

Optimización del Rendimiento (*Performance*) de Sistemas Informáticos

Theodore Hope

Club de Investigación Tecnológica
28-octubre-2010

Introducción

- "El sistema está [muy] lento"
- ¿Comparado con qué?
 - ¿Con la semana pasada?
 - ¿Con Google?
 - ¿Con un SLA?
- ¿Qué significa "el sistema"?
 - Para los usuarios: Una unidad de trabajo.
 - Para el personal de TI: Mil cosas distintas.

¿Qué Importa?



<http://www.corbisimages.com/Enlargement/Enlargement.aspx?id=42-20714353&tab=details&caller=search>

Definiciones

- Rendimiento (*Performance*)
- Tiempo de Respuesta (*Response Time*)
- Carga (*Load*) y Utilización
- Desempeño (*Productividad, Throughput*)
- Escalabilidad
- Concurrency
- Capacidad
- Eficiencia

Rendimiento (*Performance*)

Tiempo de Respuesta (*Response Time*)

- El tiempo de ejecución de una tarea, medido como...
 - Desempeño (*throughput*)
 -
 - Tiempo de Respuesta para una tarea del usuario.
- Aquí: Rendimiento (R) = Tiempo de Respuesta
- Si hay cola (queuing), entonces
R = Tiempo de Espera + Tiempo de Servicio
- ¿Qué estoy midiendo?

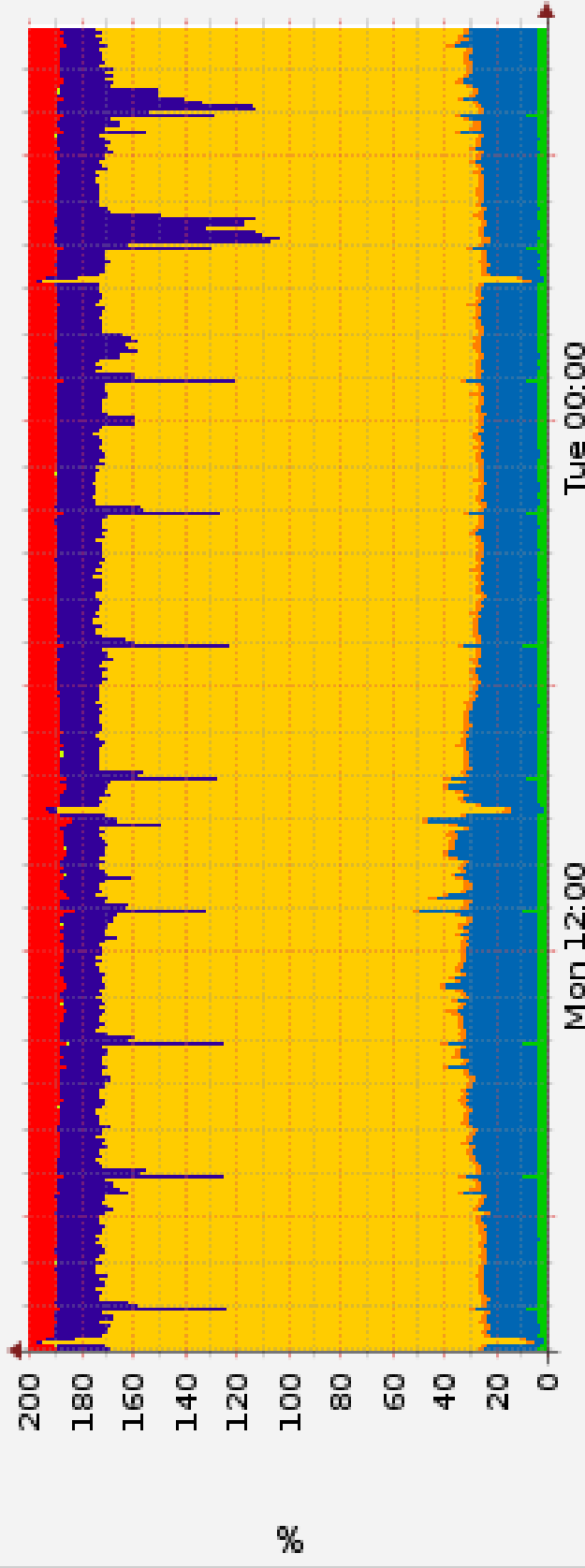
Carga (*Load*) y Utilización

- Carga:
 - # solicitudes de servicio / tiempo
 - e.g., 1000 pkts/s
- Utilización:
 - Una medida de carga.
 - Fracción del tiempo que un recurso está ocupado.
 - Ojo: Nunca mayor que 100%
- $U = 100\% \Rightarrow$ Cola (*backlog, queueing*)

Ejemplo: Utilización de CPU

RRDTOOL / TOBI OETIKER

CPU usage - by day



	Cur:	Min:	Avg:	Max:
system	3.31	1.33	3.48	9.35
user	26.58	3.15	23.80	42.21
nice	2.37	2.11	2.57	4.50
idle	142.14	77.52	137.77	186.60
iowait	13.01	1.21	20.58	86.34
irq	0.00	0.00	0.00	0.00
softirq	0.03	0.00	0.04	0.08
steal	12.60	3.47	11.78	17.99

Munin 1.4.4

Last update: Tue Oct 12 08:55:06 2010

Utilización

- ¿ A qué nivel de utilización debo operar ?
- Utilización = 100%
- Implica uso pleno del recurso.
- Ideal ...
- ... pero no viable porque el mundo no es ideal.
- Carga varía (significativamente)
- Picos en carga pueden causar encolamiento
- Necesitamos un colchón (i.e., $U < 100\%$)

Escalabilidad

- Escalabilidad es ...
- Escalabilidad != Rendimiento
- Escalabilidad
 - => Compromiso con niveles de servicio futuros.
- Rendimiento
 - => Compromiso con nivel de servicio actual.

Desempeño (Productividad, Throughput)

- Número de tareas completadas / unidad de tiempo
 - e.g., 1000 transacciones completados / segundo
- Desempeño != Rendimiento
 - No son necesariamente recíprocos.
 - 1000 tps no implica 0.001 s/t
 - Ambos son importantes por sí solos.
 - La relación entre ambos es importante.
- ¿Me importa Desempeño más que Rendimiento?

Concurrencia, Capacidad, Eficiencia

- Concurrencia
 - Número máximo de operaciones que pueden ser atendidas simultáneamente.
- Capacidad
 - El punto máximo de Carga que no implique degradación del Rendimiento.
- Eficiencia
 - Medida inversa del desperdicio.



<http://oscar.taurus.com/~jeff/2100/sven/hp3.jpg>



http://upload.wikimedia.org/wikipedia/commons/4/41/Teletype_with_papertape_punch_and_reader.jpg

Tarea lenta: ¿ por qué ?

(a) Está haciendo mucho trabajo.

(b) Algo más lo está atrasando.

Rendimiento \Leftrightarrow Desempeño

Queremos que el sistema nos dé lo siguiente:

- El mejor rendimiento (tiempo de respuesta) posible.
- El mejor desempeño (*throughput*) posible.

Pero... las dos metas son contradictorias:

- Optimizar tiempo de respuesta
 \Rightarrow Minimizar la carga.
- Optimizar el desempeño
 \Rightarrow Maximizar la carga.

Entonces: ¿ WTF ?

EXTRAORDINARY WTF COUNCIL MEETING

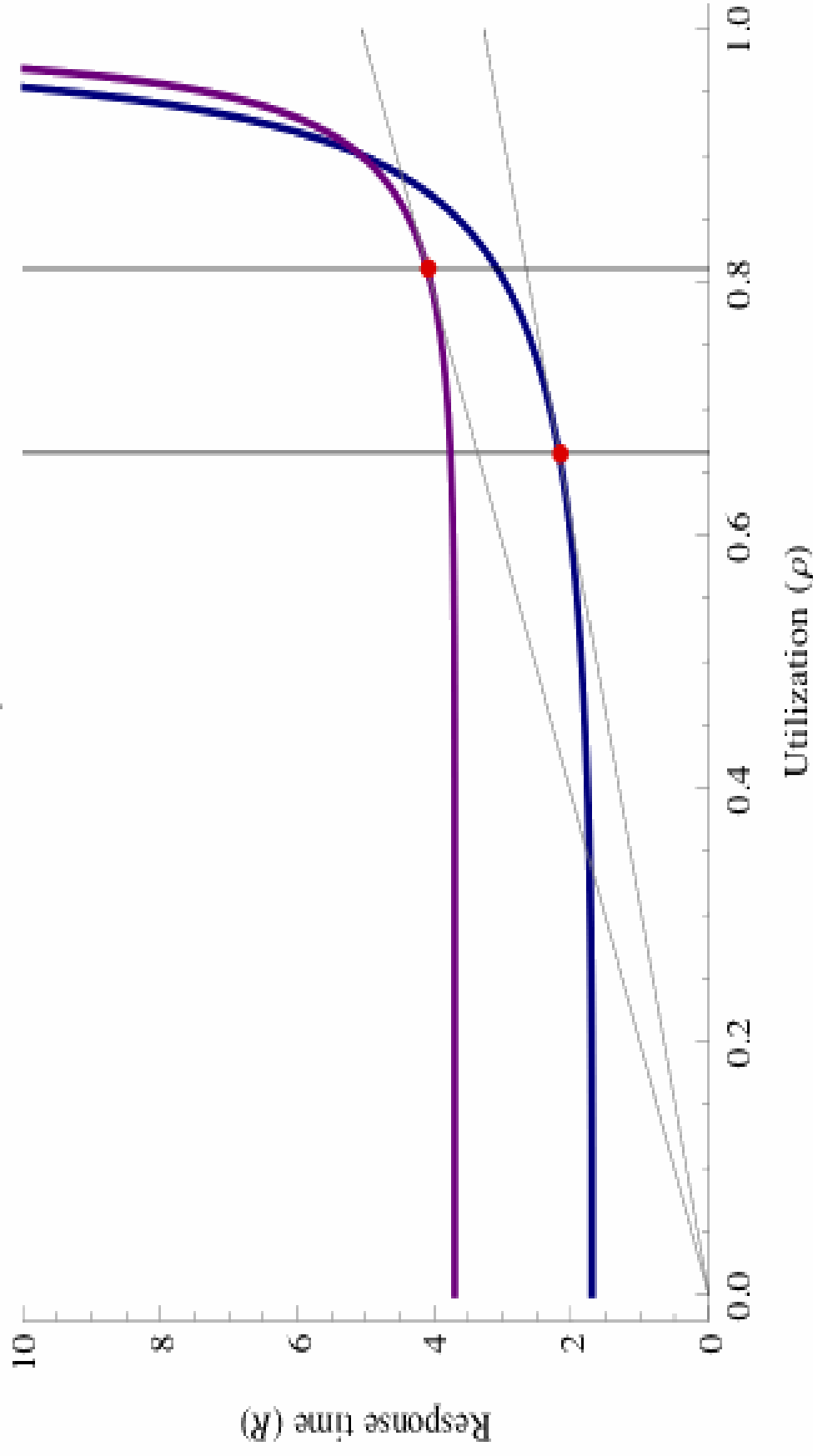
Conference Hall, Egyptian (Wafiq) Convention
Cairo, Egypt / November 24, 2009



"La Rodilla" ("The Knee")

$M/M/4, \rho^* = 0.665006$

$M/M/16, \rho^* = 0.810695$



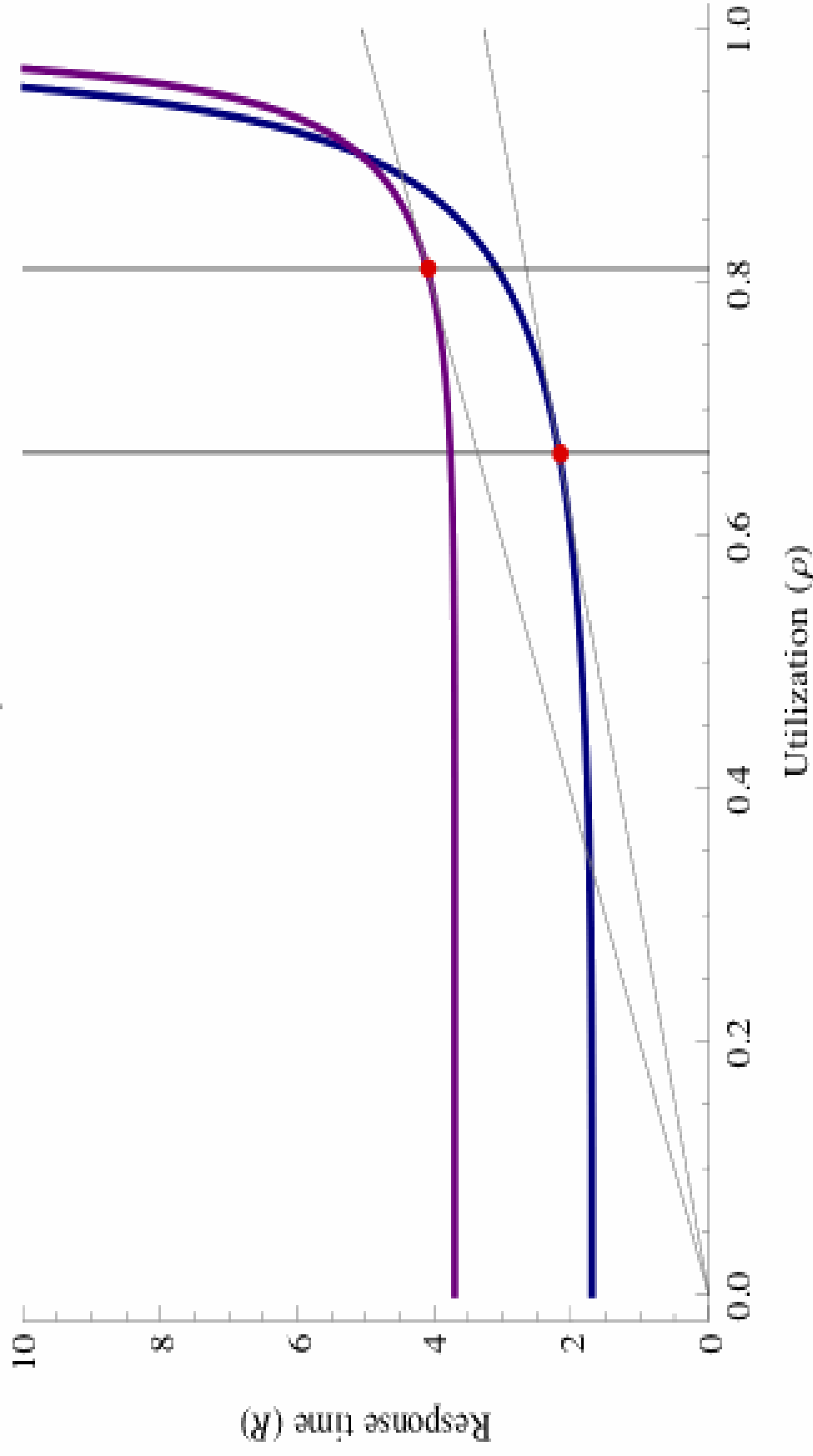
"La Rodilla" ("The Knee")

- El valor de utilización al cual se obtiene el balance óptimo entre tiempo de respuesta y utilización.
- El punto al cual se minimiza R / U
- Si la carga excede el punto de rodilla, tendremos problemas de rendimiento.

"La Rodilla" ("The Knee")

$M/M/4, \rho^* = 0.665006$

$M/M/16, \rho^* = 0.810695$



Mejorando el Rendimiento (típico)

Abordaje típico:

1. Escoger alguna métrica cuyo valor aparenta ser inaceptable.
2. Cambiar algo [incluso el hardware] con la intención de mejorar esa métrica.
3. Si los usuarios no perciben una mejora satisfactoria en rendimiento, regresar al punto #1.
4. Regresar de todas maneras al punto #1 y buscar algo más que se pueda cambiar.



<http://www.corbisimages.com/Enlargement/Enlargement.aspx?id=42-16623589&tab=details&caller=search>

"Method R" (Millsap & Holt, 1999)

Resumen:

- Identificar la[s] tarea[s] más importante[s].
- Medir su tiempo de respuesta en detalle.
- Optimizar ese tiempo de respuesta de la manera más eficiente*.
- Repetir hasta que el sistema esté óptimo.

"Method R" (Cont.)

- Seleccionar las tareas para las cuales el negocio requiere una mejora en el rendimiento.
- Recolectar datos que permiten identificar las causas de los tiempos de respuesta para cada acción de usuario (cuando el rendimiento no es óptimo).
- Seleccionar la actividad de optimización que tenga el mayor impacto neto en el negocio, y optimizarlo.
- Repetir.

"Method R" (Cont.)

- Fomenta la alineación de las prioridades de optimización con las prioridades del negocio.
- Se enfoca primero en reducir los componentes más grandes del tiempo de respuesta de las acciones más importantes del usuario.
- La forma correcta de priorizar las actividades de optimización es respondiendo primero a los síntomas más importantes.

Optimización de una Tarea

- Decidir qué voy a medir, y cómo.
- Medir (capturar datos).
- Generar y priorizar el perfil.
- Optimizar sub-tareas.
- Valorar la capacidad del sistema.

小菜類

\$29

\$29

\$27

\$26

B1

醉鷄

Drunken Chicken

B2

紅燒牛肉

House Special Stewed Beef

B3

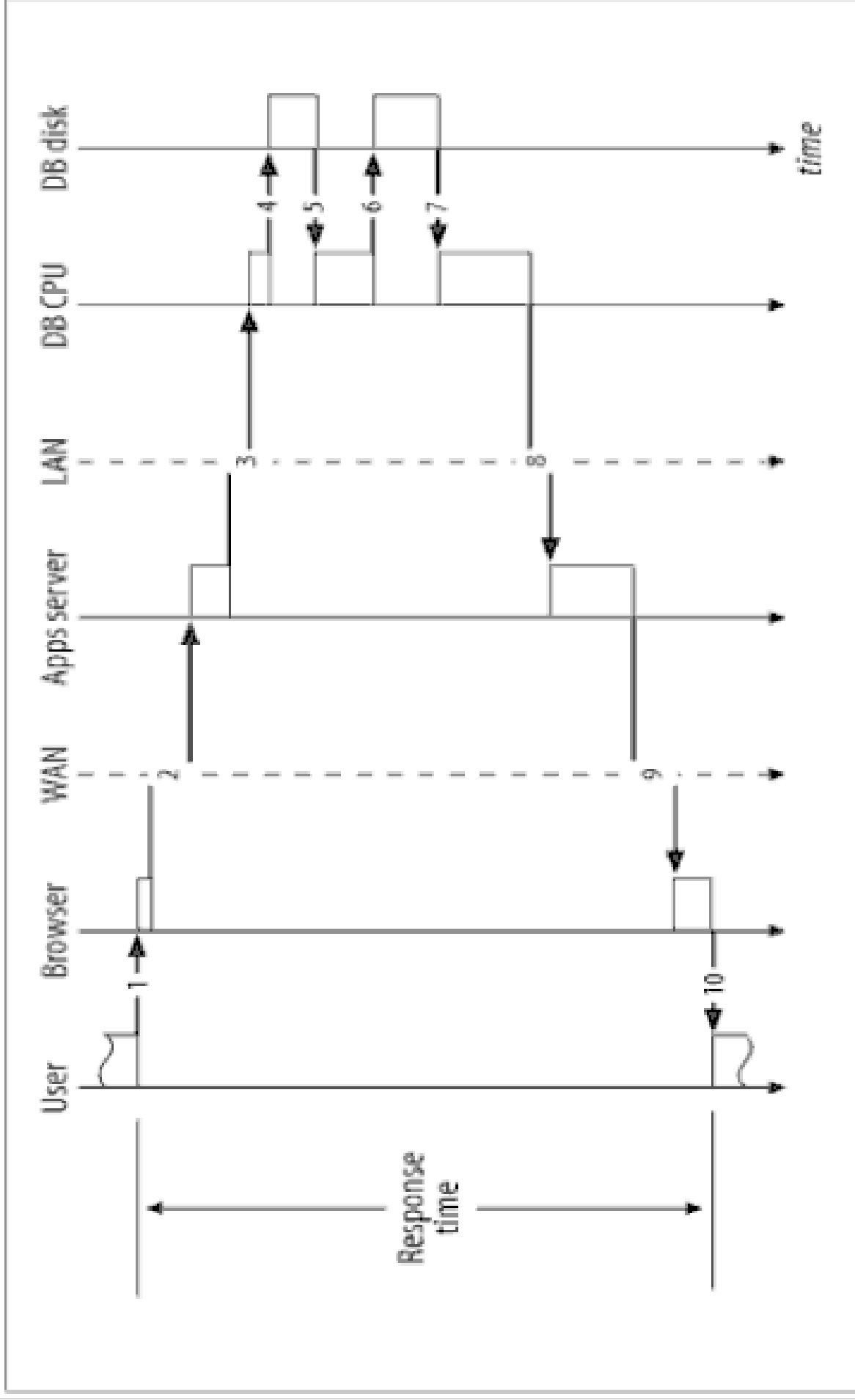
台式魯肉

Taiwanese Lu desktop meat

Diagrama de Secuencia y Perfilado / Perfil de Ejecución

- Diagrama de Secuencia desglosa los componentes de una acción.
- ¿Y el tiempo que representa cada uno?
=> ¡Perfilado (*Profiling*) !
- Nos dice qué optimizar.
 - "Ley" de Amdahl (aplicado a mejoramiento)

Diagrama de Secuencia



Perfil de Ejecución (*Profile*)

- Diagnosticar para encontrar cuellos de botella.
- Dónde está haciendo qué, y durante cuánto tiempo.
- A todo nivel:
 - Aplicaciones
 - Servicios cerrados (e.g., RDBMS)
 - Sistema operativo
- Si no lo tiene, ¡hágalo usted mismo !
 - Instrumentación

Perfil de Ejecución (prog Perl)

```
Total Elapsed Time = 9.559537 Seconds
User+System Time = 5.329537 Seconds
```

Exclusive Times

%Time	ExclSec	CumulS	#Calls	sec/call	Csec/c	Name
13.1	0.783	0.783	531	0.0013	0.0013	DBI::st::execute
5.54	0.295	0.726	189	0.0027	0.0067	base::import
5.52	0.294	0.294	18101	0.0000	0.0000	Class::Accessor::get
3.45	0.184	1.930	19443	0.0000	0.0001	Class::Accessor::__ANON__
3.13	0.167	0.970	531	0.0003	0.0018	DBIx::ContextualFetch::st::_untaint_execute
3.10	0.165	1.324	1364	0.0001	0.0010	Class::DBI::get
2.98	0.159	0.376	531	0.0003	0.0007	Class::DBI::transform_sql
2.61	0.139	0.139	74	0.0019	0.0019	Cwd::abs_path
2.23	0.119	0.119	8742	0.0000	0.0000	Class::Data::Inheritable::__ANON__
2.06	0.110	0.744	2841	0.0000	0.0003	Class::DBI::__ANON__
1.95	0.104	0.159	2669	0.0000	0.0001	Class::DBI::ColumnGroup::group_cols

Perfil de Ejecución (queries a MySQL)

```
# Profile
# Rank Query ID          Response time      Calls R/Call      Item
# =====
# 1 0x8FFEBD609B778EB2   0.6307 25.2%      67 0.0094  INSERT activity
# 2 0x1C55D6804083DB4C   0.3908 15.6%        7 0.0558  SELECT stats_cv
# 3 0x62EC2BC35CD62D85   0.3519 14.1%        7 0.0503  SELECT stats_cv
# 4 0x32AF9886FDDBAE30   0.1513  6.0%       144 0.0011  SELECT frs_file
# 5 0x1929E67B76DC55E7   0.1330  5.3%         3 0.0443  SELECT frs_dlst
# 6 0x60D6962E42C08882   0.1204  4.8%        67 0.0018  SELECT plugins
# 7 0xF2AF6DF05892D03E   0.1128  4.5%         5 0.0226  SELECT doc_data
# 8 0x64F8E6F00640AF8    0.0672  2.7%         3 0.0224  SELECT users
# 9 0x4636BFC0875521C9    0.0664  2.7%       93 0.0007  SELECT supporte
# 10 0x02CC64324ED7CA2C   0.0651  2.6%        12 0.0054  INSERT frs_dlst
```


Perfil de Ejecución (browser)

Home

Preview

HAR

About 2.0.1 Schema

Show Page Timeline | Hide Statistics | Clear



DNS
SSL/TLS
Connect
Blocked
Send
Wait
Receive



HTML/Text
JavaScript
CSS
Image
Flash
Others



Headers Sent
Bodies Sent
Headers Received
Bodies Received



Downloaded
Partial
From Cache

#1 Cuzillion

Request	Status	Size	Time
GET cuzillion?ex=10	200 OK	2.7 KB	525ms
GET logo - 32x32.gif	200 OK	1 KB	193ms
GET resource.cgi?tyf	200 OK	1.1 KB	1.43s
GET resource.cgi?tyf	200 OK	1.5 KB	1.4s
4 undefined			6.96s (onload: 6.97s)

Digresión...

<http://www.alvago.com.ar/wp-content/uploads/2010/05/desvio-desvio.gif>

"Abstracciones Agujereadas"

- "Leaky Abstractions" - Joel Spolsky, 2002.
- "Towards a New Model of Abstraction in Software Engineering" - Gregor Kiczales, 1992.
- Abstraemos para ocultar detalles de implementación.
 - e.g., TCP/IP, lenguajes de progra, SQL, VM, etc.
- Pero no siempre es posible ni deseable hacerlo.
 - Las abstracciones no son infalibles.
 - Conducen a implementaciones ineficientes.
- Es importante y valioso entender todo el "stack".

5

Jabón en Polvo

Dust Soap

Suavizantes

Softeners

Blanqueador

Launderer

Burro de Planchar

Donkey to Iron

Más Cosas a Mejores Precios

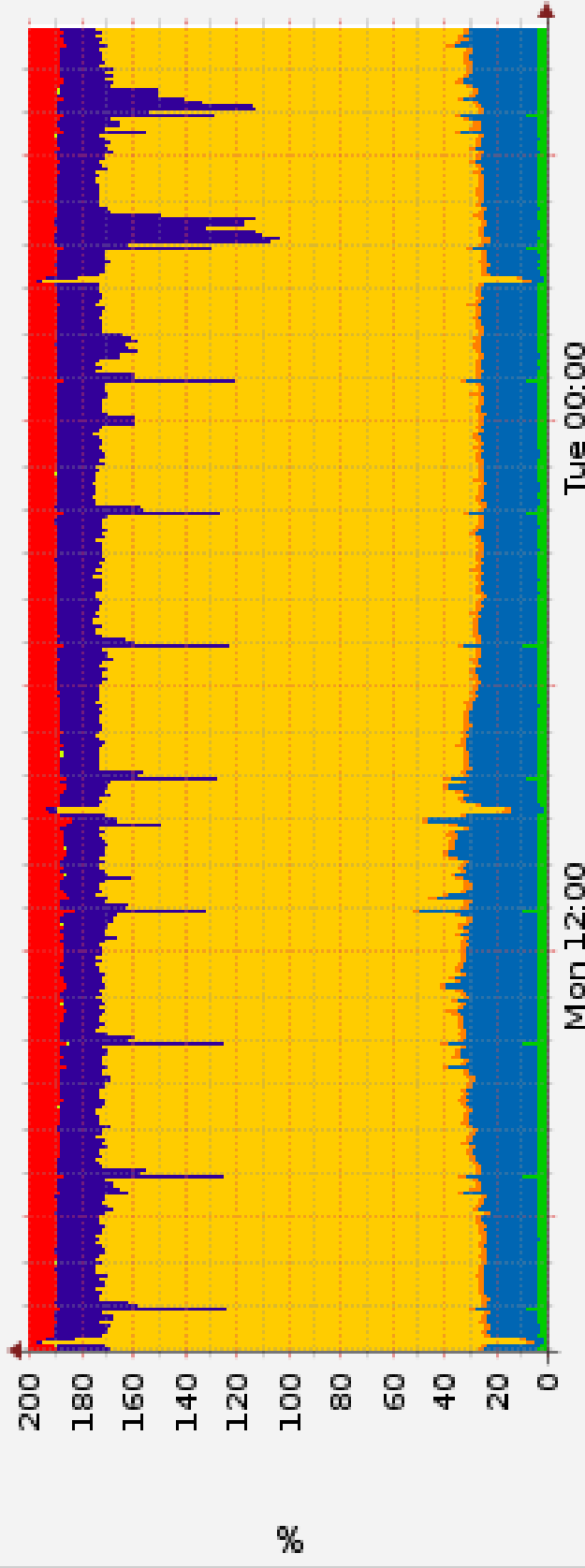
Monitoreo (de Tendencias)

- Gráficas de utilización (%) o contadores.
- Indican operación "normal" versus "anormal".
 - Muestra cambios en *tendencias*.
- Es una agregación de datos, no datos individuales.
 - Importante, pero ¡cuidado!
- ¿Qué debemos graficar?
 - ¡Todo lo que se pueda!

Ejemplo: Utilización de CPU

RRDTOOL / TOBI OETIKER

CPU usage - by day



system	3.31	1.33	3.48	9.35
user	26.58	3.15	23.80	42.21
nice	2.37	2.11	2.57	4.50
idle	142.14	77.52	137.77	186.60
iowait	13.01	1.21	20.58	86.34
irq	0.00	0.00	0.00	0.00
softirq	0.03	0.00	0.04	0.08
steal	12.60	3.47	11.78	17.99

Cur: Min: Avg: Max:

Munin 1.4.4 Last update: Tue Oct 12 08:55:06 2010

Ejemplo: Utilización de Memoria

Ejemplo: Operaciones SQL (día)

Ejemplo: Operaciones SQL (año)



<http://www.english.com/wp-content/uploads/2009/12/freezer-out-of-control.jpg>

Ejemplo: Latencia de Disco

Ejemplo: Utilización de Disco (día)

Ejemplo: Utilización de Disco (año)

Benchmarks y Pruebas de Carga

- Benchmark
 - Programas que realizan pruebas estandarizadas sobre un sistema. (e.g., TPC-C, iometer, etc.)
 - Se usan para comparar el rendimiento de diferentes sistemas.
 - Problema: benchmarks no simulan el mundo real :-(
- Prueba de Carga (*Load Testing*)
 - Someter el sistema a ciertos niveles de carga para medir su rendimiento.
 - Importante: Permite encontrar "La Rodilla".

Planificación de Capacidad

- Determinar cuánto h/w y s/w necesito para dar servicio a cierto nivel de carga.
 - ¿Cuál será mi nivel futuro de carga?
 - ¿Cuál es el rendimiento y desempeño del sistema a ese nivel de carga?
- Problema(s):
 - Estimar* niveles futuros de carga.
 - Medir* el rendimiento & desempeño de un sistema a diferentes niveles de carga.

Planificación de Capacidad

- ¿Podemos extrapolar de capacidad actual?
e.g., $1 \text{ serv} = X \text{ tpm} \quad ?? \quad \Rightarrow \quad 4 \text{ serv} = 4X \text{ tpm}$
- ¡ No necesariamente !
 - Contienda (*Contention*)
 - "Mutex Contention", "Coherency Delay"
 - "Ley" de Amdahl
- Volumen de Datos
- Escalamiento no es siempre lineal.



Planificación de Capacidad

- Contienda (*Contention*)
 - "Mutex Contention" / "Coherency Delay"
 - "Ley" de Amdahl (aplicado a escalabilidad)
- Volumen de Datos
 - Escalamiento no es siempre lineal: $O(n)$, $O(\log n)$
 - Tráfico puede ser lineal, pero...
 - Ciertas operaciones tienen complejidad polinomial: e.g., Redes sociales: $O(n^k)$

Planificación de Capacidad

- Sólo con Pruebas de Carga podemos saber con seguridad.
- No tengo el sistema nuevo... pero tengo La Nube :-)
 - Para tener ambiente de prueba [fidel al real].
 - Para generar Pruebas de Carga.
 - ¡No más excusas !
- ¿ Problema estimando niveles futuros de carga ?
 - Compromiso con la medición, análisis y gestión de utilización del sistema (actual y futuro).

胃肠科	Stomach and intestines section	儿科	Pediatrics	肝病科	Liver disease sec
耳鼻喉科	E.N.T. department	乳腺科	Mammary glands section	皮肤科	Skin section
收费处	Cashier's office	西药房	Pharmacy	妇科	Gynecology
B超室	Super room of B	不孕不育科	The barren does not teach the sec		
泌尿科	Urology	配药室	Fill a prescription the roomr	输液室	Lose the liquid roo
通液室	A liquid room	体外短波室	Short-wave room outside the body		
“离子钛”导融室	“LZT” leads to melt the room	微波室	Microwave room		
计划生育科	Planned parenthood section				
妇产科	The person of the gynecology flows the operating room				
妇科观察室	The gynecology observation room	妇科治疗室	The gynecology treatment roo		
阴道镜检查室	The vagina mirror check room	短波治疗室	The short-wave treatment roo		
海扶刀治疗室	The sea supports the knife treatment room	奥克微治疗室	Tiny treatment room of OK		
微波室	Microwave room	自凝刀治疗室	From congeals the knife treatment room		
手术室	Operating room	外科病区	Surgery disease area	换药室	Change the medicine ro
医生办公室	Doctor's office	护士站	The nurse's station		

Comentarios: Mediciones

- Tendemos a medir lo que es fácil medir, no necesariamente lo que deberíamos medir.
 - Mediciones sucedáneas (*surrogate measurements*) tienden a no servir.
 - Las mediciones por muestreo son imprecisas.
- Lo que importa (para el usuario) es el tiempo de respuesta.

Comentarios: En el mundo real...

- Los sistemas cambian continuamente.
 - Optimización de rendimiento es proceso continuo.
- Es difícil predecir el comportamiento de una aplicación.
 - Hay que programarla con eso en mente.
- No podemos subirle la velocidad, pero sí mejorar el rendimiento.
 - Eliminando instrucciones innecesarias e ineficientes.
- Tratemus de encontrar problemas durante pruebas.
 - Pero sepamos que no los vamos a encontrar todos allí.
- El Rendimiento (*Performance*) es una característica (*feature*) del sistema.

BAÑOS CABALLEROS
BATH HORSEMEN

188065200



0506.