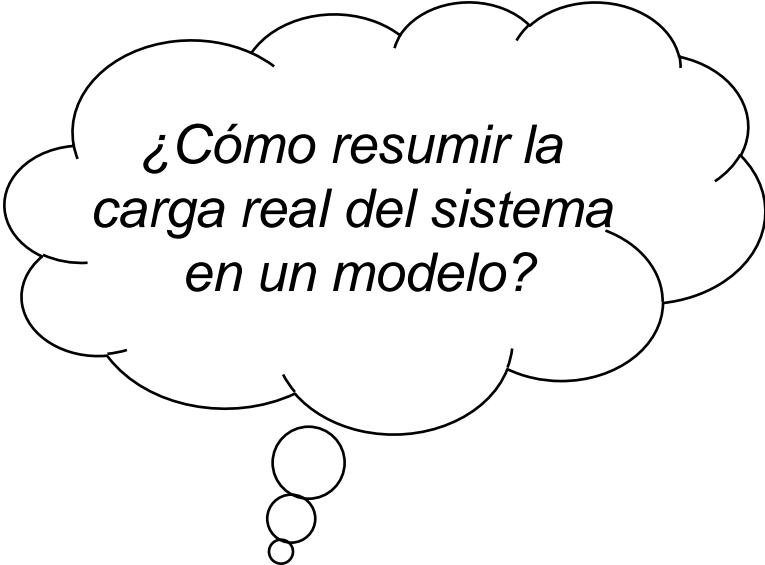


---

# Caracterización de la carga



*¿Cómo resumir la  
carga real del sistema  
en un modelo?*

Administradores de sistemas

# Contenido

## 1. Introducción

Carga de trabajo

Representatividad de la carga

Metodología de caracterización de la carga

## 2. Técnicas de agrupamiento

*Clustering*

Método del árbol de extensión mínima

Agrupamientos cualitativos

## 3. Carga web

CBMG, CVM y CSID



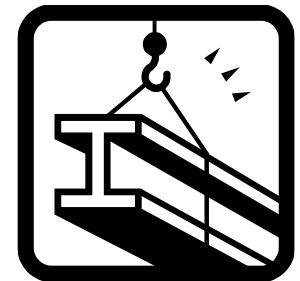
---

# 1. Introducción

Carga de trabajo  
Representatividad de la carga  
Metodología de la caracterización

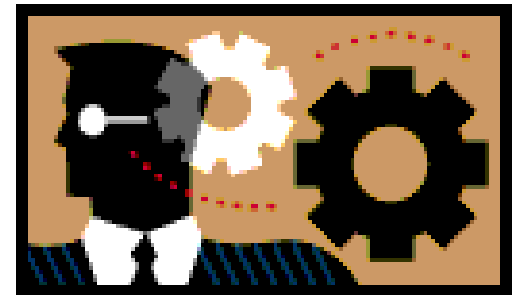
# Carga de trabajo (*workload*)

- ⇒ Conjunto de todos los inputs que el sistema recibe de su entorno durante un cierto periodo de tiempo
- ⇒ Sería deseable que la carga de trabajo fuera repetible
  - De este modo se podrían comparar sistemas diferentes bajo condiciones idénticas.
- ⇒ Desgraciadamente es muy difícil obtener idéntica carga en un entorno real...



# Carga de trabajo: caracterización

- ⇒ En lugar de ello lo que se realiza es un estudio del sistema con dos partes esenciales:
  - Observar las características clave del rendimiento de una carga de trabajo
  - Desarrollar un modelo que puede usarse posteriormente para estudiar la carga
- ⇒ La caracterización de la carga es la construcción de un modelo:
  - El modelo de carga es una representación que imita la carga real bajo estudio



# Componentes y parámetros

- ⇒ Componentes básicos de la carga: unidades genéricas de trabajo que provienen de fuentes externas = son las entidades que realizan peticiones de servicio al sistema.
  - Aplicaciones: e-mail, edición, programación, peticiones HTTP, sesiones de conexión de usuarios, transacciones
  - ... dependen de la naturaleza del servicio que provee el sistema
- ⇒ Parámetros de carga: se usan para modelar o caracterizar la carga
  - Instrucciones, tamaño de paquetes de red, patrones de referencia a páginas
  - ... se deben elegir parámetros dependientes de la carga, no dependientes del sistema.

# Descripciones de carga de trabajo

- ⇒ Existen tres descripciones diferentes de la carga
- Descripción orientada al negocio (usuario)
    - Se describe en términos empresariales como p.e. el número de empleados o clientes, la facturación, etc. Si la carga se describe así, se suelen conocer como unidades naturales o de predicción natural
  - Descripción funcional (software)
    - Constituida por programas, comandos, peticiones, ... que constituyen la carga de trabajo
  - Descripción orientada a los recursos (hardware)
    - Se describe el consumo de recursos del sistema durante la carga, p.e. uso de CPU, operaciones de disco, ocupación de memoria, etc.

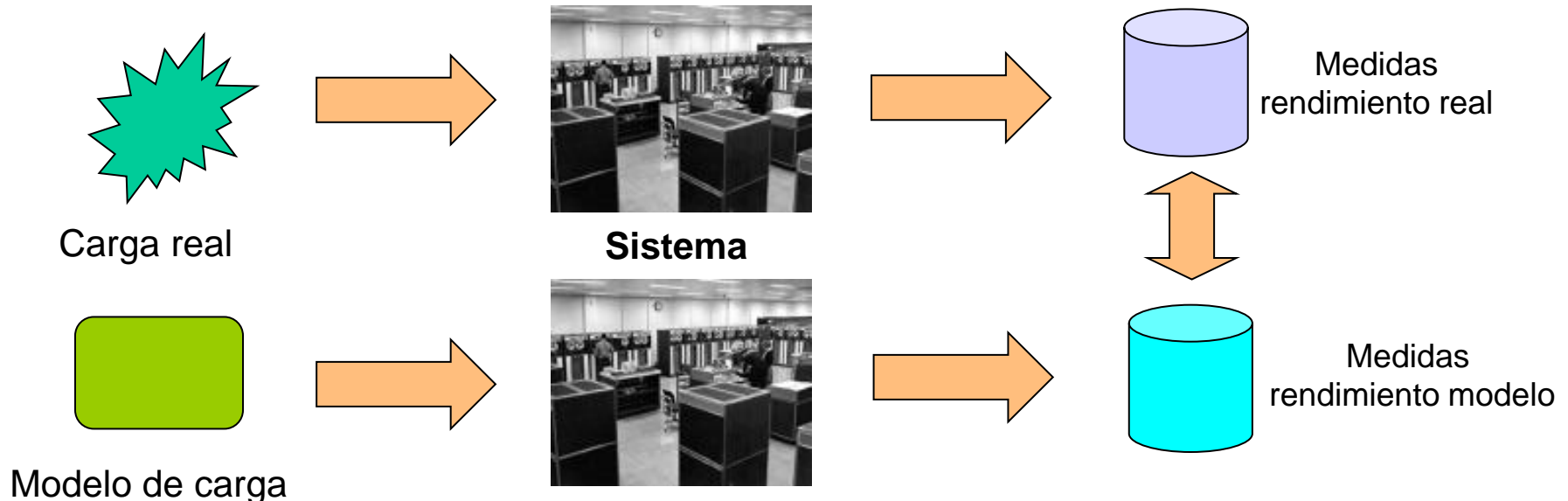
# Algunas afirmaciones y cuestiones

- ⇒ La visión crítica y la reflexión son fundamentales
- «Un servidor web soporta el acceso de 1000 usuarios para descargar documentos pdf, de los que hay 30000 en el servidor»
    - ¿Cuándo se producen las descargas? ¿Afectan los otros tipos de fichero al consumo de recursos?
  - «Se han recibido 250 peticiones HTTP en un minuto»
    - ¿Son todas iguales? ¿Es suficiente la muestra?
  - «50 de las peticiones que se han realizado sobre un mismo documento de 150 KB han tardado una media de un segundo»
    - ¿Son una muestra representativa? ¿La media es un parámetro válido para la carga?
  - «Para caracterizar el sistema tomaremos el par (%CPU, número de operaciones de E/S»
    - ¿Por qué?



# Representatividad del modelo de carga

- ⇒ Los modelos de carga son aproximaciones que representan una abstracción del trabajo que se pretende representar. La representación de la carga:
- Debe de ser lo más fidedigna posible.
  - Es una medida de la similitud entre el modelo y la carga real



# Modelos de carga

- ⇒ Un modelo de carga debe de ser representativo y compacto
- ⇒ Los modelos naturales se construyen usando componentes básicos de la carga real o utilizando trazas de la ejecución de la carga real
- ⇒ Los modelos artificiales o sintéticos no usan componentes básicos de la carga real de trabajo
  - Modelos ejecutables, p.e. *benchmarks*, que son programas que cargan al sistema con un trabajo similar al que quieren reproducir
  - Modelos no ejecutables que describen una serie de valores paramétricos que reproducen el mismo uso del sistema que la carga real

# Metodología de caracterización

- ⇒ Usualmente la caracterización de la carga de un sistema se realiza siguiendo los siguientes pasos:
- Elección del objetivo de estudio de carga
  - Identificación de los componentes básicos de la carga
  - Elección de los parámetros característicos de los componentes
  - Recolección de datos
  - Fraccionamiento la carga de trabajo
  - Cálculo de los parámetros de clase

# Selección de parámetros

- ⇒ Cada componente de la carga se caracteriza por dos grupos de información
- La intensidad de la carga
    - Frecuencia de llegada de trabajos
    - Número de clientes y tiempo de reflexión
    - Número de procesos o trazas de ejecución simultáneas
    - ...
  - Las demandas de los distintos tipos de servicios en los distintos recursos

# Recolección de datos

- ⇒ Se asignan valores a cada componente del modelo de carga
- Identificar las ventanas temporales que definen las sesiones de medida
  - Monitorizar y medir las actividades del sistema durante la ventanas temporales definidas
  - A partir de los datos recogidos, asignar valores a los parámetros de caracterización de cada componente de la carga

# Fraccionamiento de la carga de trabajo

- ⇒ La carga real puede verse como una colección heterogénea de componentes
- ⇒ Las técnicas de fraccionamiento dividen la carga de trabajo en series de clases de tal forma que sus poblaciones contengan componentes homogéneos
- ⇒ Atributos para fraccionar una carga de trabajo
  - Uso de recursos (tiempo de CPU, tiempo de E/S,...)
  - Aplicaciones (los MB transmitidos por www, ftp, telnet,...)
  - Objetos utilizados (porcentaje de acceso a HTML, gif, mpeg, pdf, ...)
  - Situación geográfica de los usuarios
  - Características funcionales, unidades organizacionales, modo de uso...

# Cálculo de los parámetros de clase

- ⇒ Hay diversas técnicas para el cálculo de los valores que representan cada clase
  - Utilización de medias
  - Especificación de la dispersión
  - Histogramas de uno o múltiples parámetros
  - Análisis de componentes principales
  - Modelos markovianos
  - Agrupamiento (*clustering*).
- ⇒ Nos centraremos en la utilización de medias y el agrupamiento o *clustering*

---

## 2. Técnicas de agrupamiento

Clustering

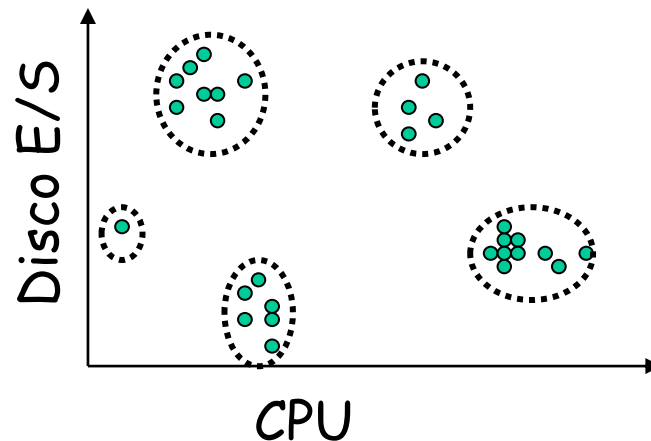
Método del árbol de extensión mínima

Agrupamientos cualitativos



# Clustering

- ⇒ Su aplicabilidad es adecuada cuando se dispone de un gran número de componentes
- Un grupo o *cluster* es aquel cuyos componentes son similares
    - Por tanto se puede trabajar con un representante de cada grupo o clase de componente
  - Ejemplo de 30 trabajos (componentes) y su consumo de CPU y E/S en 5 grupos



# Procedimiento de clustering

1. Tomar una muestra
2. Seleccionar parámetros
3. Transformar parámetros si fuese necesario
4. Eliminar valores extremos
5. Escalar las observaciones
6. Seleccionar una métrica para la distancia
7. Construir los grupos o *clusters*
8. Interpretar los grupos
9. Cambiar la agrupación si fuese necesario y repetir desde 3 a 7
10. Seleccionar los componentes representativos

# Clustering: muestreo

- ⇒ Usualmente la carga se compone de demasiados elementos para su análisis
  - Esa es la razón por la cual se agrupa
- ⇒ Se debe seleccionar un conjunto reducido de grupos
  - Si se elige con acierto los componentes de cada grupo presentan un comportamiento similar
- ⇒ Se podrían escoger los grupos aleatoriamente
  - Sin embargo, cualquier estudio de rendimiento tiene unos objetivos particulares, por lo que se deben escoger aquellos componentes interesantes
    - Por ejemplo, si interesa ver el consumo de disco, se debería elegir la carga de momentos con elevada E/S

# Clustering: selección de parámetros

- ⇒ Muchos componentes tienen un gran número de parámetros (demanda de recursos)
  - Algunos son importantes pero otros no
  - Se deben eliminar aquellos que no son de interés
- ⇒ Los criterios clave son el impacto en el rendimiento y su varianza
  - Si no hay impacto, se deben omitir
  - Si casi no tienen varianza, se deben omitir
- ⇒ Método
  - Rehacer el agrupamiento con un parámetro menos
  - Contar la fracción de cambio en los miembros de los grupos
  - Si no hay mucho cambio, eliminar el parámetro

# Clustering: transformación

- ⇒ Si la distribución de un parámetro está muy sesgada, se debería transformar la medida del parámetro
- Por ejemplo, dos programas que usan la CPU durante 1 y 2 segundos, respectivamente, son igual de diferentes que dos programas que tardan 1 y 2 milisegundos. Sin embargo la diferencia entre ellos no: uno tarda 1 segundo más, en el primer caso y otro 1 milisegundo más en el segundo.
  - Se puede tomar el ratio de CPU en forma logarítmica para mantener esas diferencias y no tomar el valor absoluto del tiempo de CPU

# Clustering: valores extremos

- ⇒ En algunos casos los valores extremos pueden producir efectos no deseados en la agrupación, afectando al máximo o al mínimo, la media o la varianza
  - Su inclusión o exclusión tiene que ser considerada
- ⇒ Solamente se excluirán los extremos si no consumen una parte significativa de los recursos

# Clustering: escalado (1 de 3)

- ⇒ Los resultados finales de agrupación dependen de los rangos relativos
  - Típicamente se trata de escalar para que los rangos relativos sean iguales
  - Hay varios modos de realizar el escalado
- ⇒ Normalizar a cero la media y a uno la varianza
  - Media  $\bar{x}_k$ , desviación estándar  $s_k$  del parámetro  $k$

$$x'_{ik} = \frac{x_{ik} - \bar{x}_k}{s_k}$$

Hacer lo mismo para cada uno de los  $k$  parámetros

# Clustering: escalado (2 de 3)

## ⇒ Pesos

- Asignar pesos basados en la importancia relativa del parámetro

## ⇒ Normalización del rango

- Cambiar de  $[x_{\min,k}, x_{\max,k}]$  a  $[0,1]$

$$x'_{ik} = w_k x_{ik}$$

$$x'_{ik} = \frac{x_{ik} - x_{\min,k}}{x_{\max,k} - x_{\min,k}}$$

## ⇒ Por ejemplo: $x_{i1} \{1, 6, 5, 11\}$

- $1 \rightarrow 0, 11 \rightarrow 1, 6 \rightarrow .5, 4 \rightarrow .4$

## ⇒ Pero es sensible a los extremos



# Clustering: escalado (3 de 3)

⇒ Normalización de percentiles

- Escalar de tal modo que el 95% de los valores caigan entre 0 y 1

$$x'_{ik} = \frac{x_{ik} - x_{2.5,k}}{x_{97.5,k} - x_{2.5,k}}$$

⇒ Es menos sensible a los extremos

# Clustering: distancia o métrica (1 de 2)

⇒ Realiza un mapa de cada componente en un espacio de  $n$  dimensiones y muestra su cercanía

- La distancia euclídea entre dos componentes  $\{x_{i1}, x_{i2}, \dots, x_{in}\}$  y  $\{x_{j1}, x_{j2}, \dots, x_{jn}\}$  se define como

$$d = \left\{ \sum_{k=1}^n (x_{ik} - x_{jk})^2 \right\}^{0.5}$$

- Distancia euclídea ponderada
  - Asignar pesos  $a_k$  para  $n$  parámetros
  - Utilizarla si los valores no están escalados o tienen importancia significativamente diferente

$$d = \sum_{k=1}^n \left\{ a_k (x_{ik} - x_{jk})^2 \right\}^{0.5}$$

# Clustering: distancia o métrica (2 de 2)

## ⇒ Distancia chi-cuadrado

- Se utiliza para distribuciones proporcionadas
- Es necesario usarla normalizada para no influir en las distancias

$$d = \sum_{k=1}^n \left\{ \frac{(x_{ik} - x_{jk})^2}{x_{ik}} \right\}$$

# Clustering: técnicas de agrupamiento

- ⇒ Particiona en grupos cuyos miembros sean lo más similares entre ellos y lo más diferentes a otros grupos
  - Minimizar la varianza intra-grupo
  - Maximizar la varianza inter-grupo
- ⇒ Dos clases de técnicas
  - No jerárquica, empezando con  $k$  grupos; mover componentes hasta que la varianza intra-grupo es mínima
  - Jerárquica
    - Empezar con un grupo, dividirlo hasta  $k$
    - Empezar con  $n$  grupos, combinarlos hasta  $k$ 
      - Ejemplo: árbol de extensión mínima (MST, *minimum spanning tree*)

# Clustering: *minimum spanning tree*

1. Empezar con  $k = n$  clases
2. Para toda clase  $i$ , encontrar el centroide de la clase
3. Para toda clase  $i$  y  $j$ , calcular la matriz de distancia de pares  $(i, j)$  entre centroides de esas clases
4. Encontrar la distancia mínima en la matriz y fusionar las clases entre las cuales esa distancia es mínima
5. Repetir 2 a 4 hasta que todos los componentes pertenezcan a la misma clase

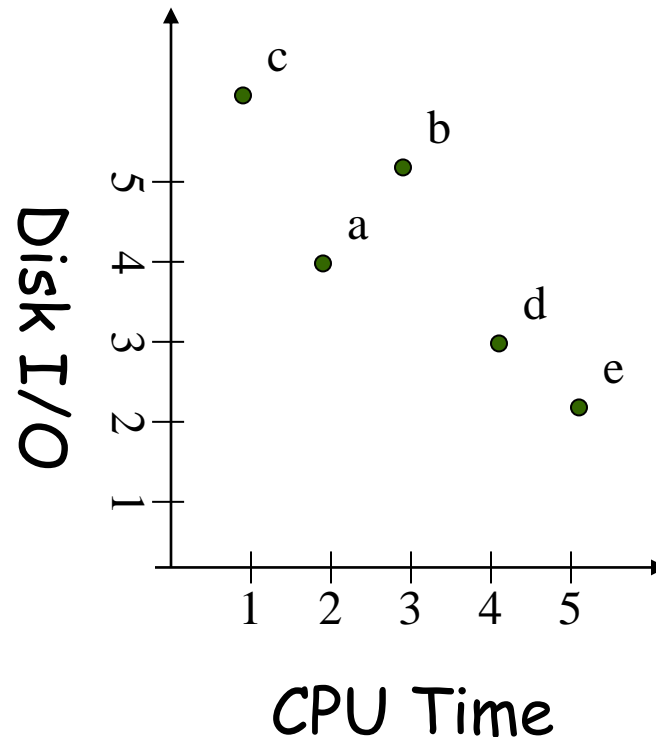
# MST: ejemplo (1 de 5)

⇒ Carga de trabajo con 5 componentes y 2 parámetros

Program	CPU Time	Disk I/O
A	2	4
B	3	5
C	1	6
D	4	3
E	5	2

## MST: ejemplo (2 de 5)

- ⇒ Consideramos que tenemos 5 grupos de un solo miembro por clase
- ⇒ Los centroides son  $\{2,4\}$ ,  $\{3,5\}$ ,  $\{1,6\}$ ,  $\{4,3\}$  y  $\{5,2\}$

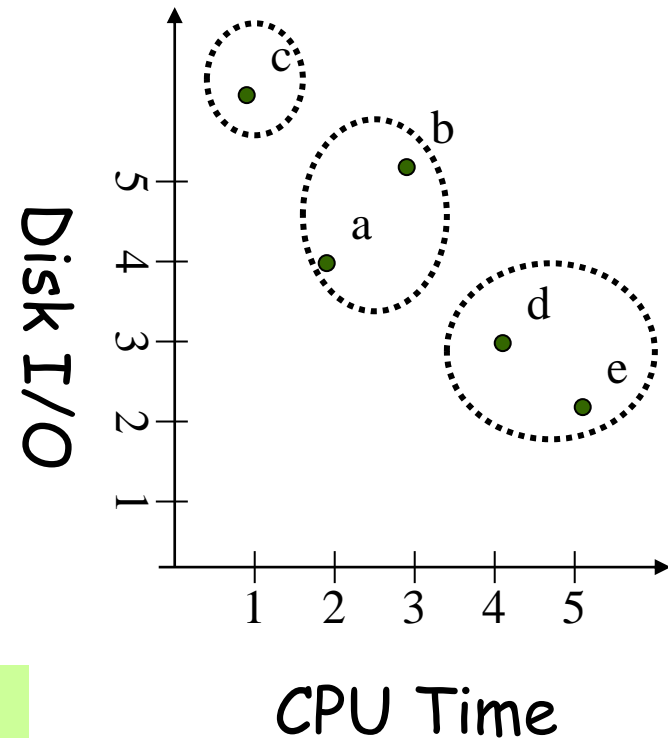


## MST: ejemplo (3 de 5)

⇒ Calculamos la distancia euclídea

Program	Program				
	A	B	C	D	E
A	0	$\sqrt{2}$	$\sqrt{5}$	$\sqrt{5}$	$\sqrt{13}$
B		0	$\sqrt{5}$	$\sqrt{5}$	$\sqrt{13}$
C			0	$\sqrt{18}$	$\sqrt{32}$
D				0	$\sqrt{2}$
E					0

Mínima distancia → fusión de clases

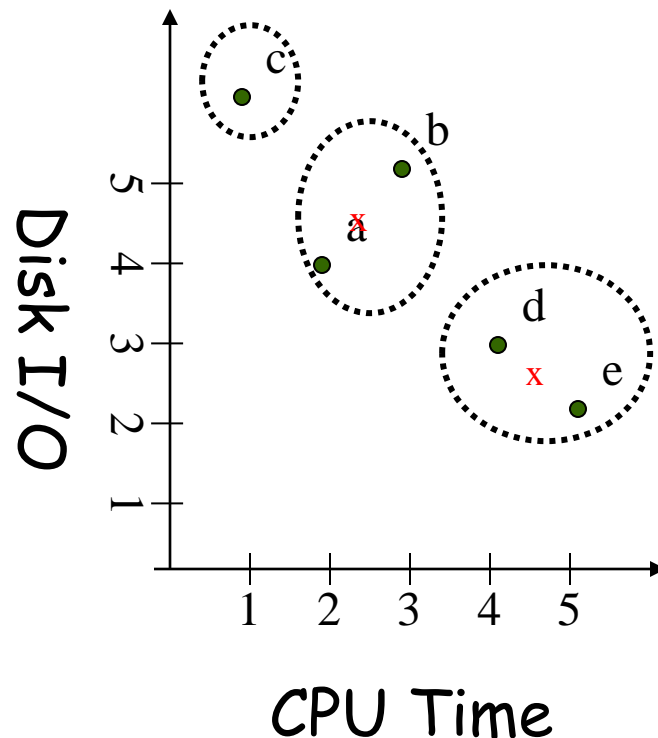




## MST: ejemplo (4 de 5)

⇒ Repetimos el proceso

- El centroide de AB es  $\{(2+3)/2, (4+5)/2\} = \{2.5, 4.5\}$
- $DE = \{4.5, 2.5\}$



	Program		
Program	AB	C	DE
AB	0	$\sqrt{4.5}$	$\sqrt{10.25}$
C		0	$\sqrt{24.4}$
DE			0

Mínimo → fusión

## MST: ejemplo (5 de 5)

⇒ Volvemos a repetir el proceso

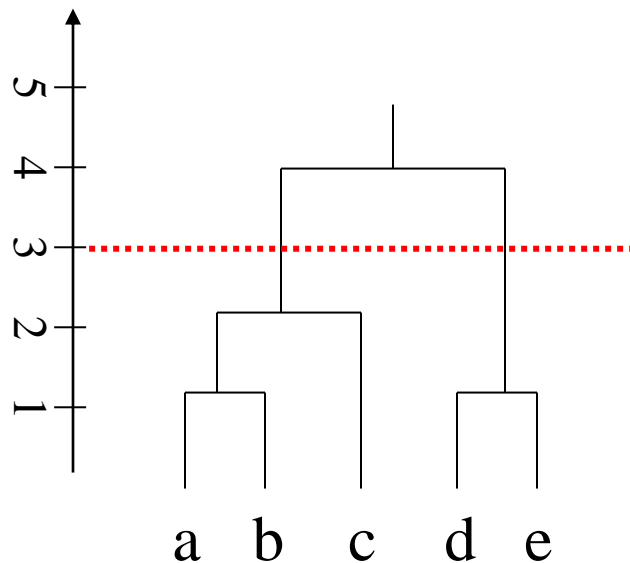
- Centroide ABC  $\{(2+3+1)/3, (4+5+6)/3\} = \{2,5\}$

Program	Program	
	ABC	DE
ABC	0	$\sqrt{12.5}$
DE		0

Mínimo → fusión → fin

# Representación de la agrupación

- ⇒ El árbol de extensión mínima se conoce con el nombre de dendograma
- Cada rama es un grupo o cluster, y su altura finaliza donde se fusiona con otro grupo



Se puede cortar el árbol a cualquier altura, lo que representa un criterio de máxima distancia inter-grupo

# Interpretación de la agrupación

- ⇒ Los grupos pequeños pueden descartarse si usan pocos recursos
  - Sin embargo, un grupo con un solo componente que usa gran parte de los recursos no se podría descartar.
- ⇒ Los grupos se pueden nombrar o etiquetar, por lo que representan, p.e. “consumo de CPU intensivo”
- ⇒ Los grupos que quedan para un determinado criterio caracterizan la carga del sistema y pueden usarse como *benchmark*

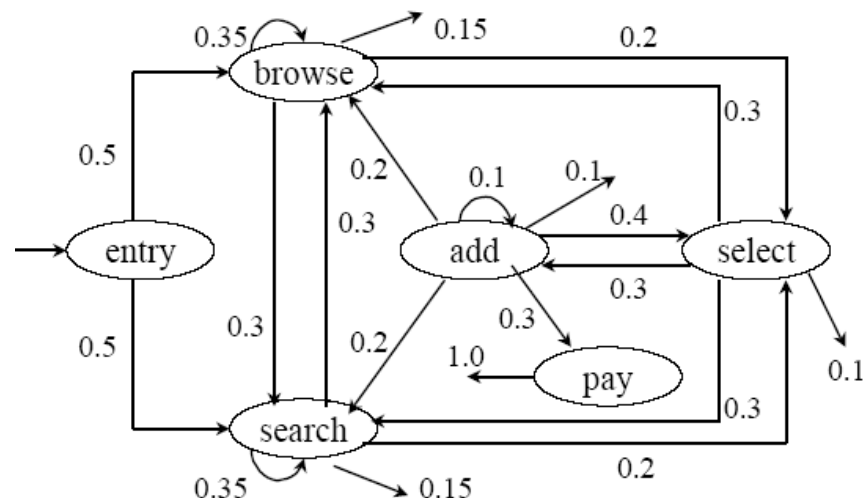
---

## 3. Carga web

CBMG, CVM y CSID

# CBMG: grafo de comportamiento

- ⇒ En el comercio electrónico los clientes interaccionan con los sitios web a través de sesiones, que son secuencias de peticiones interrelacionadas
- ⇒ Las sesiones web pueden ser caracterizadas por los grafos CBMG (*Customer Behaviour Model Graph*)



# CBMG: construcción de modelos

- ⇒ Se pueden diseñar sistemas de ecuaciones que relacionen las visitas a cada uno de los estados del CBMG y las probabilidades de transición entre estados
- ⇒ Las métricas que pueden derivarse de un CBMG son el número de visitas por estado, el ratio de compra y la longitud de sesión por visita

$$V_{entry} = 1$$

$$V_j = \sum_{k=1}^{n-1} V_k \times p_{k,j}$$

# CVM: modelo de visitas

- ⇒ Otra forma de calcular las tasas de visita a partir de un fichero bitácora HTTP es a través del modelo de visitas de cliente, CVM (*Customer Visit Model*)
- ⇒ A partir de un CVM se agrupan las sesiones más representativas y se aplica clustering hasta reducir el número de clases a un conjunto manejable

Sesión	$V_{\text{Muestra}}$	$V_{\text{Búsqueda}}$	$V_{\text{Añade al carro}}$	$V_{\text{Selecciona}}$	$V_{\text{Paga}}$
1	5	12	2	5	1
2	10	15	1	14	0
3	4	7	2	4	1
4	18	20	3	15	0
5	4	12	2	7	1
6	6	11	3	7	1
7	7	12	2	7	1
8	5	4	1	2	1
9	7	10	1	8	1
10	15	20	1	18	0