ISCAS 中国科学院软件研究所学术年会
暨重点实验室科技活动周
2025 第十届
学术论文

关系型数据库中的隔离异常检测方法

杨睿，崔紫玉，窦文生，高钰，宋建森，谢旭东，魏峻

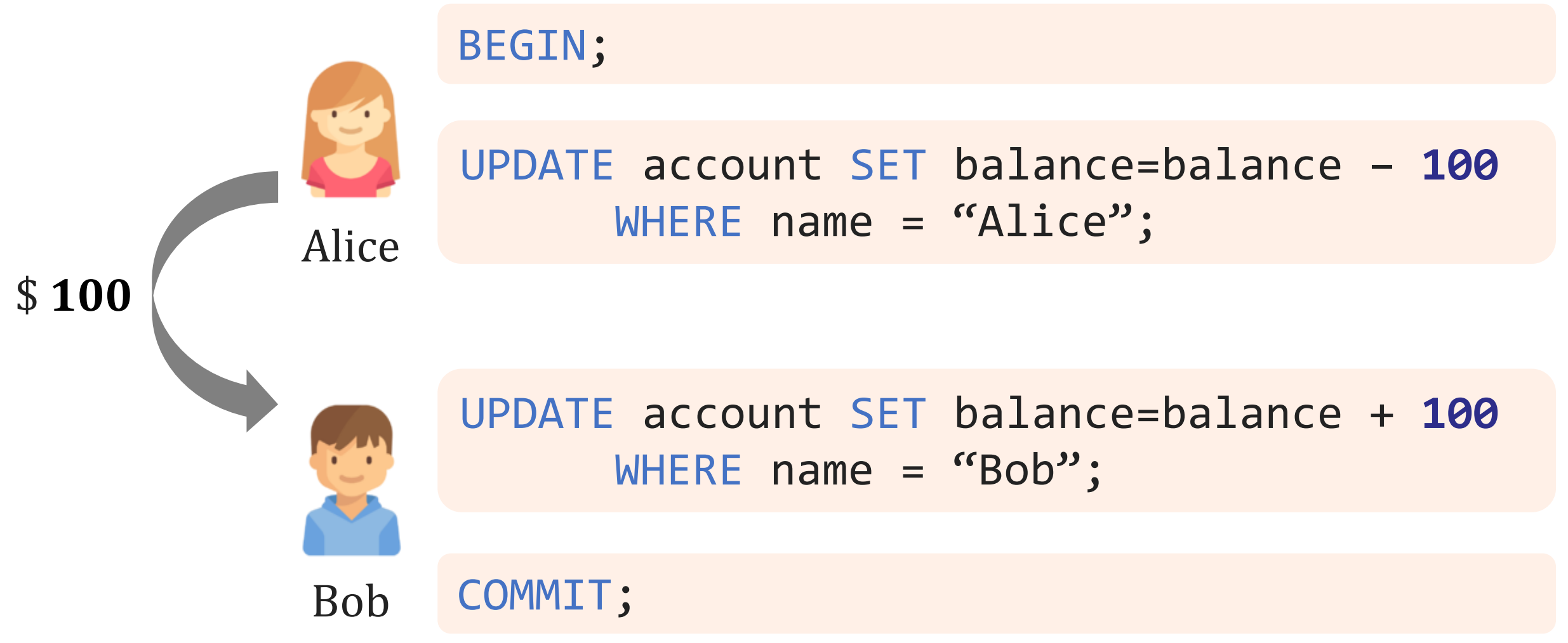
Detecting Isolation Anomalies in Relational DBMSs

The 34th ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA 2025)

联系方式：杨睿，yangrui22@otcaix.iscas.ac.cn

Transaction

□ **Relational DBMSs utilize transactions to ensure data consistency and integrity**



```
BEGIN;
UPDATE account SET balance=balance – 100
      WHERE name = “Alice”;
UPDATE account SET balance=balance + 100
      WHERE name = “Bob”;
COMMIT;
```

Isolation Anomaly

□ **Incorrect transaction implementations can cause isolation anomalies**

□ **Isolation anomalies cause incorrect query results, incorrect database states and constraint violation**

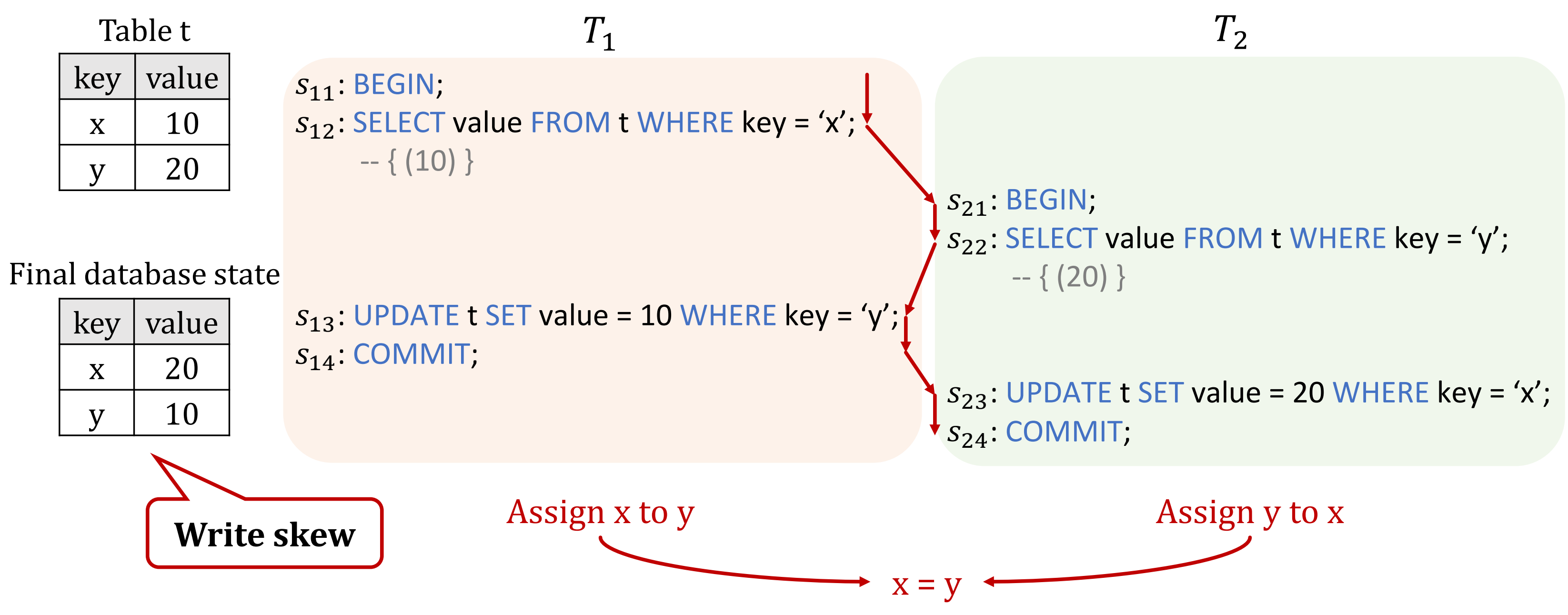


Table t

key	value
x	10
y	20

Final database state

key	value
x	20
y	10

T_1

```
s11: BEGIN;
s12: SELECT value FROM t WHERE key = ‘x’;
       -- { (10) }
s13: UPDATE t SET value = 10 WHERE key = ‘y’;
s14: COMMIT;
```

T_2

```
s21: BEGIN;
s22: SELECT value FROM t WHERE key = ‘y’;
       -- { (20) }
s23: UPDATE t SET value = 20 WHERE key = ‘x’;
s24: COMMIT;
```

Key Challenge

□ **Cannot infer which rows the transaction reads or writes**

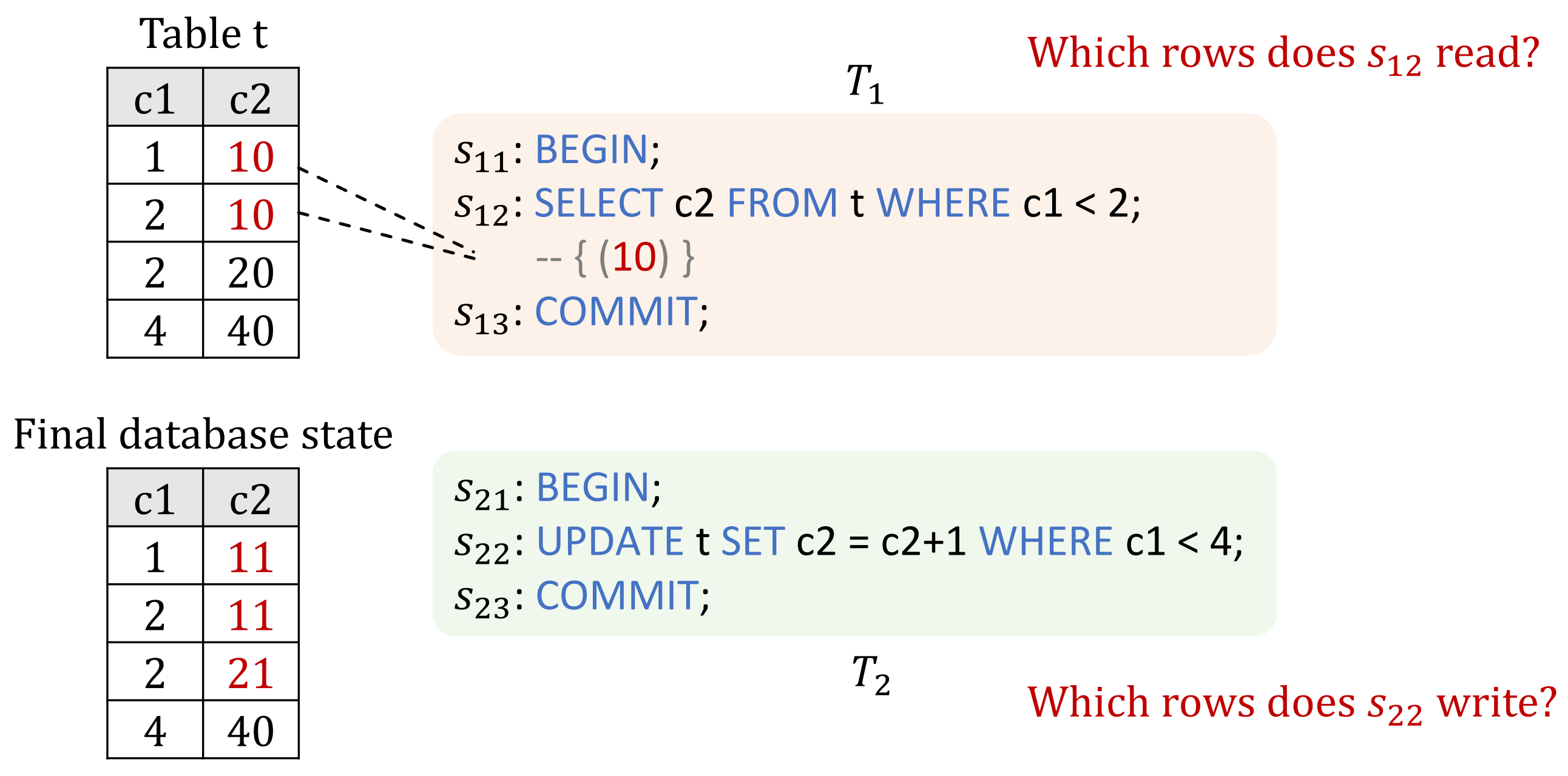


Table t

c1	c2
1	10
2	10
2	20
4	40

Final database state

c1	c2
1	11
2	11
2	21
4	40

T_1

```
s11: BEGIN;
s12: SELECT c2 FROM t WHERE c1 < 2;
       -- { (10) }
s13: COMMIT;
```

Which rows does s_{12} read?

T_2

```
s21: BEGIN;
s22: UPDATE t SET c2 = c2+1 WHERE c1 < 4;
s23: COMMIT;
```

Which rows does s_{22} write?

Key Idea

□ **Uniquely identify each row by an auxiliary column *rowId***

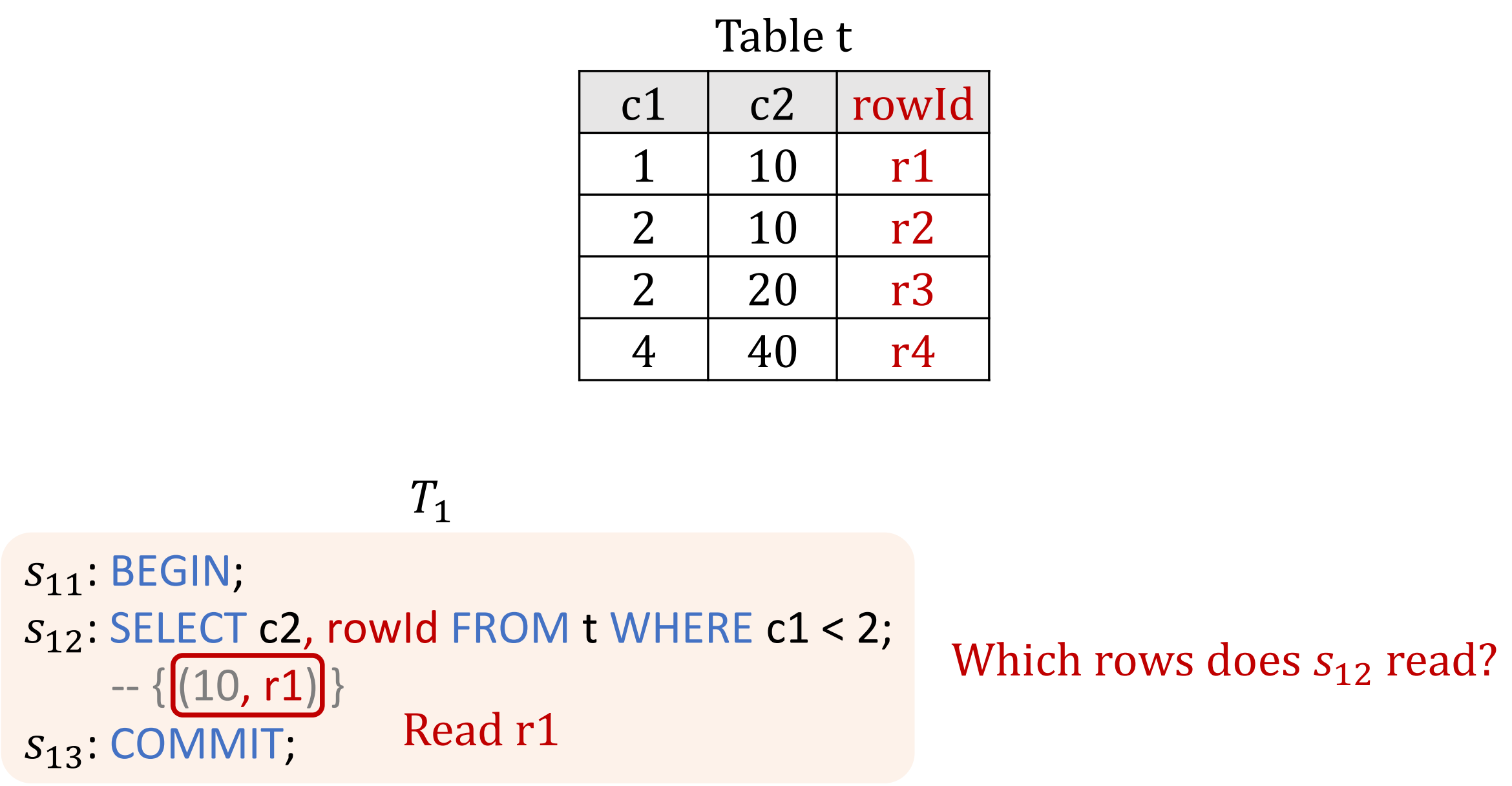


Table t

c1	c2	rowId
1	10	r1
2	10	r2
2	20	r3
4	40	r4

T_1

```
s11: BEGIN;
s12: SELECT c2, rowId FROM t WHERE c1 < 2;
       -- {(10, r1)}
s13: COMMIT;
```

Read r1

Which rows does s_{12} read?

Key Idea

□ **Maintain a transaction list that modify a row by an auxiliary column *wList***

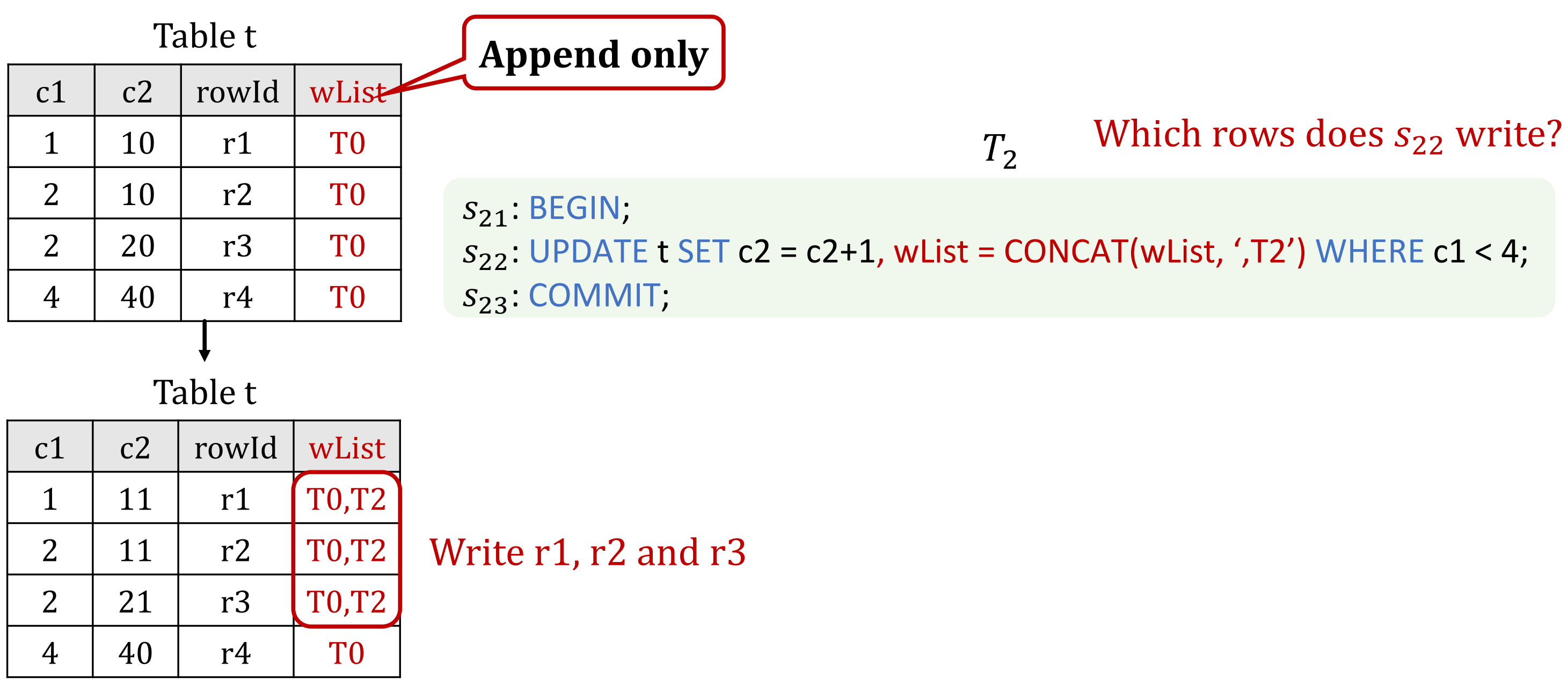


Table t

c1	c2	rowId	wList
1	10	r1	T0
2	10	r2	T0
2	20	r3	T0
4	40	r4	T0

Append only

T_2

```
s21: BEGIN;
s22: UPDATE t SET c2 = c2+1, wList = CONCAT(wList, ‘,T2’) WHERE c1 < 4;
s23: COMMIT;
```

Which rows does s_{22} write?

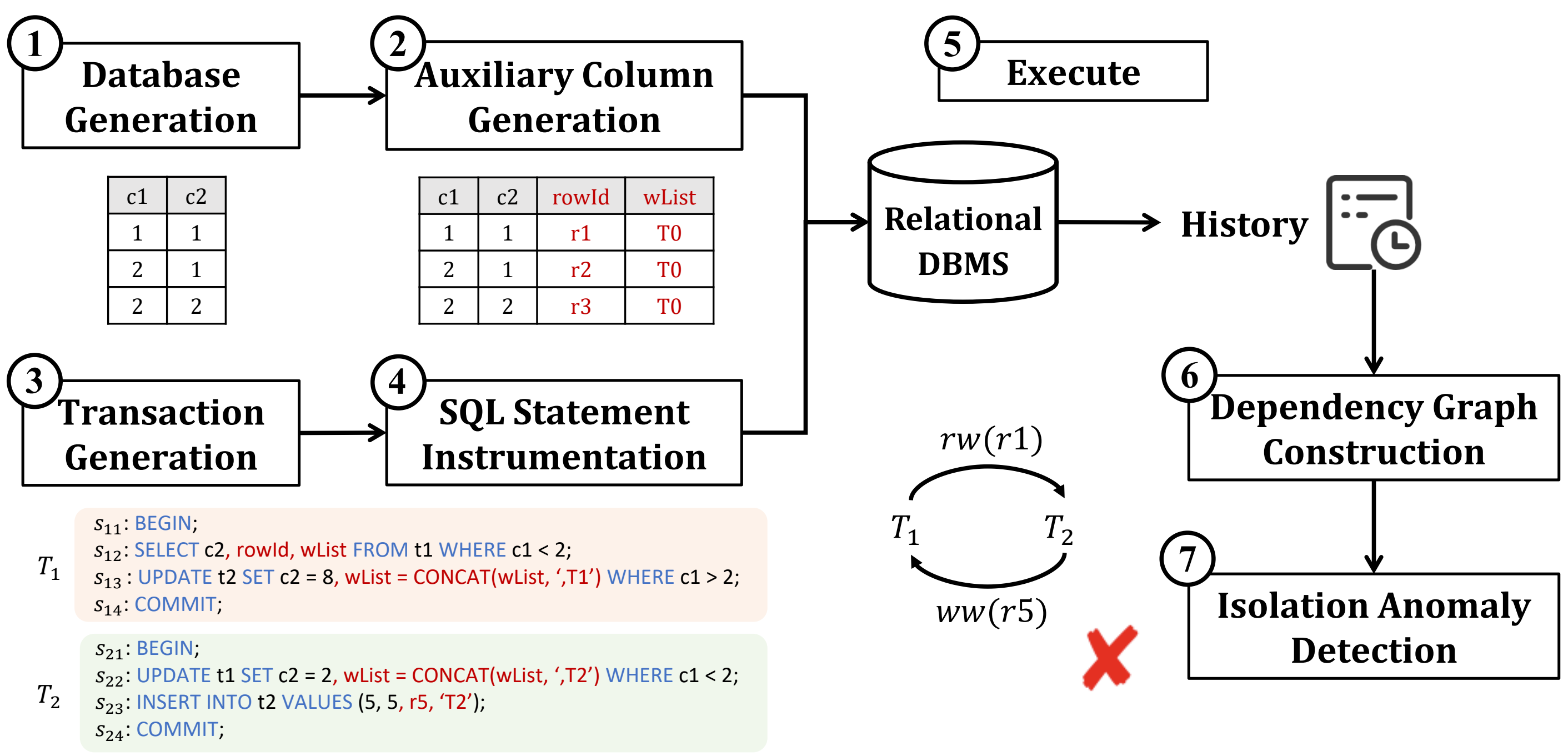
Table t

c1	c2	rowId	wList
1	11	r1	T0,T2
2	11	r2	T0,T2
2	21	r3	T0,T2
4	40	r4	T0

Write r1, r2 and r3

IsoRel Overview

□ **A black-box isolation checker for relational DBMSs**



c1	c2
1	1
2	1
2	2

c1	c2	rowId	wList
1	1	r1	T0
2	1	r2	T0
2	2	r3	T0

T_1

```
s11: BEGIN;
s12: SELECT c2, rowId, wList FROM t1 WHERE c1 < 2;
s13 : UPDATE t2 SET c2 = 8, wList = CONCAT(wList, ‘,T1’) WHERE c1 > 2;
s14: COMMIT;
```

T_2

```
s21: BEGIN;
s22: UPDATE t1 SET c2 = 2, wList = CONCAT(wList, ‘,T2’) WHERE c1 < 2;
s23: INSERT INTO t2 VALUES (5, 5, r5, ‘T2’);
s24: COMMIT;
```

Anomaly Detection Results

□ **IsoRel reveals 48 unique isolation anomalies**

Relational DBMS	#Txn	Isolation Level				#Isolation Anomaly		Lost Update	Read-Write Skew	Write Skew
		RU	RC	RR	SER	Total	Unique			
MySQL	5774	0	0	57	0	57	14	4	4	6
PostgreSQL	2536	-	0	8	0	8	6	0	0	6
MariaDB	4515	0	0	92	0	92	14	4	4	6
CockroachDB	4090	-	-	-	0	0	0	0	0	0
TiDB	5802	-	0	168	-	168	14	4	4	6
Total	22717	0	0	325	0	325	48	12	12	24

Comparison with Other Checkers

□ **Only 2 key-value anomalies can be detected by existing isolation checkers**

Detection Capabilities of Existing Isolation Checkers

	IsoRel	Elle	Cobra	PolySI
Isolation Anomaly	48	2	0	2