

2. Indica el mecanismo de recuperación de un sistema ante las siguientes situaciones:

- Fallo en (2)
- Fallo en (1) con savepoint en (2)

T_id	Estado	Dato	V. anterior	V. nuevo
T ₁	<i>starts</i>			
T ₁	<i>update</i>	S	60	61
T ₁	<i>commits</i>			
T ₂	<i>starts</i>			
T ₂	<i>update</i>	A	61	62
T ₃	<i>starts</i>			
T ₃	<i>update</i>	B	20	21
T ₄	<i>starts</i>			
T ₃	<i>update</i>	D	40	41
T ₄	<i>update</i>	F	70	71
T ₃	<i>commits</i>			
T ₂	<i>update</i>	E	50	51
(2)				
T ₂	<i>commits</i>			
(1)				
T ₄	<i>update</i>	B	21	22
T ₄	<i>commits</i>			

Si ocurre un fallo de sistema en la línea (2) de la tabla, cuando el sistema gestor se recupere, recorrerá todo el fichero de log.

Aquellas transacciones con un *commit* registrado en el fichero de logs, se volverán a hacer (*redo*). En concreto, se reharán T₁ y T₃.

Aquellas transacciones sin un *commit* registrado en el fichero de logs, se desharán (*undo*). En concreto, se desharán T₂ y T₄.

Sin embargo, si encontramos una entrada *savepoint* en la posición (2) de la tabla, y se produce un fallo cuando toca escribir la posición (1), cuando el sistema se recupere del fallo, actúa de forma ligeramente diferente.

La entrada *savepoint* garantiza que todas las transacciones cuyos *commits* hayan ocurrido antes, han sido guardadas y volcadas a disco, por lo que se ahorra el volver a hacerlas. De ese modo, no sería necesario rehacer (*redo*) T₁ ni T₃, pero sí se reharía T₂, cuyo *commit* aparece detrás del *savepoint* pero antes del fallo.

Dado que, en esta situación, tampoco hay un *commit* para T₄, sería necesario deshacerla (*undo*).