

INDUSTRIA 4.0

Processamento de
Big Data



MARMELADA NO RING, FONTE: ESTADAO

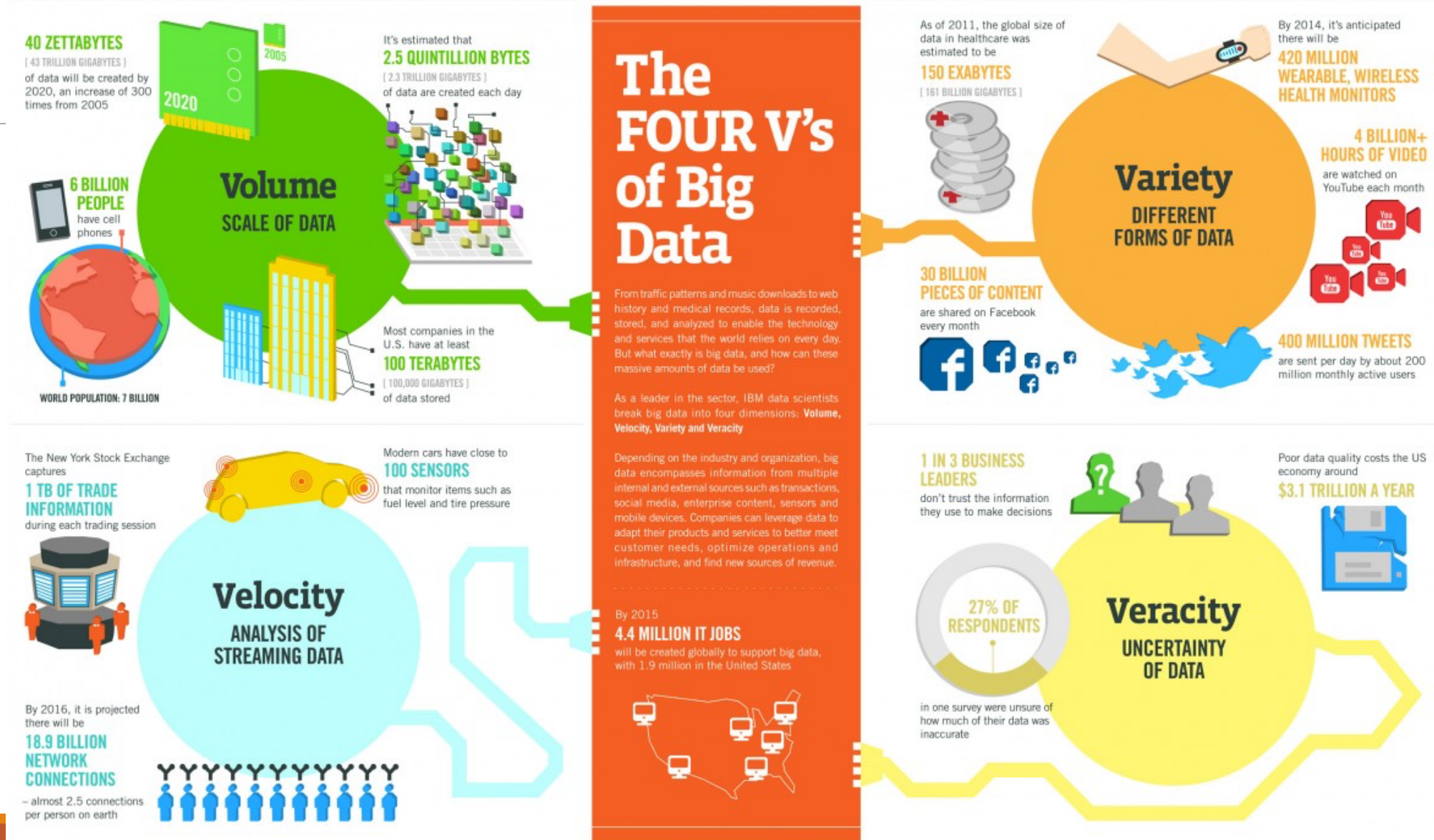
Aula #8 - Transações e concorrência

EDUARDO CUNHA DE ALMEIDA

Agenda

- Transações
- Atomicidade
- Consistência
- Isolamento
- Durabilidade

Transações criam e atualizam dados



Transação

“Uma transação é uma sequência de uma ou mais operações de acesso em um banco de dados compartilhado.”

Transação

“Uma transação é uma sequência de uma ou mais operações de acesso em um banco de dados compartilhado.”

ex: depositar R\$ 100,00



Transação

“Uma transação é uma sequência de uma ou mais operações de acesso em um banco de dados compartilhado.”



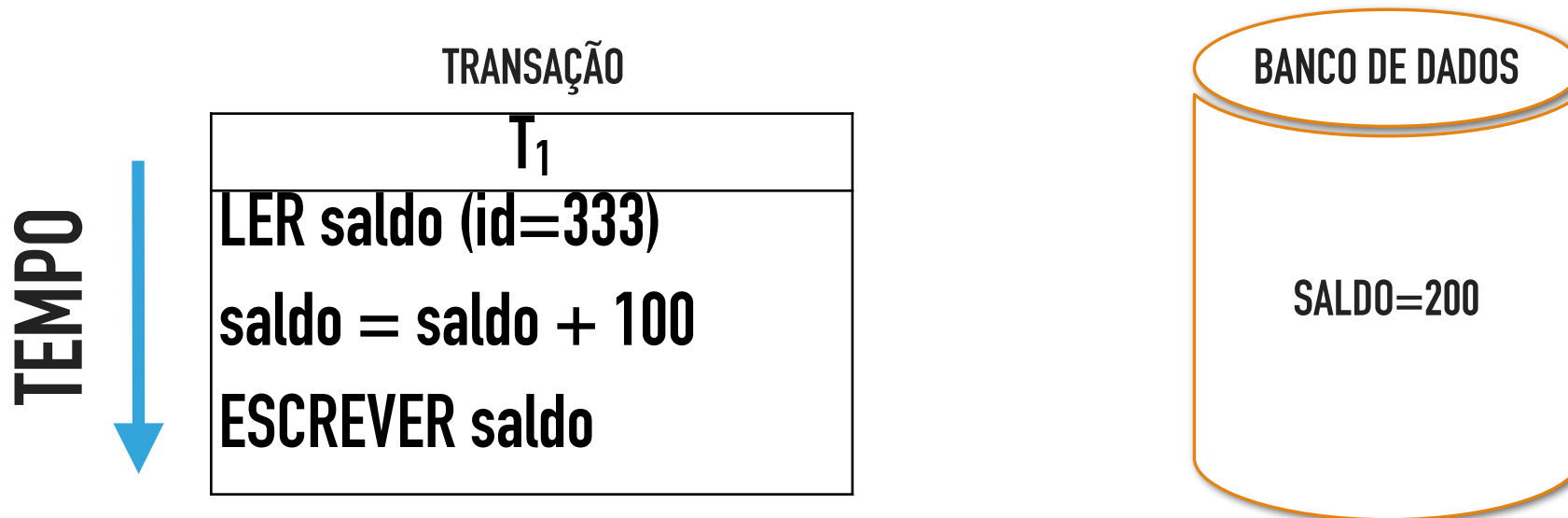
Transação

“Uma transação é uma sequência de uma ou mais operações de acesso em um banco de dados compartilhado.”



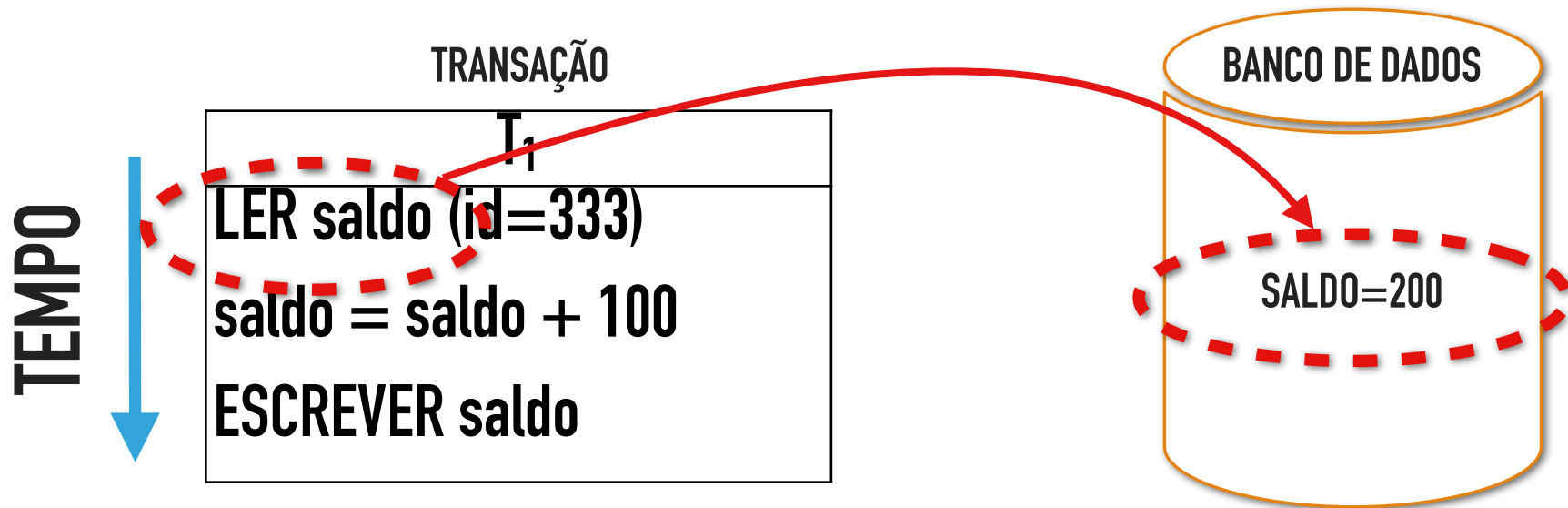
Transação

“Uma transação é uma sequência de uma ou mais operações de acesso em um banco de dados compartilhado.”



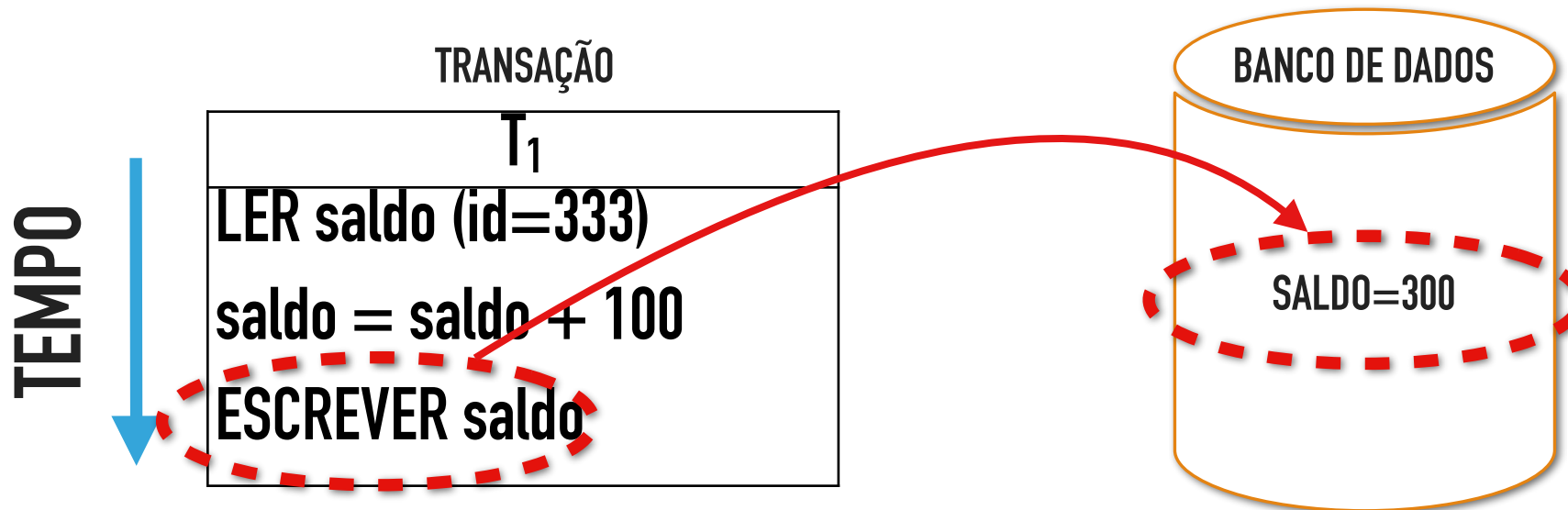
Transação

“Uma transação é uma sequência de uma ou mais operações de acesso em um banco de dados compartilhado.”

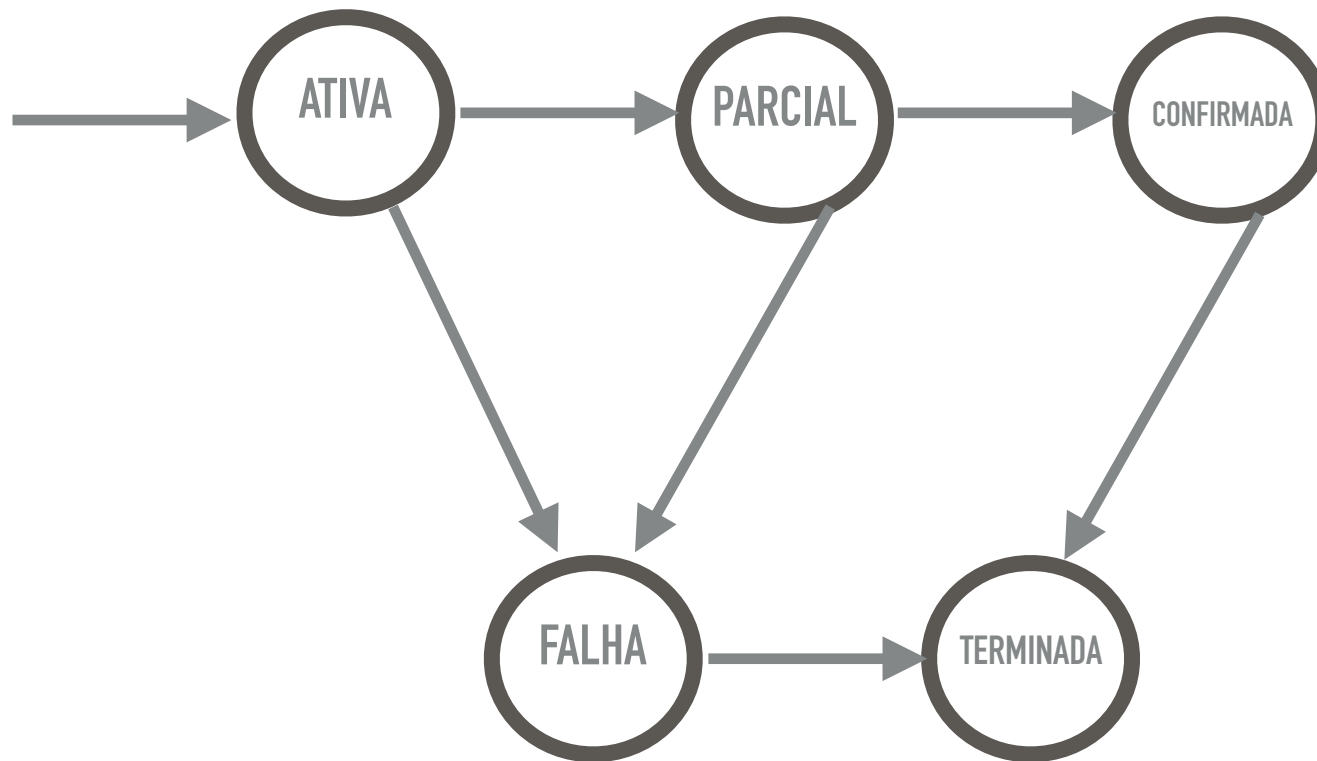


Transação

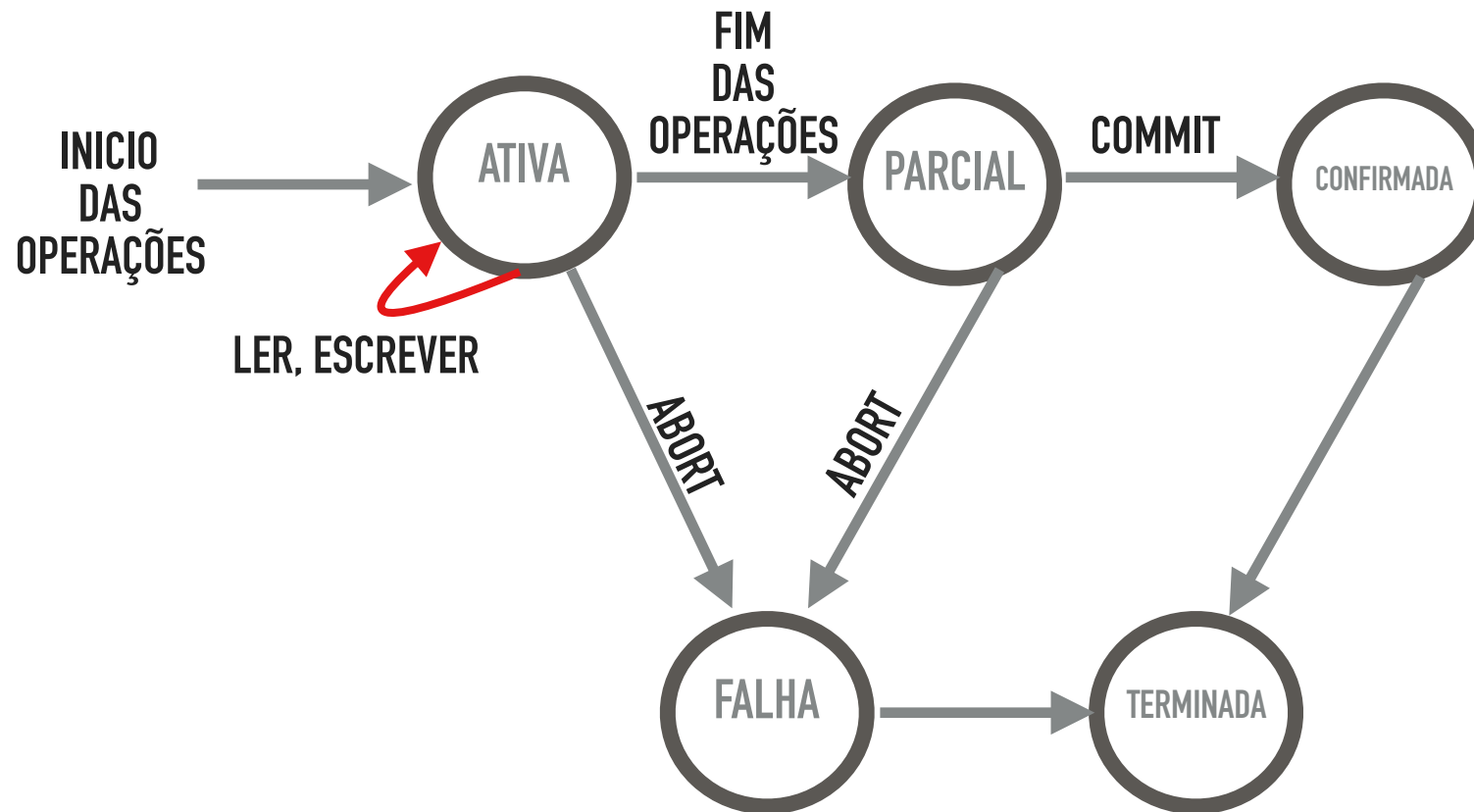
“Uma transação é uma sequência de uma ou mais operações de acesso em um banco de dados compartilhado.”



Estados de uma Transação



Estados de uma Transação



Propriedades de exatidão **ACID**

- **Atomicidade:** “tudo ou nada”
- **Consistência:** “parece correto”
- **Isolamento:** “como se estivesse sozinho”
- **Durabilidade:** “modificações persistem após commit”

Mecanismo pra manter **Atomicidade**

- **Logging:** “caixa preta do avião”

ESCRITAS



Arquivo LOG

INSERT id=333, saldo=200

INSERT id=334, saldo=70

INSERT id=335, saldo=350

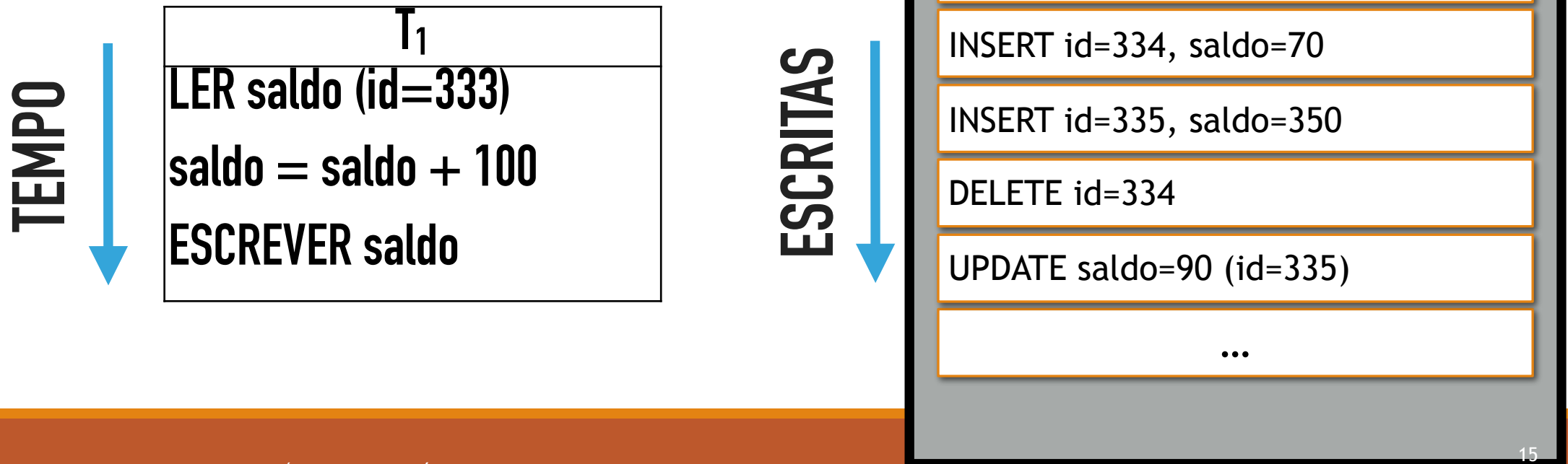
DELETE id=334

UPDATE saldo=90 (id=335)

...

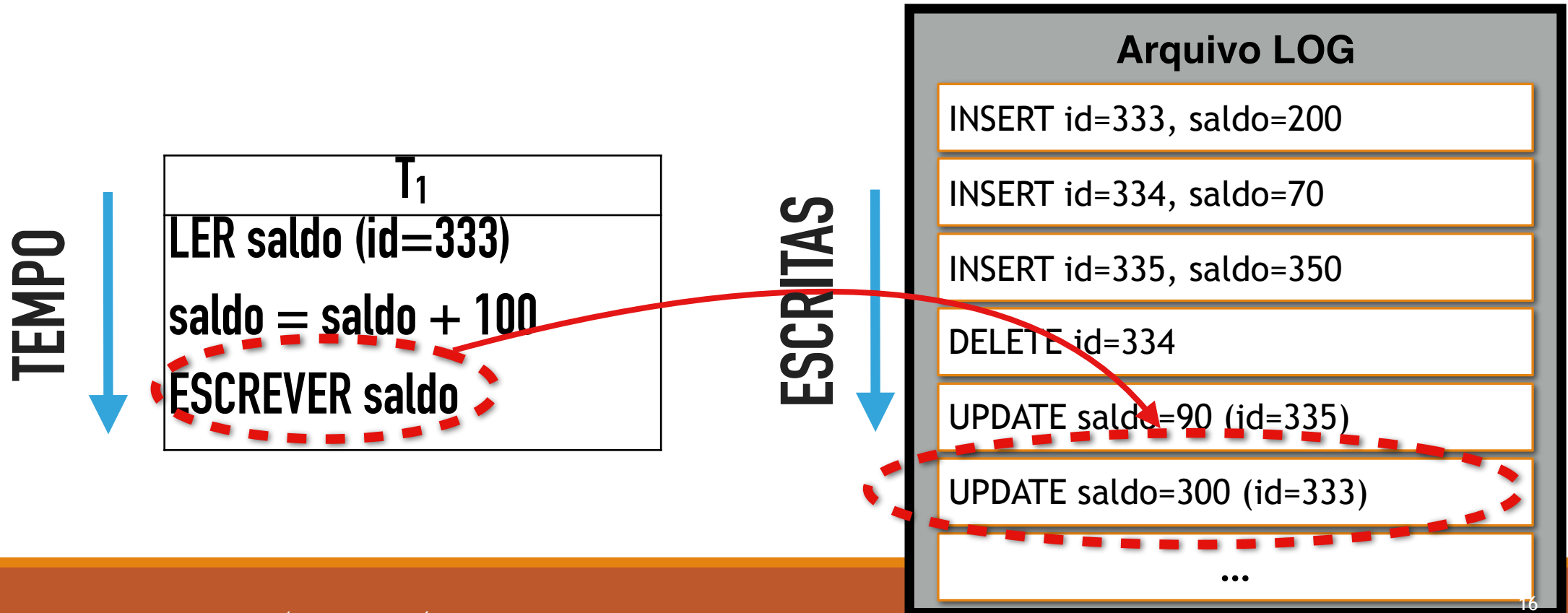
Mecanismo pra manter **Atomicidade**

- **Logging:** “caixa preta do avião”



Mecanismo pra manter **Atomicidade**

- **Logging:** “caixa preta do avião”



Mecanismo pra manter **Atomicidade**

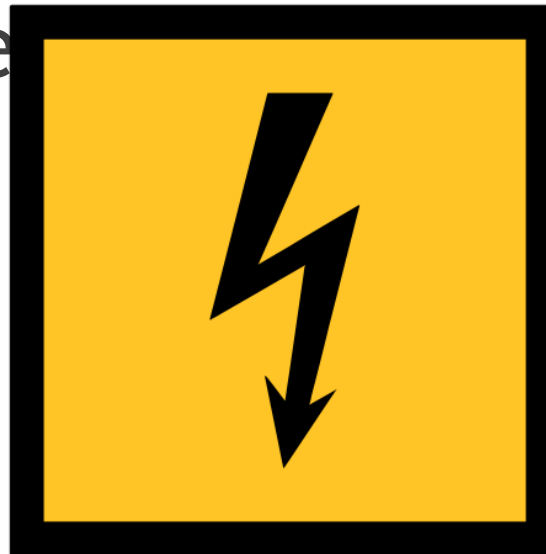
- Logging: “caixa preta”

ex: falta de energia elétrica

TABELA



id	saldo
333	200
334	70
335	90



Arquivo LOG

INSERT id=333, saldo=200

INSERT id=334, saldo=70

INSERT id=335, saldo=350

DELETE id=334

UPDATE saldo=90 (id=335)

UPDATE saldo=300 (id=333)

...

Mecanismo pra manter **Atomicidade**

- **Logging:** “caixa preta do avião”



TABELA

id	saldo
333	200
334	70
335	90

LEITURA



Arquivo LOG

INSERT id=333, saldo=200

INSERT id=334, saldo=70

INSERT id=335, saldo=350

DELETE id=334

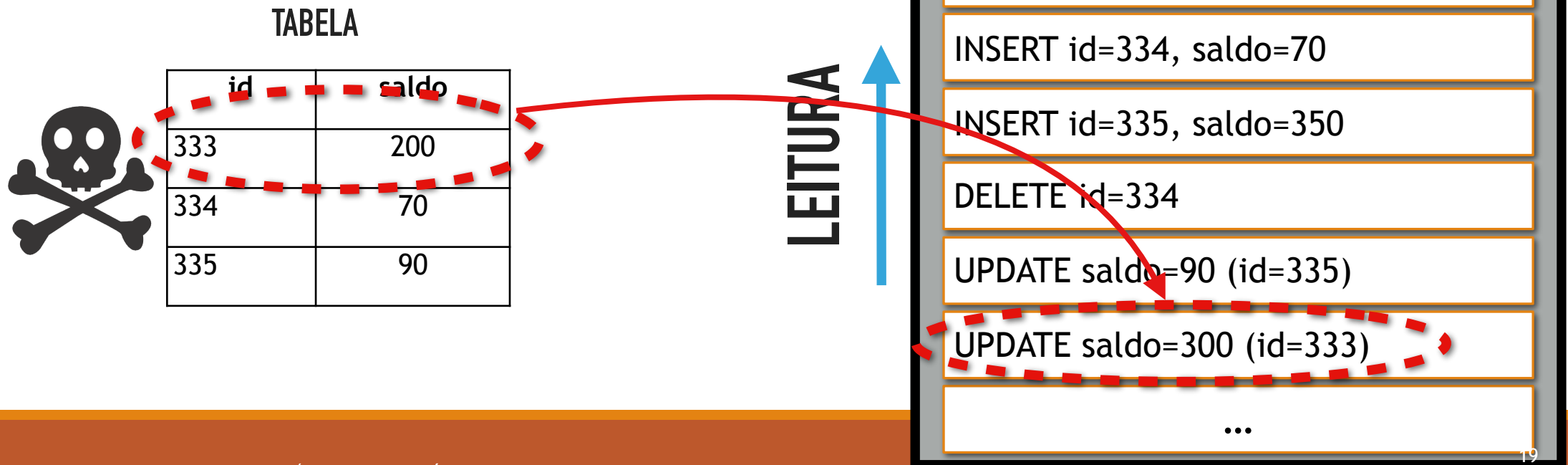
UPDATE saldo=90 (id=335)

UPDATE saldo=300 (id=333)

...

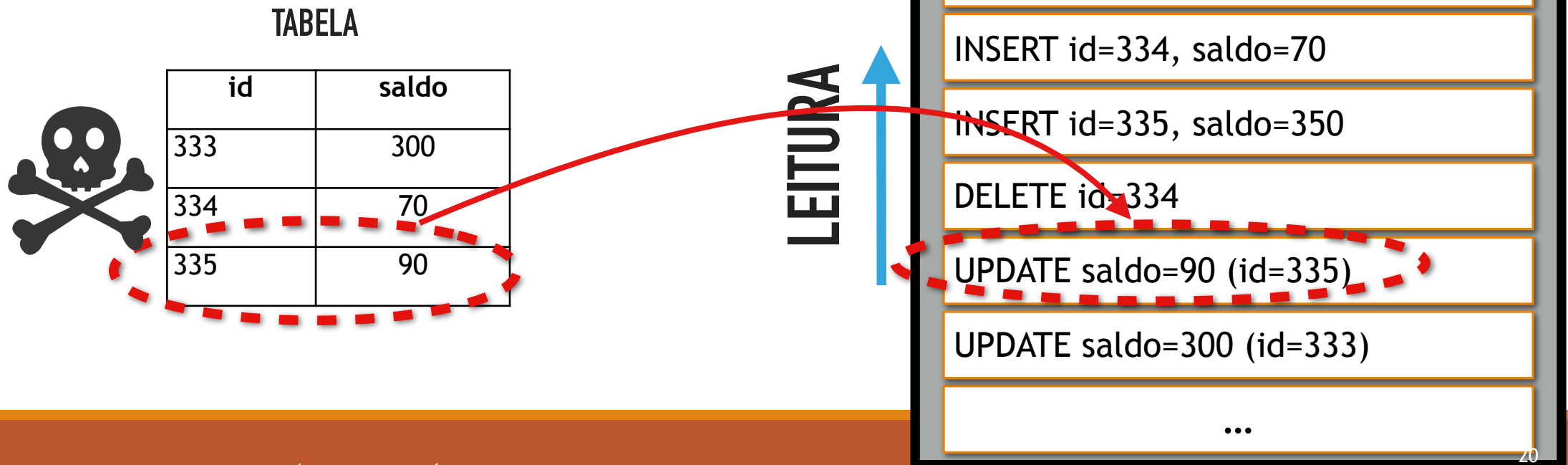
Mecanismo pra manter **Atomicidade**

- **Logging:** “caixa preta do avião”



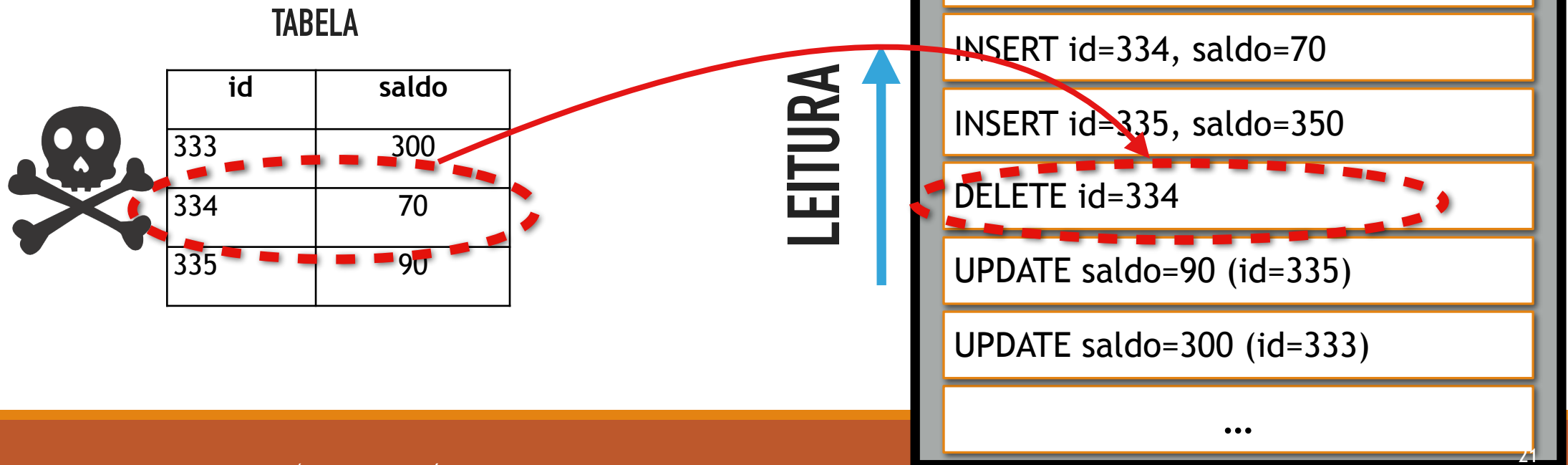
Mecanismo pra manter **Atomicidade**

- **Logging:** “caixa preta do avião”



Mecanismo pra manter **Atomicidade**

- **Logging:** “caixa preta do avião”



Mecanismo pra manter **Atomicidade**

- **Logging:** “caixa preta do avião”

TABELA

id	saldo
333	300
335	90

LEITURA



Arquivo LOG

INSERT id=333, saldo=200

INSERT id=334, saldo=70

INSERT id=335, saldo=350

DELETE id=334

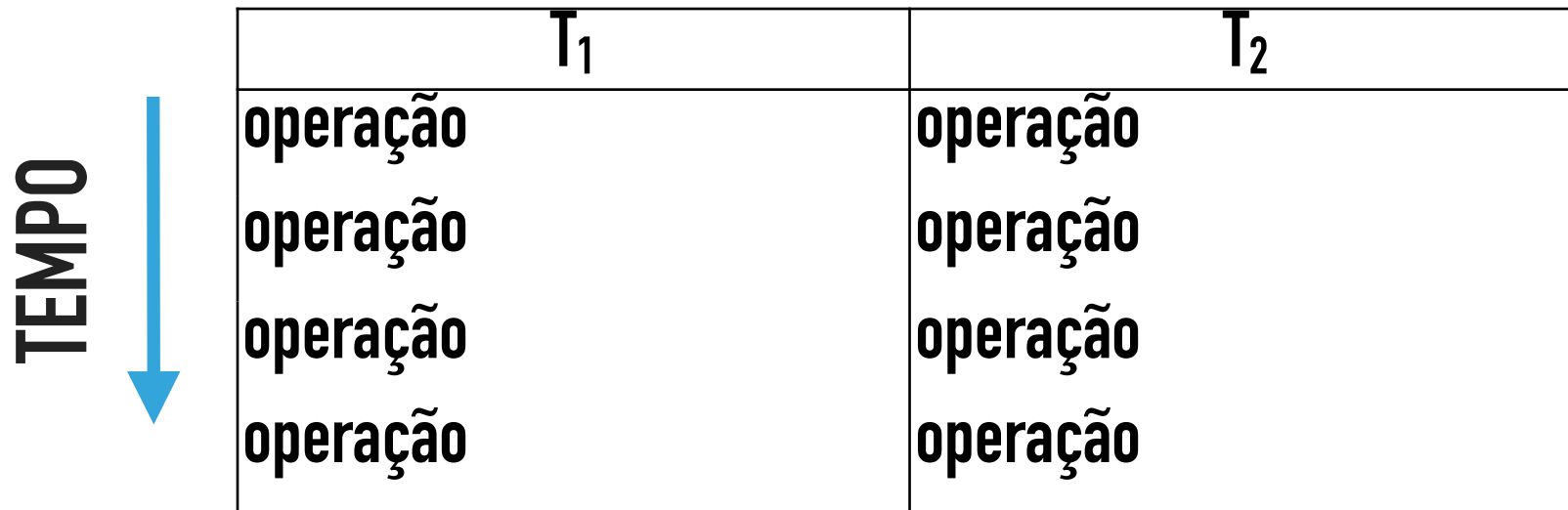
UPDATE saldo=90 (id=335)

UPDATE saldo=300 (id=333)

...

Isolamento

Transações concorrentes não se interferem



	T_1	T_2
TEMPO ↓	operação	operação
	operação	operação
	operação	operação
	operação	operação

Problemas de **Isolamento** (1)

DEPÓSITO DE R\$ 100,00

RETIRADA DE R\$ 100,00

T_1	T_2
LER saldo (id=333) saldo = saldo + 100 ESCREVER saldo	LER saldo (id=333) saldo = saldo - 100 ESCREVER saldo

TEMPO

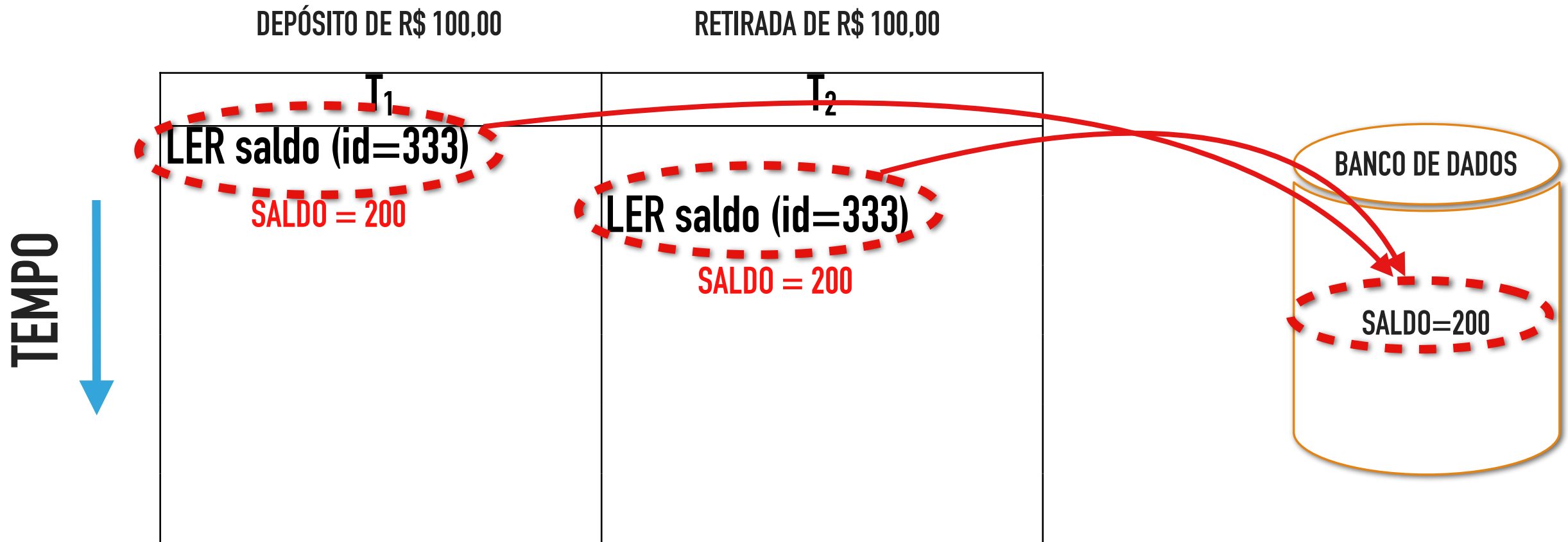


BANCO DE DADOS

SALDO=200

ESPERADO APÓS T_1 E T_2
SALDO=200

Problemas de Isolamento (1)



Problemas de **Isolamento** (1)

DEPÓSITO DE R\$ 100,00

RETIRADA DE R\$ 100,00

T ₁	T ₂
LER saldo (id=333) SALDO = 200 saldo = saldo + 100 SALDO = 300	LER saldo (id=333) SALDO = 200 saldo = saldo - 100

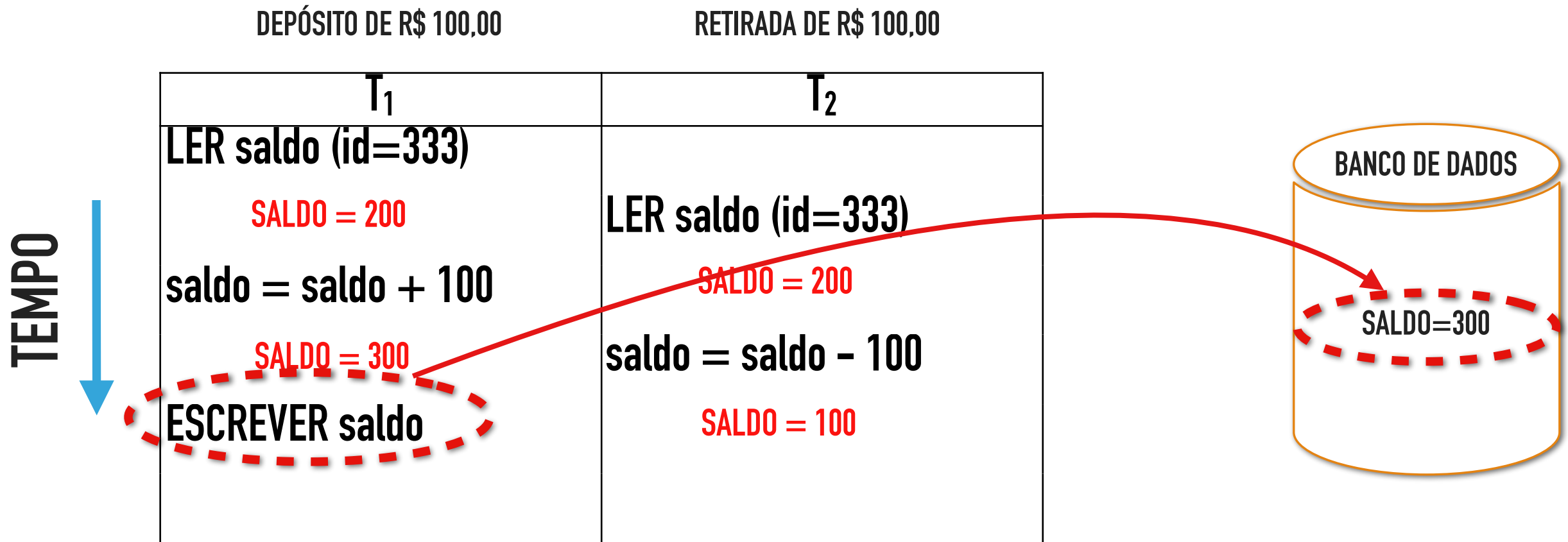
TEMPO



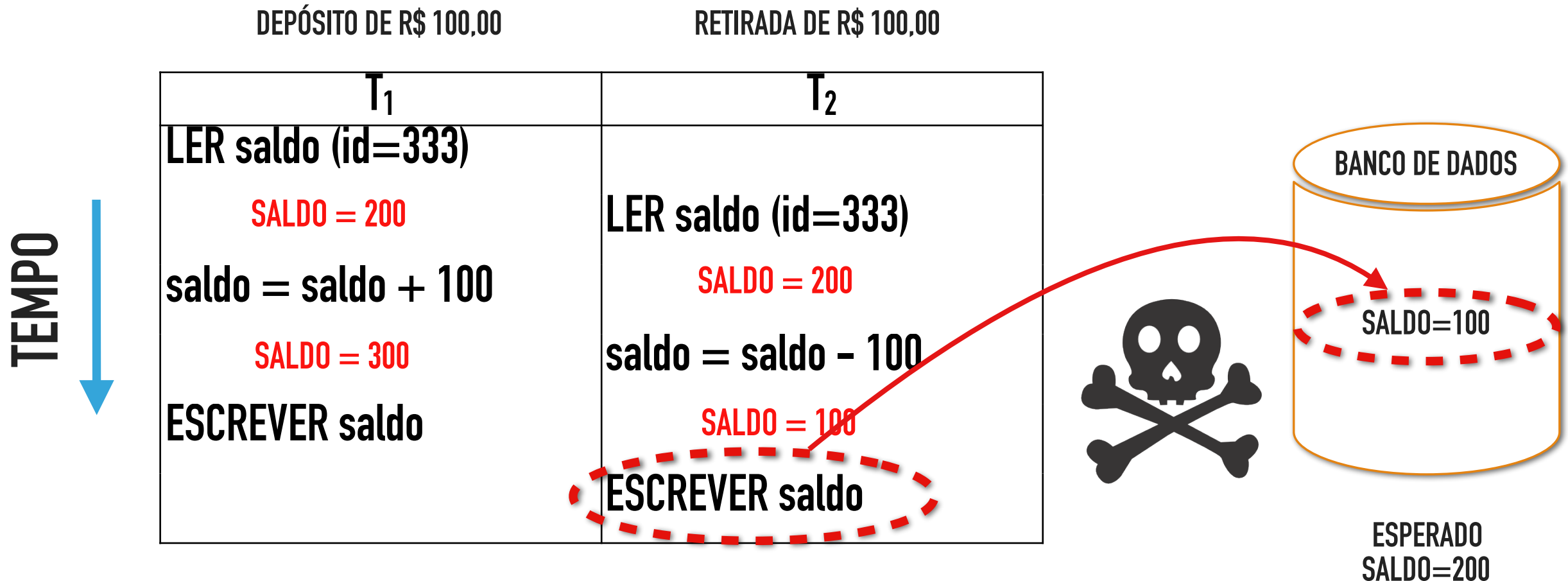
BANCO DE DADOS

SALDO=200

Problemas de Isolamento (1)



Perda de atualização



Problemas de **Isolamento** (2)

DEPÓSITO DE R\$ 100,00

RETIRADA DE R\$ 300,00

T₁	T₂
LER saldo (id=333) saldo = saldo + 100 ESCREVER saldo ABORT	LER saldo (id=333) saldo = saldo - 300 ESCREVER saldo

TEMPO

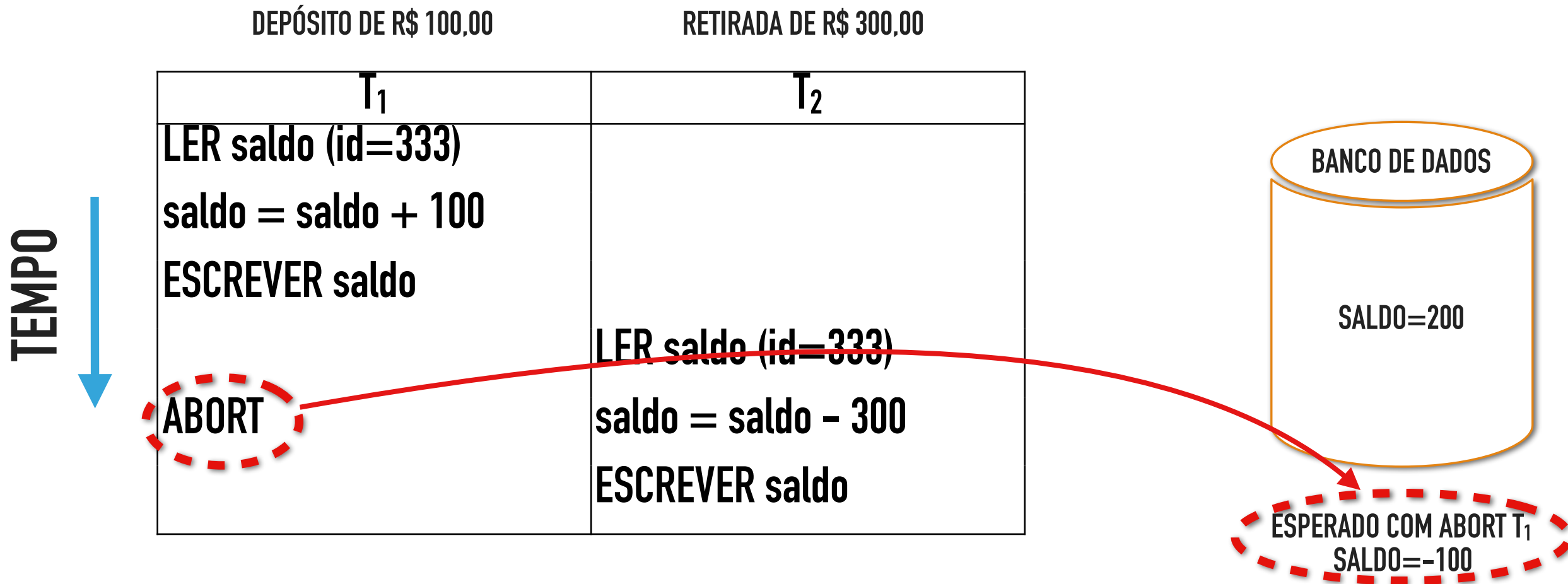


BANCO DE DADOS

SALDO=200

ESPERADO COM ABORT T₁
SALDO=-100

Problemas de Isolamento (2)



Problemas de Isolamento (2)

DEPÓSITO DE R\$ 100,00

RETIRADA DE R\$ 300,00

T ₁	T ₂
LER saldo (id=333) saldo = saldo + 100 ESCREVER saldo	

TEMPO



Problemas de **Isolamento** (2)

DEPÓSITO DE R\$ 100,00

RETIRADA DE R\$ 300,00

T ₁	T ₂
LER saldo (id=333) saldo = saldo + 100 ESCREVER saldo	LER saldo (id=333) SALDO = 300 saldo = saldo - 300

TEMPO



BANCO DE DADOS

SALDO=300

Perda de leitura

DEPÓSITO DE R\$ 100,00

RETIRADA DE R\$ 300,00

T ₁	T ₂
LER saldo (id=333) saldo = saldo + 100 ESCREVER saldo	LER saldo (id=333) saldo = saldo - 300 ESCREVER saldo
ABORT	

TEMPO



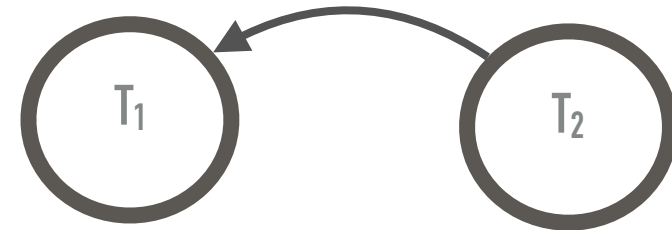
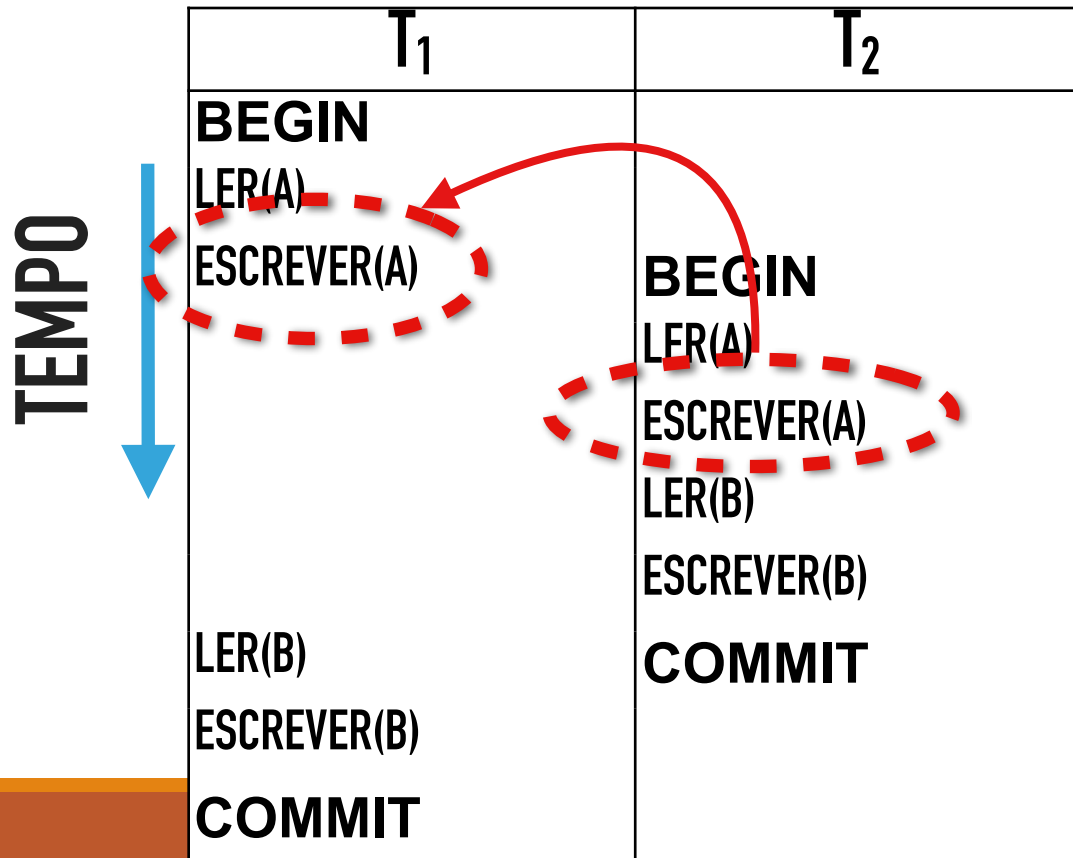
ESPERADO COM ABORT T₁
SALDO=-100

Mecanismo pra manter **Isolamento**

- **Pessimista:** “evitar que aconteça um problema”
- **Otimista:** “tratar o problema quando acontecer”

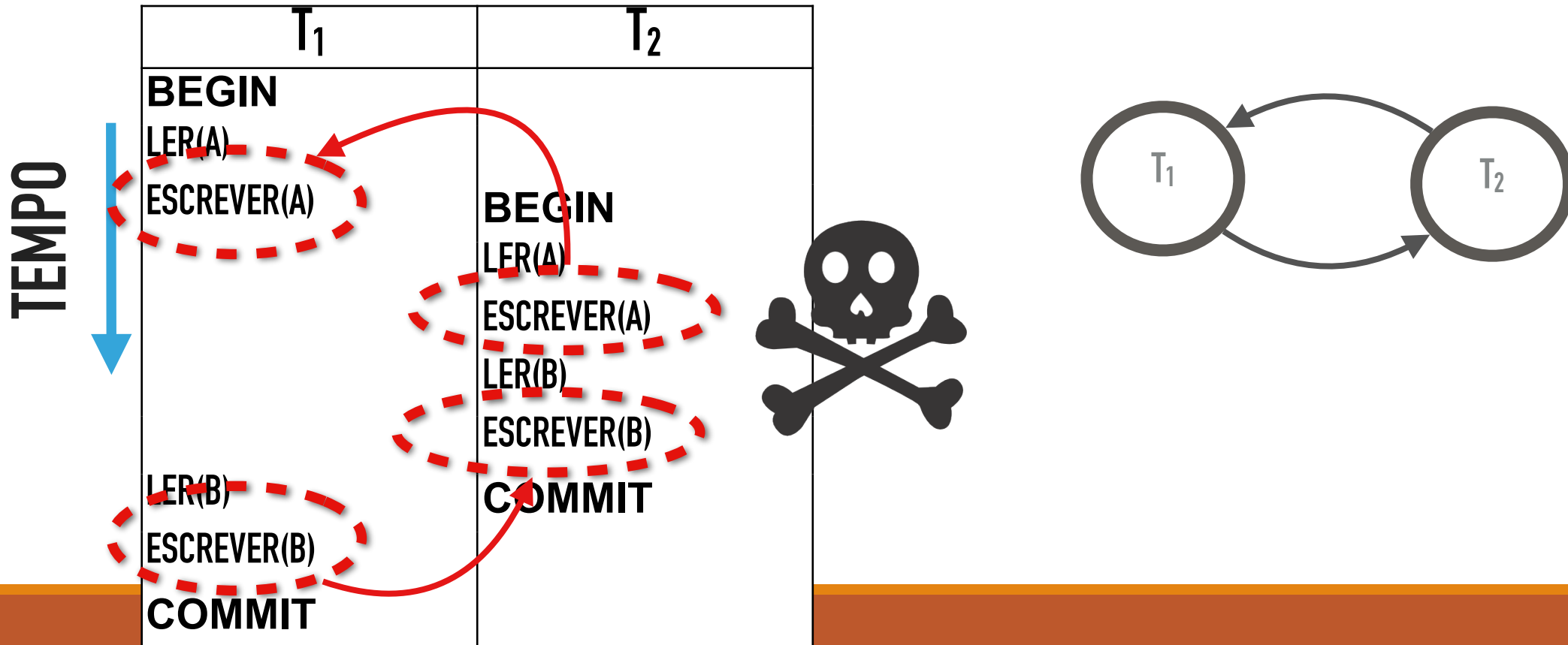
Mecanismo pra manter **Isolamento**

Reorganizar agenda de operações



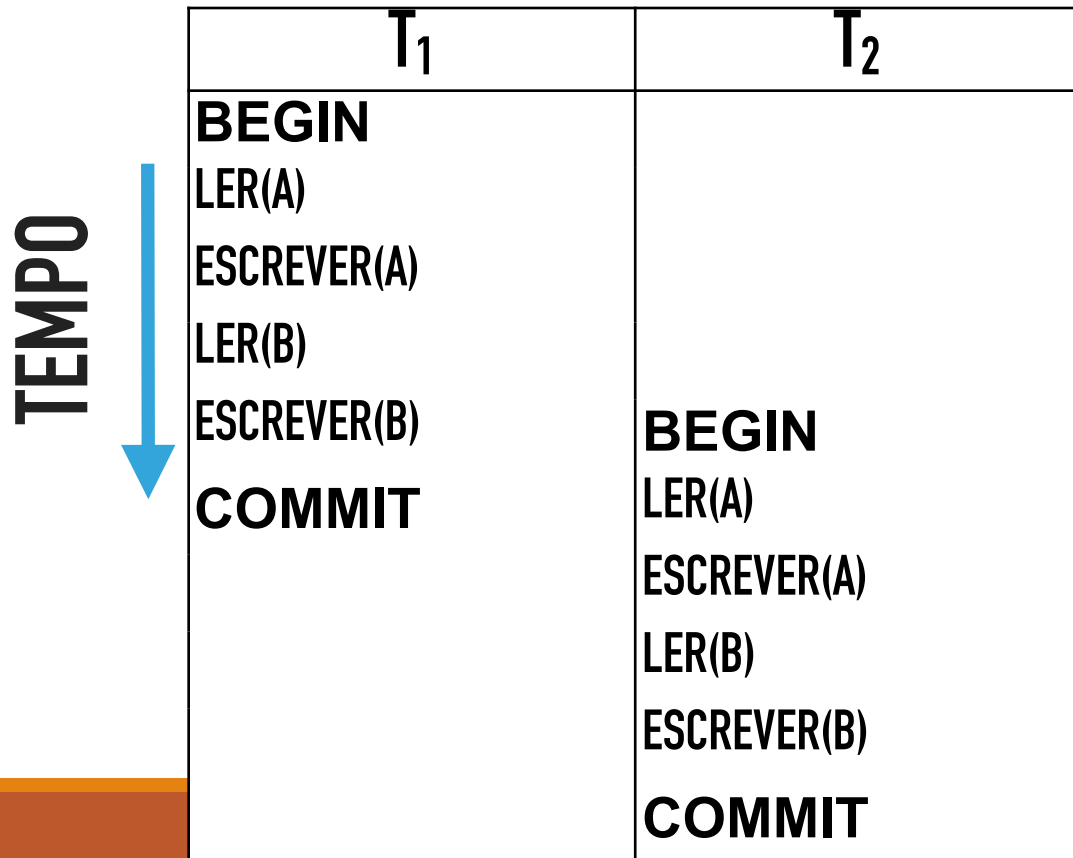
Mecanismo pra manter **Isolamento**

Reorganizar agenda de operações



Mecanismo pra manter **Isolamento**

Reorganizar agenda de operações



Execução serial!!!

Teste de Conflito de Serialidade

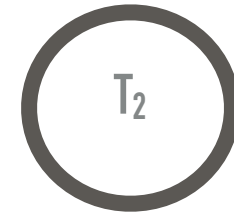
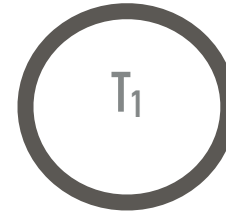
- Criar nó para cada T_i do agendamento S
- Criar aresta $T_i \rightarrow T_j$ para cada $r(x)$ em T_i depois de $w(x)$ em T_j
- Criar aresta $T_i \rightarrow T_j$ para cada $w(x)$ em T_i depois de $r(x)$ em T_j
- Criar aresta $T_i \rightarrow T_j$ para cada $w(x)$ em T_i depois de $w(x)$ em T_j
- Agendamento S é serializável se não existe ciclo

Legenda: $r(x)$ leitura do atributo “x”, ex. $r(\text{saldo})$
 $w(x)$ escrita do atributo “x”, ex. $w(\text{saldo})$

Exemplo

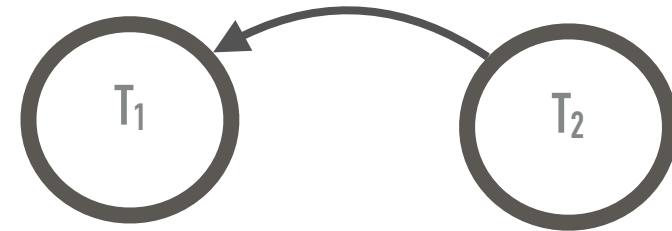
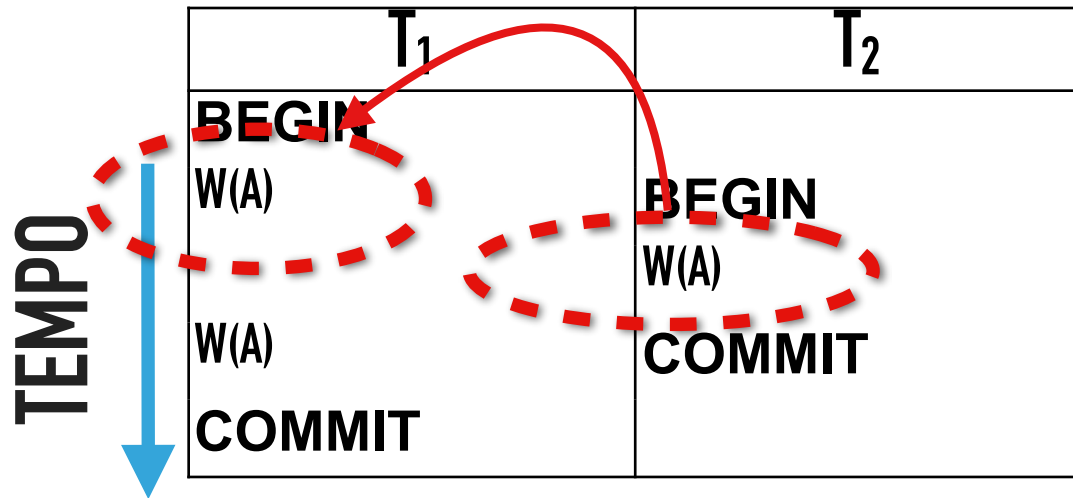
Teste a serialidade do agendamento abaixo

	T_1	T_2
TEMPO ↓	BEGIN	
	W(A)	BEGIN
	W(A)	W(A)
	COMMIT	COMMIT



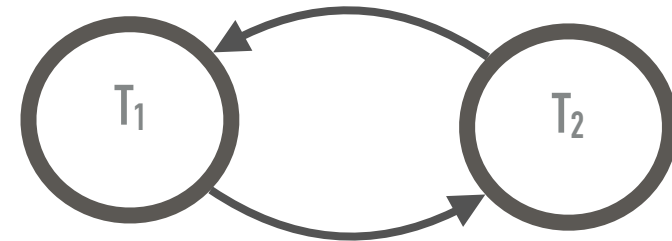
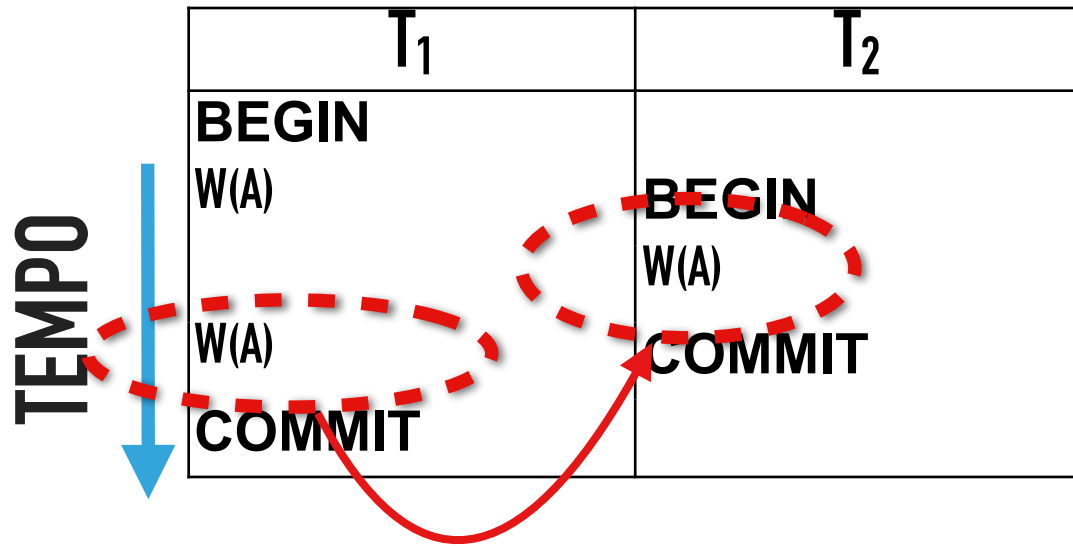
Exemplo

Teste a serialidade do agendamento abaixo



Exemplo

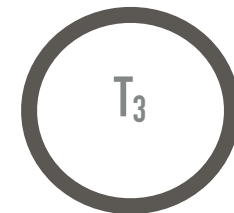
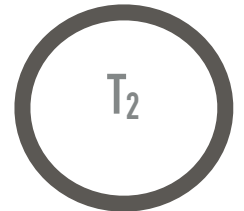
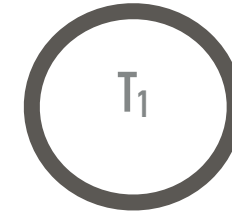
Teste a serialidade do agendamento abaixo



Exercício

Teste a serialidade do agendamento abaixo

	T_1	T_2	T_3
TEMPO ↓	BEGIN R(A)		BEGIN R(A)
	W(A)	BEGIN R(A)	
	COMMIT	COMMIT	W(A) COMMIT



Mecanismo pra manter **Isolamento**

Protocolos para manter a “serialidade” dos agendamentos

- Bloqueio
- MVCC (timestamp)
- Certificação

Consistência do banco de dados

Banco de dados representa o mundo real de forma precisa e segue restrições de integridade.

Transações no futuro observam os efeitos das transações passadas no banco de dados.

Restrições NOT NULL

```
CREATE TABLE DEPARTAMENTO(  
  DEPT_NAME VARCHAR(8),  
  PREDIO VARCHAR(4),  
  ORCAMENTO FLOAT NOT NULL  
);
```

TABELA: DEPARTAMENTO

DEPT_NAME	PREDIO	ORCAMENTO
INFORMATICA	INF	400.000,00
MECANICA	PG	700.000,00
ESTATISTICA	PA	200.000,00

Restrições NOT NULL

```
CREATE TABLE DEPARTAMENTO(  
  DEPT_NAME VARCHAR(8),  
  PREDIO VARCHAR(4),  
  ORCAMENTO FLOAT NOT NULL  
);
```

TABELA: DEPARTAMENTO

DEPT_NAME	PREDIO	ORCAMENTO
INFORMATICA	INF	400.000,00
MECANICA	PG	700.000,00
ESTATISTICA	PA	200.000,00
FISICA	PF	NULL



NAO PERMITIDO PELO BD

Restrições: Cláusula CHECK

```
CREATE TABLE DEPARTAMENTO(  
  DEPT_NAME VARCHAR(8),  
  PREDIO VARCHAR(4),  
  ORCAMENTO FLOAT CHECK (ORCAMENTO > 0.0),  
);
```

TABELA: DEPARTAMENTO

DEPT_NAME	PREDIO	ORCAMENTO
INFORMATICA	INF	400.000,00
MECANICA	PG	700.000,00
ESTATISTICA	PA	200.000,00

Restrições integridade referencial

```
CREATE TABLE DEPARTAMENTO(  
  DEPT_NAME VARCHAR(8) PRIMARY KEY,  
  PREDIO VARCHAR(4),  
  ORCAMENTO FLOAT NOT NULL  
);
```

TABELA: DEPARTAMENTO

DEPT_NAME	PREDIO	ORCAMENTO
INFORMATICA	INF	400.000,00
MECANICA	PG	700.000,00
ESTATISTICA	PA	200.000,00

Restrições integridade referencial

```
CREATE TABLE DEPARTAMENTO(  
  DEPT_NAME VARCHAR(8) PRIMARY KEY,  
  PREDIO VARCHAR(4)
```

```
CREATE TABLE PROFESSOR(  
  ID VARCHAR(5) PRIMARY KEY,  
  NOME VARCHAR(40),  
  SALARIO CHECK (SALARIO > 900.0),  
  FOREIGN KEY DEPT_NAME REFERENCES (DEPARTAMENTO)  
);
```

TABELA: DEPARTAMENTO

DEPT_NAME	PREDIO	ORCAMENTO	
INFORMATICA	INF	400.000,00	
MECANICA			
ESTATISTICA			
	ID	NOME	SALARIO
	1	JOSE	1200,00
	2	MARIA	3000,00
	3	PEDRO	3000,00
	4	PABLO	8000,00

TABELA: PROFESSOR

Restrições integridade referencial

```
CREATE TABLE DEPARTAMENTO(  
  DEPT_NAME VARCHAR(8) PRIMARY KEY,  
  PREDIO VARCHAR(4)
```

```
CREATE TABLE PROFESSOR(  
  ID VARCHAR(5) PRIMARY KEY,  
  NOME VARCHAR(40),  
  SALARIO CHECK (SALARIO > 900.0),  
  FOREIGN KEY DEPT_NAME REFERENCES (DEPARTAMENTO)  
);
```

TABELA: DEPARTAMENTO

DEPT_NAME	PREDIO	ORCAMENTO	
INFORMATICA	INF	400.000,00	
MECANICA			
ESTATISTICA			
ID	NOME	SALARIO	DEPT_NAME
1	JOSE	1200,00	ESTATISTICA
2	MARIA	3000,00	INFORMATICA
3	PEDRO	3000,00	INFORMATICA
4	PABLO	8000,00	MECANICA
5	MARCUS	8000,00	FISICA



NAO PERMITIDO PELO BD

TABELA: PROFESSOR

Restrições: Asserções

Asserções são generalizações das restrições anteriores.

```
CREATE TABLE DEPARTAMENTO(  
    DEPT_NAME VARCHAR(8) PRIMARY KEY,  
    PREDIO VARCHAR(4),  
    ORCAMENTO  
);
```

```
CREATE ASSERTION ORCAMENTO_CONST CHECK(  
    NOT EXISTS (SELECT DEPT_NAME  
        FROM DEPARTAMENTO  
        WHERE  
        ORCAMENTO < 0.0)  
);
```

Durabilidade das transações

Atualizações de **transações falhas** não são escritas no banco de dados.

Registros de arquivos de log (e shadow paging) podem ser usados para garantir a durabilidade.

Transações no mundo real?



1800 txn/sec



1 milhão txn/sec



9 bilhões txn/dia

INDUSTRIA 4.0

Processamento de
Big Data



MARMELADA NO RING, FONTE: ESTADAO

Aula #8 - Transações e concorrência

EDUARDO CUNHA DE ALMEIDA