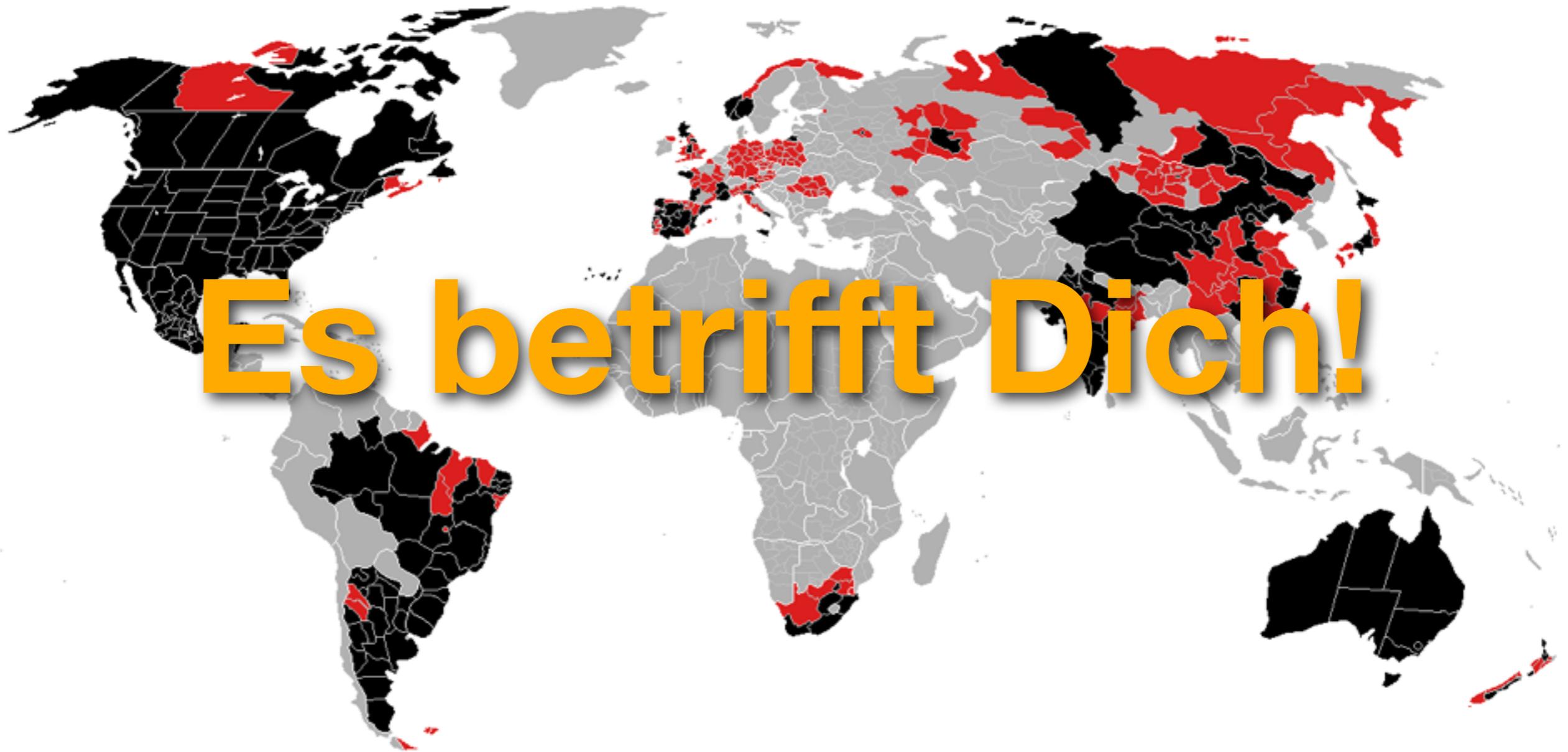


Volkskrankheit „Stiefmütterliche Indizierung“



Take-away #1: Pandemisches Ausmaß



Es betrifft Dich!

(Symbol-Bild; keine echten Daten)

http://upload.wikimedia.org/wikipedia/commons/c/c7/2009_world_subdivisions_flu_pandemic.png

Takeaway #2: Durch Erfolg verursacht



SEPARATION OF CONCERNS

Don't let your plumbing code pollute your software.

Takeaway #3: Nicht Dein Fehler



Das Problem

Unsachgemäße Index-Nutzung

Problem: Unsachgemäße Index-Nutzung

“A very common cause of performance problems is lack of proper indexes or the use of queries that are not using existing indexes.”

— Buda Consulting

http://www.budaconsulting.com/Portals/52677/docs/top_5_tech_brief.pdf

Problem: Unsachgemäße Index-Nutzung

“A **very common** cause of performance problems is lack of proper indexes or the use of queries that are not using existing indexes.”

—Buda Consulting

http://www.budaconsulting.com/Portals/52677/docs/top_5_tech_brief.pdf

Quantifizierung des Problem

Percona White Paper:

*Gründe für Performance-Probleme,
die einen Produktionsausfall verursachen:*

38% bad SQL

15% schema and indexing

Quantifizierung des Problememes

Umfrage auf sqlskills.com:

*Ursachen der letzten SQL
Server Performance-Probleme:*

27% T-SQL

19% Poor indexing

Quantifizierung des Problems

Craig S. Mullins (DB Stratege und Forscher):

„As much as 75% of poor relational performance is caused by "bad" SQL and application code.”

Noel Yuhanna (Forrester Research):

„The key difficulties surrounding performance continue to be poorly written SQL statements, improper DBMS configuration and a lack of clear understanding of how to tune databases to solve performance issues.”

Quantifizierung des Problem

Meine Beobachtung:

~50% der SQL-Performance-Probleme
werden durch Unsachgemäße
Index-Nutzung verursacht

Die Wurzel des Problems

Admins Indizieren

Wurzel des Problems: DBAs Indizieren

Wie haben Datenbanken
vor SQL funktioniert?

Wurzel des Problems: DBAs Indizieren

Die Index-Nutzung war
untrennbar
mit den Abfragen verbunden.

Wurzel des Problems: DBAs Indizieren

Beispiel: dBase

Entwickler mussten...

...Indizes bei Suchen explizit nutzen:

```
set index to last_name  
find Winand
```

...Index-Wartung berücksichtigen:

```
set index to last_name, idx2  
append
```

Wurzel des Problems: DBAs Indizieren

SQL ist eine Abstraktion:

Es definiert nur die logische Sicht

Die Implementierung muss sich
um alles andere kümmern.

Wurzel des Problems: DBAs Indizieren

SQL (Sprache)

bietet:

Tables

Constraints

Views

Transactions

Queries

Data
manipulation

SQL Datenbanken (Software)

bieten:

Storage
management

Backup
& recovery

High
Availability

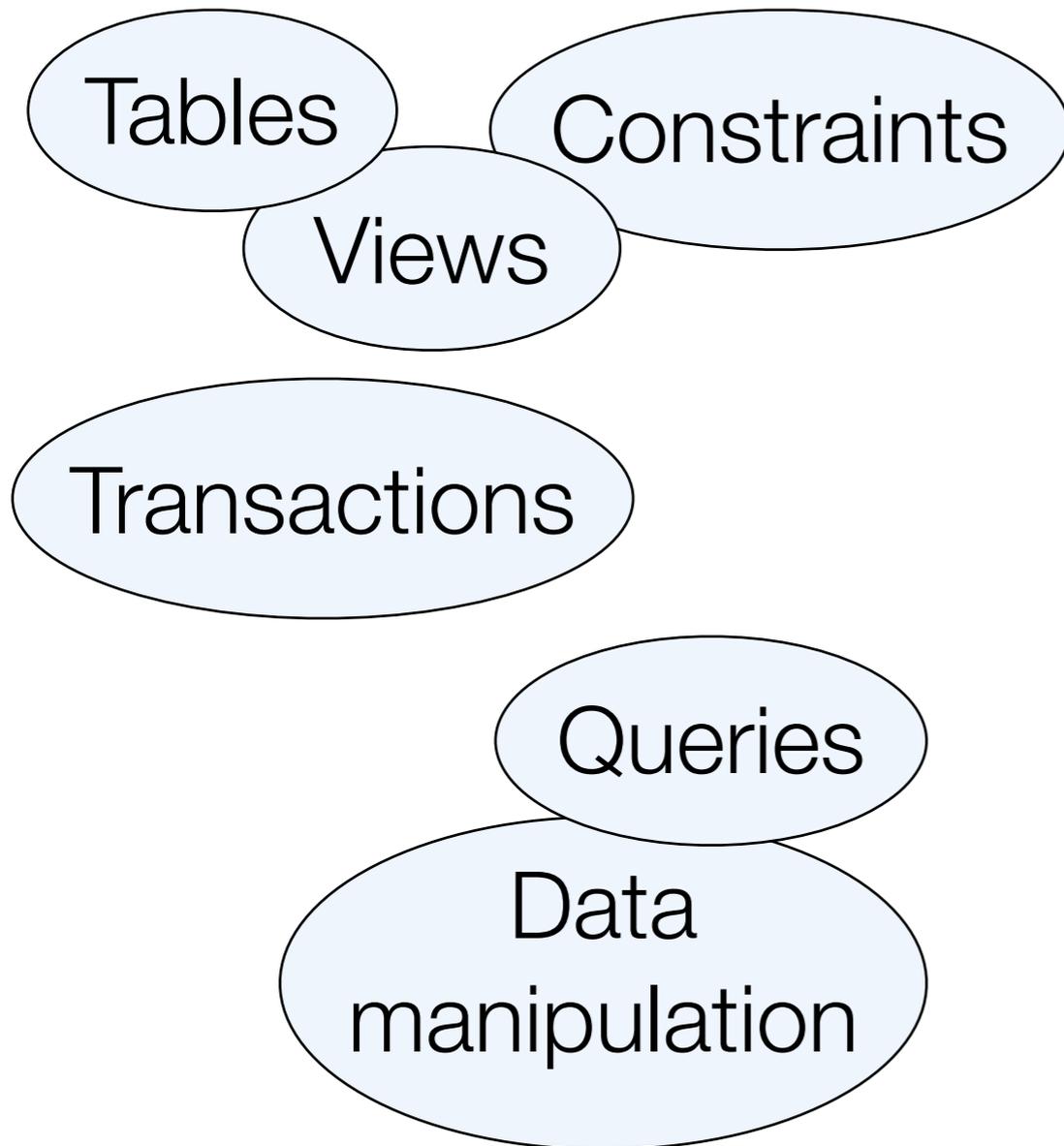
Bugs &
patches

Indexes

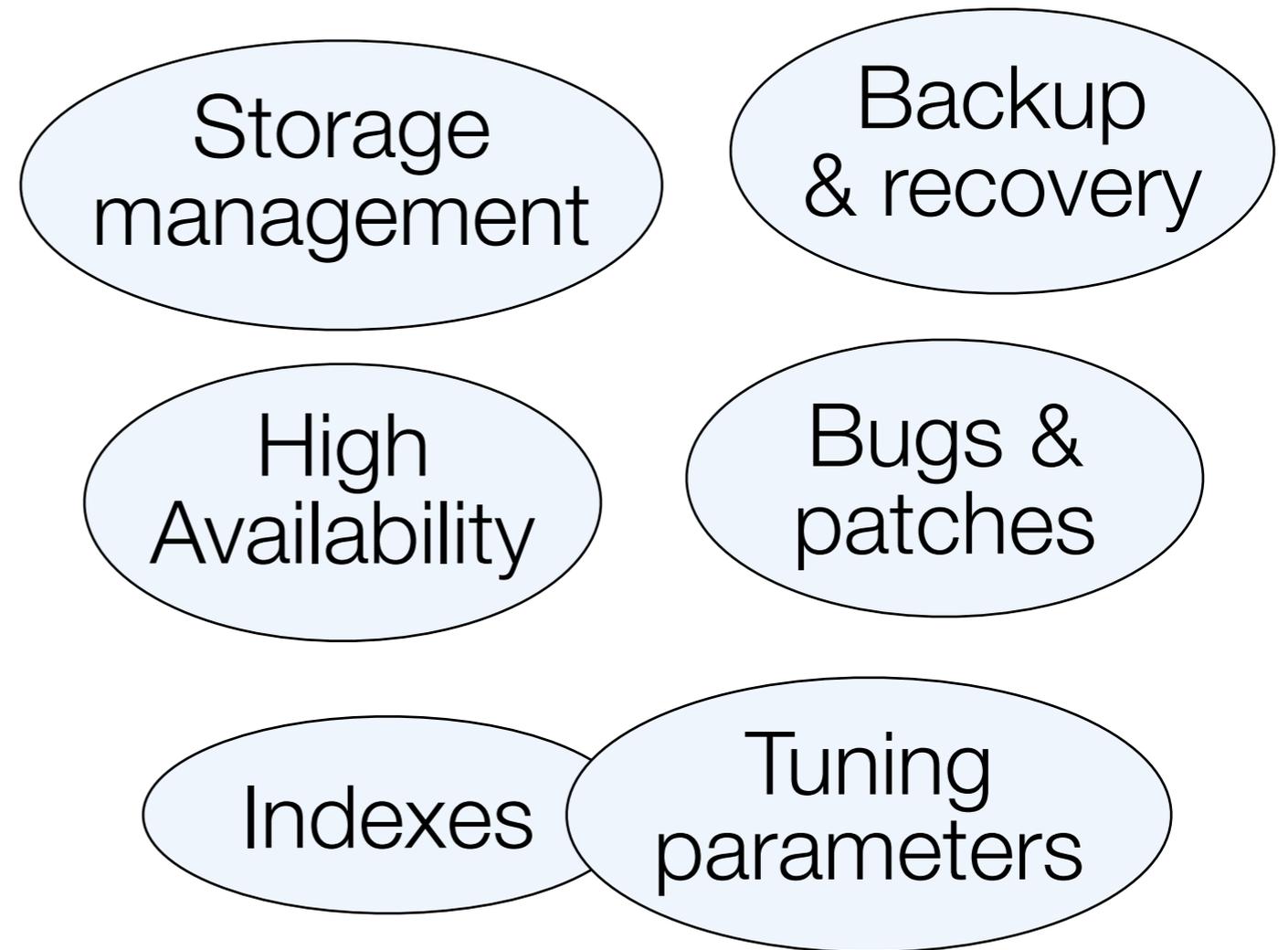
Tuning
parameters

Wurzel des Problems: DBAs Indizieren

Entwickler



SQL Datenbanken (Software) bieten:



Wurzel des Problems: DBAs Indizieren

Entwickler

Tables

Constraints

Views

Transactions

Queries

Data
manipulation

Administratoren

Storage
management

Backup
& recovery

High
Availability

Bugs &
patches

Indexes

Tuning
parameters

Wurzel des Problems: DBAs Indizieren

Indizierung wird heute als Tuning-Aufgabe betrachtet, die in die Verantwortung der Admins fällt.

Wurzel des Problems: DBAs Indizieren

Ein Missverständnis das neue Probleme bringt:

DBAs kennen die
Abfragen nicht

Müssen die Abfragen
erst eruieren.

Das ist Zeitaufwendig und
meistens Unvollständig.



Wurzel des Problems: DBAs Indizieren

Ein Missverständnis das neue Probleme bringt:

DBAs kennen die
Abfragen nicht

Müssen die Abfragen
erst eruiieren.

Das ist Zeitaufwendig und
meistens Unvollständig.

DBAs können die
Abfragen nicht ändern

Sie können den Index and
die Abfrage anpassen.

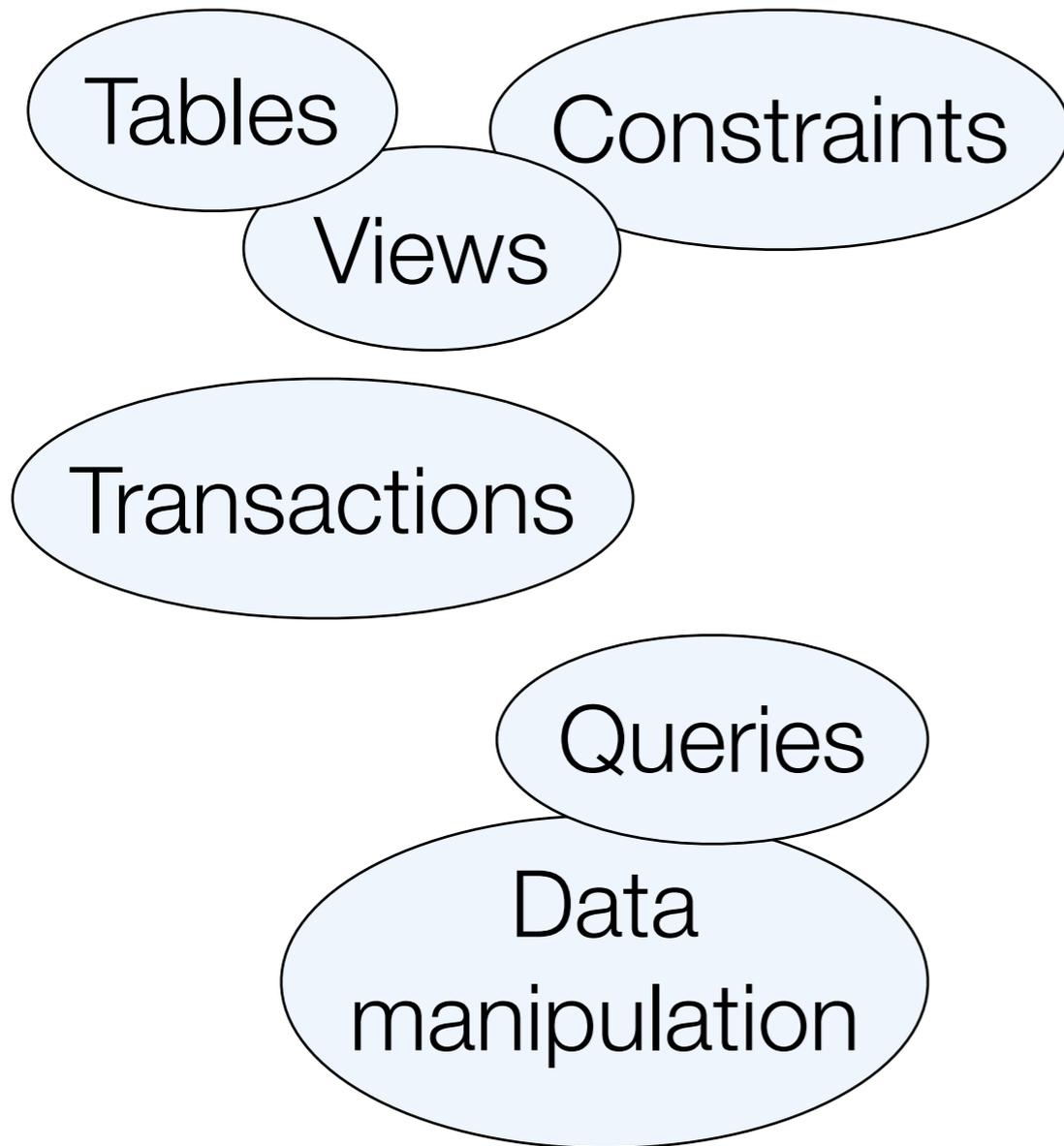
Aber nicht die Abfrage
an den Index.

Die Lösung

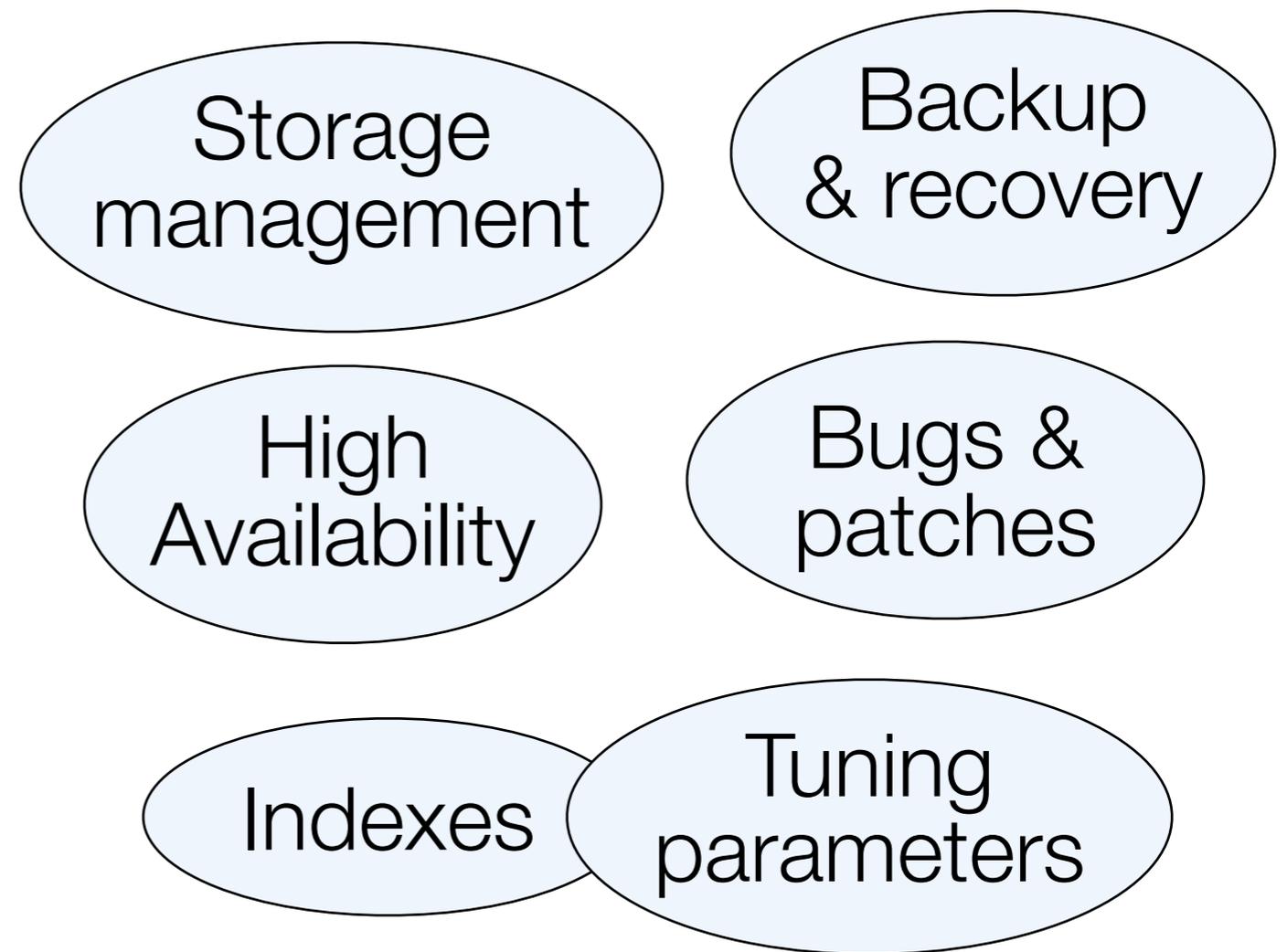
Indizierung ist eine
Entwicklungsaufgabe

Lösung: Entwickler müssen Indizieren

Developers



Administrators



Lösung: Entwickler müssen Indizieren

Developers

Tables

Constraints

Views

Transactions

Queries

Indexes

Data
manipulation

Must match!

Administrators

Storage
management

Backup
& recovery

High
Availability

Bugs &
patches

Tuning
parameters

Neues Problem: es wird nicht gelehrt

Indizierung ist nicht Teil des SQL-Standards und wird in Büchern über die Sprache SQL daher nicht abgedeckt.

11 SQL-Bücher analysiert: nur **1.0%** der Seiten sind über Indizierung (70 von 7330 Seiten).

Beispiele:

Oracle SQL by Example: **2.0%** (19/960)

Beginning DBs with PostgreSQL: **0.8%** (5/664)

Learning SQL: **3.3%** (11/336 — höchste Rate dieser Klasse)

Neues Problem: es wird nicht gelehrt

Manche Tuning-Bücher vertiefen Indizierung, verstecken dass aber immer zwischen hunderten Seiten über HW, OS and DB-Parametriesierung.

14 Datenbank-Admin-Bücher analysiert: **5.1%** der Seiten sind über Indizierung (307 von 6069 Seiten).

Beispiele:

Oracle Performance Survival Guide: **5.2%** (38/730)

High Performance MySQL: **8%** (55/684)

PostgreSQL 9 High Performance: **5.8%** (27/468)

Neues Problem: es wird nicht gelehrt

Konsequenz:

Entwickler wissen nicht, wie man Indizes richtig einsetzt.

Ergebnisse unsers 3-Minuten online Tests:

<http://use-the-index-luke.com/de/3-minuten-test>

5 Fragen: jeder über ein spezielles
Index-Verhalten

Nicht repräsentativ!

3-Minuten Quiz: Indizierungs-Wissen

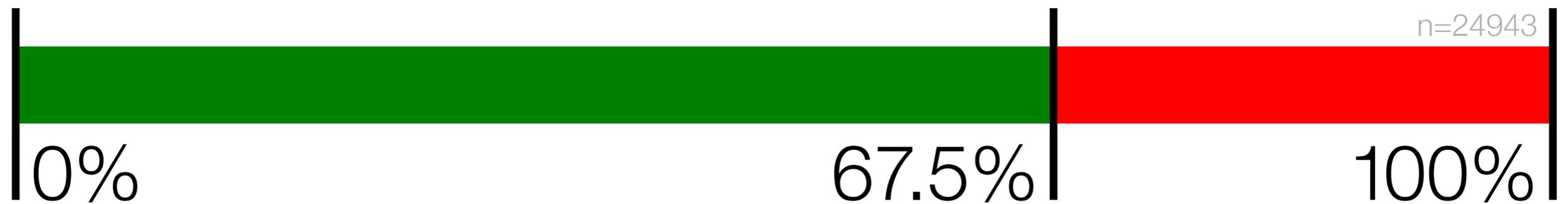
Q1: Gut oder schlecht? *(Funktionen)*

```
CREATE INDEX tbl_idx ON tbl (date_column);  
SELECT text, date_column  
FROM tbl  
WHERE TO_CHAR(date_column, 'YYYY') = '2013';
```

3-Minuten Quiz: Indizierungs-Wissen

Q1: Gut oder schlecht? *(Funktionen)*

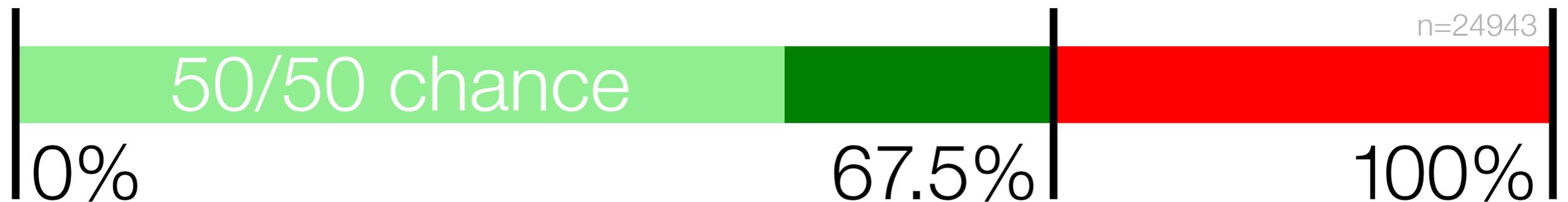
```
CREATE INDEX tbl_idx ON tbl (date_column);  
SELECT text, date_column  
FROM tbl  
WHERE TO_CHAR(date_column, 'YYYY') = '2013';
```



3-Minuten Quiz: Indizierungs-Wissen

Q1: Gut oder schlecht? *(Funktionen)*

```
CREATE INDEX tbl_idx ON tbl (date_column);  
SELECT text, date_column  
FROM tbl  
WHERE TO_CHAR(date_column, 'YYYY') = '2013';
```

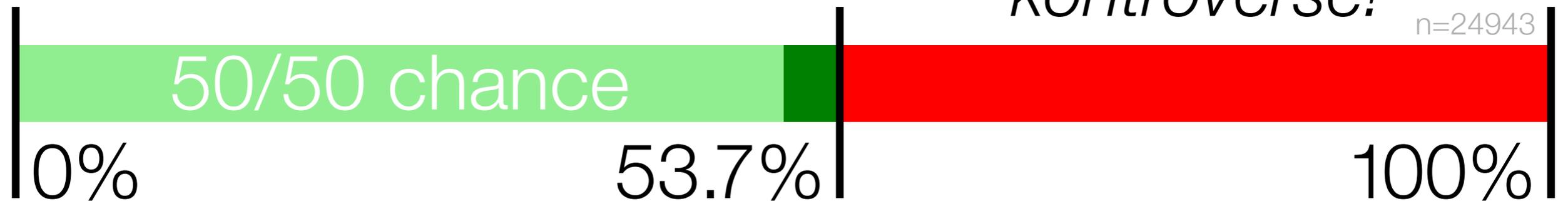


3-Minuten Quiz: Indizierungs-Wissen

Q2: Gut oder schelcht? *(Top-N, kein IOS)*

```
CREATE INDEX tbl_idx ON tbl (a, date_col);  
SELECT id, a, date_col  
  FROM tbl  
 WHERE a = $1  
 ORDER BY date_col DESC  
LIMIT 1;
```

*Nachvollziehbare
kontroverse!*



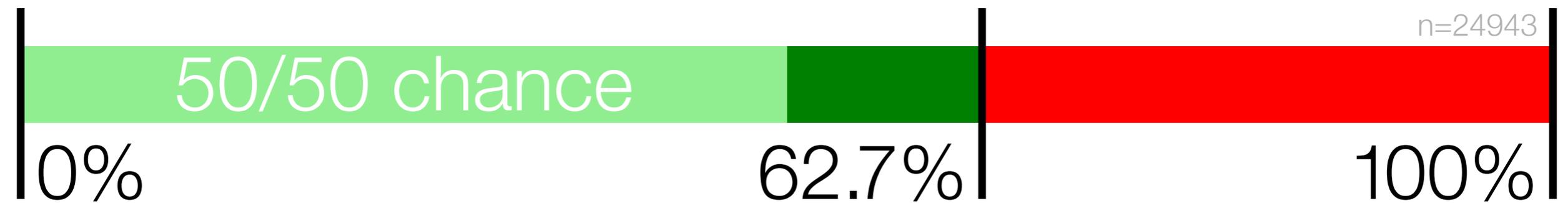
3-Minuten Quiz: Indizierungs-Wissen

Q3: Gut oder schlecht? (*Spaltenreihenfolge*)

```
CREATE INDEX tbl_idx ON tbl (a, b);
```

```
SELECT id, a, b FROM tbl  
WHERE a = $1 AND b = $2;
```

```
SELECT id, a, b FROM tbl  
WHERE b = $1;
```

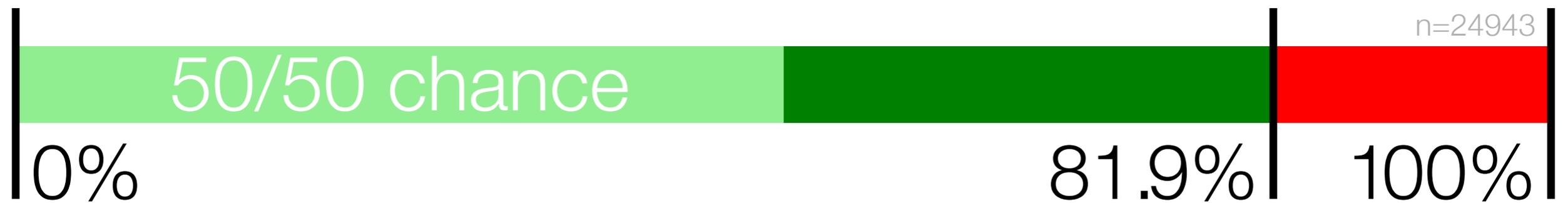


3-Minuten Quiz: Indizierungs-Wissen

Q4: Gut oder schlecht? (*LIKE* indizieren)

```
CREATE INDEX tbl_idx
  ON tbl (text varchar_pattern_ops);

SELECT id, text
  FROM tbl
 WHERE text LIKE '%TERM%';
```

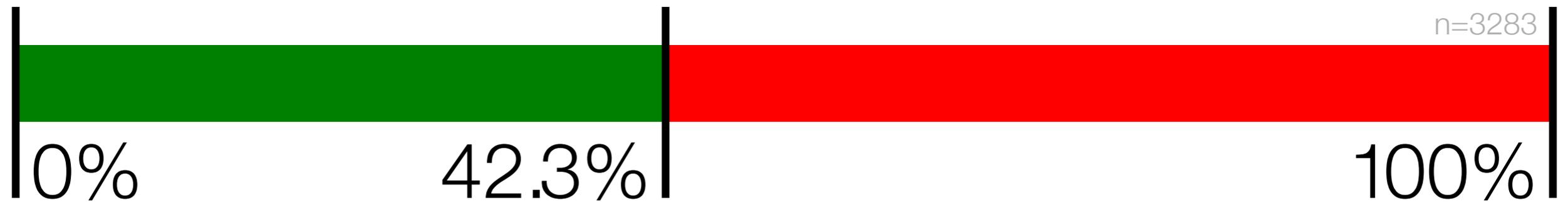


3-Minuten Quiz: Indizierungs-Wissen

Q5: Gut oder schlecht? (= vs. <>)

```
CREATE INDEX tbl_idx
      ON tbl (date_col, state);

SELECT id, date_col, state FROM tbl
WHERE date_col >=
      CURRENT_DATE - INTERVAL '5' YEAR
      AND state = 'X';
```



Volkskrankheit: Stiefmütterliche Indizierung

Jeder weiss, dass Indizierung wichtig für Performance ist, aber niemand nimmt sich Zeit es ordentlich zu lernen und durchzuführen.

Volkskrankheit: Stiefmütterliche Indizierung

Index-Details sind kaum bekannt

➔ „Details“ wie die Spaltenreihenfolge oder Bereichsuchen müssen gelernt und verstanden werden.

Nur eine Funktion wird verwendet: schnell suchen.

➔ Indizes haben drei Funktionen (Mächte/Powers):
Daten finden, Daten gruppieren, Daten sortieren.

Abfragen werden einzeln indiziert

➔ Man muss aus Applikationssicht indizieren (alle Abfragen beachten). Das ist eine Design-Aufgabe!

Volkskrankheit: Stiefmütterliche Indizierung

Legst Du Indizes einfach an

oder

designst du Indizes?

Über Markus Winand

Ich tune Entwickler auf
SQL-Performance

Training & co:
winand.at

Geeky blog:
use-the-index-luke.com

Autor von:
SQL Performance Explained

