SURETE DE FONCTIONNEMENT ET REPRISE APRES PANNE

INTRODUCTION

Un autre rôle des SGBD

- Assurer la cohérence de la BD en dépit des pannes matérielles et logicielles par un principe de récupération des données.

Les mécanismes de reprise sur panne ont le double objectif de faire respecter les propriétés d'atomicité et de durabilité des transactions.

CLASSIFICATION DES PANNES

Un SGBD doit être capable de faire face à quatre types de pannes

1. **Panne d'action**
2. **Panne de transaction**
3. **Panne système**
4. **Panne disque**

Types de Pannes

- **Panne d'une action**

Intervient lorsqu'une commande au SGBD est mal exécutée ce qui provoque la génération d'un code erreur.

Généralement, il y a correction de l'erreur et continuation de la transaction.

TYPES DE PANNES

Panne de transaction

- correspond à l'interruption d'une transaction en cours d'exécution.

Elle peut se produire dans trois cas

1. sur décision de l'utilisateur par le biais de l'ordre **Abort**
2. sur décision de l'organe de contrôle de concurrence (verrou mortel, mauvais ordonnancement des accès concurrents, panne d'action non corrigible)
3. suite à une erreur du logiciel d'application

PANNE SYSTÈME

Correspond à une interruption anormale du SGBD suite à

- une erreur logiciel
- une coupure de courant

Elle se traduit par la perte des données contenues dans le cas du mécanisme de mémoire virtuelle (Mémoire centrale).

La seule action à entreprendre consiste à déclarer les transactions en cours d'exécution lors de la panne comme abandonnées.

La base de données sur disque est dans un état cohérent correspondant à celui produit par la dernière transaction validée.

PANNE DISQUE

Correspond à la détérioration de l'espace disque contenant la base de données

Elle se traduit par la perte de tout ou partie de la base de données

Il est nécessaire de restaurer la base de données dans l'état cohérent correspondant à celui dans lequel elle était lors de la dernière validation de transaction.

Cette action peut consister à refaire toutes les transactions validées depuis la création de la base de données.

Certains systèmes permettent de partitionner l'espace disque contenant la base de données, de façon à pouvoir ne reconstituer que la partition endommagée en cas de panne disque

PROCESSUS DE REPRISE APRÈS PANNE

L'objectif principal de la résistance aux pannes est de minimiser le travail perdu en exécutant un processus de restauration de la base de données basé sur le concept de transaction.

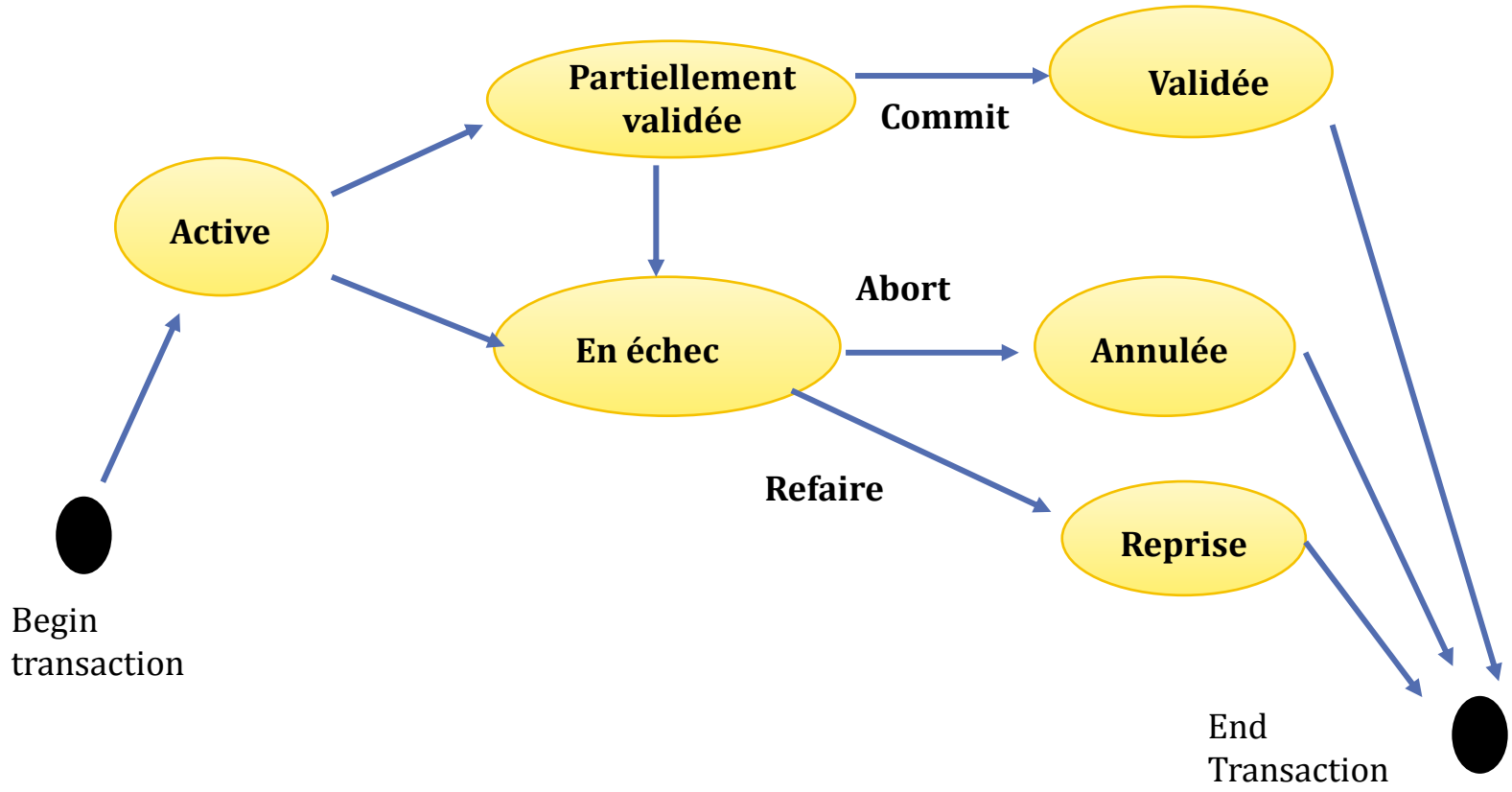
Il permet aussi de fournir un protocole permettant de : Faire, Défaire et Reprendre une transaction en se basant sur trois actions

Commit (validation) : rend effectives toutes les mises à jour de la transaction

Abort (annulation) : annule toutes les mises à jour d'une transaction

Refaire : reprise d'une transaction validée.

ETATS D'UNE TRANSACTION



CLASSIFICATION DES MÉCANISMES DE REPRISE

Les mécanismes de reprises dépendent de la méthode de propagation des mises à jour dans la base de données : Immédiates ou Différées.

Mises à jour immédiates

- Mises à jour intégrées directement dans la base de données au fil de l'exécution des transactions

Mises à jour différées

- Mises à jour effectuées dans un espace séparé, propre à chaque transaction, puis sont intégrées de façon atomique à la base de données lorsque la transaction est validée.
- Il existe deux techniques de mises à jour différées.
  - Technique des pages ombres
  - Technique des fichiers différentiels.

TECHNIQUE DES PAGES OMBRES

Dupliquer chaque page mise à jour, de façon à garder une version cohérente de la base de données.

L'ensemble des pages modifiées par une transaction constitue un espace de travail qui lui est propre.

Chaque transaction possède une table de correspondance lui permettant d'accéder aux pages faisant partie de son espace de travail, à la place des pages ombres constituant une version antérieure de la base de données.

Lorsqu'une transaction est validée, l'ensemble des pages modifiées est intégré de façon atomique à la base de données et les pages ombres correspondantes sont recyclées.

TECHNIQUE DES FICHIERS DIFFÉRENTIELS

Décomposer chaque mise à jour d'instance en une suppression de l'instance suivie d'une insertion de l'instance avec sa nouvelle valeur.

Constitution de deux fichiers différentiels pour chaque transaction T

- **Fichier des instances supprimées $DF^{-}(T)$**
- **Fichier des instances insérées $DF^{+}(T)$.**
- L'espace de travail d'une transaction T est obtenu par :

  $(B - DF^{-}(T)) \cup DF^{+}(T)$. *(B la version cohérente de la base de données à un instant donné*)

Lorsqu'une transaction est validée, les mises à jour contenues dans ces deux fichiers différentiels sont intégrées à la base de données.

L'inconvénient majeur de cette technique est lié au surcoût introduit par la consultation des fichiers différentiels lors de chaque accès à la base de données.

RÉCUPÉRATION DE BASE DE DONNÉES

La récupération est un processus de restauration de la base de données à un état correct dans l'éventualité d'une défaillance.

La transaction est l'unité de récupération pour le SGBD

Problématique

En cas de mise à jour, les données de BD sont transférées à partir de zones tampons de la MC vers les mémoires secondaires. Ce n'est que quand les tampons sont vidés que les mises à jour sont considérées comme permanentes.

Si une défaillance se produit entre l'écriture dans les tampons et le vidage des tampons, le gestionnaire de récupération doit déterminer l'état de la transaction ayant effectuée l'écriture au moment de la défaillance.

RÉCUPÉRATION DE BASE DE DONNÉES

- REDO

Si la transaction a effectué sa validation, alors pour assurer sa durabilité, le gestionnaire de récupération doit refaire (REDO) les modifications de cette transaction

- UNDO

Si la transaction n'est pas validée au moment de la défaillance, le gestionnaire de récupération doit défaire (UNDO) tous les effets de cette transaction.

UTILITAIRES DE RÉCUPÉRATION

1. **Mécanisme de sauvegarde**

   - S'occupe des sauvegardes périodiques de copies de la BD

2. **Outil de journalisation**

   - Conserve la trace de l'état courant des transactions et des différentes mises à jour de la BD

3. **Utilitaire de points de contrôle (Checkpoint)**

   - Permet de rendre permanentes des modifications de la BD

4. **Gestionnaire de récupération**

   - Permet de restaurer la BD dans un état cohérent

MÉCANISME DE SAUVEGARDE

Le SGBD doit proposer un mécanisme qui permet de sauvegarder les copies de la BD à des intervalles réguliers.

Les n copies (miroirs) du disque sont générées comme suit :

1. **Tout enregistrement est fait en *n* copies sur des disques indépendants**

   - par le SGBD pour créer la copie primaire
   - par le SGF pour propager l'enregistrement sur les copies.
   - Chaque copie est sauvegardée sur un disque indépendant

2. ***Les n* copies permettent à la BD de survivre sans perte à toute panne simultanée de (*n* - 1) volumes**

MÉCANISME DE SAUVEGARDE

Si une panne arrive à un volume, alors le système lit une autre copie de l'enregistrement. Le gestionnaire de reprise recrée alors le volume en panne sur un autre disque.

Avantages

- La probabilité de panne totale décroît
- Possibilité de parallélisme ce qui réduit le coût d'exécution des transactions

Inconvénients

- Cette solution engendre un coût de stockage et de mise à jour
- *n* fois plus d'espace mémoire
- *n* fois plus d'accès en MAJ ce qui allonge le temps des transactions et peut générer des incohérences temporaires entre les copies.

TECHNIQUE DE SAUVEGARDE RAID

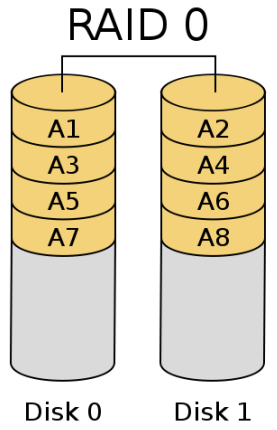
Redundant Array of Independent Disks

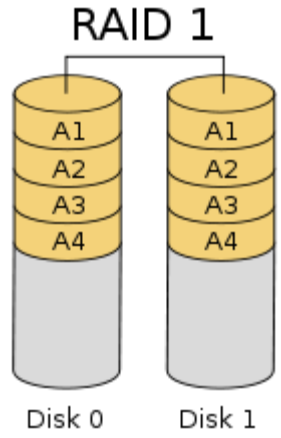
Stocker les données sur de multiples disques durs de petites tailles

Plusieurs niveaux de sauvegarde ont été définis : RAID 0, RAID 1.

RAID 0 (*entrelacement de disques*) consiste à répartir les données (sans redondance) sur plusieurs disques qui peuvent être exploités en parallèle.

RAID 1 (Disques en miroir) consiste à créer plusieurs copies du disque contenant les mêmes données





FICHIERS DE JOURNALISATION

Pour garder la trace de toutes les exécutions des transactions de la BD, le SGBD entretient un fichier Journal ou LOG.

Un journal est une séquence d'enregistrements décrivant les mises à jour effectuées par les transactions. Il représente un historique d'une exécution sur fichier séquentiel.

Journal physique

- Garde la trace des modifications au niveau octet à l'intérieur des pages.

Exemple : <Ti, Numéro Page, donnée, Image avant, Image après>

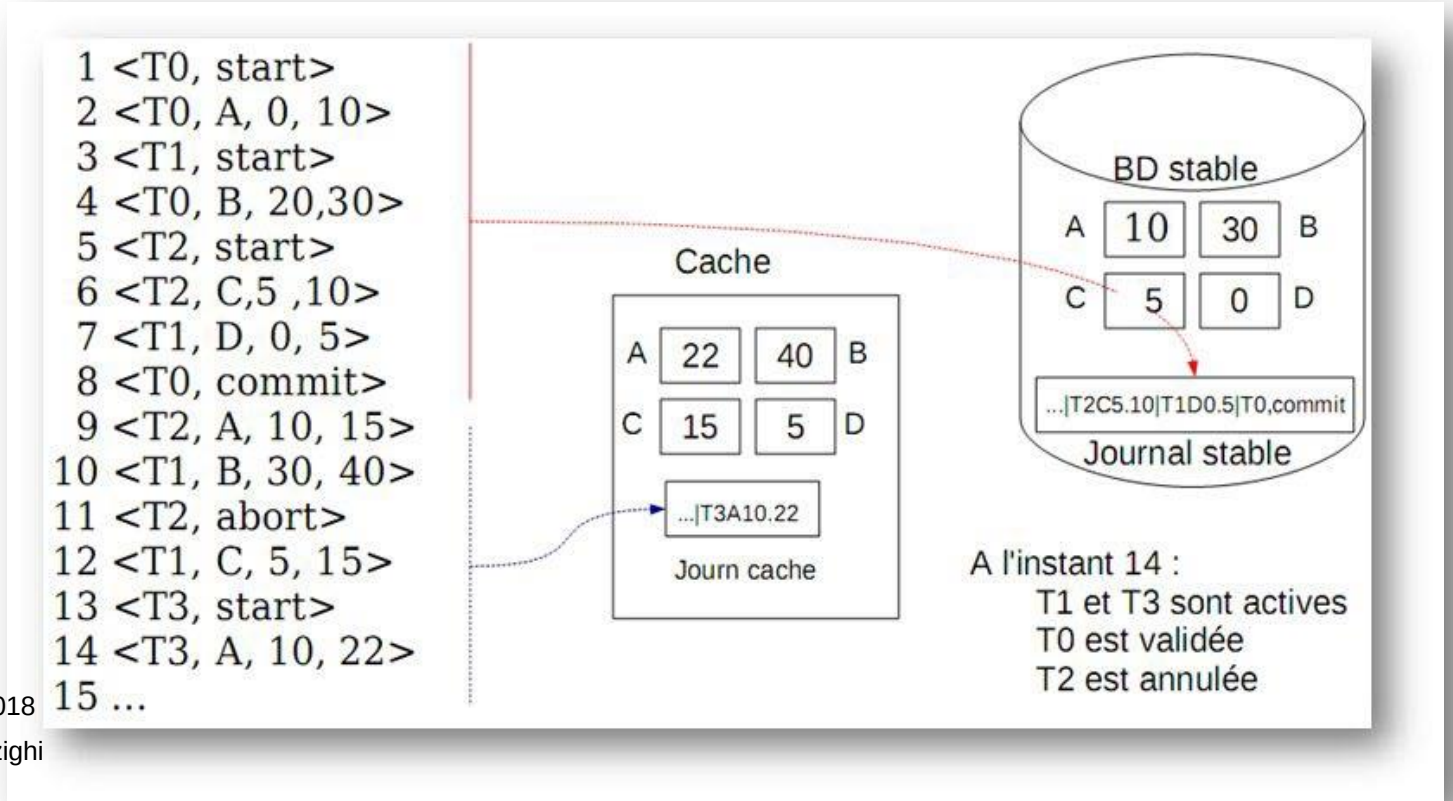
Journal logique

- Garde la trace de la description de haut niveau des opérations de mise à jour

Exemple : « insérer le tuple x dans la table T et mettre à jour les index »

INFORMATIONS DU JOURNAL

1. **<Ti start>**
2. **Identification de la transaction**
3. **Image avant de la donnée, c'est-à-dire sa valeur avant le changement soit l'ancienne valeur.**
4. **Image après de la donnée, c'est-à-dire sa valeur après le changement, soit la nouvelle valeur.**
5. **<Ti commit> lorsque la transaction s'est normalement terminée.**



RÈGLES DU UNDO ET DU REDO

Règle du Undo (ou « Write-Ahead Log Protocol »)

- Avant d'écraser une ancienne valeur par une nouvelle, dans la BD stable, sauvegarder l'ancienne (l'image avant) dans un autre emplacement stable (dans le journal stable)

Règle du Redo (ou « Force Log at commit »)

- Avant d'autoriser une transaction à valider, toutes les valeurs générées (les images-après de son journal) doivent d'abord être sauvegardées en mémoire stable (journal stable)

UTILISATION DU JOURNAL

Lors de l'annulation

- Le système exploite le journal en permettant de remettre les images avant (Undo) en le parcourant en arrière. Le journal est parcouru à partir de la fin jusqu'à la rencontre du début de la transaction annulée.

Lors de la validation

- Le système exploite le journal (à partir du début) en permettant de sauvegarder toutes les images après en mémoire stable.

Lors d'une reprise après panne

- Le système exploite le journal en permettant de refaire l'historique en exécutant les opérations du journal du début jusqu'au moment de la panne (dernier enregistrement).

POINTS DE CONTRÔLE

Un point de synchronisation entre la BD et le fichier journal.

Il permet de limiter l'ampleur de la recherche dans le fichier journal en cas de restauration.

Les points de contrôle sont planifiés à des intervalles réguliers et impliquent des opérations

- D'écriture dans les MS de tous les enregistrements de journaux présents en MC
- D'écriture en MS de tous les blocs modifiés des tampons de la BD
- D'écriture dans le fichier LOG d'un enregistrement de point de contrôle (ckpt). Cet enregistrement mentionne les identificateurs de toutes les transactions actives au moment du point de contrôle.

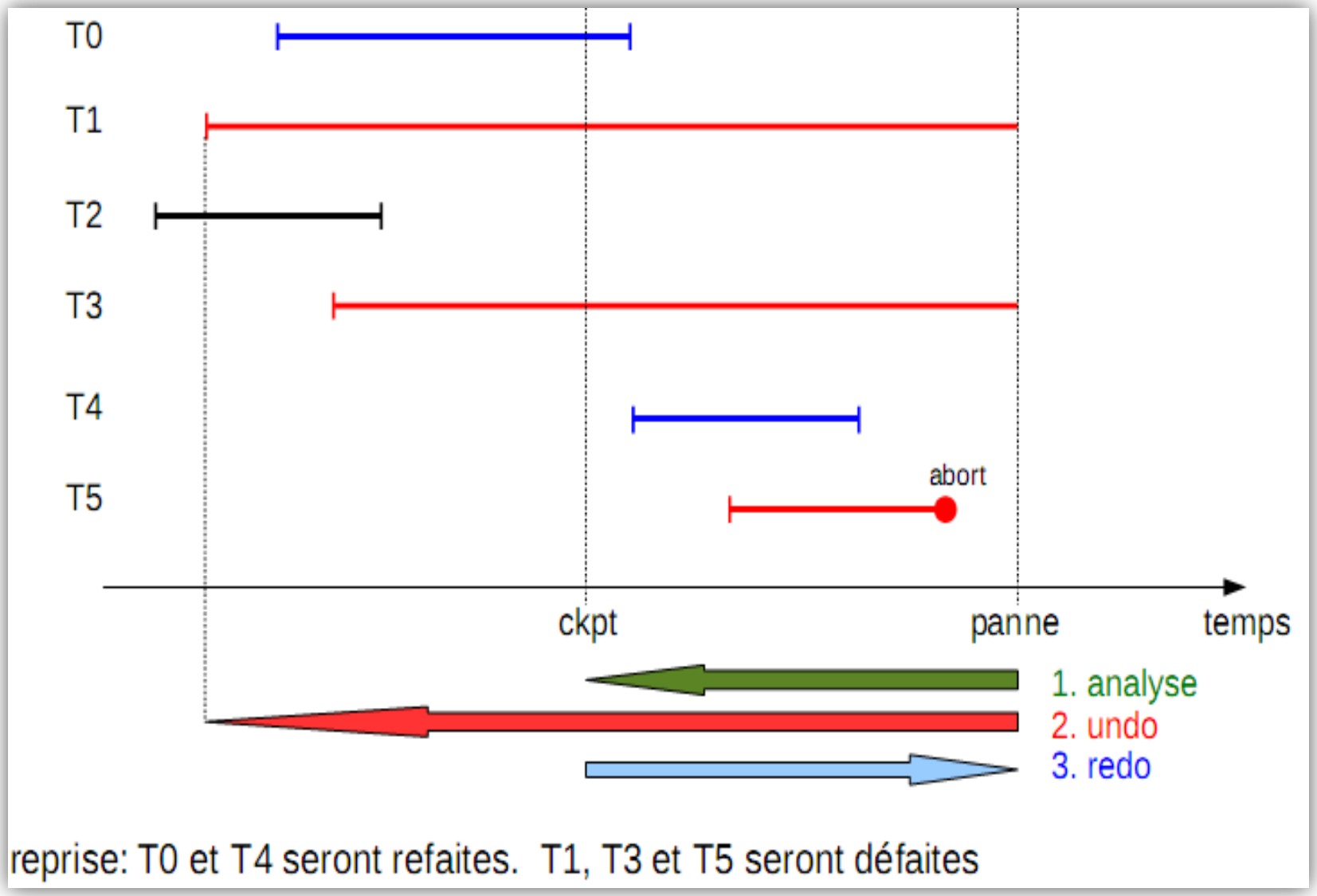
EXPLOITATION DES POINTS DE CONTRÔLE

La gestion des points de contrôle est une opération coûteuse, il est fréquent de se limiter à trois ou quatre point de contrôle par heure.

Lors d'une reprise des transactions annulées, le système parcourt le journal en avant depuis la marque du <ckpt> et refait les opérations des transactions validées. La reprise s'effectue en trois phases : Analyse, Undo et Redo.

- **Analyse :** détermine les transactions validées des autres.
- **Undo :** défaire les transactions annulées et actives.
- **Redo :** refaire les transactions validées.

EXEMPLE


T0
T1
T2
T3
T4
T5
abort
ckpt
panne
temps
1. analyse
2. undo
3. redo
reprise: T0 et T4 seront refaites. T1, T3 et T5 seront défaites

TECHNIQUE DE RÉCUPÉRATION : MISES À JOUR DIFFÉRÉES

Pas d'écriture des modifications dans la base de données tant que la transaction n'a pas atteint son point de validation

Toutes les opérations d'écriture sont différées jusqu'à validation partielle de la transaction.

Dés qu'une transaction démarre alors écriture de <Ti start> dans le journal

Quand une opération d'écriture est effectuée, écriture de la modification dans le fichier LOG.

Quand la transaction est sur le point d'être validée, écriture de <Ti commit> dans le fichier Log.

Ecriture de tous les enregistrements de cette transaction sur MS et validation de la transaction.

SUITE

Les modifications sont écrites dans le fichier journal avant de valider réellement.

En cas de défaillance, le journal est examiné puis il est déroulé jusqu'au point de contrôle le plus récent.

Processus de reprise associé :

- Exécution d'une opération REDO (Ti) qui donne à toutes les données mises-à-jour par Ti leurs nouvelles valeurs à condition que le journal contienne simultanément **<Ti start>** et **<Ti commit>.**

MISE À JOUR IMMÉDIATE

Toutes les mises-à-jour sont effectuées sur la BD au fur et à mesure de leur apparition sans attendre le point de validation.

Les informations du journal sont utilisées à la restauration de l'état de la BD.

Doivent être refaites les mises à jour des transactions validées, à la suite d'une défaillance et

Doivent être défaites les mises à jour des transactions non validées au moment de la défaillance.

MISE À JOUR IMMÉDIATE

Processus de reprise associé:

Utilisation de deux actions

- **Undo(Ti) :** restauration à leurs anciennes valeurs,
- **Redo(Ti) :** restauration des nouvelles valeurs.

Ti est annulée si le journal contient <Ti start> sans le commit, c'est-à-dire restauration de la BD aux anciennes valeurs.

Ti est restaurée aux nouvelles valeurs si le journal contient à la fois <Ti start> et <Ti commit>.

Remarque : il est essentiel d'écrire les enregistrements dans le journal avant les écritures correspondantes dans la BD.

PAGES D'OMBRE : CAS PARTICULIER MISE À JOUR DIFFÉRÉE

Cette technique entretien deux tables de pages pendant la durée de vie d'une transaction

Table de page courante contenant toutes adresses des pages de données ayant subies des mises à jour effectuées par la transaction stockée en mémoire vive.

Table des pages d'ombre ou d'arrière plan, contenant l'état avant exécution de la transaction stockée en mémoire sûre.

Processus de reprise associé:

- Quand une transaction démarre, les deux tables sont identiques.
- Si la transaction est validée, la table de page courante devient la table de page d'ombre, et une autre table de page courante est définie pour la transaction suivante.
- Si la transaction est annulée, la table courante est simplement abandonnée. La récupération est plus rapide puisqu'il n'est plus nécessaire de refaire ni de défaire des transactions.