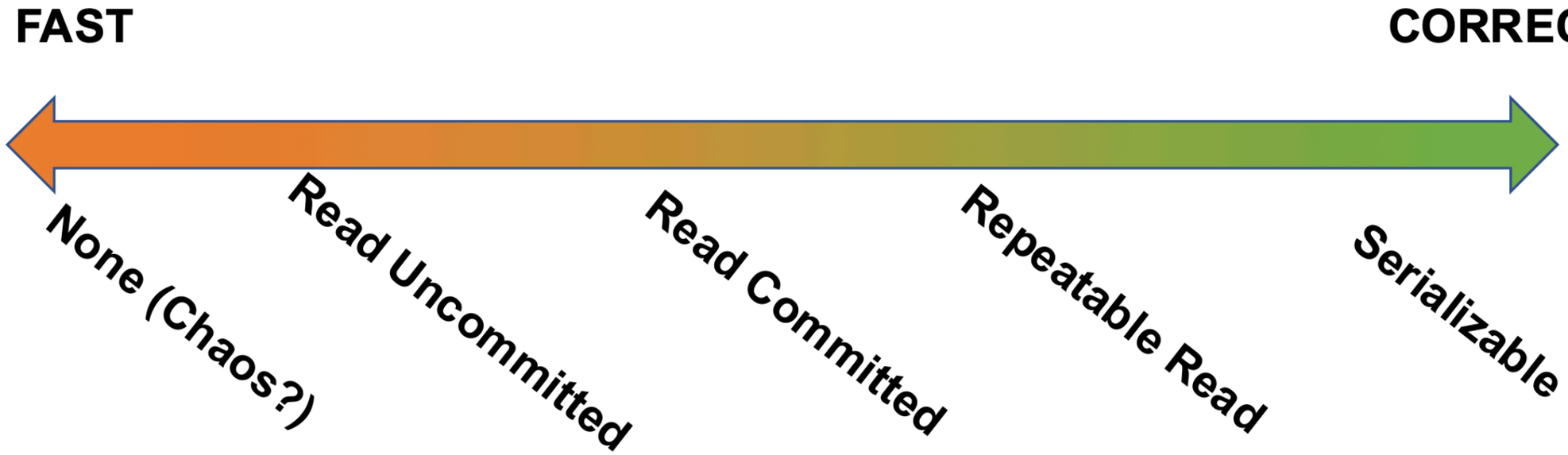
Isolation Levels & Application Programming

Remy Wang, 5/6/25

isolation levels



isolation levels

avoid certain types of problems:

dirty read/inconsistent read

lost update

unrepeatable read

dirty/inconsistent read

seeing updates from uncommitted TXN

time

Manager wants to balance project budgets

CEO wants to check company balance

SELECT SUM(budget) ...

-$10mil from project A

✅

+$7mil to project B

❌

+$3mil to project C

✅

lost update

update overwritten by another TXN

A = B = 100

W(A, 200)	
W(B, 0)	
	W(B, 200)
	W(A, 0)

A + B = 200 ✅

A = B = 100

W(A, 200)	
	W(B, 200)
W(B, 0)	
	W(A, 0)

A = B = 0 ❌

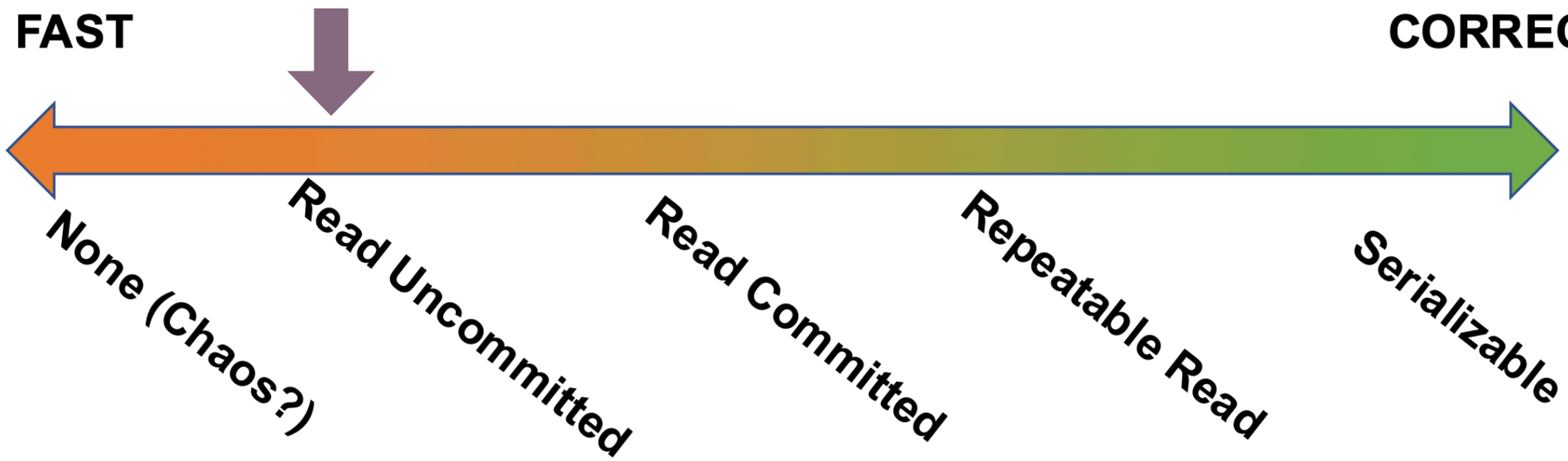
unrepeatable read

two reads give different results

A = 100

	R(A)	A = 100
W(A, 0)		
	R(A)	A = 0

isolation levels



read uncommitted

Strict 2PL for writes

no locks at all for reads!

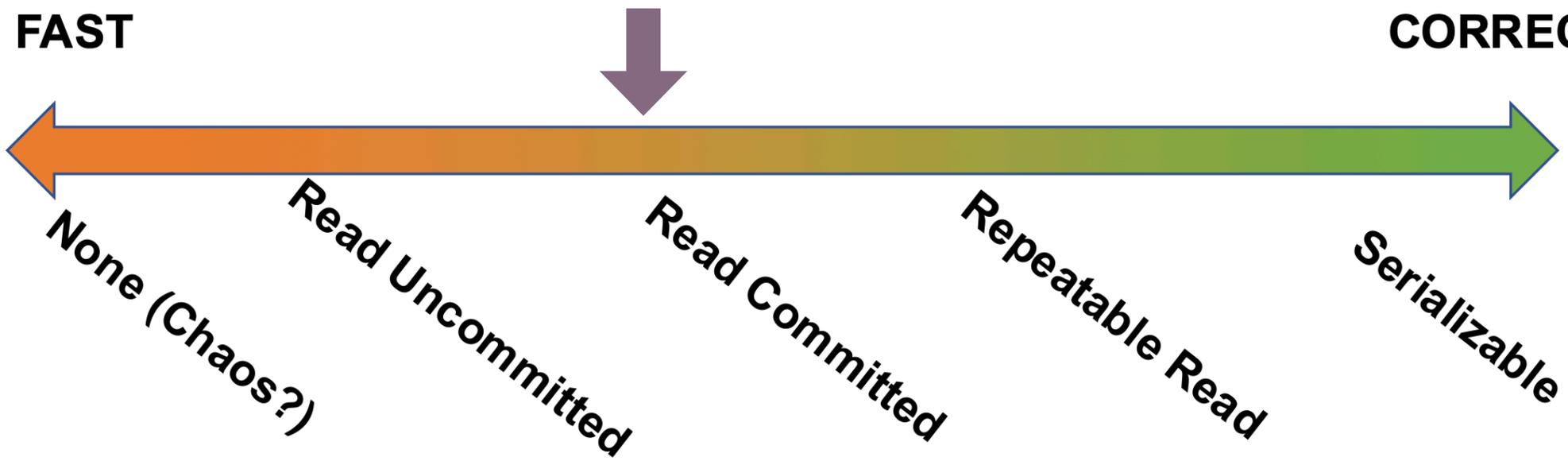
read uncommitted

very fast reads

assumes few/no writes

read accuracy is not critical

isolation levels



read committed

Strict 2PL for writes

on-demand read locks (not 2PL!)

lock → R → unlock

no dirty reads, possible unrepeatable reads

no dirty reads

W(A, 0)	
	~~L(A), R(A)~~
COMMIT, U(A)	

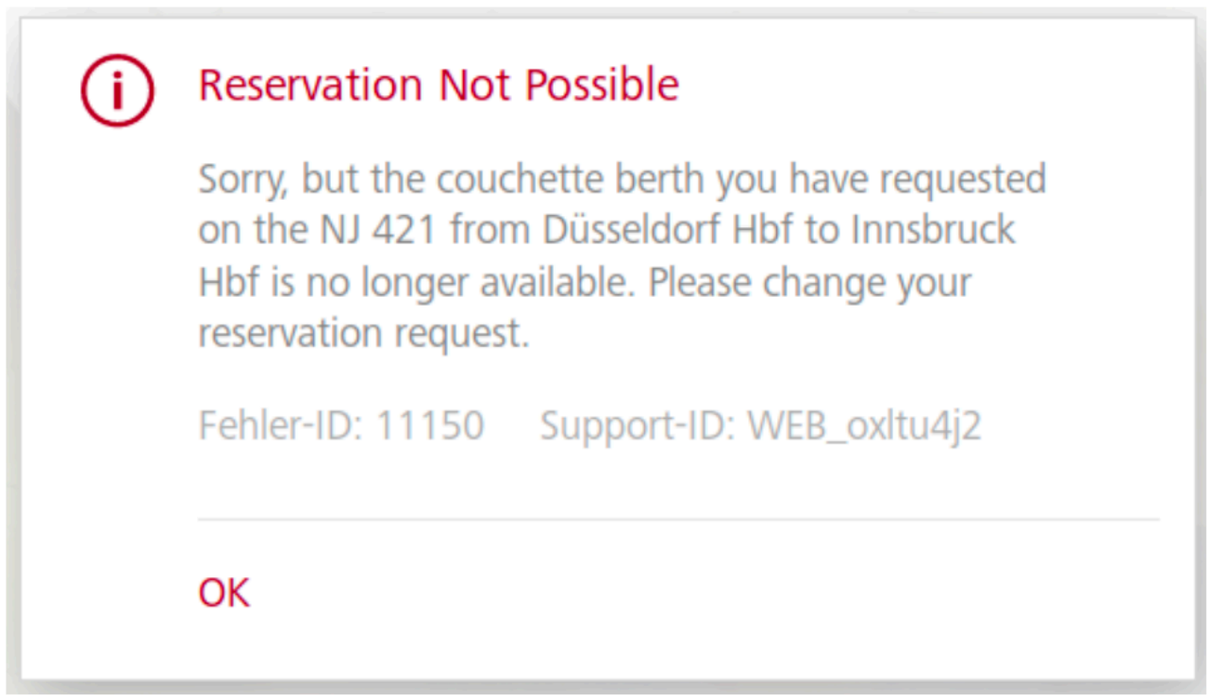
possible unrepeatable reads

	L(A),R(A),U(A)
L(A),W(A),U(A)	
	L(A),R(A),U(A)

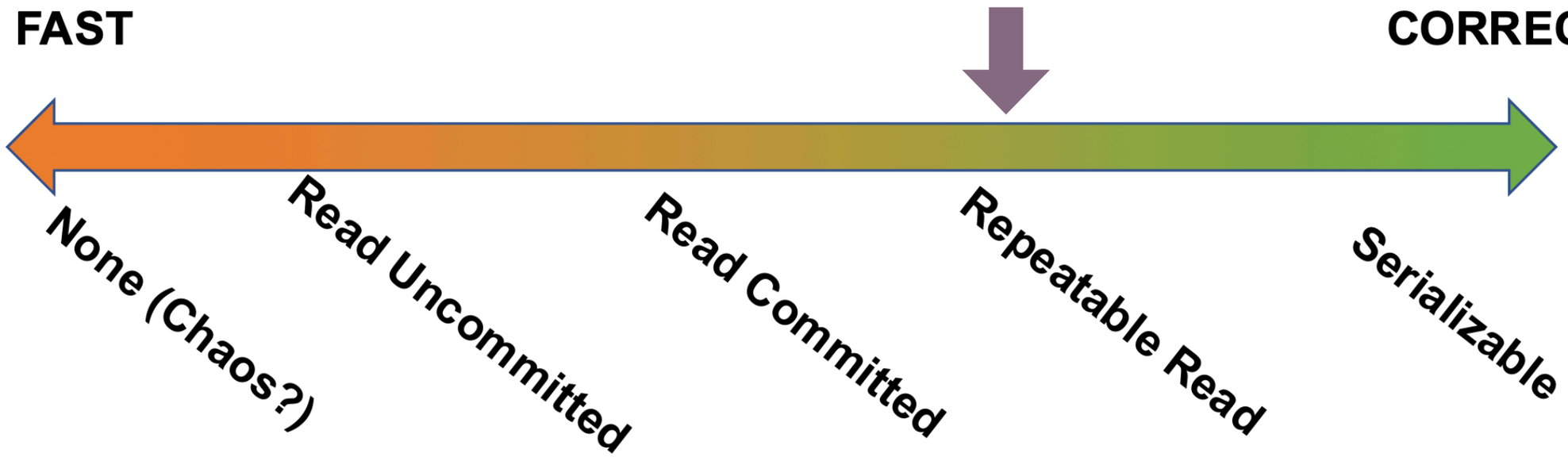
read committed

guarantee read result is valid at *some* point

useful for online shops



isolation levels



repeatable read

Strict 2PL write locks

Strict 2PL read locks

conflict serializable!

but not serializable???

The Phantom Menace

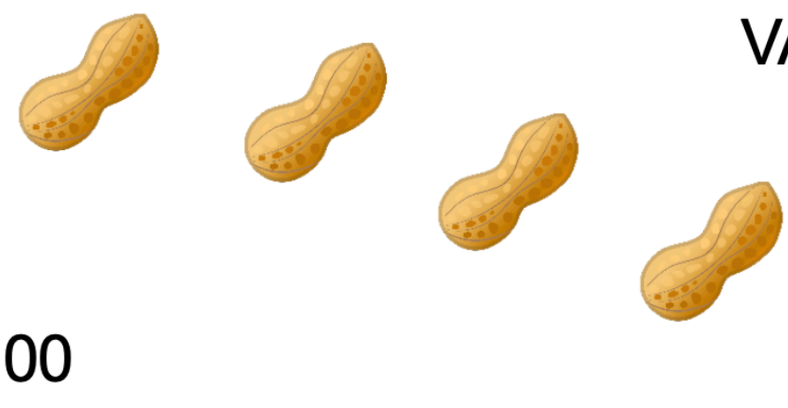
- Same read has more rows
- Asset checking scenario:

Accountant wants to check company assets

```
SELECT *
FROM products
WHERE price < 10.00
```

Warehouse catalogs new products

```
INSERT INTO Products
VALUES ('nuts', 10, 8.99)
```



time

```
SELECT *
FROM products
WHERE price < 10.00
```

the phantom problem

SELECT *	R(A), R(B)	
		W(C) INSERT
SELECT *	R(A), R(B), R(C)	

the phantom problem

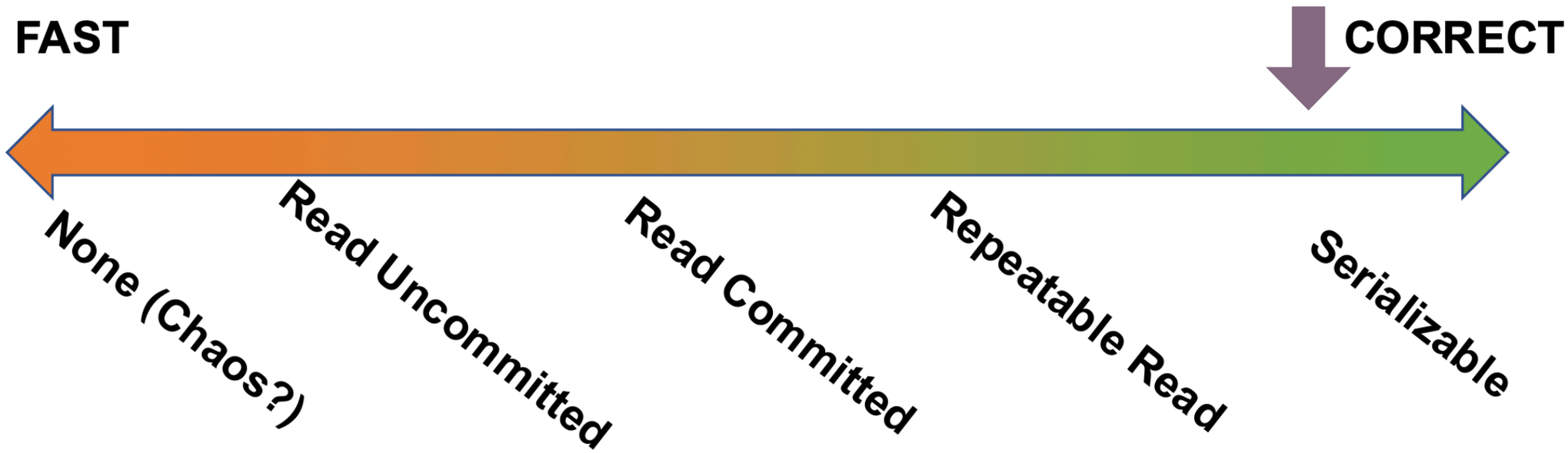
conflict serializable → serializable **w/o inserts**

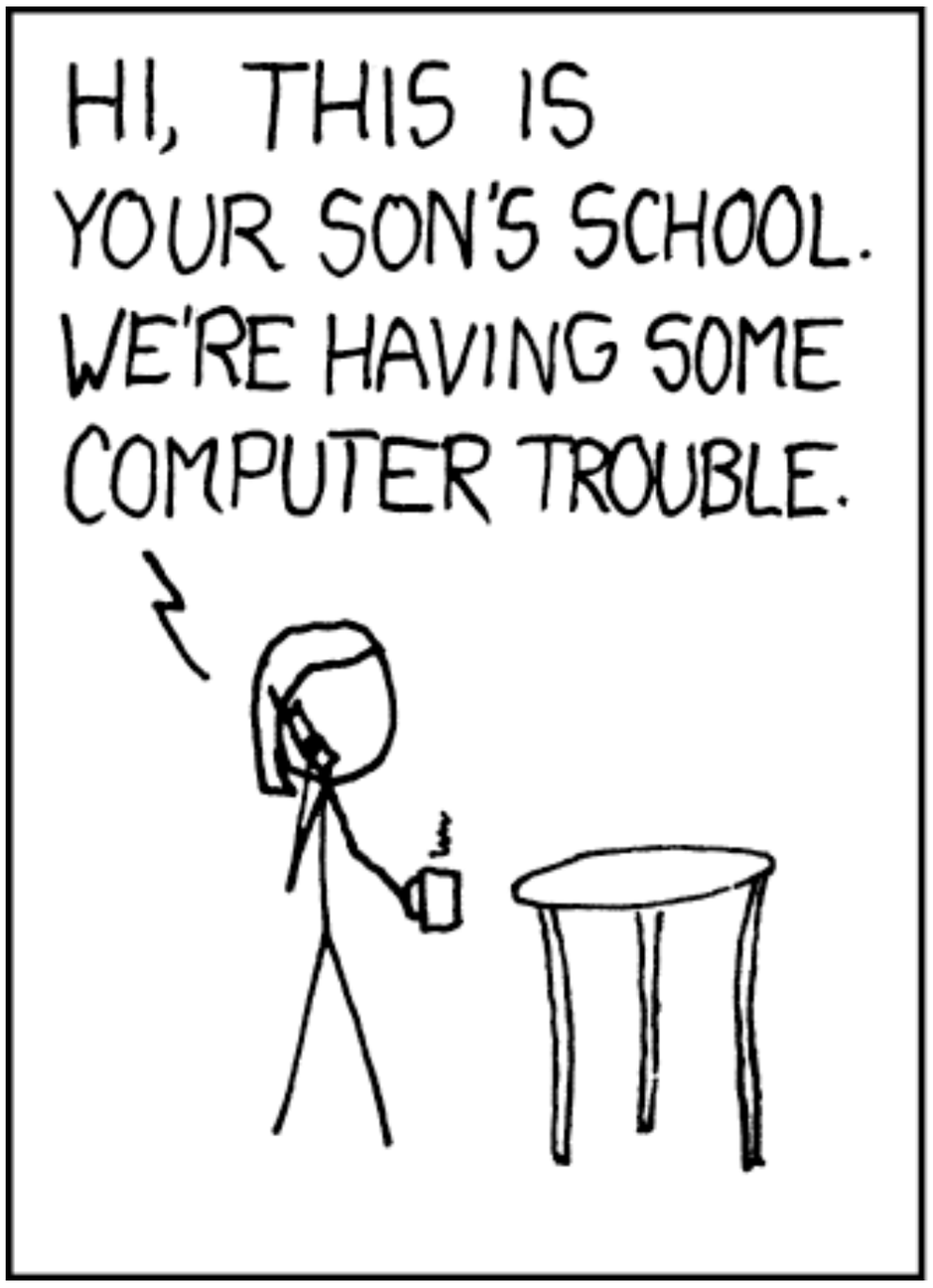
solution: lock entire table

the phantom problem

SELECT *	**L(T)**, R(T)	
		~~L(T), W(C)~~
SELECT *	R(T), C, U(T)	

isolation levels

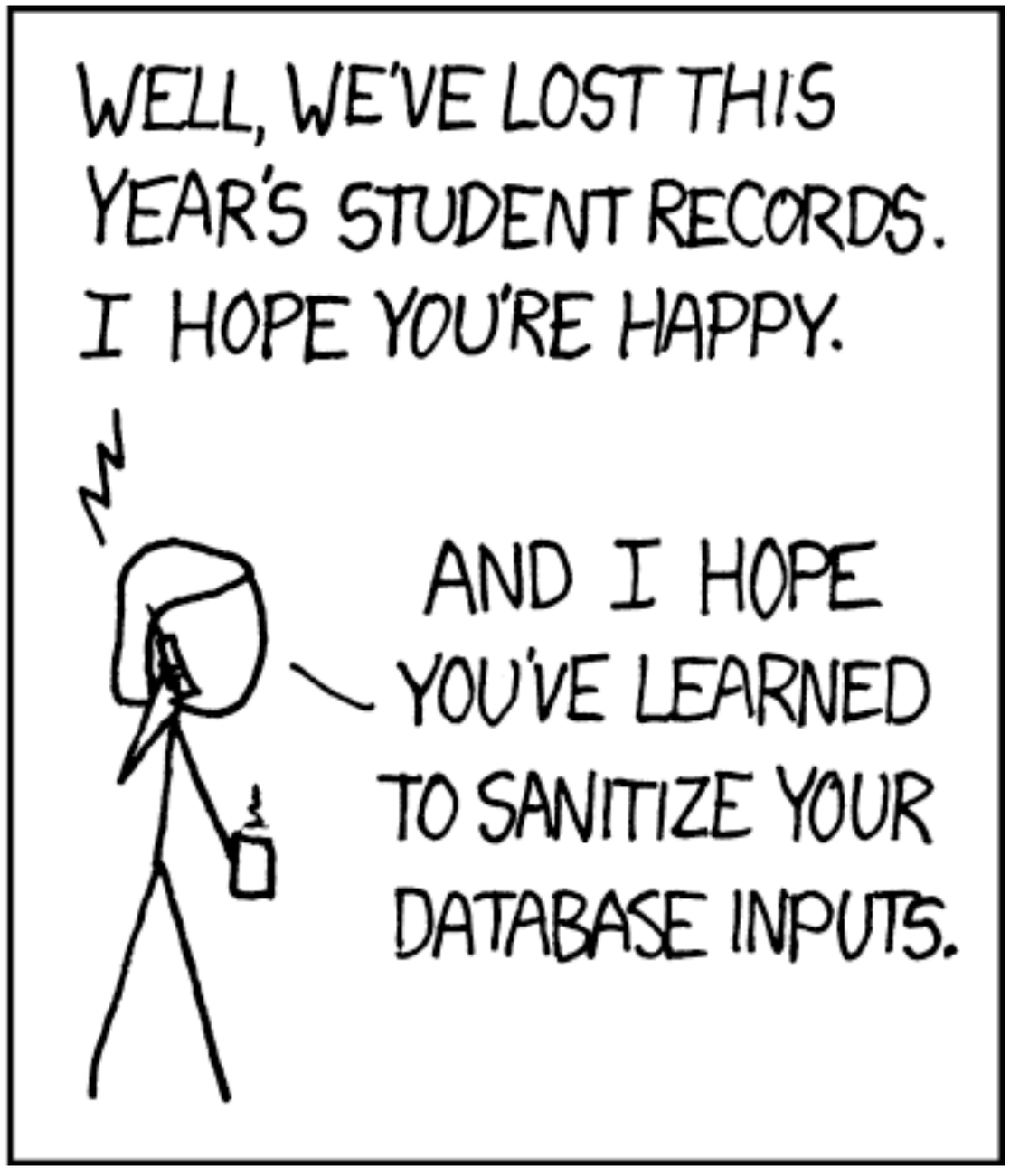



HI, THIS IS
YOUR SON'S SCHOOL.
WE'RE HAVING SOME
COMPUTER TROUBLE.


OH, DEAR - DID HE BREAK SOMETHING?
IN A WAY-

DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?

OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.


WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

```
INSERT INTO students (id, 'name');
```

id	name
111	…
123	Robert'); DROP TABLE Students;--
145	…

```
INSERT INTO students (id, 'Robert');
DROP TABLE Students;--');
```

```
SELECT * FROM user
 WHERE name = 'x'
   AND password = 'y'
```

```
attack: return all users
hint:   use UNION / OR
```

solution to SQL injection

parameterized queries

prepared statements

access control

passwords

NEVER store passwords in plain text!

hash functions

deterministic

$$x = y \implies h(x) = h(y)$$

low collision

$$x \neq y \implies h(x) \neq h(y)$$

$$(P(h(x) = h(y)) \approx 0)$$

easy to compute

$$f(x) \in O(1)$$

hard to invert

$$f^{-1}(x) \in O(2^N)$$

passwords

NEVER store passwords in plain text!

store hash instead

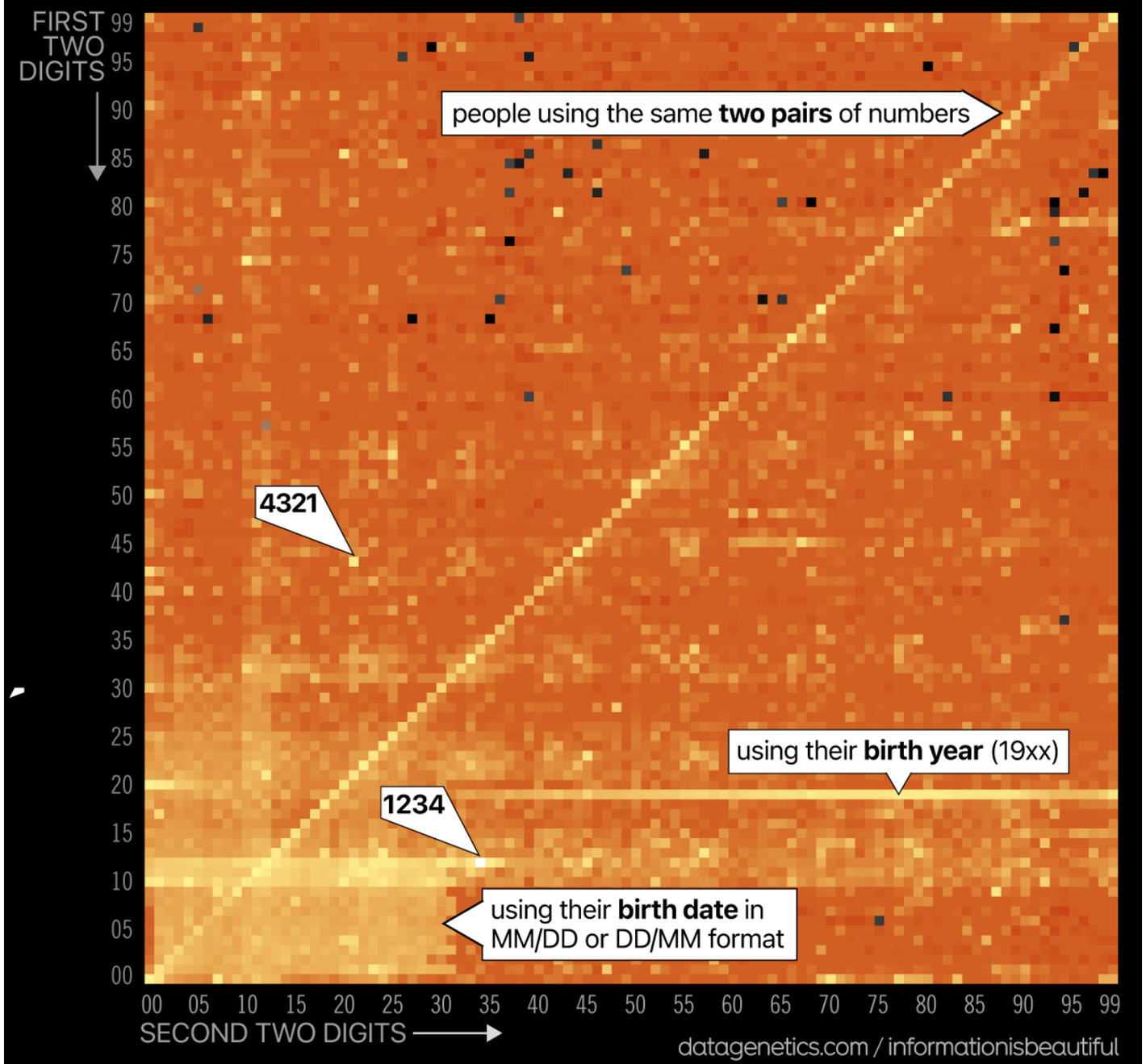
Username	Password
bobtheninja246	password
xDragonSlayerx	asdf
annabelle2001	password
lamamaster123	ilovefish
theSQLexpert234	j62ld12446
seahawksrule12	j62ld12446

Username	HashedPassword
bobtheninja246	3da541…
xDragonSlayerx	bfd361…
annabelle2001	3da541…
lamamaster123	5baa61…
theSQLexpert234	ca8612…
seahawksrule12	ca8612…

```
SELECT * FROM user
 WHERE name = 'x'
AND pw_hash = hash('y')
```

Most to Least Common 4-Digit PIN Numbers

sourced from multiple data breaches



most common

1234	0000	7777	2000	2222	9999	5555	1122	8888	2001
1111	1212	1004	4444	6969	3333	6666	1313	4321	1010

27% of all PIN numbers

least common

8557	8438	9539	7063	6827	0859	6793	0738	6835	8093
9047	0439	8196	6093	7394	9480	8398	7637	9629	8068

datagenetics.com / informationisbeautiful

Username	Password
bobtheninja246	password
xDragonSlayerx	asdf
annabelle2001	password
lamamaster123	ilovefish
theSQLexpert234	j62ld12446
seahawksrule12	j62ld12446

Username	HashedPassword
bobtheninja246	3da541…
xDragonSlayerx	bfd361…
annabelle2001	3da541…
lamamaster123	5baa61…
theSQLexpert234	ca8612…
seahawksrule12	ca8612…

people are bad at passwords!

reuse pw across sites

commonly used pw

```
salt = getRandom()
salted_pw_hash = hash(pw, salt)
```

Username	Password
bobtheninja246	password
xDragonSlayerx	asdf
annabelle2001	password
lamamaster123	ilovefish
theSQLexpert234	j62ld12446
seahawksrule12	j62ld12446

Username	Salt	HashedPassword
bobtheninja246	17	7a4959…
xDragonSlayerx	m9	59438a…
annabelle2001	23	4c812e…
lamamaster123	q7	3e0e04…
theSQLexpert234	k3	dcfea6…
seahawksrule12	ji	e840fc…

privacy laws

HIPPA

GDPR

FERPA

common approach

"de-ID": remove personal identifiable information

common approach

"de-ID": remove personal identifiable information

Latanya Sweeney: **not enough**!

Re-identification of Mass. Gov. William Weld

```
health_rec(zip, DoB, sex,
           diagnosis, procedure, ...)
```

```
voter(name, party, ...,
      zip, DoB, sex, ...)
```

```
("Weld", "R.", ...,
 12345, 02-30, M, ...)
```

Cambridge, MA Voter Data ($20)

Name	ZIP	Sex	Bday
…	…	…	…
W. Weld	12345	M	Feb 30
…	…	…	…

⋈

Anon. Insurance Data for Researchers

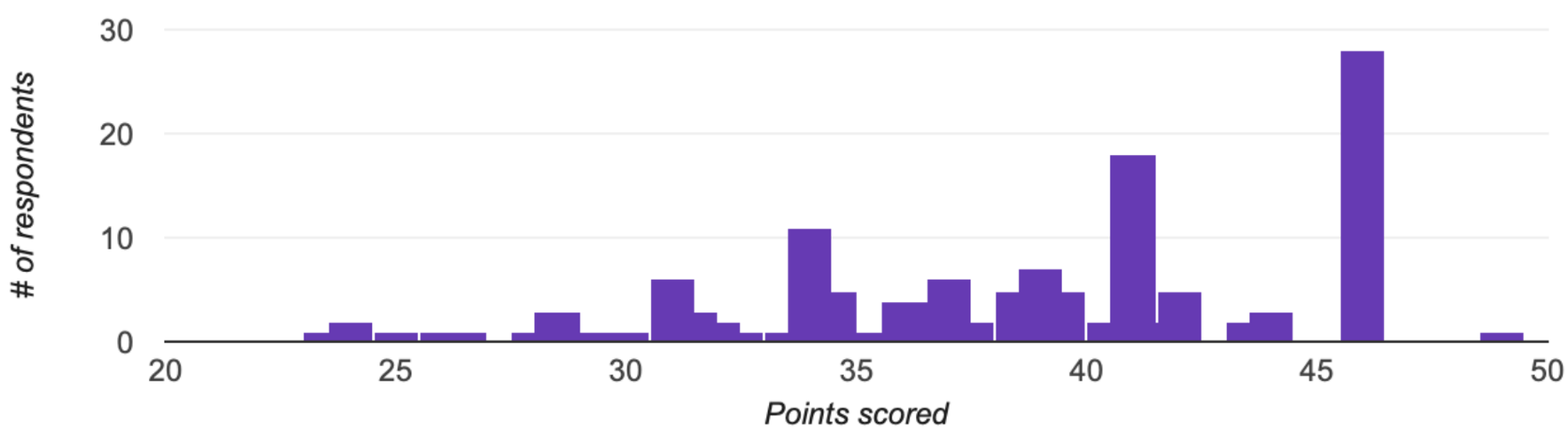
ZIP	Sex	Bday	MedInfo
…	…	…	…
12345	M	Feb 30	Afluenza
…	…	…	…

Average
38.45 / 49 points

Median
39 / 49 points

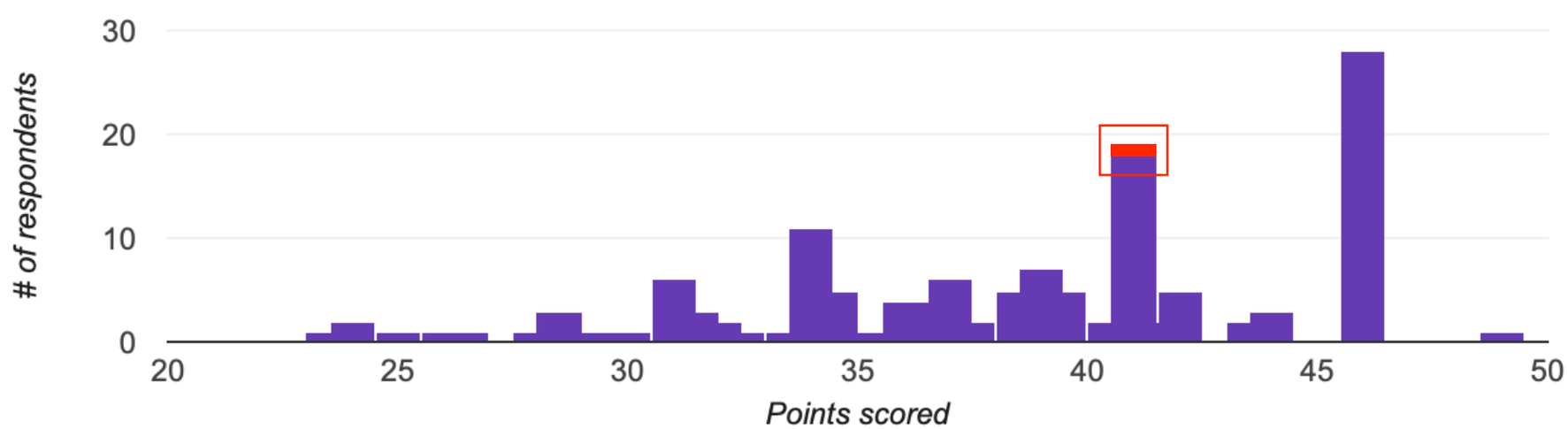
Range
23.5 - 49 points

Total points distribution



Average	Median	Range
38.45 / 49 points	39 / 49 points	23.5 - 49 points

Total points distribution



Analysis $\mathcal{M}$ satisfies differential privacy if...

For all D_1 and D_2 which **differ in one individual's data**...

Answer **A** and answer **B** are **indistinguishable**

Probability of seeing output O on input D_1

Probability of seeing output O on input D_2

$$\frac{\mathbf{Pr}[\mathcal{M}(D_1) \in O]}{\mathbf{Pr}[\mathcal{M}(D_2) \in O]} \leq e^{\varepsilon}$$

Indistinguishability: bounded ratio of probabilities