



# トランザクションとロック



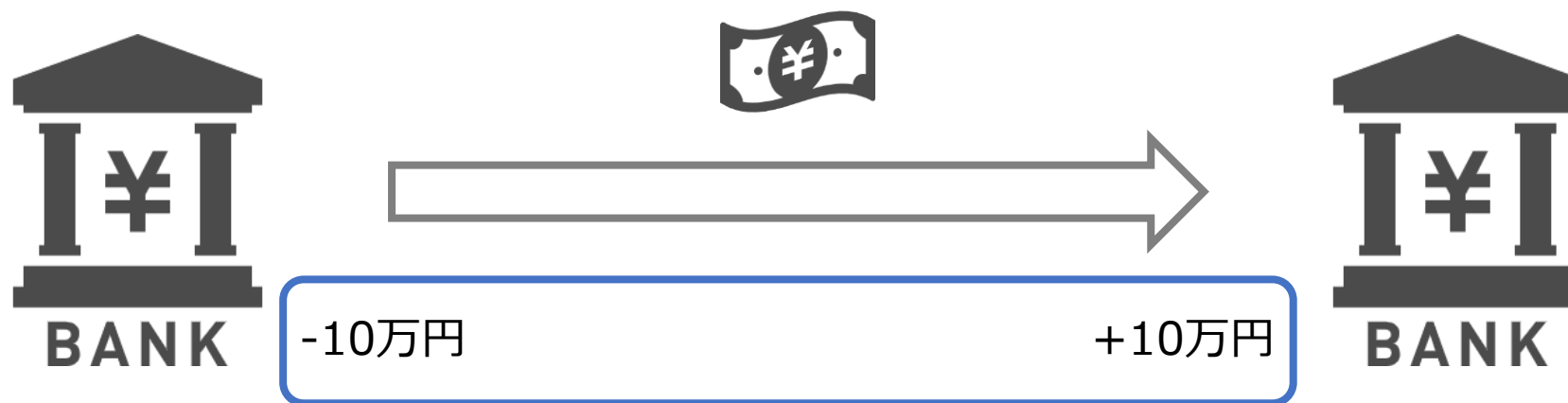
## 目次

- トランザクション
- コミット
- ロールバック
- ロック
- セーブポイント



## トランザクションとは

複数の処理をまとめて扱う機能のこと。  
データを安全に管理するための仕組みの一つ。



ひとまとめ



## トランザクションの必要性

Aさんの口座からBさんの口座に10万円振り込む場合、以下の2つの処理が行われる。

- ① Aさんの口座を-10万円
- ② Bさんの口座を+10万円



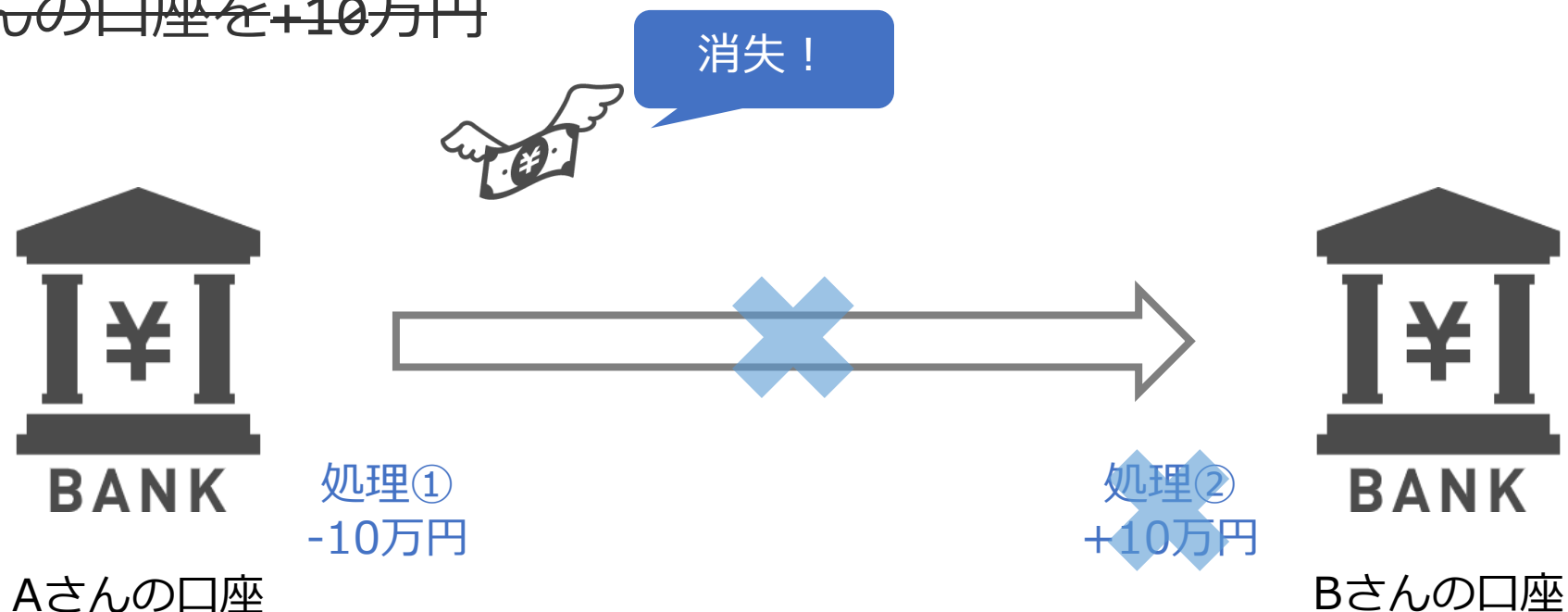


## トランザクションの必要性(続き)

②の処理でエラーが起きて①の処理だけ行われた場合、  
Aさんの10万円が消えてしまう

① Aさんの口座を-10万円

② ~~Bさんの口座を+10万円~~





## トランザクションで実現できること

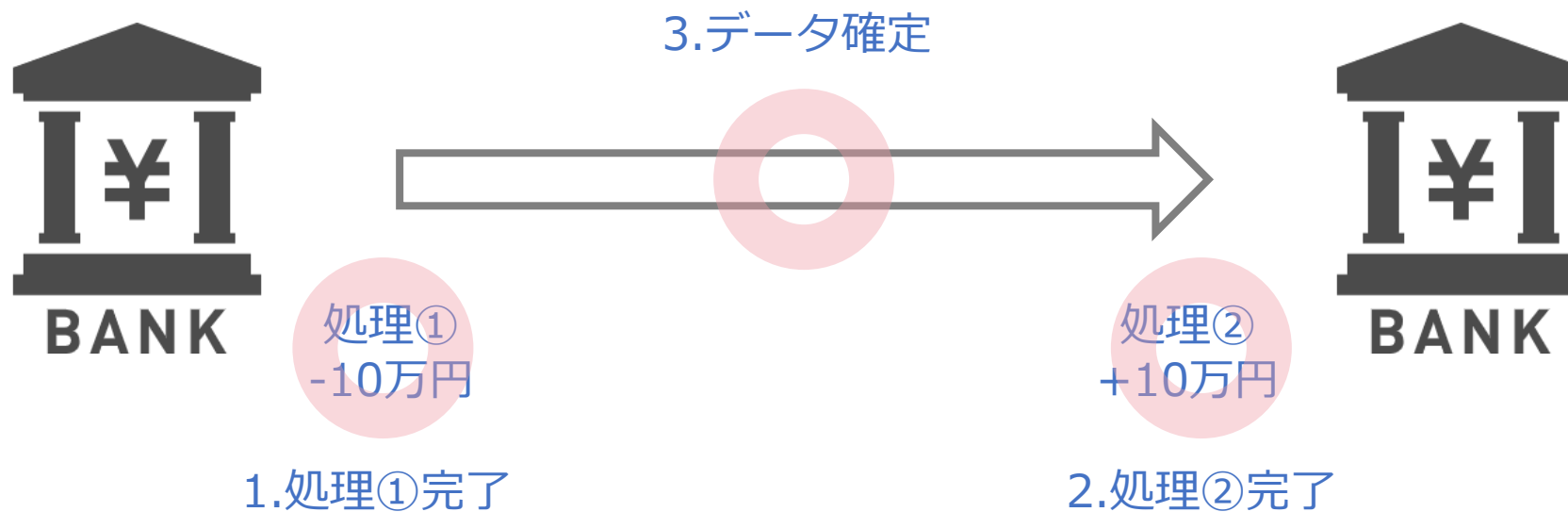
トランザクションによって処理をひとまとめにすることで、以下の2つを実現できる。

- ① 全ての処理が正常に完了したときのみ、データを確定する
- ② エラーが起きた場合、すべての処理を取り消す



## トランザクションで実現できること①

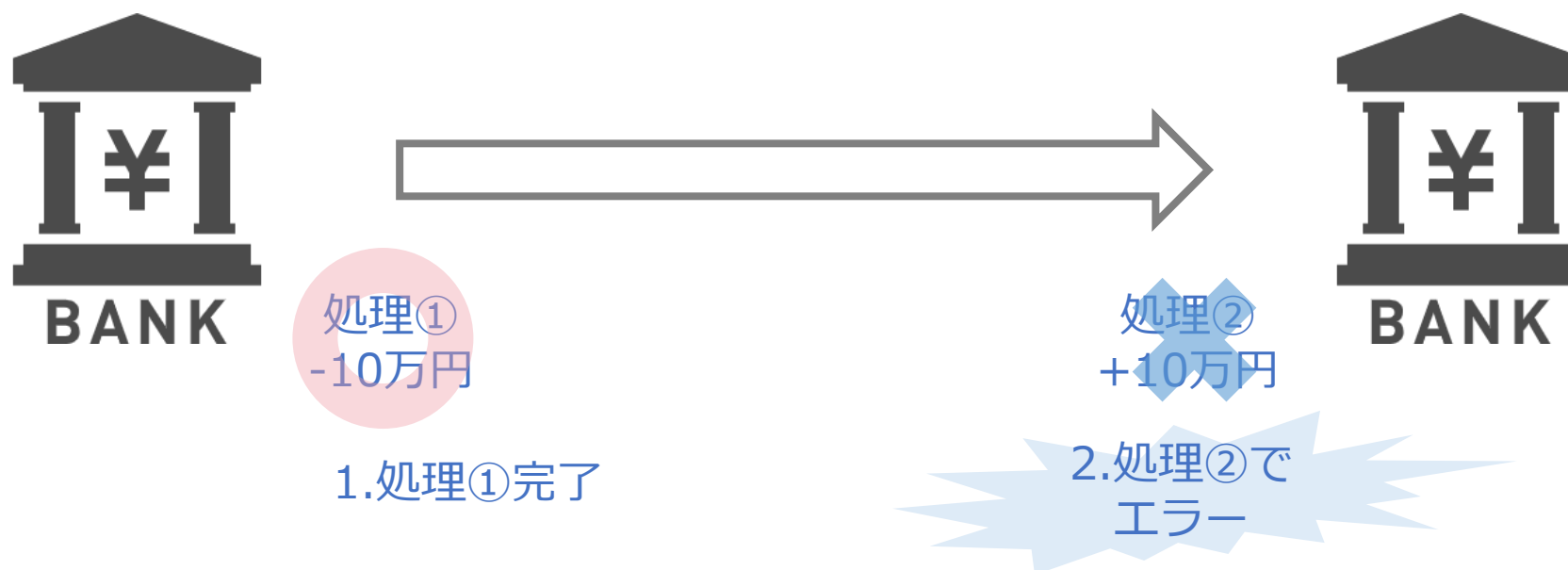
全ての処理が正常に完了したときのみ、データを確定する





## トランザクションで実現できること②

エラーが起きた場合、すべての処理を取り消す

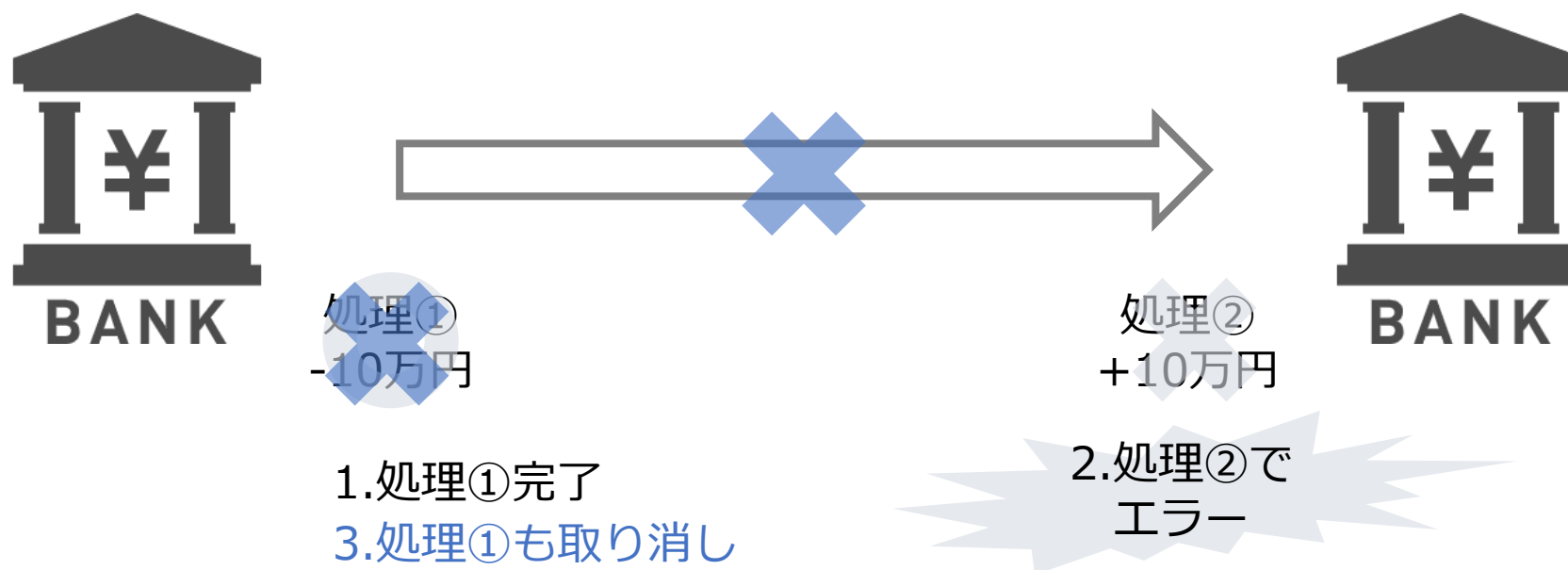






## トランザクションで実現できること②(続き)

エラーが起きた場合、すべての処理を取り消す





## コミットとロールバック

トランザクションの終了時、コミットもしくはロールバックを行う。

全ての処理が正常に完了したときのみ、データを確定する

→コミット

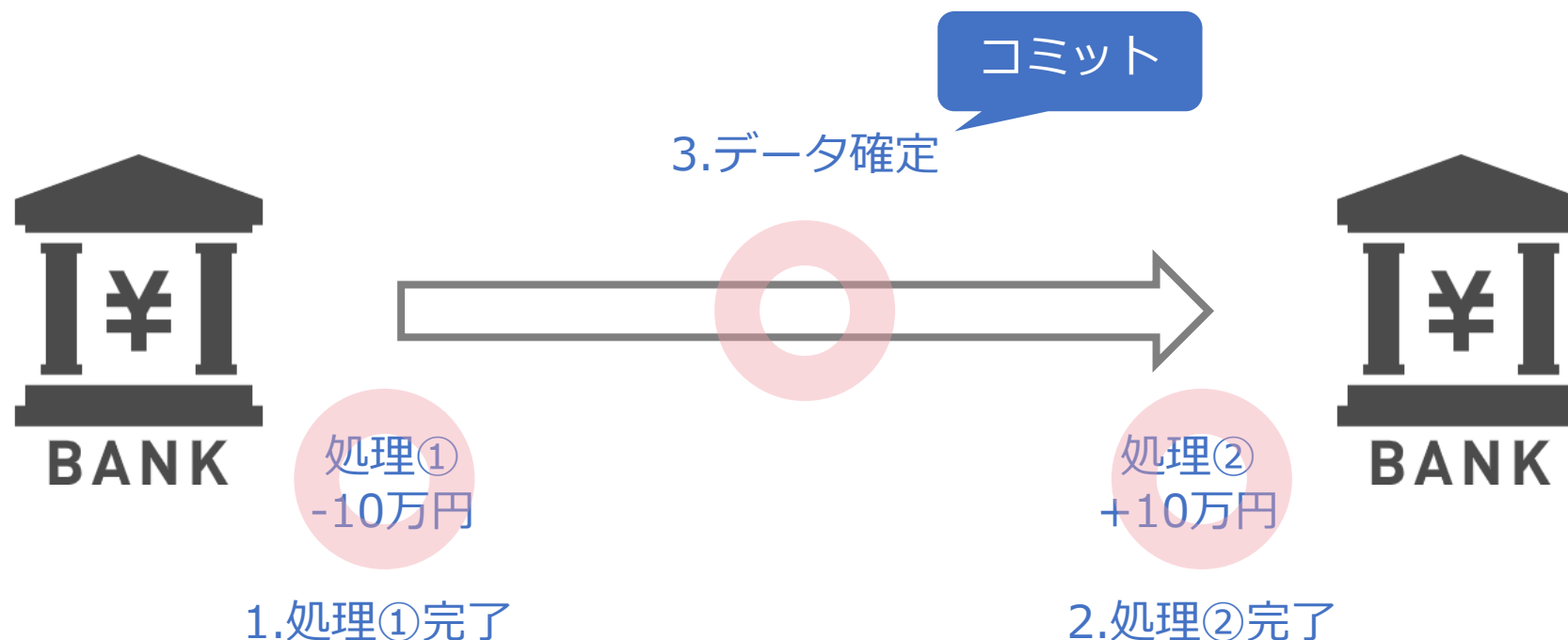
エラーが起きた場合、すべての処理を取り消す

→ロールバック



## コミットとは

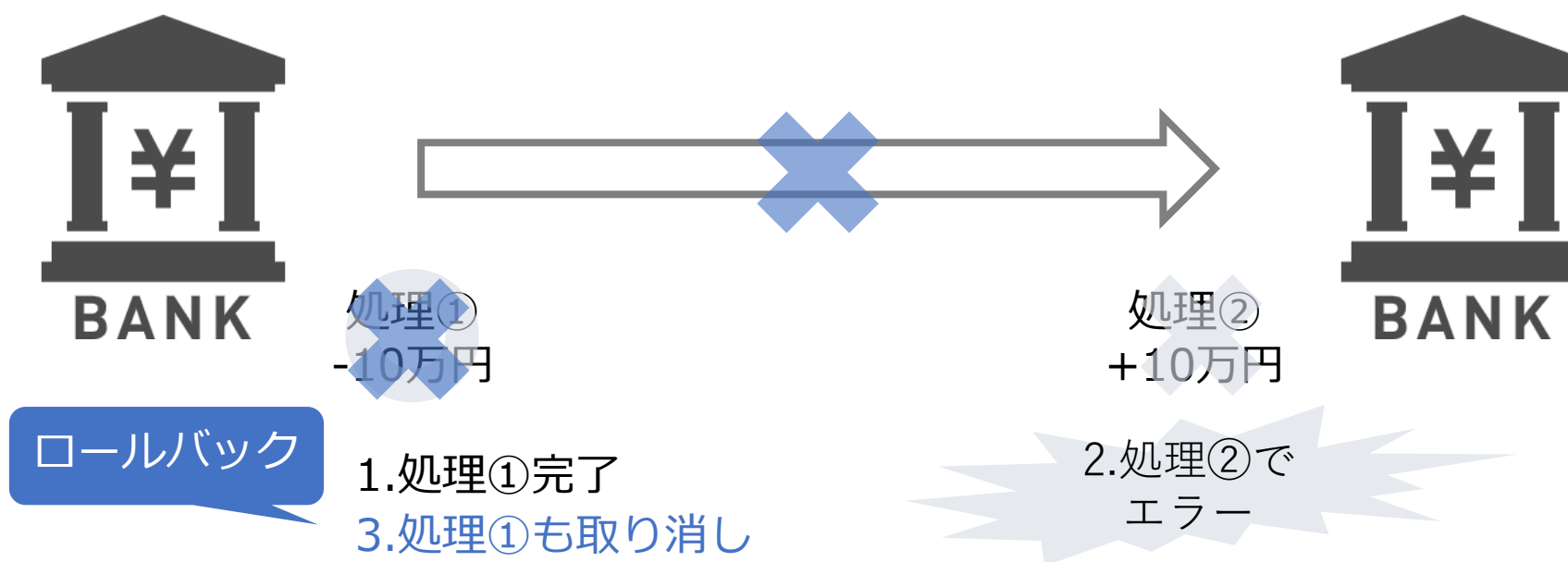
トランザクションの結果をデータベースに反映させること。





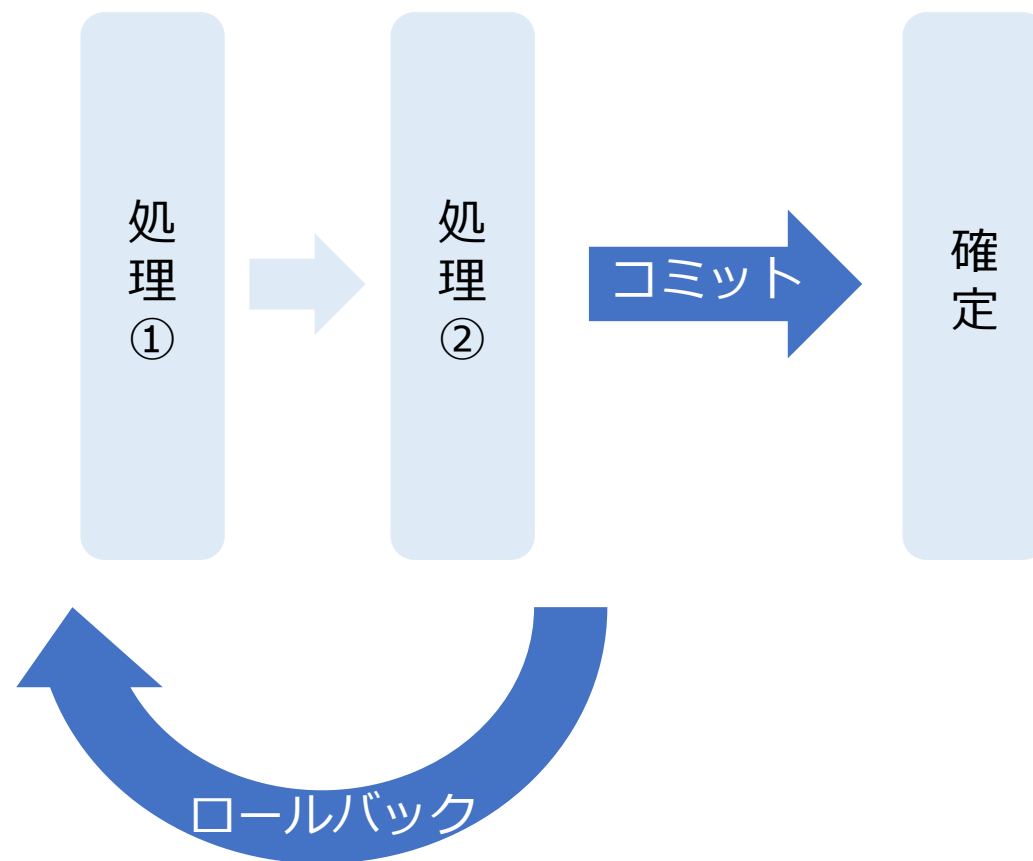
## ロールバックとは

トランザクションの結果を反映せずに、元に戻すこと。





## (図解) コミットとロールバック





## ロックとは

INSERT、UPDATE、DELETE実行後、他から変更できなくすること。  
トランザクションが終了するまで対象のレコードはロックされた状態となる。

社員ID	氏名	部署ID
1	田中太郎	D01





## 行レベルロックとは

行(レコード)単位で行うロックのこと。

このレコード  
だけロック

社員ID	氏名	部署ID
1	田中太郎	D01
2	佐藤次郎	D01
3	鈴木三郎	A02
4	吉田五郎	A01





## ロックの機能

ロック中は以下の機能が働く。

- INSERT、UPDATE、DELETEを実行した結果は、COMMIT・ROLLBACKするまで他のユーザのSELECTの結果に反映されない
- INSERT、UPDATE、DELETEを実行した行は、COMMIT・ROLLBACKするまで他のユーザは変更できない





## ロックが解除されるタイミング

- COMMITかROLLBACKを実行したとき
- DDL文 (ALTER CREATE DROP)を実行したとき
- 接続を切断(異常終了)したとき



## 読み取り一貫性とは

常に整合性のあるデータが読み取れるという特徴のこと。  
ロック機能によって実現されている。

入金前

口座	残高
A	10万
B	0

処理中

口座	残高
A	0
B	0

入金後

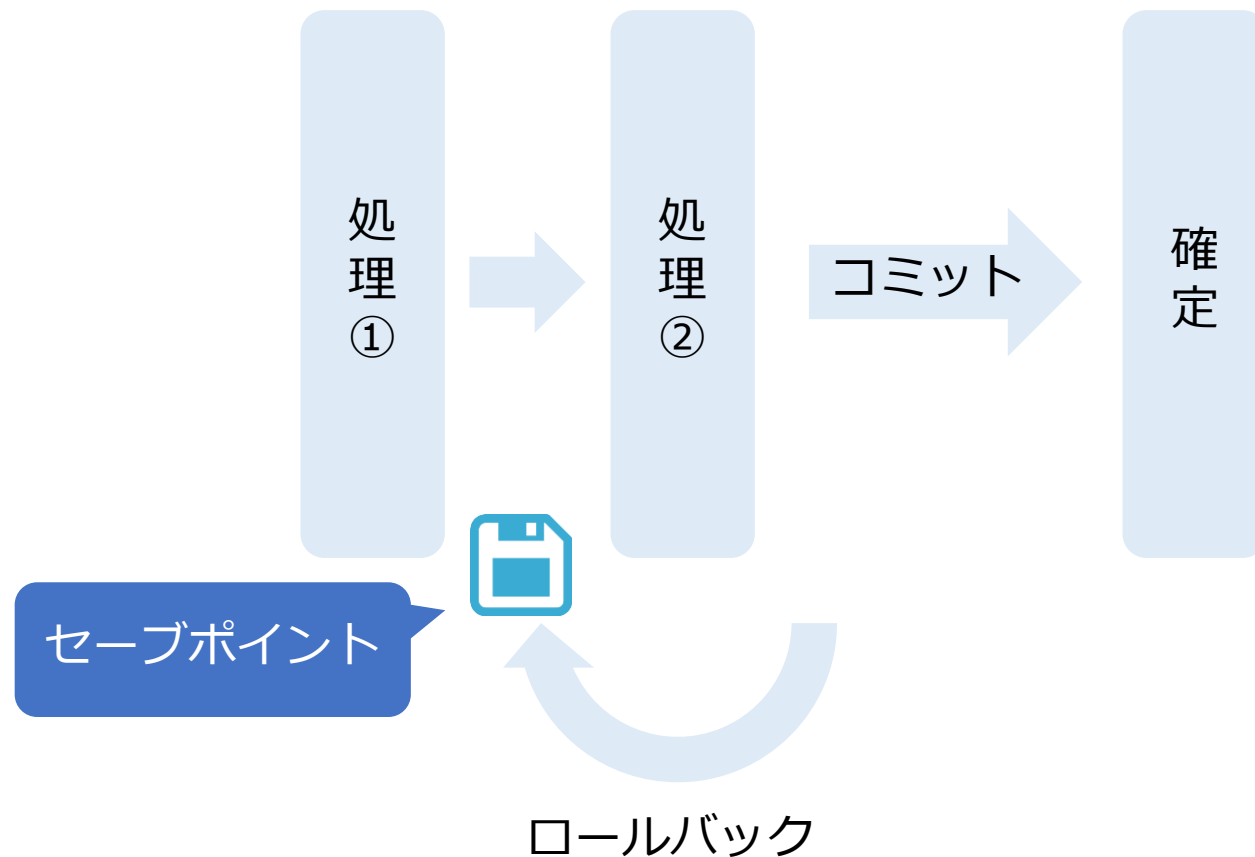
口座	残高
A	0
B	10万

ロック中のため、  
この状態のデータを  
読み取られることはない



## セーブポイントとは

ROLLBACKで処理を戻す際に、どの時点まで戻すかの指定ができる機能。





## セーブポイントの書き方

セーブポイントの設定

```
SAVEPOINT <セーブポイント名>;
```

指定するセーブポイントまでロールバック

```
ROLLBACK TO <セーブポイント名>;
```



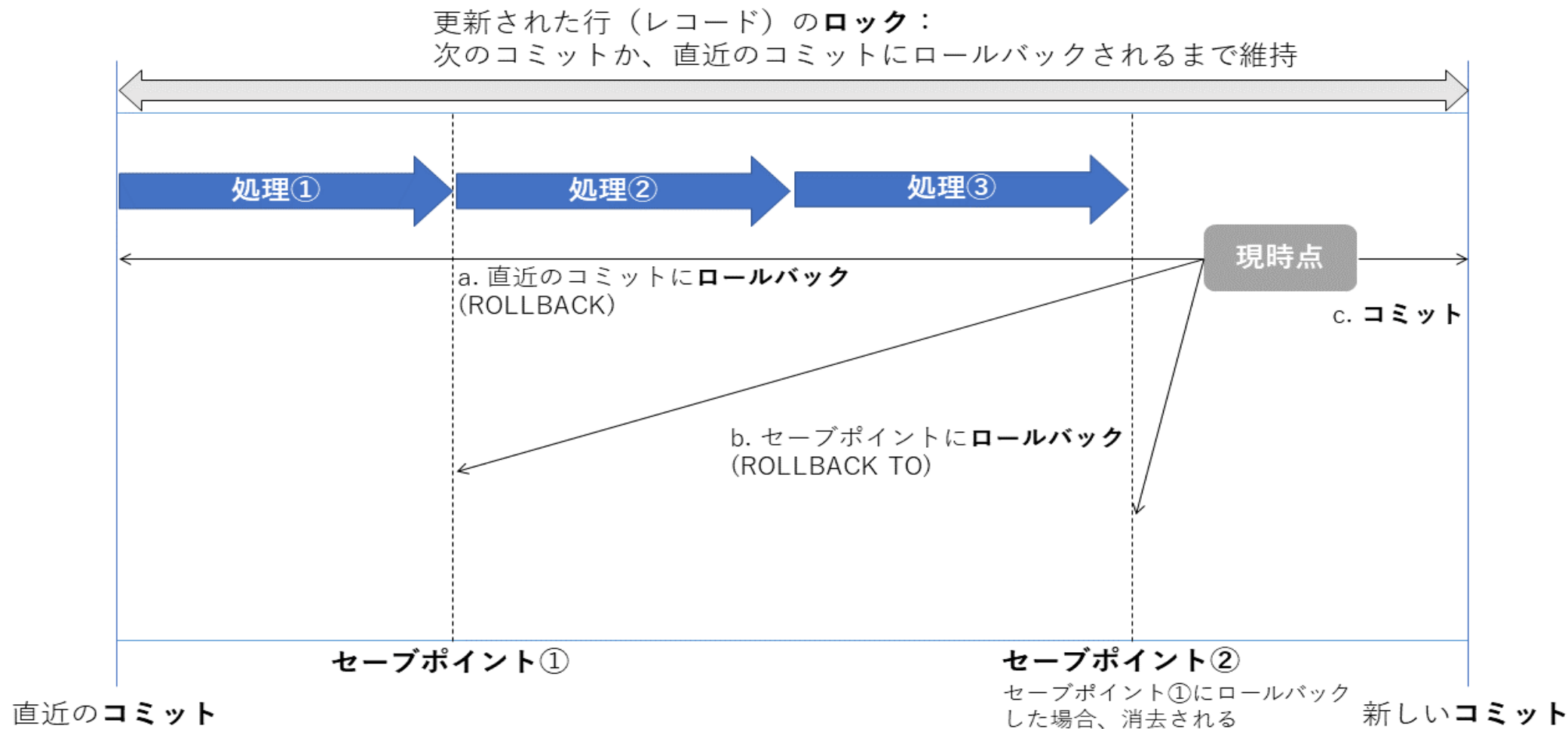
## トランザクションまとめ

1つのトランザクション内で複数の処理を行ったあと、以下の3つの選択肢がある。

- 直近のコミットにロールバック
- セーブポイントにロールバック
- コミット



## トランザクションまとめ(図解)





## トランザクションまとめ(続き)

- 行（レコード）のロック状態は新しいコミットがされるか、直近のコミットにロールバックされるまで維持される。
- ROLLBACK文を使って直近のコミットにロールバックできる。
- ROLLBACK TO文を使って特定のセーブポイントにロールバックできる。
- セーブポイントにロールバックした場合、そのセーブポイント以降に作成されたセーブポイントは消去される。