		Escuela Politécnica Superior Ingeniería Informática Prácticas de Sistemas Informáticos 2			
Grupo	2312	Práctica	2	Fecha	18/03/2015
Alumno/a		García Roqué, Mario Valdemaro			
Alumno/a		García Teodoro, Roberto			

Práctica 2: Rendimiento

Ejercicio número 1:

Siguiendo todos los pasos anteriores, defina el plan completo de pruebas para realizar las tres ejecuciones secuenciales sobre los tres proyectos definidos hasta ahora (P1-base, P1-ws-cli, P1-ejb).

Adjunte el fichero generado P2.jmx al entregable de la práctica.

Definimos el plan de pruebas y hacemos unos pagos como se muestra en las siguientes imágenes:

Resultado del Muestreador	Petición	Datos de Respuesta
<p>Nombre del hilo:P1-ejb 3-1 Comienzo de muestra:2015-03-04 17:05:50 CET Tiempo de carga:13 Latencia:12 Tamaño en bytes:1327 Headers size in bytes: 287 Body size in bytes: 1040 Conteo de muestra:1 Conteo de error:0 Código de respuesta:200 Mensaje de respuesta:OK</p> <p>Cabeceras de respuesta: HTTP/1.1 200 OK X-Powered-By: JSP/2.2 Server: GlassFish Server Open Source Edition 4.0 Set-Cookie: JSESSIONID=5899078106ac4784fb6291877de0; Path=/P1-ejb-cliente; HttpOnly Content-Type: text/html; charset=ISO-8859-1 Date: Wed, 04 Mar 2015 16:06:12 GMT Transfer-Encoding: chunked</p> <p>HTTPSampleResult campos: ContentType: text/html; charset=ISO-8859-1 DataEncoding: ISO-8859-1</p>		
<p>POST http://10.5.7.2:8080/P1-ejb-cliente/procesapago</p> <p>POST data: numero=3326 6202 5028 6651&titular=Gabriel Moss Pomares&fechaEmision=09/10&fechaCaducidad=08/19&codigoVerificacion=452&importe=979&directConnection=false&usePrepared=true&idTransaccion=4793&idComercio=3&debug=false</p> <p>[no cookies]</p> <p>Cabeceras de petición: Connection: keep-alive Content-Length: 214 Content-Type: application/x-www-form-urlencoded</p>		

```
Resultado del Muestreador  Petición  Datos de Respuesta

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<title>Sistema de Pago con tarjeta</title>
<style type="text/css">
td.error {color:red;font-size:0.65em;}
</style>
</head>
<body>
<h1>Pago con tarjeta</h1>

<p>Pago realizado con &eacute;xito. A continuaci&oacute;n se muestra el
comprobante del mismo:</p>
<table>
<tr><td>idTransaccion:</td><td>4793</td></tr>
<tr><td>idComercio:</td><td>3</td></tr>
<tr><td>importe:</td><td>979.0</td></tr>
<tr><td>codRespuesta:</td><td>000</td></tr>
<tr><td>idAutorizacion:</td><td>2959</td></tr>
</table>
<p><a href="http://10.5.7.2:8080/P1-ejb-cliente/jsessionid=5899078106ac4784fb6291877de0">Volver al com
ercio</a></p>

<hr />
<small>Pr&aacute;cticas de Sistemas Inform&aacute;ticos II</small>
</body>
</html>
```

Ejercicio número 2:

Preparar los PCs con el esquema descrito. Para ello:

- Anote en la memoria de prácticas las direcciones IP asignadas a cada PC.
- Detenga el servidor de GlassFish de los PCs físicos
- Inicie los servidores GlassFish en las máquinas virtuales
- Repliegue todas las aplicaciones o pruebas anteriores (P1-base, P1-ws, etc), para limpiar posibles versiones incorrectas.
- Revise y modifique si es necesario los ficheros build.properties (propiedad “nombre”) de cada versión, de modo que todas las versiones tengan como URL de despliegue las anteriormente indicadas (P1-base, P1-ws, P1-ejb).
- Despliegue las siguientes prácticas: P1-base, P1-ws, P1-ejb, con el siguiente esquema:
 - El destino del despliegue en todos los casos será PC2VM con IP 10.X.Y.2 (as.host o as.host.client en P1-ws)
 - La base de datos en todos ellos será la de PC1VM con IP 10.X.Y.1 (db.host)
 - En el caso particular de P1-ws, el servidor SOAP estará en 10.X.Y.1 (variable as.host.server)

Tras detener / iniciar todos los elementos indicados, anotar la salida del comando “free” así como un pantallazo del comando “nmon” (pulsaremos la tecla “m” para obtener el estado de la RAM) tanto en las máquinas virtuales como los PCs físicos. Anote sus comentarios en la memoria.

Pruebe a ejecutar un pago “de calentamiento” por cada uno de los métodos anteriores y verifique que funcionan (comprobar resultados en el árbol de resultados).

IP ORDENADORES:

Ordenador con MV1: 10.10.66.65

Ordenador con MV2: 10.10.66.64

FREE INICIALES

```
si2@si2srv01:~$ free
              total        used        free      shared    buffers     cached
Mem:          767172      445432      321740          0       19240      152836
-/+ buffers/cache:      273356      493816
Swap:         153592           0       153592
si2@si2srv01:~$ _
```

Mv1

```
si2@si2srv02:~$ free
              total        used        free      shared    buffers     cached
Mem:          767172      355064      412108          0       16680      141748
-/+ buffers/cache:      196636      570536
Swap:         153592           0       153592
si2@si2srv02:~$ _
```

Mv2

```
e280635@12-23-66-64:~/Desktop/p2/jakarta-jmeter/bin$ free
              total        used        free      shared    buffers     cached
Mem:          2964416      2208648      755768          0       44660      1596368
-/+ buffers/cache:      567620      2396796
Swap:          3071992         4528      3067464
```

Ordenador con Mv2

```
              total        used        free      shared    buffers     cached
Mem:          2964416      2616348      348068          0       34208      1949764
-/+ buffers/cache:      632376      2332040
Swap:          3071992         4008      3067984
e280635@12-24-66-65:~/Desktop/p2$ █
```

Ordenador con Mv1

FREE FINALES

```
si2@si2srv01:~$ free
              total        used        free      shared    buffers     cached
Mem:          767172      451532      315640          0       19572      158036
-/+ buffers/cache:      273924      493248
Swap:         153592           0       153592
si2@si2srv01:~$
```

Mv1

```
si2@si2srv02:~$ free
              total        used        free      shared    buffers     cached
Mem:          767172      364376      402796          0       16992      142576
-/+ buffers/cache:      204808      562364
Swap:         153592           0       153592
si2@si2srv02:~$ _
```

Mv2

```
e280635@12-23-66-64:~/Desktop/p2/jakarta-jmeter/bin$ free
              total        used        free      shared    buffers     cached
Mem:          2964416      2215112      749304          0       45588      1602180
-/+ buffers/cache:      567344      2397072
Swap:          3071992         4528      3067464
```

Ordenador con Mv2

```
e280635@12-24-66-65:~/Desktop/p2$ free
              total        used        free      shared    buffers     cached
Mem:          2964416      2636456      327960          0       35144      1964904
-/+ buffers/cache:      636408      2328008
Swap:          3071992         4008      3067984
e280635@12-24-66-65:~/Desktop/p2$ █
```

Ordenador con Mv1

Vemos que en ambas máquinas virtuales el consumo de memoria ha subido casi 10 MB, ya que se han desplegado los diferentes servicios.

NMON INICIALES

nmon-12f		Hostname=si2srv01				Refresh= 2secs		09:25.58	
Memory Stats									
		RAM	High	Low	Swap				
Total MB		749.2	0.0	749.2	150.0				
Free MB		312.5	0.0	312.5	150.0				
Free Percent		41.7%	0.0%	41.7%	100.0%				
	MB			MB		MB			
			Cached=	149.3		Active=		304.2	
Buffers=	18.8	Swapcached=		0.0		Inactive =		110.8	
Dirty =	0.0	Writeback =		0.0		Mapped =		25.0	
Slab =	12.9	Commit_AS =		962.8		PageTables=		1.5	

Mv1

nmon-12f										[H for help]										Hostname=si2srv02										Refresh= 2secs										09:26.06									
Memory Stats																																																	
										RAM					High					Low					Swap																								
Total MB										749.2					0.0					749.2					150.0																								
Free MB										401.2					0.0					401.2					150.0																								
Free Percent										53.6%					0.0%					53.6%					100.0%																								
										MB										MB										MB																			
															Cached=					138.4					Active=					224.8																			
Buffers=										16.3					Swapcached=					0.0					Inactive =					103.1																			
Dirty =										0.0					Writeback =					0.0					Mapped =					21.9																			
Slab =										12.3					Commit_AS =					823.0					PageTables=					1.2																			

Mv2

nmon-13g										[H for help]										Hostname=12-23-66-64										Refresh= 2secs										18:29.24																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											
Memory Stats																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																			
		RAM		High		Low		Swap																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											

Ordenador con Mv2

nmon-13g					Hostname=12-24-66-65					Refresh= 2secs					18:29.25				
Memory Stats																			
	RAM	High	Low	Swap															
Total MB	2894.9	2043.5	851.4	3000.0															
Free MB	331.8	143.3	188.5	2996.1															
Free Percent	11.5%	7.0%	22.1%	99.9%															
	MB		MB		MB														
		Cached=	1910.5	Active=	1086.6														
Buffers=	33.3	Swapcached=	0.1	Inactive =	1375.2														
Dirty =	1.1	Writeback =	0.0	Mapped =	653.6														
Slab =	53.1	Commit_AS =	2962.9	PageTables=	9.6														

Ordenador con Mv1

NMON FINALES

nmon-12f [H for help] Hostname=si2srv01 Refresh= 2secs 09:32.34				
Memory Stats				
	RAM	High	Low	Swap
Total MB	749.2	0.0	749.2	150.0
Free MB	311.7	0.0	311.7	150.0
Free Percent	41.6%	0.0%	41.6%	100.0%
	MB		MB	MB
		Cached=	149.3	Active= 304.7
Buffers=	19.0	Swapcached=	0.0	Inactive = 111.1
Dirty =	0.0	Writeback =	0.0	Mapped = 25.0
Slab =	12.9	Commit_AS =	962.8	PageTables= 1.5

Mv1

nmon-12f Hostname=si2srv02 Refresh= 2secs 09:32.41				
Memory Stats				
	RAM	High	Low	Swap
Total MB	749.2	0.0	749.2	150.0
Free MB	392.0	0.0	392.0	150.0
Free Percent	52.3%	0.0%	52.3%	100.0%
	MB		MB	MB
		Cached=	139.2	Active= 233.7
Buffers=	16.6	Swapcached=	0.0	Inactive = 103.1
Dirty =	0.3	Writeback =	0.0	Mapped = 21.9
Slab =	12.4	Commit_AS =	830.4	PageTables= 1.2

Mv2

nmon-13g Hostname=12-23-66-64 Refresh= 2secs 18:33.54				
Memory Stats				
	RAM	High	Low	Swap
Total MB	2894.9	2043.5	851.4	3000.0
Free MB	728.7	290.1	438.6	2995.6
Free Percent	25.2%	14.2%	51.5%	99.9%
	MB		MB	MB
		Cached=	1565.7	Active= 811.0
Buffers=	44.4	Swapcached=	0.4	Inactive = 1259.9
Dirty =	0.3	Writeback =	0.0	Mapped = 542.4
Slab =	47.0	Commit_AS =	3242.5	PageTables= 10.8

Ordenador con Mv2

nmon-13g [H for help] Hostname=12-24-66-65 Refresh= 2secs 18:33.42				
Memory Stats				
	RAM	High	Low	Swap
Total MB	2894.9	2043.5	851.4	3000.0
Free MB	307.9	121.5	186.4	2996.1
Free Percent	10.6%	5.9%	21.9%	99.9%
	MB		MB	MB
		Cached=	1926.1	Active= 1094.8
Buffers=	34.2	Swapcached=	0.1	Inactive = 1389.9
Dirty =	0.9	Writeback =	0.0	Mapped = 664.6
Slab =	53.1	Commit_AS =	2989.8	PageTables= 9.7

Ordenador con Mv1

Vemos que después aumenta el uso de ram en las máquinas virtuales en torno a un 3%

Cuestión número 3:

Ejecute el plan completo de pruebas sobre las 3 versiones de la práctica, empleando el esquema de despliegue descrito anteriormente. Realice la prueba tantas veces como necesite para eliminar ruido relacionado con procesos periódicos del sistema operativo, lentitud de la red u otros elementos.

- Compruebe que efectivamente se han realizado todos los pagos. Es decir, la siguiente consulta deberá devolver “3000”:
`SELECT COUNT(*) FROM PAGO;`
- Compruebe que ninguna de las peticiones ha producido un error. Para ello revise que la columna %Error indique 0% en todos los casos.

Una vez que los resultados han sido satisfactorios:

- Anote los resultados del informe agregado en la memoria de la práctica.
- Salve el fichero server.log que se encuentra en la ruta glassfish/domains/domain1/logs de Glassfish y adjúntelo con la práctica.
- Añada a la memoria de prácticas la siguiente información: ¿Cuál de los resultados le parece el mejor? ¿Por qué? ¿Qué columna o columnas elegiría para decidir este resultado? (Texto de la respuesta)

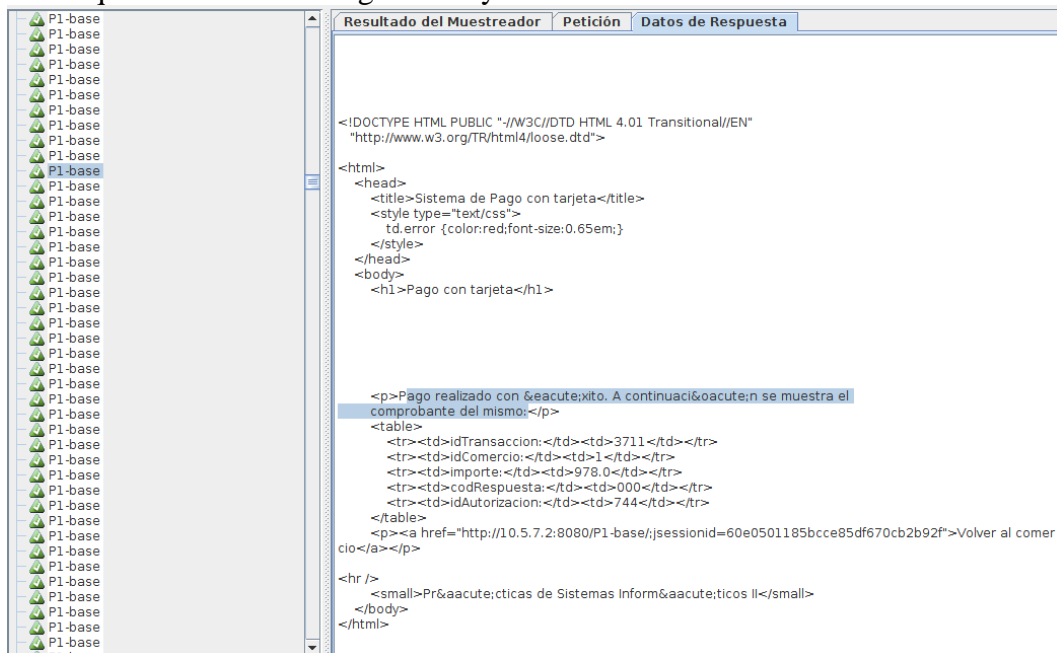
Comprobamos que se han recibido los pagos:

```
visa=# select count(*) from pago;
count
-----
3000
(1 row)
```

```
visa=# █
```

Así ha sido.

Comprobamos que no ha habido ningún fallo y así ha sido.



Resultado del Muestreador | Petición | Datos de Respuesta

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<title>Sistema de Pago con tarjeta</title>
<style type="text/css">
td.error {color:red;font-size:0.65em;}
</style>
</head>
<body>
<h1>Pago con tarjeta</h1>

<p>Pago realizado con éxito. A continuación se muestra el
comprobante del mismo:</p>
<table>
<tr><td>idTransaccion:</td><td>3711</td></tr>
<tr><td>idComercio:</td><td>1</td></tr>
<tr><td>importe:</td><td>978.0</td></tr>
<tr><td>codRespuesta:</td><td>000</td></tr>
<tr><td>idAutorizacion:</td><td>744</td></tr>
</table>
<p><a href="http://10.5.7.2:8080/P1-base/?sessionId=60e0501185bce85df670cb2b92f">Volver al comer
cio</a></p>

<hr />
<small>Prácticas de Sistemas Informáticos II</small>
</body>
</html>
```

No hay fallo en P1-base

Resultado del Muestreador Petición Datos de Respuesta

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<title>Sistema de Pago con tarjeta</title>
<style type="text/css">
    td.error {color:red;font-size:0.65em;}
</style>
</head>
<body>
<h1>Pago con tarjeta</h1>

<p>Pago realizado con éxito. A continuación se muestra el
comprobante del mismo:</p>
<table>
<tr><td>idTransaccion:</td><td>1067</td></tr>
<tr><td>idComercio:</td><td>2</td></tr>
<tr><td>importe:</td><td>871.0</td></tr>
<tr><td>codRespuesta:</td><td></td></tr>
<tr><td>idAutorizacion:</td><td></td></tr>
</table>
<p><a href="http://10.5.7.2:8080/P1-ws-cliente/?jsessionId=60e7a7019fc5c68b586ee5b835d8">Volver al c
omercio</a></p>

<hr />
<small>Prácticas de Sistemas Informáticos II</small>
</body>
</html>

```

Texto

No hay fallo en P1-ws

Resultado del Muestreador Petición Datos de Respuesta

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<title>Sistema de Pago con tarjeta</title>
<style type="text/css">
    td.error {color:red;font-size:0.65em;}
</style>
</head>
<body>
<h1>Pago con tarjeta</h1>

<p>Pago realizado con éxito. A continuación se muestra el
comprobante del mismo:</p>
<table>
<tr><td>idTransaccion:</td><td>1313</td></tr>
<tr><td>idComercio:</td><td>3</td></tr>
<tr><td>importe:</td><td>835.0</td></tr>
<tr><td>codRespuesta:</td><td>000</td></tr>
<tr><td>idAutorizacion:</td><td>2264</td></tr>
</table>
<p><a href="http://10.5.7.2:8080/P1-ejb-cliente/?jsessionId=60f8ff91f9ab527acc30036d91dd">Volver al co
mercio</a></p>

<hr />
<small>Prácticas de Sistemas Informáticos II</small>
</body>
</html>

```

Texto

Buscar: Encontrar ☐ Sensible a mayúsculas ☒ Expresión regular

No hay fallo en P1-ejb

Hacemos tres pruebas y obtenemos los siguientes valores del informe agregado:

Etiqueta	# Muestras	Media	Mediana	Línea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
P1-base	1000	8	8	10	5	377	0,00%	113,3/sec	145,1
P1-ws-cliente	1000	63	60	74	49	404	0,00%	15,6/sec	20,1
P1-ejb-cliente	1000	10	9	13	6	748	0,00%	88,0/sec	114,1
Total	3000	27	10	65	5	748	0,00%	35,6/sec	45,8

Etiqueta	# Muestras	Media	Mediana	Línea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
P1-base	1000	8	7	10	5	817	0,00%	109,8/sec	140,6
P1-ws-cliente	1000	64	60	75	49	859	0,00%	15,5/sec	19,9
P1-ejb-cliente	1000	10	9	13	6	653	0,00%	92,0/sec	119,2
Total	3000	27	10	63	5	859	0,00%	34,9/sec	45,0

Etiqueta	# Muestras	Media	Mediana	Línea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
P1-base	1000	8	7	9	4	783	0,00%	116,7/sec	149,5
P1-ws-cliente	1000	64	61	76	49	805	0,00%	15,3/sec	19,6
P1-ejb-cliente	1000	10	9	13	6	658	0,00%	91,1/sec	118,0
Total	3000	27	10	65	4	805	0,00%	34,9/sec	44,9

Vemos el 0.00% de error

El mejor resultado parece ser el P1-base, para basarnos en esa elección seleccionamos la columna rendimiento y vemos que el mayor valor es el de P1-base.

Ya que el base es muy sencillo, mientras que los otros implementan colas de mensajes o web services, que inevitablemente afectan al rendimiento.

Ejercicio número 4:

Adaptar la configuración del servidor de aplicaciones a los valores indicados. Guardar, como referencia, la configuración resultante, contenida en el archivo de configuración localizado en la máquina virtual en \$J2EE_HOME/domains/domain1/config/domain.xml1. Para obtener la versión correcta de este archivo es necesario detener el servidor de aplicaciones. Incluir este fichero en el entregable de la práctica. Se puede copiar al PC del laboratorio con scp

Copiamos el fichero y vemos que, por ejemplo, se ha añadido el -Xms512m

```
<jvm-options>-Xmx512m</jvm-options>
<jvm-options>-Djdbc.drivers=org.apache.derby.jdbc.ClientDriver</jvm-options>
<jvm-options>-Dosgi.shell.telnet.port=6666</jvm-options>
<jvm-options>-Dosgi.shell.telnet.maxconn=1</jvm-options>
<jvm-options>-Djava.ext.dirs=${com.sun.aas.javaRoot}/lib/ext${path.separator}${com.sun.aas.javaRoot}/
<jvm-options>-Djava.security.policy=${com.sun.aas.instanceRoot}/config/server.policy</jvm-options>
<jvm-options>-Dgosh.args=-nointeractive</jvm-options>
<jvm-options>-Xms512m</jvm-options>
```

Ejercicio número 5:

Registrar en la hoja de cálculo de resultados los valores de configuración que tienen estos Parámetros

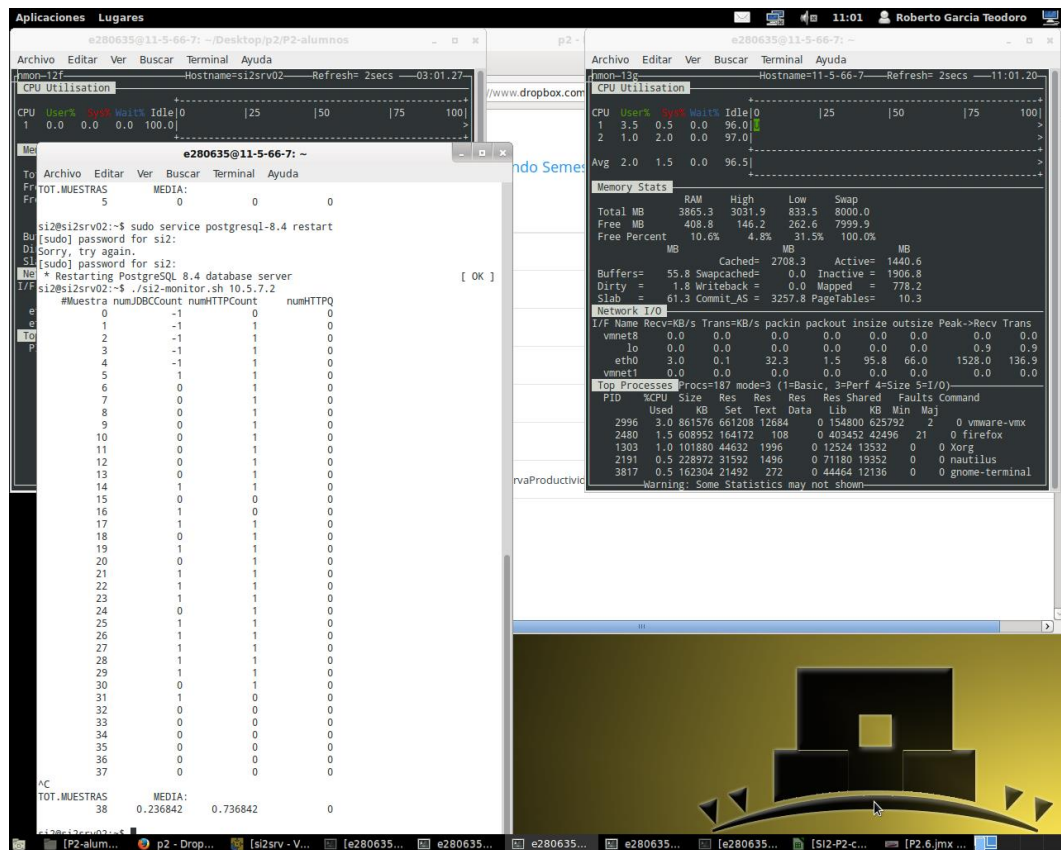
Se han registrado en la hoja de cálculo

Ejercicio número 6:

Tras habilitar la monitorización en el servidor, repita la ejecución del plan de pruebas anterior. Durante la prueba, vigile cada uno de los elementos de monitorización descritos hasta ahora. Responda a las siguientes cuestiones:

- **A la vista de los resultados, ¿qué elemento de proceso le parece más costosa? ¿Red? ¿CPU? ¿Acceso a datos? En otras palabras, ¿cuál fue el elemento más utilizado durante la monitorización con nmon en un entorno virtual? (CPU, Memoria, disco,...)**
- **¿Le parece una situación realista la simulada en este ejercicio? ¿Por qué?**
- **Teniendo en cuenta cuál ha sido el elemento más saturado, proponga otro esquema de despliegue que resuelva esa situación.**

9



Realizamos tres fotos de la ejecución, dos durante el proceso y una al finalizar éste.

Mientras se procesan los pagos, el uso de la CPU es máximo, se mantiene al 100% todo el tiempo, viendo que se produce un cambio en el uso de CPU aumentando el consumo de CPU por parte del sistema en lugar del usuario, debido posiblemente al intercambio de datos vía red.

En la primera foto disponemos de más memoria libre que en la segunda. Debido a que conforme avanza la ejecución, necesitamos almacenar más datos en memoria principal. Esto va a aumentar debido a la saturación dada en el sistema y se mantendrá hasta que vayan finalizando las peticiones.

Vemos también como aumenta el uso de red por parte de la máquina virtual 2, quedando reflejado en los valores bajos vistos en la primera foto, que luego crecen en la segunda, por ejemplo, la tasa de transferencia de emisión aumenta de 13.1KB/s a 122.2 KB/s, también aumenta la de recepción.

Por lo que deducimos que el componente que más estamos utilizando es la CPU, ya que está al 100% de su capacidad durante casi toda la ejecución.

No nos parece una situación realista, pues nunca tienes un servidor completamente vacío y te llegan 1000 peticiones en unos pocos segundos.

Para mejorar el rendimiento proponemos aumentar el número de núcleos del procesador en la máquina virtual 2, ya que es lo que más parece usarse.

Ejercicio número 7:

Preparar el script de JMeter para su ejecución en el entorno de pruebas. Cambiar la dirección destino del servidor para que acceda al host en el que se encuentra el servidor de aplicaciones. Crear también el directorio datagen en el mismo directorio donde se encuentre el script, y copiar en él el archivo listado.csv, ya que de dicho archivo, al igual que en las prácticas anteriores, se obtienen los datos necesarios para simular el pago.

A continuación, realizar una ejecución del plan de pruebas, con un único usuario, una única

ejecución, y un think time bajo (entre 1 y 2 segundos) para verificar que el sistema funciona correctamente. Comprobar, mediante el listener View Results Tree que las peticiones se ejecutan correctamente, no se produce ningún tipo de error y los resultados que se obtienen son los adecuados.

Una vez comprobado que todo el proceso funciona correctamente, desactivar dicho listener del plan de pruebas para que no aumente la carga de proceso de JMeter durante el resto de la prueba.

Este ejercicio no genera información en la memoria de la práctica, realízelo únicamente para garantizar que la siguiente prueba va a funcionar.

Este ejercicio no genera información en la memoria.

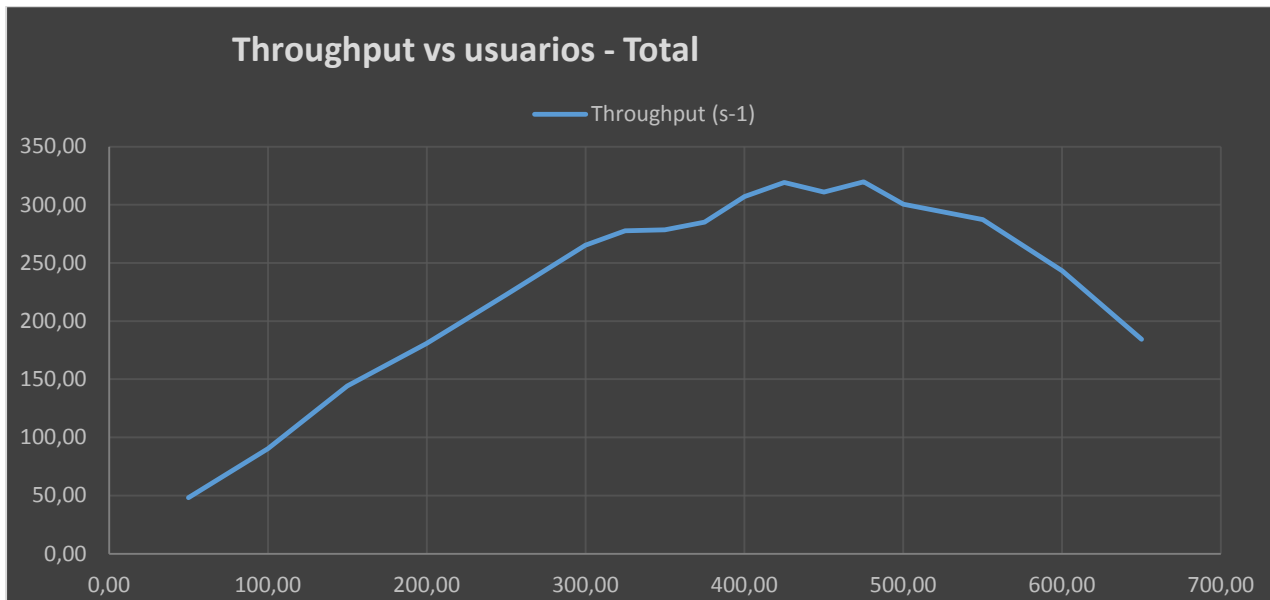
Ejercicio número 8:

Obtener la curva de productividad, siguiendo los pasos que se detallan a continuación:

- **Previamente a la ejecución de la prueba se lanzará una ejecución del script de pruebas (unas 10 ejecuciones de un único usuario) de la que no se tomarán resultados, para iniciar el sistema y preparar medidas consistentes a lo largo de todo el proceso. Borrar los resultados de la ejecución anterior. En la barra de acción de JMeter, seleccionar Run -> Clear All.**
- **Borrar los datos de pagos en la base de datos VISA.**
- **Seleccionar el número de usuarios para la prueba en JMeter (parámetro C de la prueba)**
- **Conmutar en JMeter a la pantalla de presentación de resultados, Aggregate Report.**
- **Ejecutar la prueba. En la barra de acción de JMeter, seleccionar Run -> Start.**
- **Ejecutar el programa de monitorización si2-monitor.sh.**
 - **Arrancarlo cuando haya pasado el tiempo definido como rampa de subida de usuarios en JMeter.**
 - **Detenerlo cuando esté a punto de terminar la ejecución de la prueba.**
 - **Registrar los resultados que proporciona la monitorización en la hoja de cálculo.**
- **Durante el periodo de monitorización anterior, vigilar que los recursos del servidor si2srv02 y del ordenador que se emplea para realizar la prueba no se saturen, mediante inspección del programa de monitorización nmon que se ejecuta en ambas máquinas.**
- **Finalizada la prueba, salvar el resultado de la ejecución del Aggregate Report en un archivo, y registrar en la hoja de cálculo de resultados los valores Average, 90% line y Throughput para las siguientes peticiones:**
 - **ProcesaPago.**
 - **Total.**

Una vez realizadas las iteraciones necesarias para alcanzar la saturación, representar la curva de Throughput versus usuarios.

Se ha representado la curva en la hoja de Excel

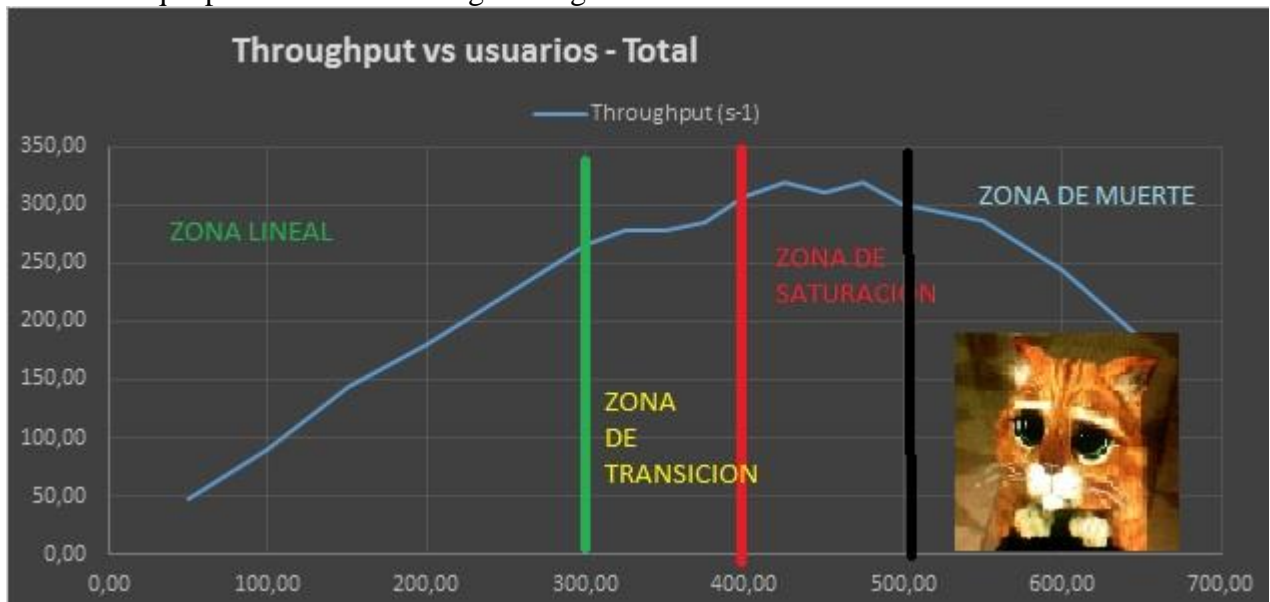


Vemos la Zona Lineal hasta unos 300 usuarios

La zona de Transición se encuentra entre 300 y 425 usuarios

La Zona de Saturación se encuentra a partir de 425 usuarios, vemos incluso la zona donde se comienza a rechazar peticiones y por eso decae el Throughput (aproximadamente a partir de 475 usuarios)

Divisiones que podemos ver en la siguiente gráfica:



La cuarta zona, denominada por nosotros zona de muerte, es una zona donde comienza a bajar el throughput debido a que el sistema comienza a descartar peticiones, ya que está excesivamente saturado. Fue nuestra profesora quien nos hizo llegar a esa conclusión, y en su honor hemos querido poner el gatito.

Ejercicio número 9:

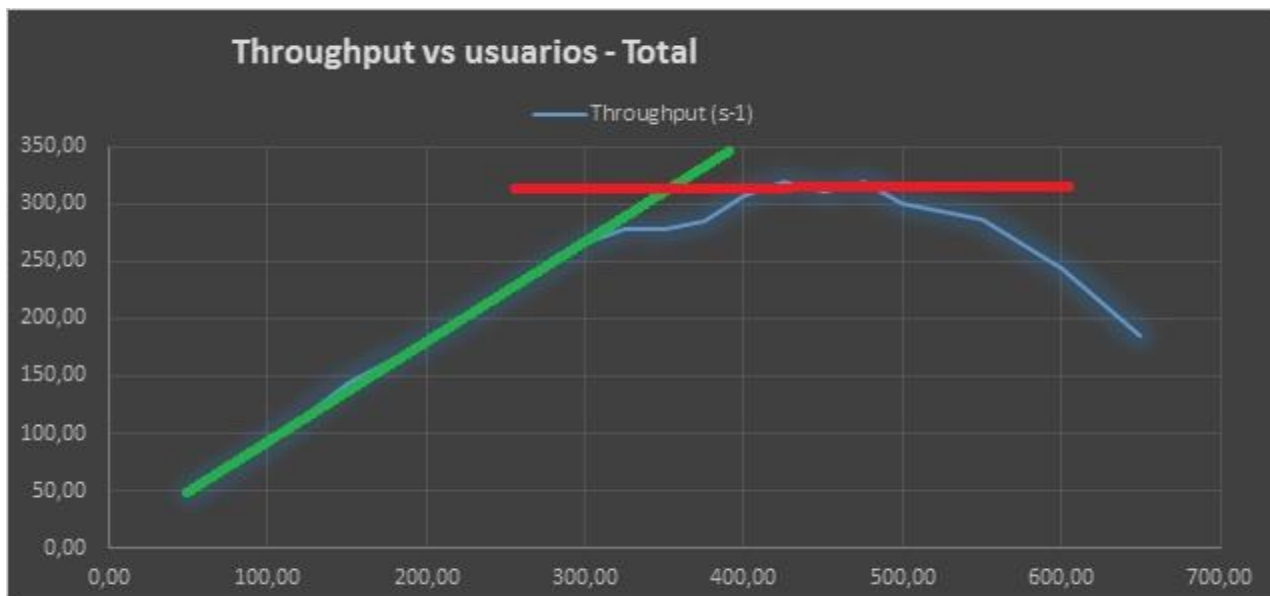
Responda a las siguientes cuestiones:

- **A partir de la curva obtenida, determinar para cuántos usuarios conectados se produce el punto de saturación, cuál es el máximo throughput que se alcanza en el mismo, y el throughput máximo que se obtiene en zona de saturación.**
- **Analizando los valores de monitorización que se han ido obteniendo durante la**

elaboración de la curva, sugerir el parámetro del servidor de aplicaciones que se cambiaría para obtener el punto de saturación en un número mayor de usuarios.

- Realizar el ajuste correspondiente en el servidor de aplicaciones, reiniciarlo y tomar una nueva muestra cercana al punto de saturación. ¿Ha mejorado el rendimiento del sistema? Documente en la memoria de prácticas el cambio realizado y la mejora obtenida.

El punto de saturación se encuentra en el corte de la línea que extiende la zona lineal con la línea horizontal a la que tiende la cumbre de la zona de saturación.



Como podemos observar, el punto de saturación se encuentra en unos 350 usuarios, cuando el Throughput vale ~275

En un principio todo apuntaba a que teníamos que aumentar el tamaño del Thread pool size para conseguir que se conectaran más clientes al servidor, pero tras probar con 10 y 50, hemos visto que todo el rato quedaba saturado el servidor independientemente de ese número (tal como podíamos observar en que el procesador de la MV2 siempre se encontraba al 100% de uso), por lo que hemos creído más conveniente aumentar el número de núcleos de la máquina virtual 2.

Sin embargo, más tarde nos pusimos a pensar si debíamos mirar el throughput en lugar de exclusivamente el uso de CPU. Así que montamos de nuevo y pusimos el Thread pool size a 20, ejecutamos y nos dispusimos a ver el nuevo throughput para 650 hilos, valor para el que antes estaba saturado, obteniendo los siguientes valores.

Aplicaciones Lugares

SI2-P2-curvaProductividad.ods - LibreOffice Calc

Archivo Editar Ver Insertar Formato Herramientas Datos Ventana Ayuda

Calibri 11

A15:H15 $f(x)$ Σ = 650

Parámetros de configuración										
Elemento	Parámetro	Valor	Descripción de la prueba							
VM Settings	Heap Máx. (MB)	512								
VM Settings	Heap Mín. (MB)	512								
HTTP Service	Max.Thread Count	5								
HTTP Service	Queue size	4096								
Web Container	Max.Sessions	-1								
Visa Pool	Max.Pool Size	32								

Prueba de rendimiento										
Usuarios	Sistema	Monitores				Total			ProcesaPago	
		CPU, average (%)	Visa Pool used Conns	HTTP Current Threads	Conn queued, instant	Average (ms)	90% line (ms)	Throughput (s ⁻¹)	Average (ms)	90%line (ms)
650,00	100,00	0,00	5,00	490,77	2436,00	3294,00	184,40	2461,00	3316,00	92,70

P2-curvaProductividad.jmx (/home/alumnos/e280635/Desktop/p2/P2-alumnos/P2-curvaProductividad.jmx) - Apache JMeter (2.5.1 r1177103)

Editar Lanzar Opciones Ayuda

0 / 650

Informe Agregado

Nombre: Aggregate Report

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo Navegar... Log/Mostrar sólo: ☐ Escribir en Log Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
/P1-base/comienzapago	6500	1009	1019	1412	3	9376	0,00%	153,0/sec	296,9
/P1-base/proc...	6500	1035	1066	1441	9	9391	0,00%	152,0/sec	185,8
Total	13000	1022	1042	1424	3	9391	0,00%	302,2/sec	477,8

va de rendimiento aplicación Vis
Consultas

Aggregate Report

View Results Tree

User Defined Variables

IdTransaccion

Importe compra

Configuración del CSV Data

HTTP Request Defaults

HTTP Cookie Manager

HTTP Cache Manager

Pago

/P1-base/comienzapago

HTTP Header Manager

Uniform Random Time

/P1-base/proc...

HTTP Header Manager

Uniform Random Time

o de Trabajo

[e280635@12...]

[p2]

[si2srv - VMwa...]

[SI2-P2-curvaP...]

[e280635@12...]

P2-curvaProdu...

e280635@12...

SI2-P2-curvaPr...

302.2 de nuevo rendimiento frente al 184.4 anterior, por lo que vemos que ya no satura en esa posición, pese a que el uso de CPU sigue siendo 100%.

Tomamos valores para 850 hilos:

Etiqueta	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
/P1-base/comi...	8500	699	794	1042	2	2294	0,00%	243,7/sec	472,8
/P1-base/proc...	8500	741	837	1078	9	2343	0,00%	242,6/sec	296,5
Total	17000	720	815	1063	2	2343	0,00%	480,7/sec	760,1

Y 1000 hilos:

Etiqueta	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
/P1-base/comi...	17000	1072	1094	1351	2	31814	0,00%	93,4/sec	181,2
/P1-base/proc...	17000	1164	1139	1422	4	31858	0,00%	93,5/sec	93,6
Total	34000	1118	1116	1392	2	31858	0,00%	186,6/sec	274,4

Por lo que vemos, el Throughput de 850 mejora, pero el de 1000 es peor, por lo que a 1000 volvería a saturar.