



# Transactions

---

**ENSIMAG 2<sup>ème</sup> année**

*Equipe Pédagogique BD Ensimag*

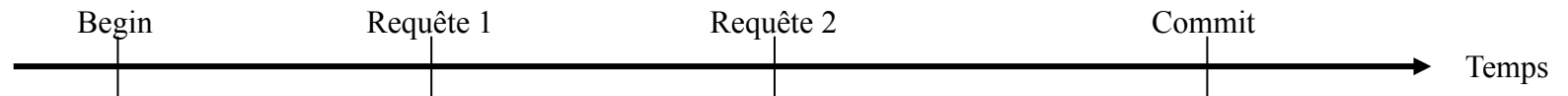
# Transactions

---

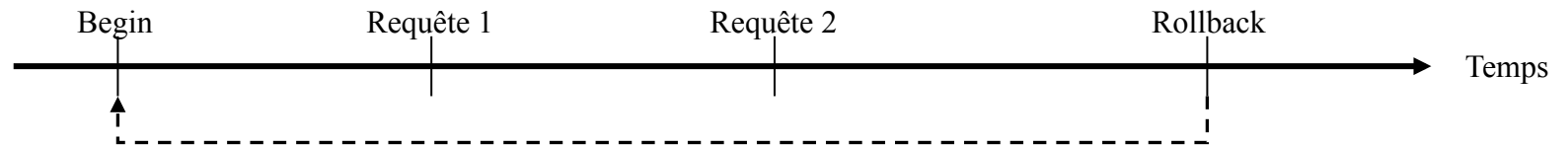
- Séquence de requêtes SQL (*select, insert, update, delete*)
- Unité logique de traitement
  - ◆ Fait passer la BD d'un état cohérent à un autre état cohérent
  - ◆ État cohérent :
    - ★ Schéma conceptuel global
    - ★ Contraintes d'intégrité fonctionnelles
- Propriétés ACID
  - ◆ Atomicité : tout ou rien
  - ◆ Cohérence : contraintes d'intégrité (différées)
  - ◆ Isolation : vue mono-utilisateur
  - ◆ Durabilité : résultats validés jamais perdus
- Instructions SQL
  - ◆ `(begin;), commit;, rollback;`

# Déroulement d'une transaction

## ■ Tout se passe bien

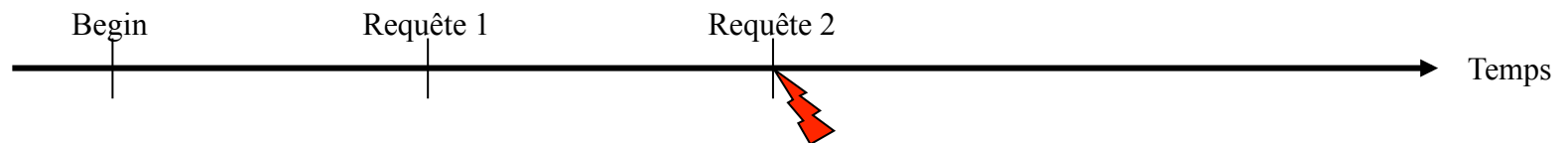


## ■ Abandon explicite



## ■ Erreur dans l'exécution d'une requête

- ◆ Norme : Abandon automatique
- ◆ En pratique : le SGBD rend la main



## ■ Principe

- ◆ Soit toutes les requêtes s'exécutent correctement
- ◆ Soit aucune

## ■ En pratique

- ◆ Utilisation d'un journal d'image avant
  - ★ Sauvegarde de l'état de la base de données avant la modification
- ◆ Opérations sur une copie des données
  - ★ Remplacement en cas de validation

## ■ Principe

- ◆ Toutes les contraintes d'intégrité doivent être vérifiées
  - ★ Vérification au fur et à mesure de l'exécution
  - ★ Vérification différée à la validation

## ■ Syntaxe SQL

- ◆ Ajout du mot clef **DEFERRABLE** après la définition d'une contrainte d'intégrité
- ◆ Dans une transaction :
  - ★ **SET CONSTRAINT [S]**  
(**ALL** | <constraint\_name>)  
(**IMMEDIATE** | **DEFERRED**) ;

## ■ Principe

- ◆ L'utilisateur doit avoir l'impression d'être le seul connecté à la BD
- ◆ Une transaction ne doit pas pouvoir voir les résultats intermédiaires des autres transactions

## ■ En pratique

- ◆ Exécution concurrente de transactions
  - ★ Entrelacement des opérations des différentes transaction (séquence réelle)
- ◆ **Sérialisabilité**
  - ★ Séquence réelle équivalente à une exécution en série

# Exécutions en série

	T1	T2	A	B		T1	T2	A	B
<i>Temps</i>			25	25				25	25
	lire(A,t)					lire(A,s)			
	t:=t+100					s:=s*2			
	écrire(A,t)		125			écrire(A,s)	50		
	lire(B,t)					lire(B,s)			
	t:=t+100					s:=s*2			
	écrire(B,t)			125		écrire(B,s)			50
		lire(A,s)				lire(A,t)			
		s:=s*2				t:=t+100			
		écrire(A,s)	250			écrire(A,t)		150	
		lire(B,s)				lire(B,t)			
		s:=s*2				t:=t+100			
		écrire(B,s)		250		écrire(B,t)			150

# Exécutions sérialisables

T1	T2	A	B
		25	25
lire(A,t)			
t:=t+100			
écrire(A,t)		125	
	lire(A,s)		
	s:=s*2		
	écrire(A,s)	250	
lire(B,t)			
t:=t+100			
écrire(B,t)			125
	lire(B,s)		
	s:=s*2		
	écrire(B,s)		250

T1	T2	A	B
		25	25
	lire(A,s)		
	s:=s*2		
	écrire(A,s)	50	
lire(A,t)			
t:=t+100			
écrire(A,t)		150	
	lire(B,s)		
	s:=s*2		
	écrire(B,s)		50
lire(B,t)			
t:=t+100			
écrire(B,t)			150



# Exécution non sérialisable

T1	T2	A	B
		25	25
lire(A,t)			
t:=t+100			
écrire(A,t)		125	
	lire(A,s)		
	s:=s*2		
	écrire(A,s)	250	
	lire(B,s)		
	s:=s*2		
	écrire(B,s)		50
lire(B,t)			
t:=t+100			
écrire(B,t)			150

## ■ Pessimistes

- ◆ Interdiction des séquences potentiellement non sérialisables
- ◆ Exemples
  - ★ Verrouillage
  - ★ Estampillage

## ■ Optimistes

- ◆ Exécution libre et vérification à la validation
- ◆ Exemple
  - ★ Certification

# Niveaux d'isolation

---

- La sérialisabilité peut s'avérer trop contraignante et pas toujours nécessaire
- Relâchement de l'isolation, 4 niveaux
  - ◆ Read uncommitted (plus faible)
  - ◆ Read committed
  - ◆ Repeatable Read
  - ◆ Serializable (plus fort)

Standardisé par SQL2,  
Set Transaction Isolation Level

## ...Niveaux d'isolation...

---

### ■ Read uncommitted

- ◆ Lecture des résultats intermédiaires (lectures sales)
  - ★  $T_1$  lit le tuple A
  - ★  $T_1$  écrit le tuple A
  - ★  $T_2$  lit le tuple A
  - ★  $T_1$  abandonne
  - ★  $T_2$  exploite la valeur du tuple A qui est incorrecte
  
- ◆ Contrôle de concurrence
  - ★ Séquences écriture → écriture

## ...Niveaux d'isolation...

---

### ■ Read committed

- ◆ Lecture de valeurs validées uniquement (pas de lectures sales)
- ◆ Lectures non reproductibles possibles
  - ★  $T_1$  lit le tuple A
  - ★  $T_1$  écrit le tuple A
  - ★  $T_2$  lit le tuple A (valeur avant  $T_1$ )
  - ★  $T_1$  valide
  - ★  $T_2$  relit le tuple A (nouvelle valeur validée)
- ◆ Fantômes possibles

## ■ Repeatabe Read

- ◆ Lecture reproductible
- ◆ Possibilité d'apparition de fantômes (nouvelle donnée)
  - ★  $T_1$  calcule la moyenne de R.A
  - ★  $T_2$  insert un nouveau tuple dans R
  - ★  $T_2$  valide
  - ★ Le calcul de  $T_1$  n'est plus à jour
- ◆ Contrôle de concurrence
  - ★ Séquences écriture→écriture, écriture→lecture et lecture→écriture

# ...Niveaux d'isolation

## Serializable

- Pas d'anomalies
- Contrôle de concurrence
  - ◆ Séquences écriture→écriture, écriture→lecture, lecture→écriture et lecture→insertion

Isolation level	Lectures sales	Lectures non reproductibles	Fantômes
Serializable			
Repeatable Read			Possible
Read Committed		Possible	Possible
Read Uncommitted	Possible	Possible	Possible

## ■ Principe

- ◆ Les effets d'une transaction validée ne doivent **jamais** être perdus

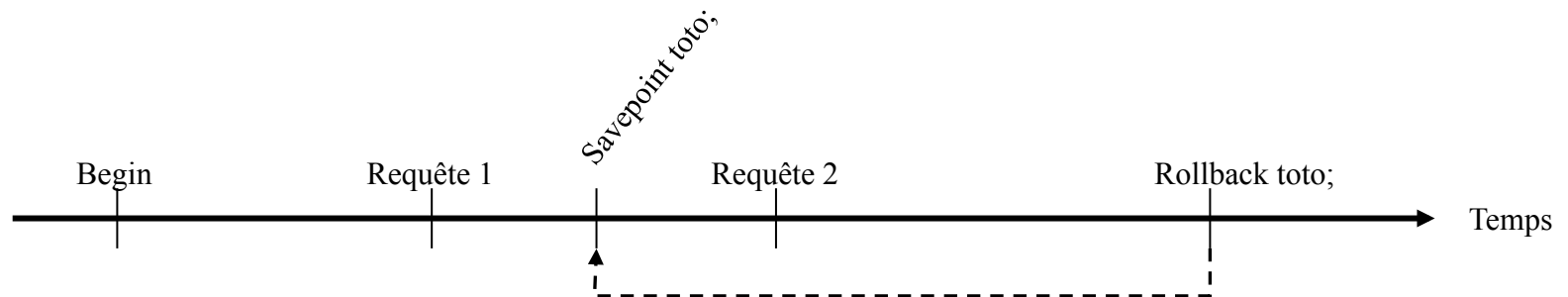
## ■ En pratique

- ◆ Journaux
  - ★ Opérations exécutées au fur et à mesure
  - ★ Terminaison de transaction
  - ★ (Images après)
- ◆ Sauvegardes externalisées



# Points de sauvegarde

- **Utilité**
  - ◆ Abandon partiel
- **Propriété**
  - ◆ L'exécution reste linéaire
- **Syntaxe SQL**
  - ◆ **Savepoint** <nom>;
  - ◆ **Rollback** <nom>;



- **Verrouillage multi-versions**

- ◆ Chaque utilisateur (session) travaille sur une copie des données

- **Principe**

- ◆ Verrouillage en lecture sur la copie locale
- ◆ Verrouillage en écriture sur toutes les copies

- **Modes d'isolation**

- ◆ Read committed (par défaut)
  - ★ Synchronisation de la copie locale dès qu'une transaction valide, quelle qu'elle soit (mode *push*)
- ◆ Serializable
  - ★ Synchronisation de la copie locale uniquement à la terminaison de la transaction locale (mode *pull*)

- **Notion de transaction, concept très important**
- **Choix du niveau d'isolation à faire selon les besoins applicatifs**
- **Reste à mettre cela en pratique**
  - ◆ TP transactions...