



INICIO GRABACIÓN



SANJOSÉ
FUNDACIÓN DE EDUCACIÓN SUPERIOR

The background features a photograph of several hands of different skin tones holding a small blue ball. This image is overlaid with a large, semi-transparent dark blue circle on the right side and a smaller, semi-transparent light blue circle on the left side, which contains the text.

TOLERANCIA A FALLOS



INTRODUCCION

Una característica donde los sistemas distribuidos difieren de un sistema de una sola máquina es la noción de fallo parcial.

En un sistema distribuido una falla puede afectar la operación de algunos componentes y otros no verse afectados.

En un sistema no distribuido, una falla afecta todos los componentes.

Un objetivo importante en el diseño de Sistemas Distribuidos es construirlos de manera que puedan recuperarse de fallas parciales sin que se afecte.

Siempre que ocurra una falla, el Sistema Distribuido debe continuar operando de modo aceptable mientras se soluciona, es decir, tolerar las fallas.





Conceptos básicos

Ser tolerante a fallas está fuertemente relacionado a los sistemas fiables, y ello involucra lo siguiente:

DISPONIBILIDAD

Propiedad por la cual, una sistema está listo para ser utilizado inmediatamente. Probabilidad de que un sistema esté operando correctamente en cualquier momento dado y disponible a realizar sus funciones.





Conceptos básicos

CONFIABILIDAD

Propiedad por la cual, una sistema es capaz de funcionar de manera continua sin fallar.

A diferencia de la Disponibilidad, esta se define en función de un intervalo de tiempo en lugar de un instante de tiempo.

Este lapso de tiempo es relativamente largo.





Conceptos básicos

SEGURIDAD

Situación en la que no sucede nada catastrófico cuando un sistema deja de funcionar por algún tiempo.



MANTENIMIENTO

Se refiere a cuan fácil puede ser reparado un sistema que fallo.

Un sistema altamente mantenible también puede ser altamente disponible, en especial si las fallas pueden ser detectadas y reparadas automáticamente.

Dicha recuperación automática es difícil de realizar.





LOS FALLOS

Un sistema falla cuando deja de proporcionar sus funcionalidades a sus usuarios.
Un error es una parte del estado del sistema que puede conducir a que dicha funcionalidad deje de proporcionarse.
La causa de un error se llama falla.
Indagar la causa de un error es importante para la solución.





LOS FALLOS



CLASIFICACIÓN DE FALLOS

Existen 03 tipos de fallas:

Transitorias

Falla una vez y luego funciona correctamente.

Si la operación se repite la falla desaparece.

Intermitentes

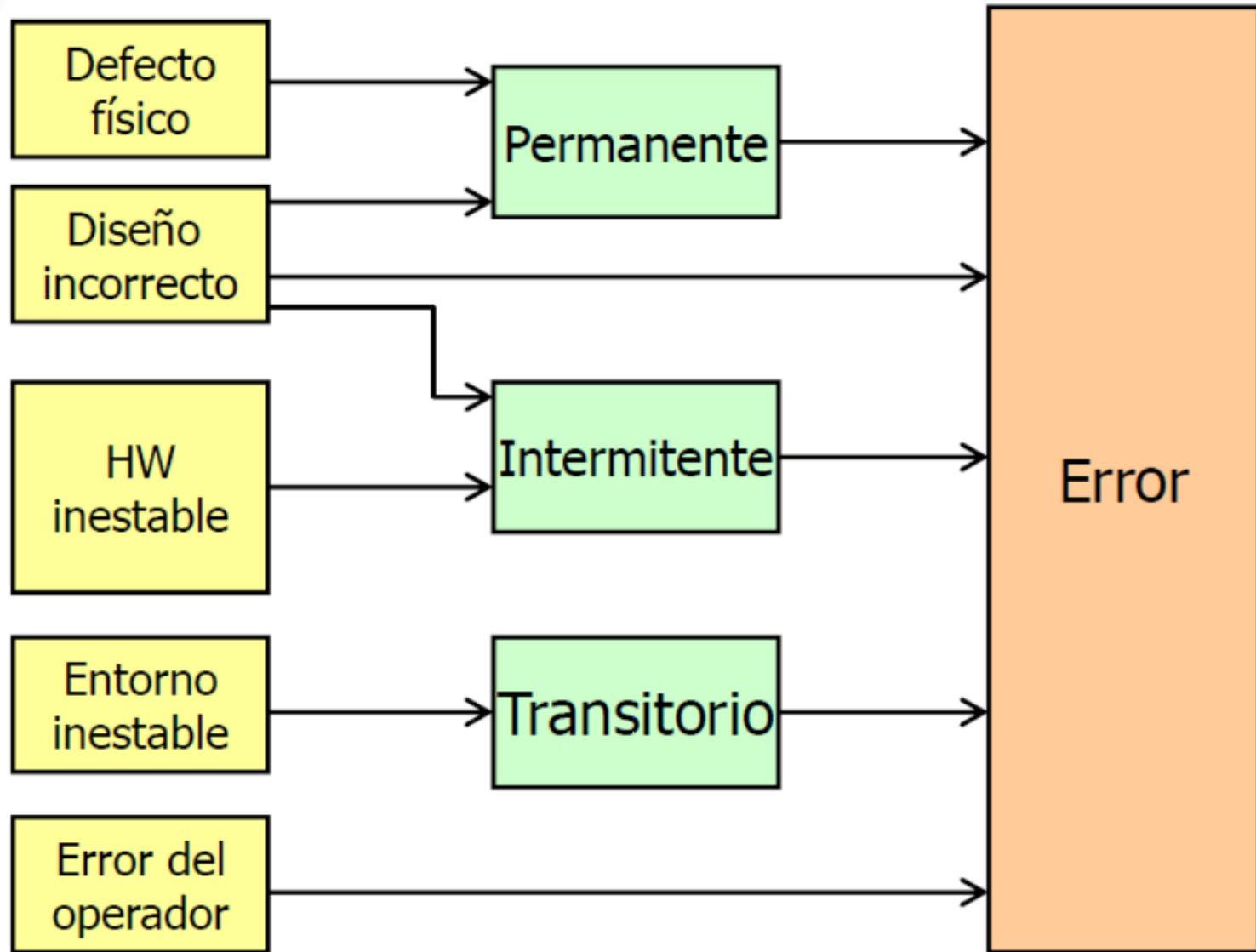
La falla ocurre, luego desaparece por sí sola, después reaparece, y así sucesivamente.

Permanentes

Una vez falla el elemento, ya no se recupera.

La falla continua existiendo hasta que el componente defectuoso es reemplazado.

LOS FALLOS





MODELOS DE FALLA

Para tener una mejor idea de que tan seria en realidad es una falla, se han realizado esquemas de clasificación.

Tipo de falla	Descripción
Falla de congelación	<ul style="list-style-type: none">• Un servidor se detiene pero estaba trabajando correctamente hasta que se detuvo.
Falla de omisión Omisión de recepción Omisión de envío	<ul style="list-style-type: none">• Un servidor no responde a las peticiones entrantes.• Un servidor no recibe los mensajes entrantes.• Un servidor no envía mensajes.
Falla de tiempo	<ul style="list-style-type: none">• La respuesta de un servidor queda afuera del intervalo de tiempo especificado
Falla de respuesta Falla de valor Falla de transición de estado	<ul style="list-style-type: none">• La respuesta de un servidor es incorrecta.• El valor de la respuesta esta equivocado.• El servidor se desvía del flujo de control correcto.
Falla arbitraria	<ul style="list-style-type: none">• El servidor puede producir respuestas arbitrarias en tiempos arbitrarios.



MODELOS DE FALLA



Falla de congelación

Ocurre cuando un servidor se detiene prematuramente, el cual estuvo funcionando

correctamente hasta antes de la falla.

Una característica de la falla de congelación es que al detenerse el servidor, ya no responde.

Falla de omisión

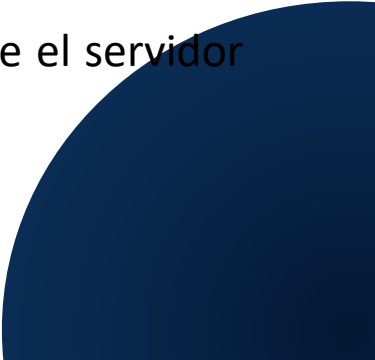
Ocurre cuando un servidor no responde una petición.

Puede darse por las siguientes razones:

Falla de recepción: El servidor nunca obtuvo la petición

Falla de emisión: Cuando el servidor hizo correctamente su trabajo pero algo salió mal al enviar una respuesta.

Mala gestión de memoria: Por la mala gestión de memoria por la que el servidor se "colgó".





LOS FALLOS

DISFRAZADO DE FALLAS POR REDUNDANCIA

Para que un sistema sea tolerante a fallas, lo mejor que puede hacer es tratar de ocultar ante otros procesos la ocurrencia de fallas, para ello usamos la REDUNDANCIA.

Tres clases de redundancia son posibles:

Redundancia de información

Se agregan bits adicionales para recuperar los bits mutilados (perdidos por el ruido durante la transmisión).

Redundancia de tiempo

Se realiza una acción, luego si es necesario, se vuelve a realizar.

Redundancia física

Se agregan equipos o procesos para que tolere la pérdida o mal funcionamiento de algunos componentes.





ATENUACIÓN DE UN PROCESO




TEMAS DE DISEÑO

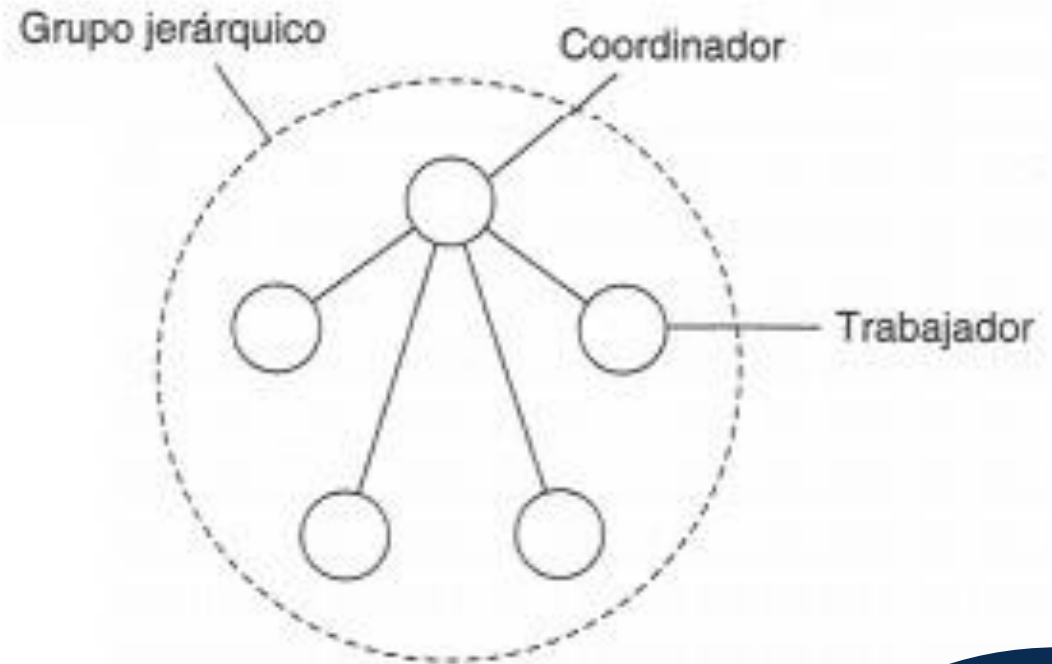
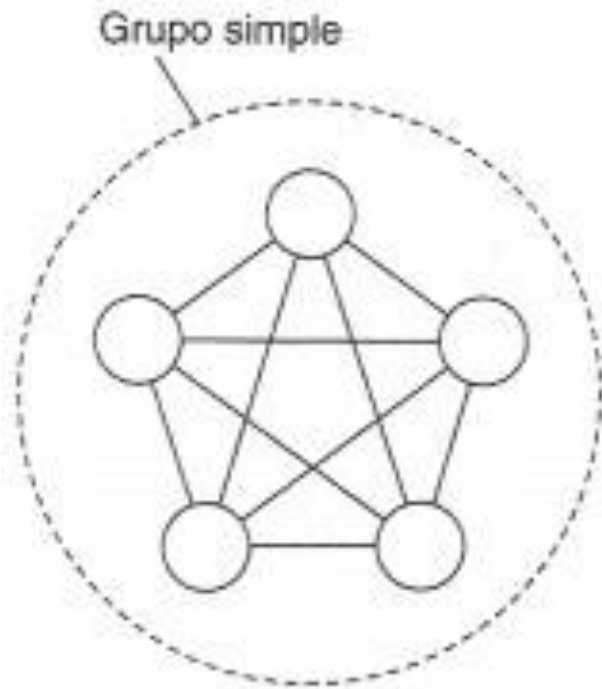
La clave para afrontar la tolerancia a un proceso defectuoso es con la replicación de proceso en grupos.

- Estos grupos pueden ser dinámicos en su gestión.
- Se pueden crear nuevos grupos y destruir los viejos.

Un proceso puede ser miembro de varios grupos, por ellos se requieren mecanismos para una adecuada gestión de los grupos y la membresía a un grupo.



ATENUACIÓN DE UN PROCESO





ATENUACIÓN DE UN PROCESO




Existen dos tipos de grupos:

Grupo simple

Si uno de sus nodos se congela, el grupo se vuelve más pequeño pero continua. La toma de decisiones resulta más complicada ya que deben participar todos.

Grupo jerárquico

Aquí la pérdida del coordinador hace que todo el grupo se detenga bruscamente, pero mientras esta funcionando puede tomar decisiones sin molestar a los demás.





MEMBRESÍA DE UN GRUPO



La comunicación en grupo requiere cierto método para:

- Creación y eliminación de grupos.
- Unión y separación de procesos a grupos.

Existen dos métodos posibles:

Tener un servidor de grupos al cual enviar todas las solicitudes

EL servidor puede mantener una base de datos completa de todos los grupos y su membresía exacta.

Es un método directo, eficiente y fácil de implementar.

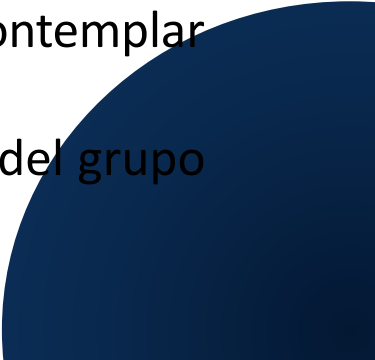
La desventaja es el punto de fallo que representa la administración centralizada de los grupos.

La administración distribuida de las membresías a grupos

En un grupo abierto, un proceso extraño puede enviar un mensaje a los integrantes del grupo para anunciar su presencia.

En un grupo cerrado se precisa algo similar, ya que se debe contemplar la admisión de nuevos miembros al grupo cerrado.

Al salir de un grupo, el proceso debe comunicarlo a los demás del grupo que deja.





MEMBRESÍA DE UN GRUPO



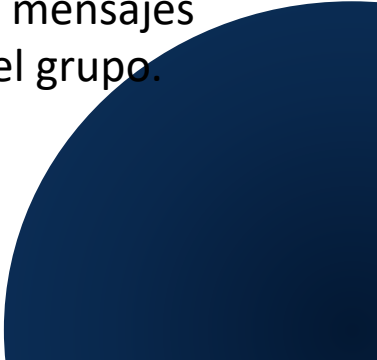
Un aspecto problemático se presenta cuando un miembro falla, saliendo por lo tanto del grupo:

- No hay un anuncio apropiado de este hecho.
- Los demás miembros del grupo lo deben descubrir de forma experimental; luego se lo puede eliminar del grupo.

Otro aspecto importante es que la entrada y salida al grupo debe sincronizarse con el envío de mensajes:

- Un proceso que se unió a un grupo debe recibir todos los mensajes que se envíen al mismo.
- Un proceso que ha salido de un grupo:
 - No debe recibir más mensajes del grupo.
 - El grupo no debe recibir más mensajes del proceso.
 - Los otros miembros no deben recibir más mensajes del proceso saliente.

Una forma de garantizar que una entrada o salida se integra al flujo de mensajes en el lugar correcto es convertir esta operación en un mensaje a todo el grupo.





ENMASCARAMIENTO DE FALLAS Y REPLICACIÓN




Los grupos de procesos son parte de la solución.

Tener un grupo de procesos idénticos permite disfrazar uno o más procesos defectuosos presentes en dicho grupo.

Es decir, podemos replicar los procesos y organizarlos en grupo para reemplazar uno o varios procesos (tolerante a fallas).

Un punto importante en éste tema con la utilización de grupos de procesos para tolerar fallas es cuanta replicación se requiere.






ACUERDO EN SISTEMAS DEFECTUOSOS



El tener procesos replicados en un grupo incrementa la tolerancia de fallos. Las cosas se complican cuando un grupo tiene que llegar a un acuerdo. El objetivo general de los algoritmos de acuerdo distribuidos es hacer que los procesos no defectuosos alcancen un consenso en algún tema. El problema se complica por el hecho de que diferentes suposiciones sobre el sistema requieren diferentes soluciones, para ello se distinguen los siguientes casos:

- Sistemas síncronos contra sistemas asíncronos
 - El retraso de la comunicación está o no limitado
 - La entrega del mensaje se hace o no en orden
 - Los mensajes se transmiten unitransmisión o multitransmisión
- 

ACUERDO EN SISTEMAS DEFECTUOSOS



Solo es posible llegar a un acuerdo en la siguiente situación:

Comportamiento de un proceso

Ordenamiento de los mensajes

No ordenados

Ordenados

Síncrono

Asíncrono



DETECCIÓN DE FALLAS

Para enmascarar fallas, es necesario detectarlas antes. Existen solo dos mecanismos.

- Los procesos constantemente se envían mensajes del tipo ¿estas activo? entre si y esperan respuesta.
- Esperan pasivamente hasta que lleguen mensajes de los diferentes procesos. Esto es posible si se puede garantizar que existe suficiente comunicación entre los procesos.

La detección de fallas también puede ocurrir como un efecto colateral del intercambio irregular de información con los vecinos, igual que en el caso de la diseminación de información basada en gossip.

Es importante distinguir las fallas de red de las de nodos.





RECUPERACIÓN

La recuperación de errores es fundamental para la tolerancia a fallas ya que este puede conducir a una falla.

Existen dos formas de recuperarse de un error:

Recuperación hacia atrás.

Lo principal es hacer que el sistema regrese de su estado actual erróneo a su estado previamente correcto.

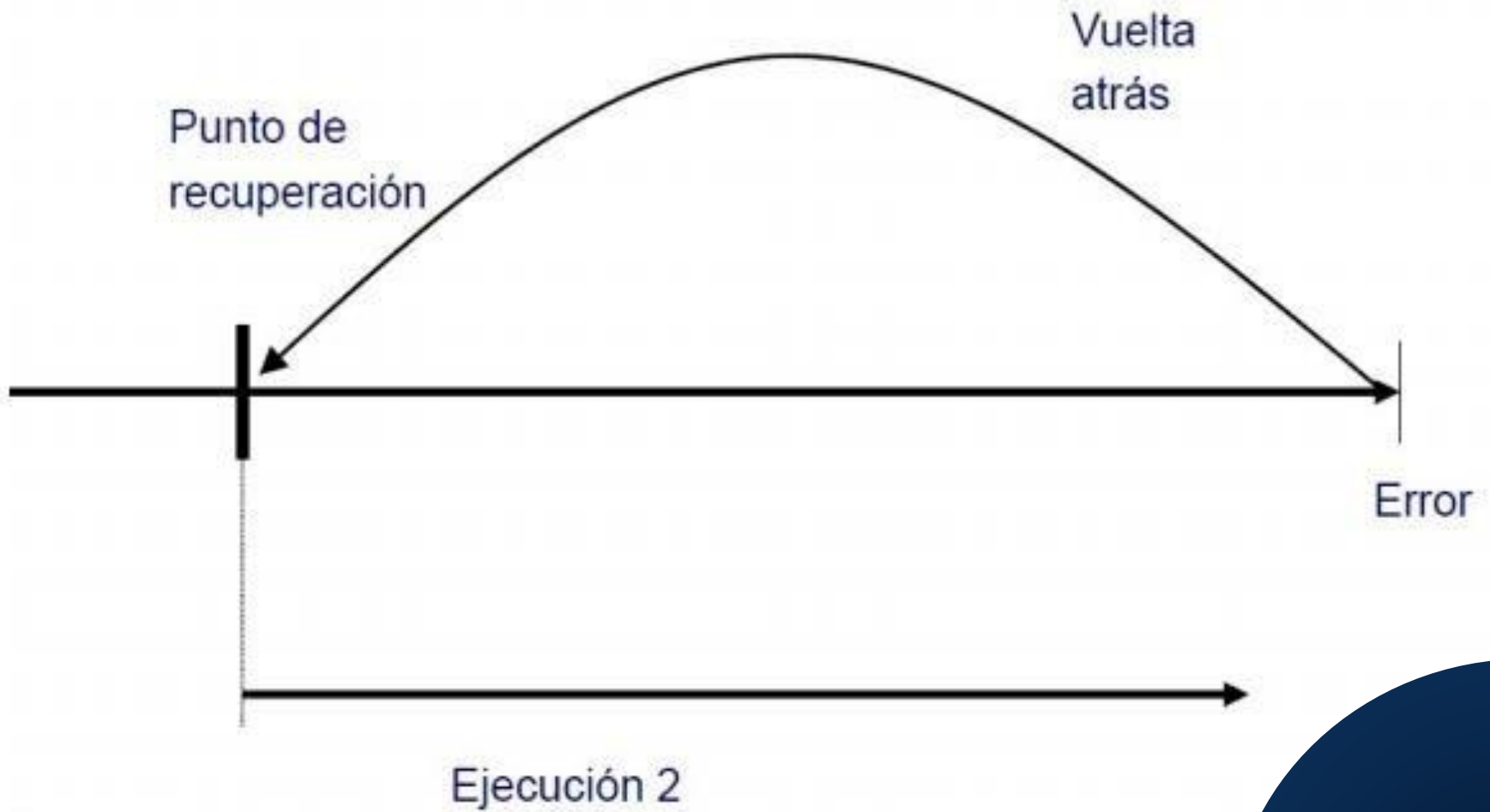
Para ello, será necesario registrar el estado del sistema de vez en cuando; y cuando se genere un error, restaurar el estado registrado.

Cada vez que se registra un estado se dice que se marca un punto de control.

Su beneficio principal es que se trata de un método aplicable e independiente de cualquier sistema o proceso específico.

Entre sus desventajas tenemos que es que es costoso en cuanto a desempeño, no hay garantía de que el error no se repita y en algunos estados, es imposible volver hacia atrás.

RECUPERACIÓN HACIA ATRÁS





RECUPERACIÓN

En este caso, cuando el sistema ha entrado en estado erróneo, en lugar de regresarlo a un estado de punto de control previo, se intenta llevarlo a un nuevo estado correcto a partir del cual se pueda continuar ejecutando.

El principal problema es que se tiene que saber de antemano que errores pueden ocurrir, solo en ese caso es posible corregir los errores y trasladarse a un nuevo estado.

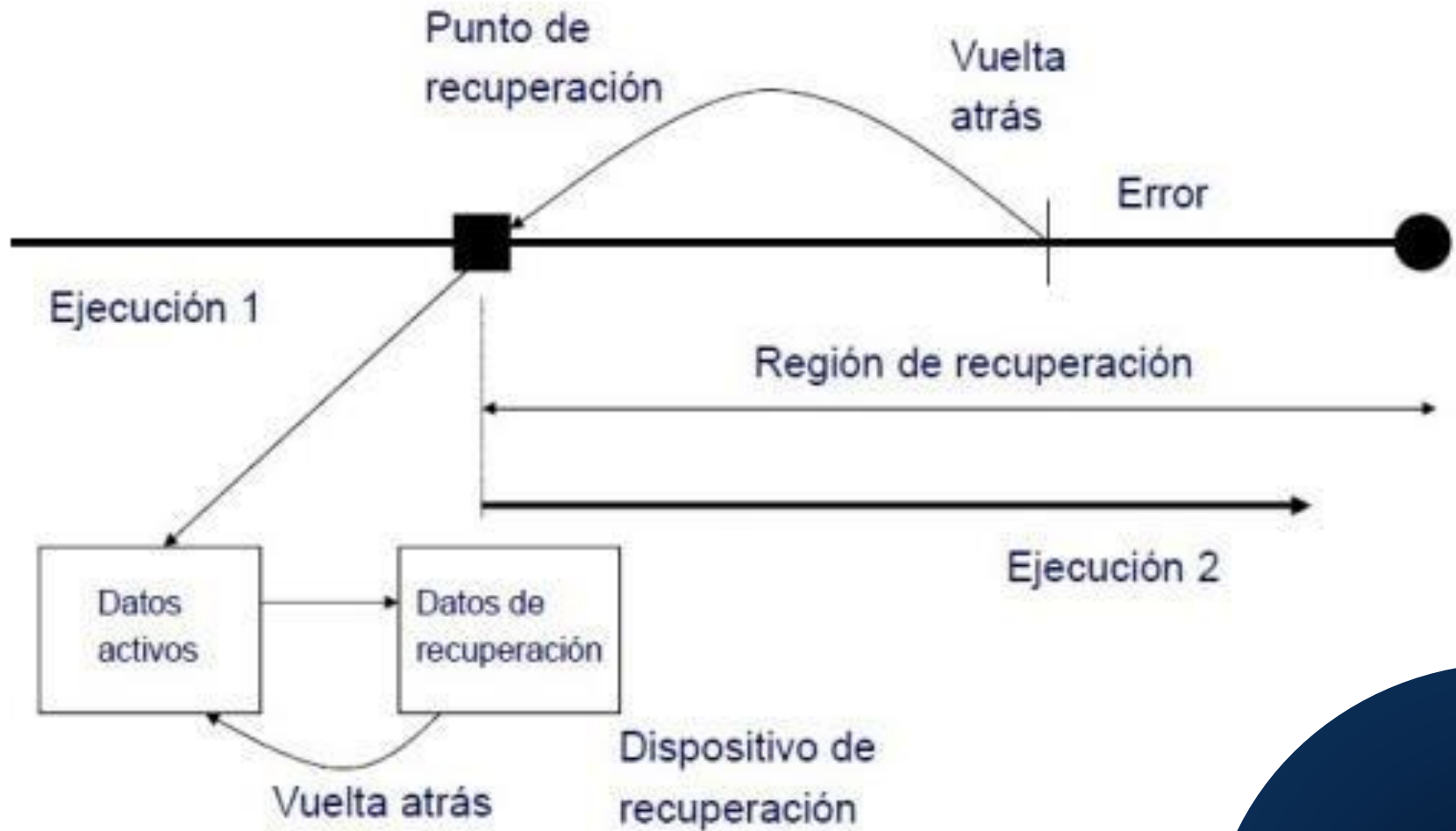
ALMACENAMIENTO ESTABLE

Desempeña un rol muy importante cuando se trata de recuperación de sistemas distribuidos.

Es necesaria que la información requerida para habilitar la recuperación sea guardada con seguridad.



RECUPERACIÓN HACIA ADELANTE





FUNDACIÓN DE EDUCACIÓN SUPERIOR

SAN JOSÉ

INSTITUCIÓN TECNOLÓGICA

FIN DE
GRABACIÓN