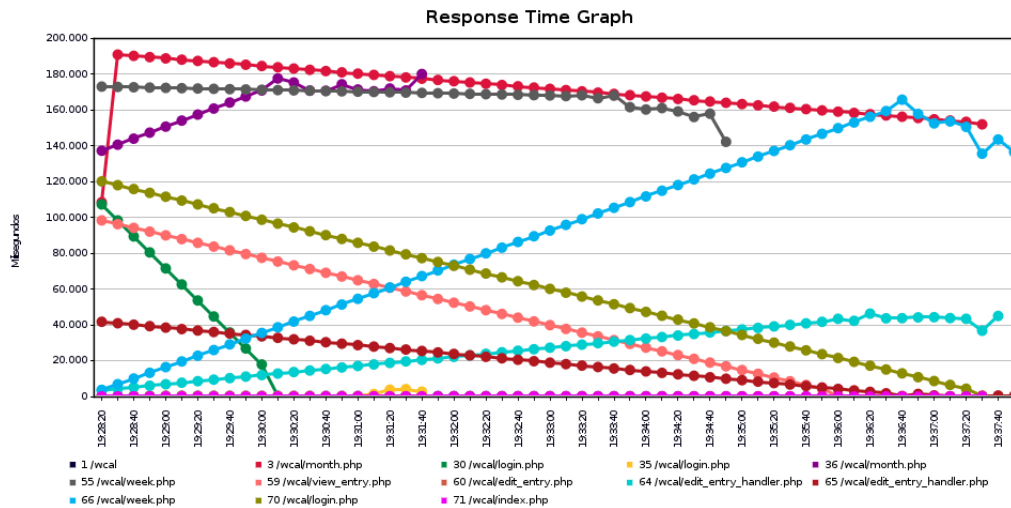


Figura 39: Resultados tiempo respuesta escenario 2 con 70 hilos

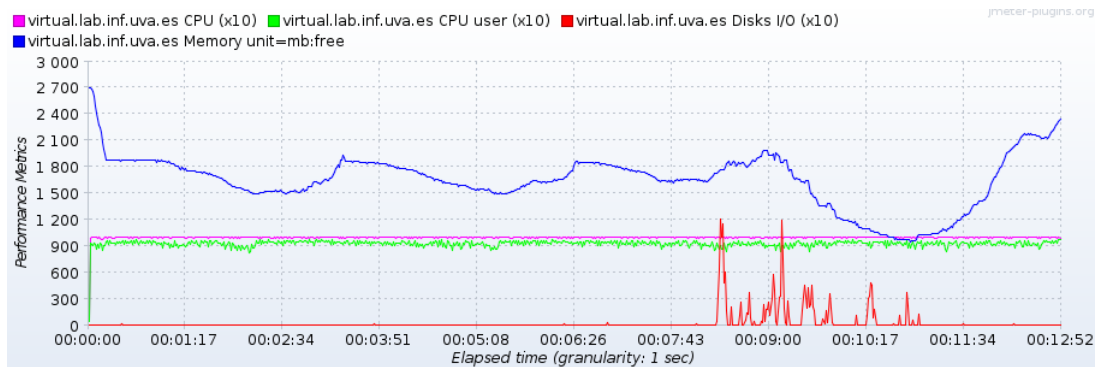


Figura 40: Resultados rendimiento servidor escenario 2 con 70 hilos

## ANEXO III.

Las tablas de resultados se han mostrado en los apartados de análisis correspondientes porque se creía necesario que se tuviera una vista de ellas mientras se describían.