

Schedules

Índice

► **Introducción**

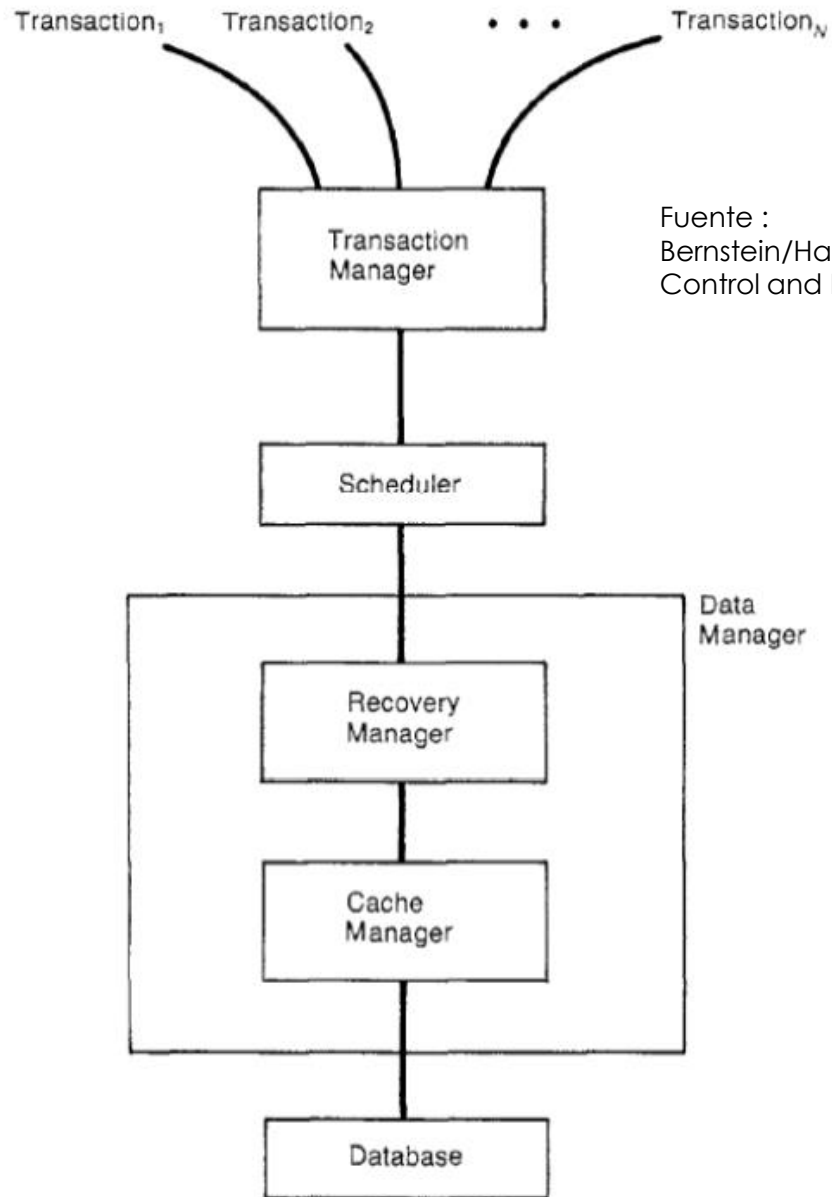
- Principales módulos
- Definición formal de transacción
- Definición de historia o plan
- Historias equivalentes (ejercicio de historias)
- Teorema de seriabilidad, grafo
- Tipos de historias

Introducción

- ▶ Que es una transacción?
- ▶ Principales Propiedades
- ▶ Analicemos el siguiente conjunto de transacciones
 - ▶ $T1 = \{r1(x), r1(y), w1(x), c1\}$
 - ▶ $T2 = \{r2(y), r2(z), w2(z), c2\}$
 - ▶ $T3 = \{r3(z), w3(z), c3\}$
 - ▶ $T4 = \{r4(u), r4(v), r4(x), w4(v), c4\}$
- ▶ Suponiendo que se hubieran iniciado en el orden que indica el subíndice, como convendría resolverlas?

Índice

- ▶ Introducción
- ▶ **Principales módulos**
- ▶ Definición formal de transacción
- ▶ Definición de historia o plan
- ▶ Historias equivalentes (ejercicio de historias)
- ▶ Teorema de seriabilidad, grafo
- ▶ Tipos de historias



Fuente :
Bernstein/Hadzilacos/Goodman - Concurrency
Control and Recovery in Database Systems

FIGURE 1-1
Centralized Database System

Principales Módulos

- ▶ Transaction Manager: cualquier pre proceso para ejecutar las transacciones
- ▶ Scheduler: controla el orden en el que se ejecutan las transacciones
- ▶ Recovery Manager: es el encargado de asegurar la durabilidad y atomicidad de las transacciones.
- ▶ Cache Manager: transferencia desde y hacia el disco

Índice

- ▶ Introducción
- ▶ Principales módulos
- ▶ **Definición formal de transacción**
- ▶ Definición de historia o plan
- ▶ Historias equivalentes (ejercicio de historias)
- ▶ Teorema de seriabilidad, grafo
- ▶ Tipos de historias

Que es formalmente una transacción?

- ▶ Una transacción T_i es un orden parcial con una relación de ordenamiento $<_i$ que satisface las siguientes condiciones:
 - ▶ $T_i \subseteq \{ri(x), wi(x)/x \text{ es un data ítem}\} \cup \{ci, ai\}$
 - ▶ $ai \in T_i \Leftrightarrow ci \notin T_i$
 - ▶ Si $ti \in \{ai, ci\} \Rightarrow$ para todo $pi \in T_i$, $pi <_i ti$
 - ▶ Si $ri(x), wi(x) \in T_i \Rightarrow ri(x) <_i wi(x)$ o $wi(x) <_i ri(x)$

Índice

- ▶ Introducción
- ▶ Principales módulos
- ▶ Definición formal de transacción
- ▶ **Definición de historia o plan**
- ▶ Historias equivalentes (ejercicio de historias)
- ▶ Teorema de seriabilidad, grafo
- ▶ Tipos de historias

Historia o plan

- ▶ Una historia es la forma en que se ejecuta un conjunto de transacciones, con la única restricción de que las operaciones de cada transacción respetan su ordenamiento parcial
- ▶ Dos operaciones entran en conflicto cuando
 - ▶ Pertenecen a transacciones distintas
 - ▶ Operan sobre un mismo data item
 - ▶ Al menos una de ellas es un write
- ▶ Formalmente, dado un conjunto de transacciones $T = \{T1, T2, T3...Tn\}$ una historia completa H sobre T es un orden parcial $<_H$ tal que
 - ▶ $H = U T_i$
 - ▶ $U <_i \subseteq <_H$
 - ▶ Para todo par de operaciones p y q en conflicto, $p <_H q$ o $q <_H p$

Índice

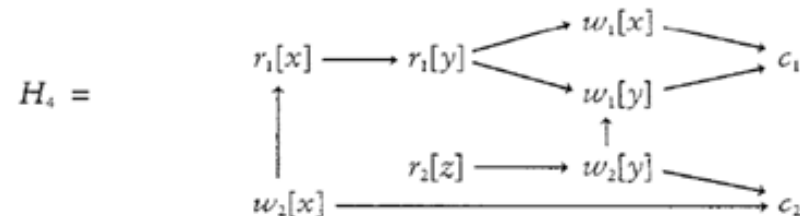
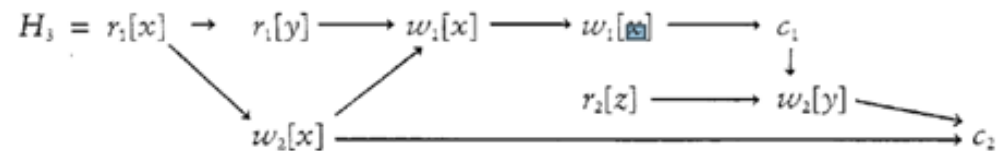
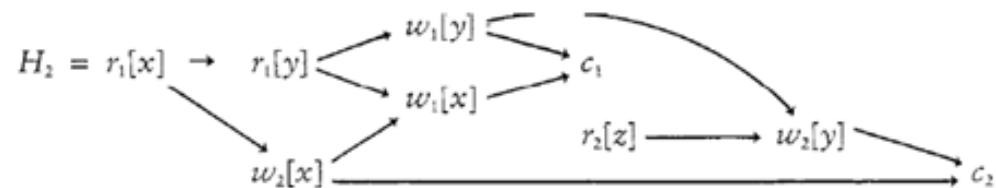
- ▶ Introducción
- ▶ Principales módulos
- ▶ Definición formal de transacción
- ▶ Definición de historia o plan
- ▶ **Historias equivalentes (ejercicio de historias)**
- ▶ Teorema de seriabilidad, grafo
- ▶ Tipos de historias

Historias equivalentes

- ▶ Dos historias H y H' son equivalentes si
 - ▶ Están definidas sobre el mismo conjunto de transacciones y contienen las mismas operaciones
 - ▶ Ordenan las operaciones en conflicto de las transacciones no abortadas en el mismo orden

Historias equivalentes 2

Ejemplos de Historias



Historias equivalentes 3

- ▶ Cuales de las historias del ejemplo son equivalentes?
- ▶ Una historia es **serializable** cuando es equivalente a una serial.
- ▶ Por que les parece que es importante este concepto?

Índice

- ▶ Introducción
- ▶ Principales módulos
- ▶ Definición formal de transacción
- ▶ Definición de historia o plan
- ▶ Historias equivalentes (ejercicio de historias)
- ▶ **Teorema de seriabilidad, grafo**
- ▶ Tipos de historias

Teorema de seriabilidad

- ▶ Dada una historia H para analizar si la misma es serializable o no se construye el grafo de seriabilidad : $SG (H)$
- ▶ Como construir el grafo:
 - ▶ Se agrega un nodo por cada transacción
 - ▶ Se agrega un arco de T_i a T_j , si tienen una operación en conflicto y T_i ocurre antes que T_j
- ▶ Una historia es serializable si $SG(H)$ es aciclico
- ▶ Analicemos el siguiente caso
 $r_2(a), r_1(b), w_2(a), r_3(a), w_1(b), w_3(a), r_2(b), w_2(b)$

Índice

- ▶ Introducción
- ▶ Principales módulos
- ▶ Definición formal de transacción
- ▶ Definición de historia o plan
- ▶ Historias equivalentes (ejercicio de historias)
- ▶ Teorema de seriabilidad, grafo
- ▶ **Tipos de historias**

Tipos de historias

- ▶ Analicemos la siguiente historia
 - ▶ $H1 = \{r1(a), w1(a), r2(a), w2(a), c2, \dots\}$
 - ▶ Que pasa si T1 hace commit y si aborta?
- ▶ Decimos que T_i lee X de T_j si
 - ▶ $w_j(x) < r_i(x)$
 - ▶ $A_j \sim < r_i(x)$
 - ▶ Si existe k tq $w_j(x) < w_k(x) < r_i(x)$, $a_k < r_i(x)$
- ▶ T_i lee de T_j si lee algún x de T_j en H

Tipos de historias 2

- ▶ Una historia se llama **recuperable** (RC) si cuando T_i lee de T_j ($i < j$) en H , si $c_i \in H$, $c_j < c_i$
- ▶ Una historia **evita abort en cascada** (ACA) si cuando T_i lee x de T_j en H , $c_j < r_i(x)$
- ▶ Una historia es **estricta** (ST) si cuando $w_j(x) < o_i(x)$ ($i < j$) o bien $a_j < o_i(x)$ o $c_j < o_i(x)$, $o_i(x) \in \{w_i(x), r_i(x)\}$

Tarea para el hogar

- ▶ Probar que $ST \subseteq ACA \subseteq RC$

Schedules

MUCHAS GRACIAS!