

3. Datenbankschema

3.1 Relationen

Partei: {name: varchar(40)}

Bundesland: {name: varchar(30)}

Sitzkontingent: {id: serial unique, bundesland: varchar(30) references Bundesland(name), wahljahr: int, kontingent: int}

Wahlkreis: {nummer: int, name: varchar(100), bundesland: varchar(30) references Bundesland(name)}

Wahlberechtigte: {id: serial unique, wahlkreis: int references Wahlkreis(nummer), wahljahr: int, wahlberechtigte: int}

Bewerber: {id: int, titel: varchar(30), vorname: varchar(50), nachname: varchar(60), partei: varchar(40) references Partei(name), jahrgang: int}

Direktkandidat: {wahljahr: int, wahlkreis: int references Wahlkreis(nummer), partei: varchar(40) references Partei(name), bewerber: int references Bewerber(id)}

Landesliste: {id: int, wahljahr: int, partei: varchar(40) references Partei(name), bundesland: varchar(30) references Bundesland(name)}

Listenplaetze: {listen_id: int references Landesliste(id), bewerber_id: int references Bewerber(id), listenplatz: int}

Stimmzettel: {id: serial unique, wahljahr: int, direktkandidat: int [*nullable*] references Bewerber(id), landesliste: int [*nullable*] references Landesliste(id), wahlkreis: int references Wahlkreis(nummer)}

Alle Attribute der Relationen sind implizit als *not null* zu verstehen, außer sie wurden explizit mit dem Zusatz *nullable* versehen.

4 Ergebnisberechnung

Die Berechnung des Wahlergebnisses erfolgt mittels *materialized views*, welche die Ergebnisse persistieren, sich bei Änderungen an den Wahldaten aber sehr simpel über den Aufruf der Funktion *RefreshAllMaterializedViews(schema: text)* aktualisieren lassen.

4.1 Materialized Views

DirektmandateProLand: {wahljahr: int, bundesland: varchar(30), partei: varchar(40), count: int}

Berechnet die Anzahl der Direktmandate einer Partei pro Bundesland und Wahljahr.

DirektmandateProPartei: {partei: varchar(40), anzahl: int, wahljahr: int}

Berechnet die Anzahl der Direktmandate einer Partei auf Bundesebene pro Wahljahr.

ErlaubteListen: {bundesland: varchar(30), landesliste: int, stimmen: int, wahljahr: int}

Berechnet die Landeslisten der Parteien, welche über die Fünf-Prozent-Hürde gekommen sind oder mindestens zwei Direktmandate haben, pro Bundesland und Wahljahr.

ErlaubteParteien: {partei: varchar(40), wahljahr: int}

Berechnet die Parteien, welche über die Fünf-Prozent-Hürde gekommen sind oder mindestens zwei Direktmandate haben, pro Wahljahr.

Oberverteilung: {partei: varchar(40), sitze: int, wahljahr: int}

Berechnet die Zusammensetzung des Deutschen Bundestags für 2013 komplett und für 2009 ohne Überhangmandate.

StimmenGueltigkeit: {wahlkreis: int, wahljahr: int, waehler: int, g_erststimmen: int, g_zweitstimmen: int, ug_erststimmen: int, ug_zweitstimmen: int}

Berechnet die gültigen (g) bzw. ungültigen (ug) Erst- und Zweitstimmen, sowie die Anzahl der abgegebenen Stimmzettel (waehler) pro Wahlkreis und Wahljahr.

StimmenProKandidat: {wahlkreis: int, direktkandidat: int, stimmen: int, wahljahr: int}

Berechnet die Stimmen eines Bewerbers in seinem Wahlkreis pro Wahljahr.

StimmenProLandesliste: {bundesland: varchar(30), landesliste: int, stimmen: int, prozent: numeric, wahljahr: int}

Berechnet die Zweitstimmen einer Partei in einem Bundesland absolut und prozentual pro Wahljahr.

StimmenProPartei: { landesliste: int, stimmen: int, prozent: numeric, wahljahr: int}

Berechnet die Zweitstimmen einer Partei auf Bundesebene absolut und prozentual pro Wahljahr.

Ungerade: {zahl: int}

Berechnet alle ungeraden Zahlen bis 600 für das Höchstzahlverfahren.

Wahlkreissieger: {wahlkreis: int, direktkandidat: int, stimmen: int, wahljahr: int}

Berechnet den Direktkandidaten eines Wahlkreises und dessen erhaltene Stimmen pro Wahljahr.

WahlkreisstimmenProLandesliste: {wahlkreis: int, landesliste: int, stimmen: int, prozent: numeric, wahljahr: int}

Berechnet die Zweitstimmen einer Partei in einem Wahlkreis absolut und prozentual pro Wahljahr.

5. Zugriffsschutz

Die Datenbank verfügt über keine REST-API, ist also auch durch die darüberliegende Applikationslogik geschützt. Zusätzlich wurden folgende User mit unterschiedlichen Berechtigungen angelegt:

- **Wahladmin:** OWNER der Datenbank; Wird nur für die Aktualisierung der Materialized Views verwendet
- **Waeher:** Standard-Nutzer der Web-Schnittstelle, ausschließlich SELECT auf allen Tables und Views
- **Stimmegeber:** Nutzer zum Abgeben einer Stimme in der Stimmabgabe-Ansicht; SELECT auf allen Tables und Views, INSERT auf Table Stimmzettel