

## Laboratorio 8 - Trumpsacciones

En el servidor del curso, usted encontrará el esquema **eleccion** con datos relativos a la elección presidencial realizada en EE.UU. en 2016. (Alerta de spoiler: Trump gana).

Cuando se trata de votaciones, la integridad de los datos es crucial. Hoy trabajaremos en agregar transacciones y restricciones a las siguientes tablas del esquema **eleccion**:

- **estado**(nombre, voto\_electoral, cierre, num\_candidatos)
- **condado**(nombre, e.estado, reportado)
- **candidato**(nombre, partido)
- **votosPorCondado**(ca.candidato, co.condado, co.estado, votos)

La tabla **estado** contiene los nombres de los estados EE.UU., la cantidad de votos electorales que poseen, la hora de cierre de la votación y el número de candidatos que corren en él. La tabla **condado** contiene los condados de un estado y la fracción de votos reportados hasta el momento (inicialmente 0). Puede haber dos condados con el mismo nombre en diferentes estados. En **candidato** se registran los candidatos a la presidencia y sus partidos. En **votosPorCondado** está la cantidad de votos por cada candidato en cada condado (inicialmente 0).

En la mañana del día de la elección, las tablas comienzan con 0 votos. En la base de datos hay actualizaciones de los datos de la votación por cada hora. Las tablas **votosPorCondado1**, ..., **votosPorCondado9** tienen la misma estructura que **votosPorCondado** y representan la cantidad **total** de votos en ese momento (**votosPorCondado<sub>n</sub>** corresponde a las votaciones totales **hasta** la *n*-ésima hora incluyendo horas anteriores). Las tablas **condado1**, ..., **condado9** representan las fracciones de votos emitidos hasta cada hora de la elección, en cada condado, con la misma estructura que **condado**. Estas tablas con actualizaciones contienen una tupla por cada vez que cambia algo; si no cambia el valor de una tupla en la *n*-ésima hora, la tupla no estará en **votosPorCondado<sub>n</sub>** o **condado<sub>n</sub>**, pero se mantiene. Dado que son valores totales hasta la hora respectiva, para actualizar **votosPorCondado** y **condado**, es necesario reemplazar (no sumar) los valores de las tuplas con la misma llave a partir de las tablas de actualizaciones.

Al final del laboratorio, usted debe entregar un archivo **.txt** con las respuestas de las siguientes preguntas (incluyendo comandos SQL que usted escriba o modifica).

- P1.** 10 PUNTOS Para practicar con las actualizaciones y restricciones, es necesario que copie las tablas base. Para ahorrar tiempo, descargue el archivo **tr\_carga.txt** desde Material Docente. ¡No llegue y corra los comandos! Debe hacer las siguientes modificaciones a los comandos en **tr\_carga.txt** antes. Primero, reemplace “GGGGG” por un nombre único para su grupo en todo el archivo. Luego agregue las siguientes modificaciones, tomando en cuenta las llaves primarias y foráneas definidas antes para las cuatro tablas del esquema **elección**.

- (a) Agregue la llave primaria a la tabla `grupo_estado` (ahora `GRUPO` debería ser el nombre de su grupo; hay que extender el comando de `CREATE TABLE` correspondiente).
- (b) Agregue la llave primaria y la llave foránea a la tabla `grupo_condado`. Agregue a la tabla una restricción que verifique que la fracción reportada sea un valor entre 0 y 1 (inclusive) con `CHECK (reportado >= 0 AND reportado <= 1)` separado por coma.
- (c) Agregue la llave primaria a la tabla `grupo_candidato`.
- (d) Agregue la llave primaria y las dos llaves foráneas a la tabla `grupo_votosPorCondado`.

Ejecute todos los comandos modificados (cuatro pares de `CREATE` y `INSERT`).

- P2.** 10 PUNTOS Ahora queremos ir actualizando los datos sobre las votaciones en su copia de las tablas, comenzando con la primera hora de elección. Revise los contenidos de la tabla `votosPorCondado1`. Construya una consulta `UPDATE` en SQL para actualizar su tabla `grupo_votosPorCondado` para cada tupla en `votosPorCondado1` según la llave primaria. Ejecute su instrucción.
- P3.** 10 PUNTOS Haremos lo mismo con la fracción de votos reportados durante la primera hora. Revise los contenidos de la tabla `condado1`; sigue la misma idea que la tabla `votosPorCondado1`. Escriba las instrucciones para actualizar `grupo_condado` con los datos de `condado1`.
- P4.** 10 PUNTOS Piense en lo que pasaría si actualizamos `grupo_votosPorCondado` y luego intentamos actualizar `grupo_condado` y ocurriera un problema. ¿Los datos estarían bien? No, pues tendríamos votos contados en el condado, pero la fracción de datos escrutados seguiría siendo desactualizada. Por lo tanto, use una transacción para actualizar ambas tablas atómicamente, pero esta vez usando los datos de la segunda hora de votación.
- P5.** 10 PUNTOS Modifique su transacción para actualizar las horas siguientes hasta la hora 9 inclusiva (sí, puede seguir *copypasteando* hasta completar todas las horas; puede usar una transacción por hora).
- P6.** 10 PUNTOS Putin envía sus hackers a cargar datos maliciosos a su base de datos para lograr la victoria de Clinton.<sup>1</sup> Es decir, envía como actualizaciones las tablas `votosPorCondadoX` y `condadoX`. Lamentablemente para Vladimir (y para Hillary), los hackers creyeron que la columna `reportado` correspondía a un porcentaje y no a una fracción. Cree una transacción para cargar `votosPorCondadoX` y `condadoX` en sus tablas atómicamente. Vea cuidadosamente la salida de Postgres. Realice una consulta para verificar el condado y candidato que trataron de hackear. ¿Putin lo logró? En su respuesta, indique si funcionó o no esta actualización, por qué, e indique las dos garantías de ACID más relevantes.

---

<sup>1</sup>... ¿imáginese que estamos en un universo alternativo?