

Bases de datos distribuidas (2019-2020)
GRADO EN INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE GRANADA

Bases de Datos distribuidas

Daniel Terol Guerrero

3 de marzo de 2020

ÍNDICE

1. Práctica 1: Diseño conceptual y lógico global de la base de datos	1
1.1. Paso a tablas	1
1.2. Absorción de tablas	2
2. Práctica 2: Diseño de la fragmentación y de la asignación	3
2.1. Determinar qué relaciones deben fragmentarse y con qué tipo de fragmentación	3
2.2. Determinar qué relaciones o fragmentos deben o pueden replicarse y por qué	4
2.3. Realizar un diseño distribuido para esta base de datos	4
2.3.1. Sucursal	4
2.3.2. Empleado	5
2.3.3. SucursalVendeVino	5
2.3.4. Cliente	6
2.3.5. ClienteSolicitaVinoASucursal	7
2.3.6. Vino	7
2.4. Determinar una asignación de fragmentos, réplicas y relaciones o tablas no fragmentadas, que se adecue a los requerimientos de las aplicaciones	9
2.4.1. Sucursal, Empleado y SucursalVendeVino	9
2.4.2. Cliente y ClienteSolicitaVinoASucursal	9
2.4.3. Vino	9
2.4.4. Productor y SucursalSolicitaSucursal	10

1. PRÁCTICA 1: DISEÑO CONCEPTUAL Y LÓGICO GLOBAL DE LA BASE DE DATOS

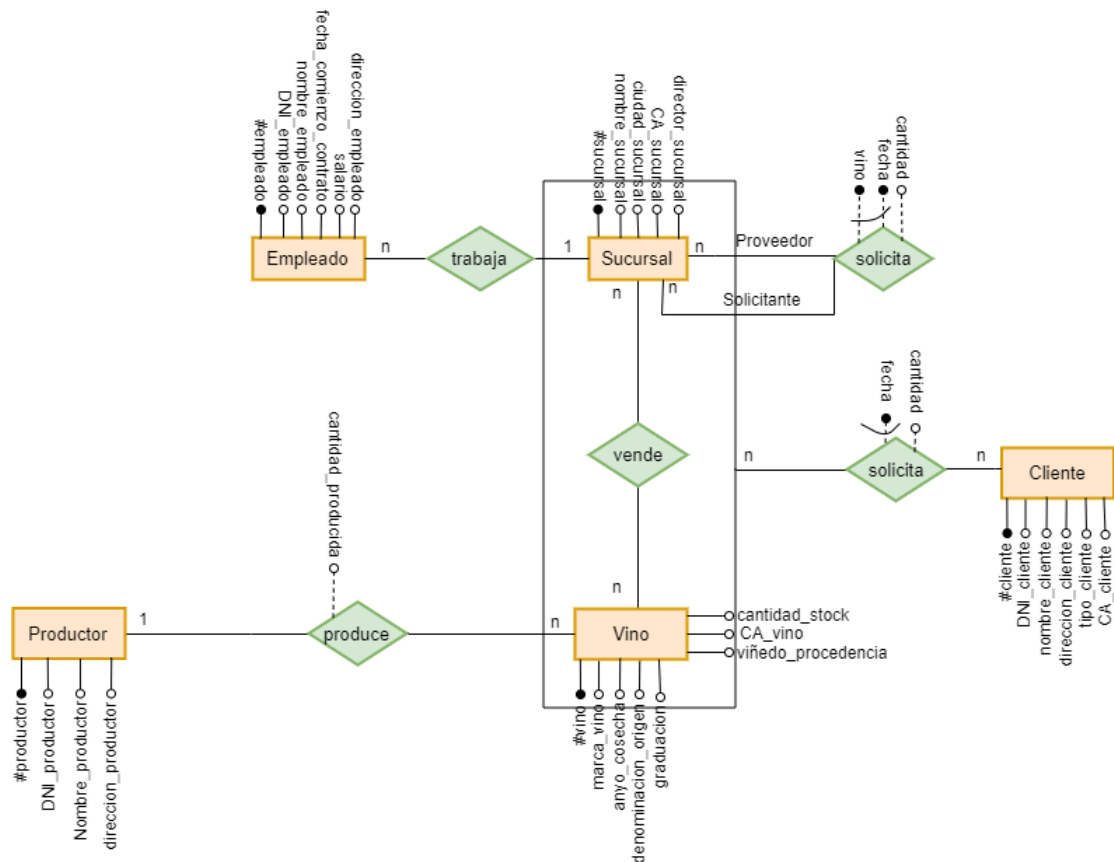


Figura 1.1: Entidad/Relación sobre la venta de vinos

HAY QUE PONER EL COMENTARIO DEL E/R Y DISCUTIR LAS DIFERENTES OPCIONES.

1.1. PASO A TABLAS

1. **Productor** (#productor, DNI_productor, nombre_productor, direccion_productor)
2. **Vino** (#vino, marca_vino, anyo_cosecha, denominacion_origen, graduacion, cantidad_stock, CA_vino, viñedo_procedencia)

3. **Vino producido** (#vino (CE2), #productor (CE1), cantidad_producida)
4. **Sucursal** (#sucursal, nombre_sucursal, ciudad_sucursal, CA_sucursal, director_sucursal (CE9))
5. **SucursalVendeVino** (#sucursal (CE4), #vino (CE2))
6. **SucursalSolicitaSucursal** (#sucursal1 (CE4), #sucursal2 (CE4), #vino (CE2), fecha, cantidad)
7. **Cliente** (#cliente, DNI_cliente, nombre_cliente, dirección_cliente, tipo_cliente, CA_cliente)
8. **ClienteSolicitaVinoASucursal** (#cliente(CE7), (#sucursal(CE4), #vino(CE2)) (CE5), fecha, cantidad)
9. **Empleado**(#empleado, DNI_empleado, nombre_empleado, fecha_comienzo_contrato, salario, direccion_empleado)
10. **EmpleadoTrabajaSucursal** (#empleado (CE9), #sucursal(CE4))

1.2. ABSORCIÓN DE TABLAS

1. **Productor** (#productor, DNI_productor, nombre_productor, direccion_productor)
2. **Vino** (#vino, marca_vino, anyo_cosecha, denominacion_origen, graduacion, cantidad_stock, CA_vino, viñedo_procedencia, cantidad_producida, #productor(CE1))
3. **Sucursal** (#sucursal, nombre_sucursal, ciudad_sucursal, CA_sucursal, director_sucursal (CE8))
4. **SucursalVendeVino** (#sucursal (CE3), #vino (CE2))
5. **SucursalSolicitaSucursal** (#sucursal1 (CE3), #sucursal2 (CE3), #vino (CE2), fecha, cantidad)
6. **Cliente** (#cliente, DNI_cliente, nombre_cliente, dirección_cliente, tipo_cliente, CA_cliente)
7. **ClienteSolicitaVinoASucursal** (#cliente(CE6), (#sucursal(CE3), #vino(CE2)) (CE4), fecha, cantidad)

8. **Empleado**(#*empleado*, DNI_*empleado*, nombre_*empleado*, fecha_*comienzo_contrato*, salario, direccion_*empleado*, #*sucursal*(CE3))

2. PRÁCTICA 2: DISEÑO DE LA FRAGMENTACIÓN Y DE LA ASIGNACIÓN

2.1. DETERMINAR QUÉ RELACIONES DEBEN FRAGMENTARSE Y CON QUÉ TIPO DE FRAGMENTACIÓN

- Según la descripción del problema, las sucursales pertenecen a delegaciones, que abarcan diferentes comunidades autónomas. Por tanto, se realizará una fragmentación horizontal primaria de la tabla **Sucursal** con el atributo *CA_sucursal*
- Como los empleados solo pueden trabajar en una sucursal, es interesante tener los empleados que trabajan en cada sucursal junto a la información de la sucursal. Por tanto, va a realizar una fragmentación horizontal derivada de la tabla **Empleado** por el atributo *#sucursal*.
- Para poder saber los vinos que ha vendido cada sucursal, la tabla **SucursalVendeVino** se va a fragmentar horizontalmente de manera derivada por el atributo *#sucursal*.
- Puesto que los clientes tienen que solicitar suministros de cualquier vino a través de sucursales de la delegación a la que pertenece su comunidad autónoma, la tabla **Cliente** se va a fragmentar horizontalmente de manera primaria con el atributo *CA_cliente*.
- Para poder tener la información de los clientes y los pedidos que ha realizado, la tabla **ClienteSolicitaVinoASucursal** se va a fragmentar horizontalmente de manera derivada por el atributo *#cliente*.
- Sobre la tabla **Vino** se va a realizar una fragmentación horizontal primaria con el atributo *CA_vino*, ya que será distribuido, mayoritariamente, por diferentes lugares de la correspondiente comunidad autónoma.

2.2. DETERMINAR QUÉ RELACIONES O FRAGMENTOS DEBEN O PUEDEN REPLICARSE Y POR QUÉ

- Puesto que los productores pueden producir vinos de diferentes comunidades autónomas, etc. La tabla **Productor** se va a replicar para poder tener la información de los productores en todos los fragmentos.
- Por otro lado, la relación **SucursalSolicitaSucursal** no es conveniente fragmentarlo porque una sucursal no puede solicitar pedidos a una sucursal de su misma delegación. Por tanto, es necesario replicar la información de esta tabla por si es necesario acceder a la información de las dos sucursales involucradas en un pedido.

2.3. REALIZAR UN DISEÑO DISTRIBUIDO PARA ESTA BASE DE DATOS

2.3.1. SUCURSAL

El atributo elegido para la fragmentación es la comunidad autónoma que contamos con 19 diferentes. Por tanto, tenemos 2^{19} términos de predicado posibles pero por la definición del problema sabemos que una sucursal solo puede estar en una comunidad autónoma. Es decir, los predicados verdaderos son los que tienen un único predicado afirmado y todos los demas, 18, negados.

$$P = \{ \begin{array}{l} p_1: CA_sucursal = 'Castilla-León', \\ p_2: CA_sucursal = 'Castilla-La Mancha', \\ p_3: CA_sucursal = 'Aragón', p_4: CA_sucursal = 'Madrid', \\ p_5: CA_sucursal = 'La Rioja', p_6: CA_sucursal = 'Cataluña', \\ p_7: CA_sucursal = 'Baleares', p_8: CA_sucursal = 'Valencia', \\ p_9: CA_sucursal = 'Murcia', p_{10}: CA_sucursal = 'Galicia', \\ p_{11}: CA_sucursal = 'Asturias', p_{12}: CA_sucursal = 'Cantabria', \\ p_{13}: CA_sucursal = 'País Vasco', p_{14}: CA_sucursal = 'Navarra', \\ p_{15}: CA_sucursal = 'Andalucía', p_{16}: CA_sucursal = 'Extremadura', \\ p_{17}: CA_sucursal = 'Canarias', p_{18}: CA_sucursal = 'Ceuta', \\ p_{19}: CA_sucursal = 'Melilla' \end{array} \}$$

P es un conjunto completo y minimal y, como se ha comentado anteriormente, cada predicado tiene la siguiente estructura:

p_1 : $CA_sucursal = 'Castilla-León' \wedge CA_sucursal \neq 'Castilla-La Mancha' \wedge CA_sucursal \neq 'Aragón' \wedge \dots$

Pero por simplificar, se especifica el afirmado y se omiten los negados. A continuación se van a establecer cuatro términos de predicados diferentes:

- $Y_1 = \{ p_1 \vee p_2 \vee p_3 \vee p_4 \vee p_5 \}$
- $Y_2 = \{ p_6 \vee p_7 \vee p_8 \vee p_9 \}$
- $Y_3 = \{ p_{10} \vee p_{11} \vee p_{12} \vee p_{13} \vee p_{14} \}$
- $Y_4 = \{ p_{15} \vee p_{16} \vee p_{17} \vee p_{18} \vee p_{19} \}$

Que corresponden con los siguientes fragmentos:

- $Sucursal_1 = SL_{Y_1} (Sucursal)$
- $Sucursal_2 = SL_{Y_2} (Sucursal)$
- $Sucursal_3 = SL_{Y_3} (Sucursal)$
- $Sucursal_4 = SL_{Y_4} (Sucursal)$

2.3.2. EMPLEADO

La relación **Empleado** se fragmenta a partir de la relación *Sucursal*, por tanto los fragmentos correspondientes son:

- $Empleado_1 = Empleado \ SJN_{\#sucursal=\#sucursal} \ Sucursal_1$
- $Empleado_2 = Empleado \ SJN_{\#sucursal=\#sucursal} \ Sucursal_2$
- $Empleado_3 = Empleado \ SJN_{\#sucursal=\#sucursal} \ Sucursal_3$
- $Empleado_4 = Empleado \ SJN_{\#sucursal=\#sucursal} \ Sucursal_4$

2.3.3. SUCURSALVENDEVINO

La relación **SucursalVendeVino** se fragmenta a partir de la relación *Sucursal*, por tanto los fragmentos correspondientes son:

- $SucursalVendeVino_1 = SucursalVendeVino S J N_{\#sucursal=\#sucursal} Sucursal_1$
- $SucursalVendeVino_2 = SucursalVendeVino S J N_{\#sucursal=\#sucursal} Sucursal_2$
- $SucursalVendeVino_3 = SucursalVendeVino S J N_{\#sucursal=\#sucursal} Sucursal_3$
- $SucursalVendeVino_4 = SucursalVendeVino S J N_{\#sucursal=\#sucursal} Sucursal_4$

2.3.4. CLIENTE

Con la relación **Cliente** se procede de manera análoga que con *Sucursal*, es decir, se fragmenta horizontalmente de manera primaria por el atributo comunidad autónoma. Por tanto, el conjunto de términos de predicado son los siguientes:

$P = \{ p_1: CA_cliente = 'Castilla-León',$
 $p_2: CA_cliente = 'Castilla-La Mancha',$
 $p_3: CA_cliente = 'Aragón', p_4: CA_cliente = 'Madrid',$
 $p_5: CA_cliente = 'La Rioja', p_6: CA_cliente = 'Cataluña',$
 $p_7: CA_cliente = 'Balears', p_8: CA_cliente = 'Valencia',$
 $p_9: CA_cliente = 'Murcia', p_{10}: CA_cliente = 'Galicia',$
 $p_{11}: CA_cliente = 'Asturias', p_{12}: CA_cliente = 'Cantabria',$
 $p_{13}: CA_cliente = 'País Vasco', p_{14}: CA_cliente = 'Navarra',$
 $p_{15}: CA_cliente = 'Andalucía', p_{16}: CA_cliente = 'Extremadura',$
 $p_{17}: CA_cliente = 'Canarias', p_{18}: CA_cliente = 'Ceuta',$
 $p_{19}: CA_cliente = 'Melilla' \}$

P es un conjunto completo y minimal y, como se ha comentado anteriormente, cada predicado tiene la siguiente estructura:

$p_1: CA_cliente = 'Castilla-León' \wedge CA_cliente \neq 'Castilla-La Mancha' \wedge CA_cliente \neq 'Aragón' \wedge \dots$

Pero por simplificar, se especifica el afirmado y se omiten los negados. A continuación se van a establecer cuatro términos de predicados diferentes:

- $Y_1 = \{ p_1 \vee p_2 \vee p_3 \vee p_4 \vee p_5 \}$

- $Y_2 = \{ p_6 \vee p_7 \vee p_8 \vee p_9 \}$
- $Y_3 = \{ p_{10} \vee p_{11} \vee p_{12} \vee p_{13} \vee p_{14} \}$
- $Y_4 = \{ p_{15} \vee p_{16} \vee p_{17} \vee p_{18} \vee p_{19} \}$

Que corresponden con los siguientes fragmentos:

- $Cliente_1 = SL_{Y_1} (Cliente)$
- $Cliente_2 = SL_{Y_2} (Cliente)$
- $Cliente_3 = SL_{Y_3} (Cliente)$
- $Cliente_4 = SL_{Y_4} (Cliente)$

2.3.5. CLIENTESOLICITA VINO ASUCURSAL

La relación **ClienteSolicitaVinoASucursal** se fragmenta a partir de la relación *Cliente*, por tanto los fragmentos correspondientes son:

- $ClienteSolicitaVinoASucursal_1 = ClienteSolicitaVinoASucursal \Join_{\#cliente=\#cliente} Cliente_1$
- $ClienteSolicitaVinoASucursal_2 = ClienteSolicitaVinoASucursal \Join_{\#cliente=\#cliente} Cliente_2$
- $ClienteSolicitaVinoASucursal_3 = ClienteSolicitaVinoASucursal \Join_{\#cliente=\#cliente} Cliente_3$
- $ClienteSolicitaVinoASucursal_4 = ClienteSolicitaVinoASucursal \Join_{\#cliente=\#cliente} Cliente_4$

2.3.6. VINO

Con la relación **Vino** se procede de manera análoga que con *Sucursal* y *Cliente*, es decir, se fragmenta horizontalmente de manera primaria por el atributo comunidad autónoma. Por tanto, el conjunto de términos de predicado son los siguientes:

$P = \{ p_1: CA_vino = 'Castilla-León',$
 $p_2: CA_vino = 'Castilla-La Mancha',$
 $p_3: CA_vino = 'Aragón', p_4: CA_vino = 'Madrid',$

$p_5: CA_vino = 'La\ Rioja', p_6: CA_vino = 'Cataluña',$
 $p_7: CA_vino = 'Balears', p_8: CA_vino = 'Valencia',$
 $p_9: CA_vino = 'Murcia', p_{10}: CA_vino = 'Galicia',$
 $p_{11}: CA_vino = 'Asturias', p_{12}: CA_vino = 'Cantabria',$
 $p_{13}: CA_vino = 'País\ Vasco', p_{14}: CA_vino = 'Navarra',$
 $p_{15}: CA_vino = 'Andalucía', p_{16}: CA_vino = 'Extremadura', p_{17}: CA_vino = 'Canarias',$
 $p_{18}: CA_vino = 'Ceuta',$
 $p_{19}: CA_vino = 'Melilla' \}$

P es un conjunto completo y minimal y, como se ha comentado anteriormente, cada predicado tiene la siguiente estructura:

$p_1: CA_vino = 'Castilla-León' \wedge CA_vino \neq 'Castilla-La\ Mancha' \wedge CA_vino \neq 'Aragón' \wedge \dots$

Pero por simplificar, se especifica el afirmado y se omiten los negados. A continuación se van a establecer cuatro términos de predicados diferentes:

- $Y_1 = \{ p_1 \vee p_2 \vee p_3 \vee p_4 \vee p_5 \}$
- $Y_2 = \{ p_6 \vee p_7 \vee p_8 \vee p_9 \}$
- $Y_3 = \{ p_{10} \vee p_{11} \vee p_{12} \vee p_{13} \vee p_{14} \}$
- $Y_4 = \{ p_{15} \vee p_{16} \vee p_{17} \vee p_{18} \vee p_{19} \}$

Que corresponden con los siguientes fragmentos:

- $Vino_1 = SL_{Y_1} (Vino)$
- $Vino_2 = SL_{Y_2} (Vino)$
- $Vino_3 = SL_{Y_3} (Vino)$
- $Vino_4 = SL_{Y_4} (Vino)$

2.4. DETERMINAR UNA ASIGNACIÓN DE FRAGMENTOS, RÉPLICAS Y RELACIONES O TABLAS NO FRAGMENTADAS, QUE SE ADECUEN A LOS REQUERIMIENTOS DE LAS APLICACIONES

2.4.1. SUCURSAL, EMPLEADO Y SUCURSALVENDEVINO

La asignación de los fragmentos a las diferentes localidades son las siguientes:

- $Sucursal_1$, $Empleado_1$ y $SucursalVendeVino_1 \Rightarrow$ Madrid
- $Sucursal_2$, $Empleado_2$ y $SucursalVendeVino_2 \Rightarrow$ Barcelona
- $Sucursal_3$, $Empleado_3$ y $SucursalVendeVino_3 \Rightarrow$ La Coruña
- $Sucursal_4$, $Empleado_4$ y $SucursalVendeVino_4 \Rightarrow$ Sevilla

2.4.2. CLIENTE Y CLIENTESOLICITAVINOASUCURSAL

La asignación de los fragmentos a las diferentes localidades son las siguientes:

- $Cliente_1$ y $ClienteSolicitaVinoASucursal_1 \Rightarrow$ Madrid
- $Cliente_2$ y $ClienteSolicitaVinoASucursal_2 \Rightarrow$ Barcelona
- $Cliente_3$ y $ClienteSolicitaVinoASucursal_3 \Rightarrow$ La Coruña
- $Cliente_4$ y $ClienteSolicitaVinoASucursal_4 \Rightarrow$ Sevilla

2.4.3. VINO

La asignación de los fragmentos a las diferentes localidades son las siguientes:

- $Vino_1 \Rightarrow$ Madrid
- $Vino_2 \Rightarrow$ Barcelona
- $Vino_3 \Rightarrow$ La Coruña
- $Vino_4 \Rightarrow$ Sevilla

2.4.4. PRODUCTOR Y SUCURSALSOLICITASUCURSAL

- Un productor puede producir vino en diferentes localidades, por lo que será interesante la presencia de la tabla productor completa en cada localidad.
- La relación SucursalSolicitaSucursal se replicará en las localidades donde haya sucursales, es decir, todas. De esta manera la sucursal que pida el encargo y la sucursal que reciba el encargo tendrán en sus localidades los datos del encargo.