



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

Laboratorio de:

Bases de datos distribuidas

Práctica No.:

5

Grupo No.:

9

Integrantes:

- Fernando Eliceo Huilca Villagómez
- Gregory Leandro Salazar Domínguez
- Mateo Nicolás Simbaña Guarnizo

Tema:

Implementación de Esquemas Locales y extracción de información

Objetivos:

- Crear los esquemas locales (Tablas) desde la base de datos centralizada para obtener los fragmentos correspondientes en los dos nodos de la base distribuida a través de la extracción de información usando consultas SQL.
- Implementar los diagramas de las 2 bases de datos locales para visualizar la correcta implementación de los esquemas de fragmentación y ubicación a través del uso de SQL Server y su herramienta para crear diagramas de una base de datos en específico.
- Determinar la integridad y consistencia de los datos distribuidas a través del uso de consultas SQL entre nodos para garantizar la correcta implementación de las operaciones de fragmentación y replicación.

Marco teórico:

En el diseño de Base de Datos Distribuida (BDD), la metodología descendente parte de un Esquema Lógico Global (GCS) para definir estrategias de fragmentación, asignación y replicación que determinan la distribución de los datos en los nodos de la red [1].



1. Esquema de fragmentación

La fragmentación divide tablas en subconjuntos lógicos (fragmentos) según patrones de acceso de las aplicaciones. Se rige por tres principios [1]:

- **Compleitud:** Todos los datos deben estar en algún fragmento.
- **Disyunción:** Datos no repetidos (excepto claves en fragmentación vertical).
- **Reconstrucción:** Posibilidad de reconstruir la tabla original desde los fragmentos.

1.1 Tipos de fragmentación [1]

Existen los siguientes tipos de fragmentación:

1.1.1 Fragmentación horizontal primaria

Selección de filas mediante predicados locales.

$$R_i = \sigma_{P_i}(R)$$

1.1.2 Fragmentación horizontal derivada

Predicados definidos sobre tablas relacionadas.

$$R_i = R \bowtie Q_i$$

1.1.3 Fragmentación vertical

División por columnas, preservando la clave primaria en todos los fragmentos:

$$R_i = \Pi_{L_i}(R)$$

1.1.4 Fragmentación mixta

Tenemos dos maneras, fragmentación horizontal seguida de una vertical (HV) y fragmentación vertical seguida de una horizontal (VH).

2. Esquema de replicación parcial

Tras la fragmentación, se implementa replicación selectiva de fragmentos (no tablas completas) en nodos específicos, optimizando [1]:

- **Disponibilidad:** Acceso local a fragmentos críticos.
- **Rendimiento:** Minimización de transferencias en red.
- **Consistencia:** Balance entre lecturas locales y actualizaciones sincronizadas.

2.1 Criterios de diseño



- **Minimizar Costos:** Evitar réplicas innecesarias en nodos con pocas consultas [1].
- **Optimizar Rendimiento:** Asignar fragmentos a nodos con alta frecuencia de acceso local [1].

3. Esquema de asignación

Define la ubicación de fragmentos en nodos físicos mediante su heurística [1]:

- **Priorización de procesamiento local:** Asignar fragmentos al nodo con mayor tasa de operaciones sobre ellos.
- **Replicación selectiva:** Replicar si predominan consultas desde múltiples nodos como evitar la replicación si las actualizaciones son frecuentes o la sincronización es costosa.

Desarrollo de la práctica:

1. Esquema de fragmentación

Para las fragmentaciones horizontales primarias se estableció un único campo junto con la condición de fragmentación, como se presenta en la Tabla 1.

Campo de fragmentación	<i>idSucursal</i>
Condición de fragmentación	<i>idSucursal</i> = i = {QUITO_NORTE, QUITO_SUR}

Tabla 1. Campos y condiciones definidas para las fragmentaciones horizontales primarias.

En la Tabla 2, se resumen las fragmentaciones realizadas en el diseño de la base de datos distribuida, organizadas según el tipo al que corresponden.

Fragmentación vertical	NOMINA_INSTRUCTOR $= \Pi_{cedulaInstructor, salario, fechaContratacion}(INSTRUCTOR)$
	INFORMACIÓN_INSTRUCTOR = $\Pi_{cedulaInstructor, idSucursal, nombre, apellido, telefono, email, fechaNacimiento, direccion}(INSTRUCTOR)$
	CEDULA_CLIENTE = $\Pi_{cedulaCliente}(CLIENTE)$
	CLIENTE = $\Pi_{cedulaCliente, idSuscripcion, idSucursal, nombre, apellido, telefono, email, fechaNacimiento, fechaRegistro, direccion}(CLIENTE)$



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

Fragmentación horizontal primaria	$CLIENTE_i = \sigma_{idSucursal=i}(CLIENTE)$ $i = \{QUITO_NORTE, QUITO_SUR\}$
	$SUPLEMENTO_i = \sigma_{idSucursal=i}(SUPLEMENTO)$ $i = \{QUITO_NORTE, QUITO_SUR\}$
Fragmentación horizontal derivada	$COMPRA_i = COMPRA \bowtie SUPLEMENTO_i$ $i = \{QUITO_NORTE, QUITO_SUR\}$
Fragmentación horizontal mixta (VH)	<p>Fragmentación vertical:</p> <p>Esta operación se realizó previamente sobre la tabla INSTRUCTOR, obteniendo como resultado el fragmento denominado INFORMACIÓN_INSTRUCTOR.</p> <p>Fragmentación horizontal:</p> $INFORMACION_INSTRUCTOR_i = \sigma_{idSucursal=i}(INFORMACION_INSTRUCTOR)$ $i = \{QUITO_NORTE, QUITO_SUR\}$

Tabla 2. Detalle de las fragmentaciones según su clasificación.

En la etapa de diseño de la base de datos distribuida, la fragmentación horizontal derivada se había definido de la siguiente manera:

$$COMPRA_i = COMPRA \bowtie CLIENTE_i$$
$$i = \{QUITO_NORTE, QUITO_SUR\}$$

Sin embargo, este planteamiento presentó el problema de que se colocaban compras según la sucursal de registro del cliente, no según la ubicación donde se realizó la transacción. Esto generaría varias inconsistencias. Por ejemplo, que un cliente registrado en QUITO_NORTE que compra en QUITO_SUR tendría sus transacciones guardadas en COMPRA_QUITO_NORTE, siendo información innecesaria para el nodo norte debido a que la operación ocurrió realmente en la sucursal sur.

Ahora bien, en la fragmentación derivada propuesta en la Tabla 2, la relación se establece entre *COMPRA* y *SUPLEMENTO*, lo que permite almacenar las compras en la sede donde se venden los productos, sin importar la sucursal en la que el cliente esté registrado y, de este modo, se cumple con uno de los requerimientos planteados en el escenario del problema.

2. Esquema de ubicación



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

En la Tabla 3, se muestra el esquema de ubicación, el cual incluye las relaciones originales y la distribución de sus réplicas y fragmentos en los nodos Norte y Sur de la empresa FITEC.

	QUITO_NORTE	QUITO_SUR
SUCURSAL	SUCURSAL	SUCURSAL
SUSCRIPCION	SUSCRIPCION	SUSCRIPCION
CLIENTE	CLIENTE_QUITO_NORTE	CLIENTE_QUITO_SUR
	CEDULA_CLIENTE	CEDULA_CLIENTE
SUPLEMENTO	SUPLEMENTO_QUITO_NORTE	SUPLEMENTO_QUITO_SUR
COMPRA	COMPRA_QUITO_NORTE	COMPRA_QUITO_SUR
INSTRUCTOR	INFORMACION_INSTRUCTOR_QUITO_NORTE	INFORMACION_INSTRUCTOR_QUITO_SUR
	NOMINA_INSTRUCTOR	

Tabla 3. Esquema de ubicación para la empresa FITEC.

3. Creación de los esquemas locales y extracción de datos

3.1 Nodo QUITO_NORTE

El nodo QUITO_NORTE corresponde al equipo DESKTOP-HSCPRKC, que posee la base de datos centralizada. Por ello, para extraer los datos fue suficiente con especificar la base de datos (FITEC), el esquema (dbo) y la tabla de origen en las consultas.

3.1.1 Tabla CLIENTE_QUITO_NORTE

Se aplicó una **fragmentación horizontal primaria** mediante una consulta que especificaba la creación de la tabla **CLIENTE_QUITO_NORTE**, en la cual se almacenaron los registros del fragmento vertical **CLIENTE** del nodo QUITO_NORTE (creado en la anterior práctica de replicación) cuyo **IDSUCURSAL** era 'QUITO_NORTE'. Luego, se procedió a eliminar el fragmento **CLIENTE**, puesto que el fragmento **CLIENTE_QUITO_NORTE** ya contenía los registros pertinentes para el nodo QUITO_NORTE. Además, **CLIENTE** poseía adicionalmente los registros relativos al nodo QUITO_SUR

Adicionalmente, se estableció como clave primaria una combinación del campo **CEDULACLIENTE** y el campo de fragmentación **IDSUCURSAL**. En la Figura 1, se observan los registros almacenados en esta nueva tabla.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

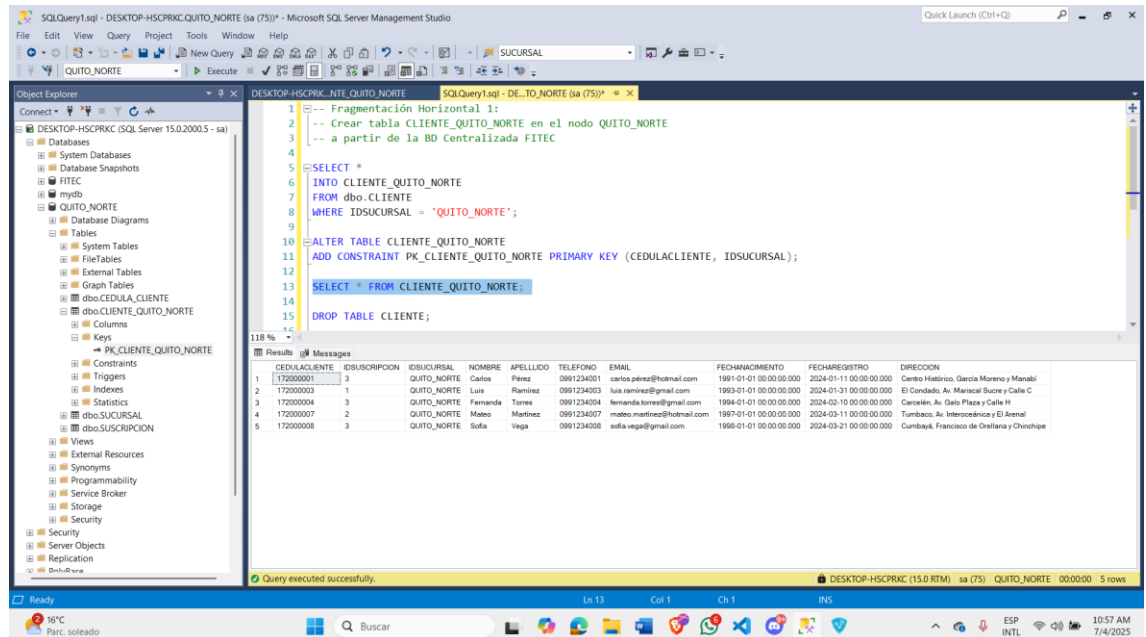


Figura 1. Creación del fragmento *CLIENTE_QUITO_NORTE*.

3.1.2 Tabla *CEDULA_CLIENTE*

Se aplicó una **fragmentación vertical** sobre la tabla **CLIENTE** para extraer la clave primaria en una nueva tabla. Esto permitió mantener una referencia de los clientes suscritos a cualquier sucursal, de modo que puedan adquirir suplementos en cualquier sede y los registros se almacenen en la sucursal donde realizaron la compra, garantizando así la consistencia de los datos.

Durante la práctica anterior de la replicación se creó la tabla **CEDULA_CLIENTE**, en la que se almacenaron las claves primarias de todos los clientes de la base de datos centralizada, como se observa en la Figura 2. Esto fue necesario para replicar la información en ambas sedes.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

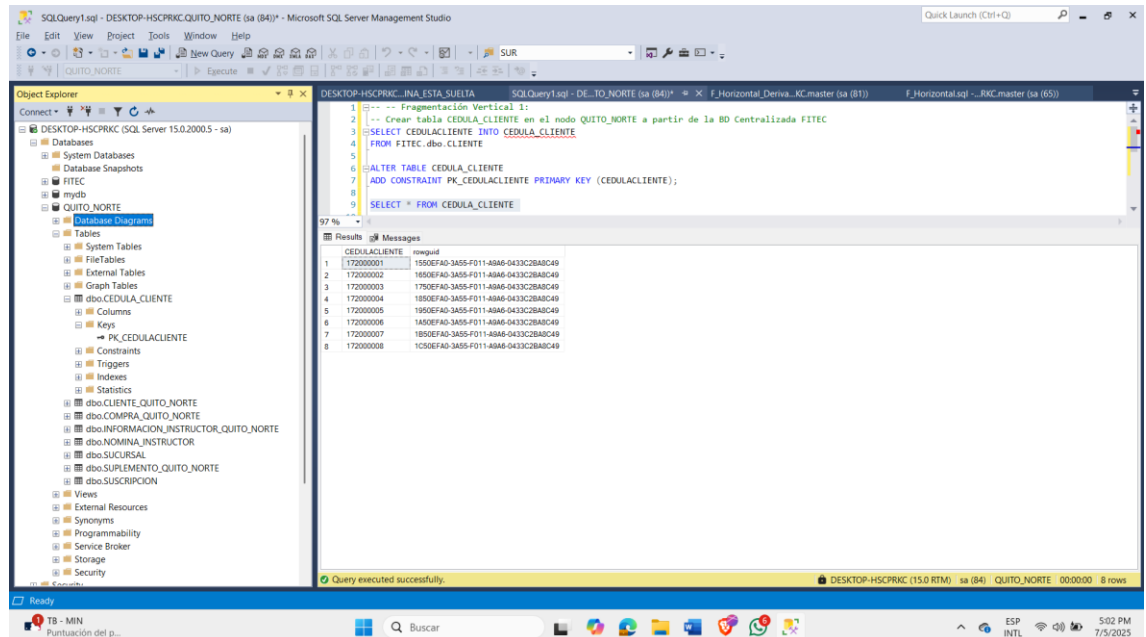


Figura 2. Creación del fragmento replicado CEDULA_CLIENTE.

3.1.3 Tabla SUPLEMENTO_QUITO_NORTE

Se realizó una **fragmentación horizontal primaria** a través de una consulta que generó la tabla **SUPLEMENTO_QUITO_NORTE**, en la que se guardaron los registros de la tabla **SUPLEMENTO** de la base de datos centralizada con **IDSUCURSAL** igual a 'QUITO_NORTE'.

Luego de esto, se definió como clave primaria la combinación de los campos **IDSUPLEMENTO** e **IDSUCURSAL**. En la Figura 3, se muestran los registros que fueron almacenados en esta nueva tabla.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

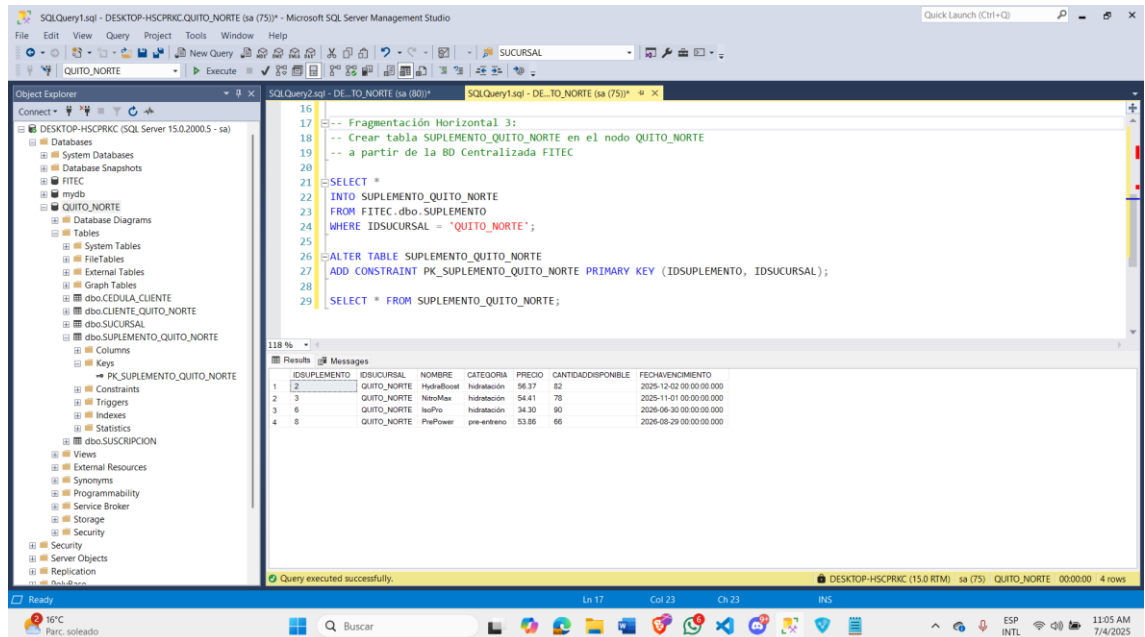


Figura 3. Creación del fragmento *SUPLEMENTO_QUITO_NORTE*.

3.1.4 Tabla *COMPRA_QUITO_NORTE*

La tabla ***COMPRA_QUITO_NORTE*** se generó aplicando una **fragmentación horizontal derivada** mediante una consulta que realizó un **SEMI JOIN** entre la tabla ***COMPRA*** de la base de datos centralizada y el fragmento previamente creado ***SUPLEMENTO_QUITO_NORTE***. Esta operación seleccionó todas las columnas y tuplas de ***COMPRA*** cuyo **IDSUPLEMENTO** existía en ***SUPLEMENTO_QUITO_NORTE***. Además, la consulta añadió el campo de fragmentación (columna) **IDSUCURSAL** mediante una subconsulta que obtiene el valor correspondiente de ***SUPLEMENTO_QUITO_NORTE*** cuando el **IDSUPLEMENTO** de ***COMPRA*** coincidía con el de dicha tabla.

Luego de almacenar únicamente los registros de las compras realizadas en el nodo QUITO_NORTE, se alteró el campo **IDSUCURSAL** para que no aceptara valores nulos, puesto que se lo consideró como parte de la clave primaria junto con el campo **IDCOMPRA**. En la Figura 4, se presentan los registros almacenados en esta nueva tabla.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

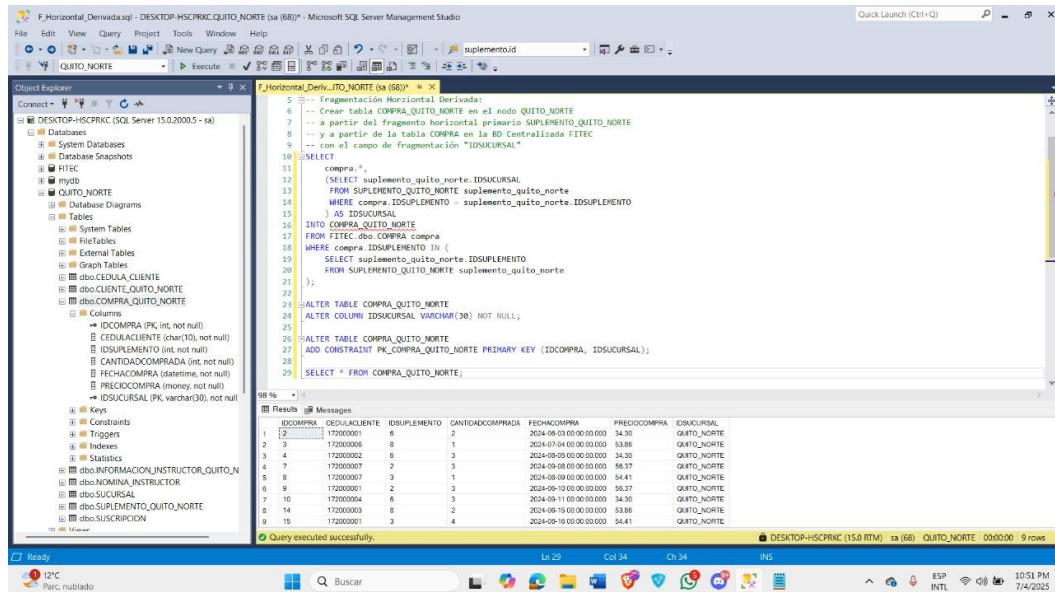


Figura 4. Creación del fragmento *COMPRA_QUITO_NORTE*.

3.1.5 Tabla *NOMINA_INSTRUCTOR*

Se aplicó una fragmentación vertical mediante una consulta que generó la tabla ***NOMINA_INSTRUCTOR***, incluyendo los campos necesarios de la tabla ***INSTRUCTOR*** de la base de datos centralizada.

Posteriormente, se asignó como clave primaria el campo ***CEDULAINSTRUCTOR***. La información correspondiente a todos los registros de los instructores se guardó en esta sede, según uno de los requerimientos proporcionados por la descripción del escenario de la empresa FITEC, como se muestra en la Figura 5.

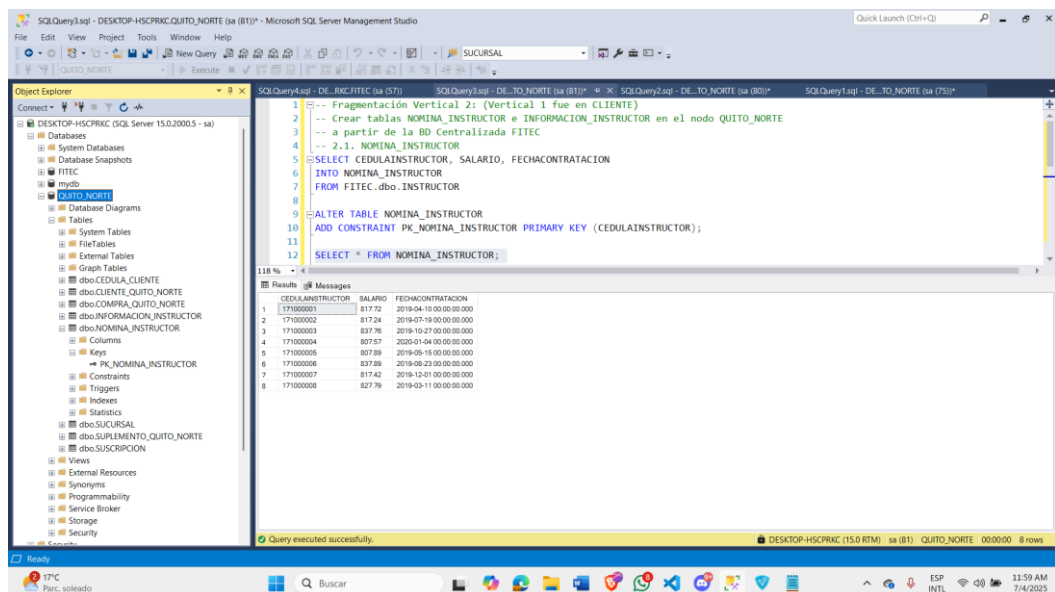


Figura 5. Creación del fragmento *NOMINA_INSTRUCTOR*.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

3.1.6 Tabla INFORMACIÓN_INSTRUCTOR_QUITO_NORTE

Para obtener esta tabla se aplicó una fragmentación mixta. En primer lugar, se realizó una fragmentación vertical mediante una consulta que generó la tabla **INFORMACION_INSTRUCTOR**, incluyendo los campos de la tabla **INSTRUCTOR** de la base de datos centralizada que no se habían considerado en la fragmentación vertical del punto 3.1.5 (**NOMINA_INSTRUCTOR**).

Una vez generada esta tabla, se asignó como clave primaria el campo **CEDULAINSTRUCTOR**. La tabla **INFORMACION_INSTRUCTOR** incluía todos los registros de los instructores, como se muestra en la Figura 6.

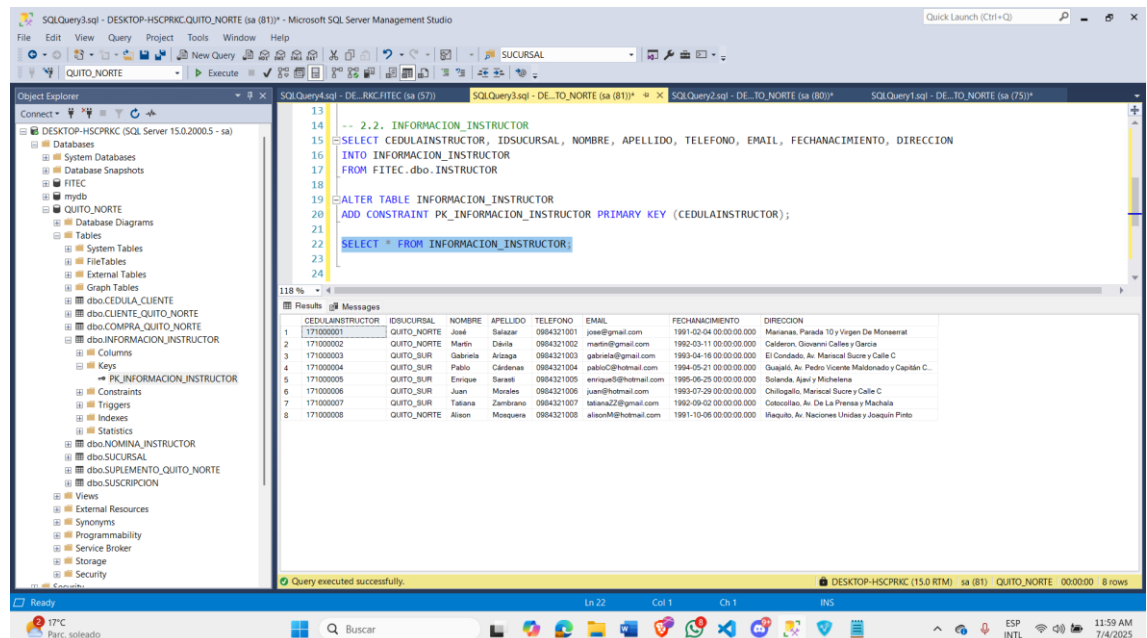


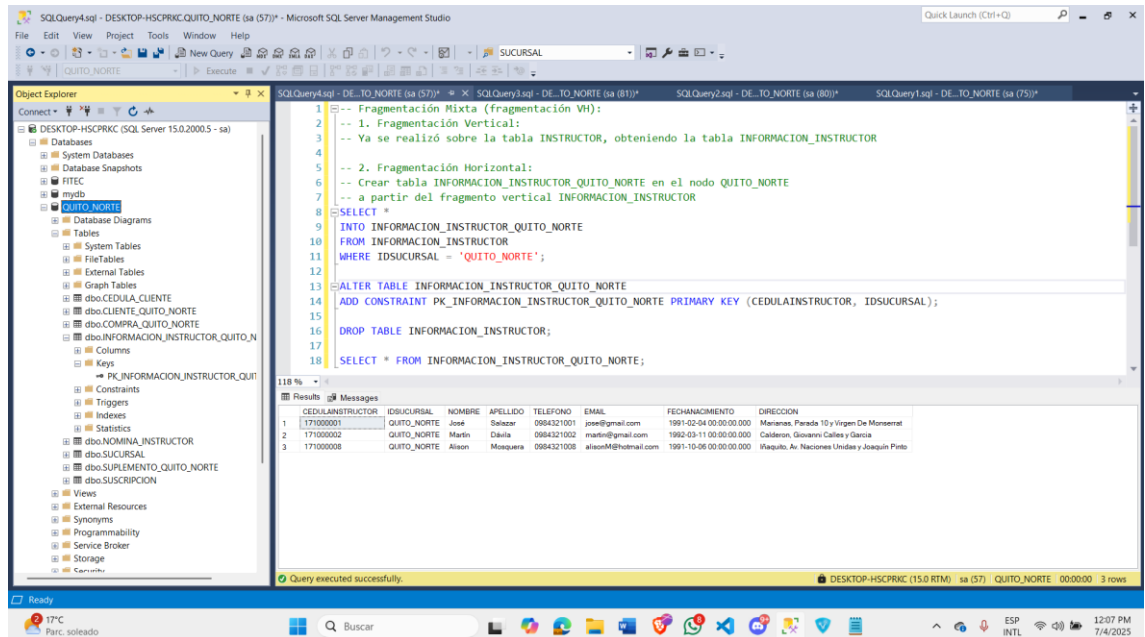
Figura 6. Creación temporal del fragmento **INFORMACION_INSTRUCTOR**.

Dado que la información de los instructores debe almacenarse únicamente en la sucursal donde trabajan, se ejecutó una consulta sobre el fragmento **INFORMACION_INSTRUCTOR** para guardar en la tabla **INFORMACION_INSTRUCTOR_QUITO_NORTE** los registros de los empleados con **IDSUCURSAL** igual a 'QUITO_NORTE'.

Por último, se estableció como clave primaria la combinación de los campos **CEDULAINSTRUCTOR** e **IDSUCURSAL** y se eliminó la tabla **INFORMACION_INSTRUCTOR** del nodo **QUITO_NORTE**, puesto que solo se utilizó como intermediaria para generar la tabla que realmente se requería. En la Figura 7, se muestran los registros de los instructores asignados a esta sucursal.



**ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN**



**Figura 7. Creación del fragmento
INFORMACION_INSTRUCTOR_QUITO_NORTE.**

3.2 Nodo QUITO_SUR

El nodo QUITO_SUR, correspondiente al equipo MATEITO, no dispone de la base de datos centralizada. Por esta razón, fue necesario establecer un linked server hacia el nodo QUITO_NORTE (computador DESKTOP-HSCPRKC) para acceder a los datos de manera remota.

Además, para extraer los datos de las tablas asociadas a este nodo, en cada consulta fue necesario especificar el servidor (DESKTOP-HSCPRKC), la base de datos (FITEC), el esquema (dbo) y el nombre de la tabla requerida (Arquitectura 4 capas).

3.2.1 Tabla CLIENTE_QUITO_SUR

Los pasos para obtener la tabla **CLIENTE_QUITO_SUR** fueron los mismos descritos en el punto 3.1.1. En la consulta se realizaron las siguientes acciones:

- Se creó la tabla **CLIENTE_QUITO_SUR**.
- Seguidamente, se indicó el servidor, la base de datos, el esquema y la tabla remota desde la cual se obtendrían los datos (Se accedió a la tabla **CLIENTE**).
- Se filtraron los registros para que solo los clientes con **IDSUCURSAL** igual a 'QUITO_SUR' se almacenen en la nueva tabla.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

Además, se estableció como clave primaria la combinación de los campos **CEDULACLIENTE** e **IDSUCURSAL**. En la Figura 8, se presentan los registros almacenados en esta tabla.

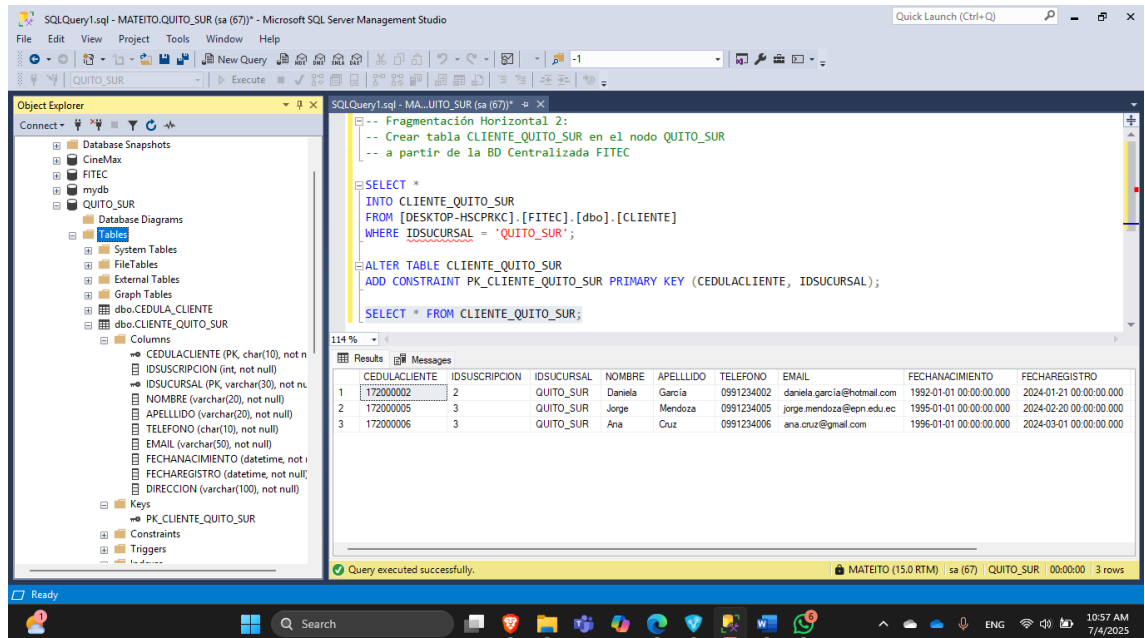


Figura 8. Creación del fragmento **CLIENTE_QUITO_SUR**.

3.2.2 Tabla CEDULA_CLIENTE

En este caso no se llevó a cabo ninguna fragmentación, dado que el resultado de la fragmentación vertical en el nodo QUITO_NORTE se replicó en el nodo QUITO_SUR conforme a la estrategia planteada en el análisis de los requerimientos de la empresa FITEC y conforme a lo realizado en la anterior práctica.

3.2.3 Tabla SUPLEMENTO_QUITO_SUR

El procedimiento para obtener la tabla **SUPLEMENTO_QUITO_SUR** fue similar al descrito en el punto 3.1.3. La consulta ejecutada contempló las siguientes acciones:

- Se creó la tabla **SUPLEMENTO_QUITO_SUR**.
- Seguidamente, se indicó el servidor, la base de datos, el esquema y la tabla remota desde donde se recuperarían los datos.
- Se filtraron los registros para que solo los suplementos con **IDSUCURSAL** igual a 'QUITO_SUR' se almacenen en la nueva tabla.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

Asimismo, se definió como clave primaria la combinación de los campos **IDSUPLEMENTO** e **IDSUCURSAL**. En la Figura 9, se visualizan los registros almacenados en esta tabla.

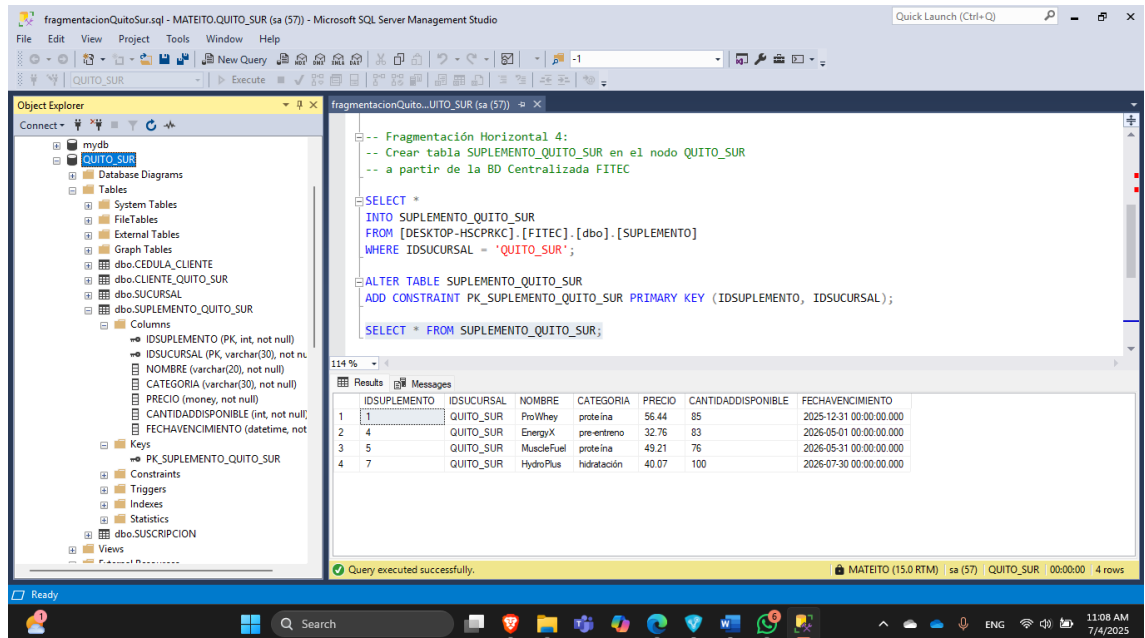


Figura 9. Creación del fragmento *SUPLEMENTO_QUITO_SUR*.

3.2.4 Tabla **COMPRA_QUITO_SUR**

El proceso para generar la tabla **COMPRA_QUITO_SUR** se realizó de forma similar a lo descrito en el apartado 3.1.4. La consulta realizada incluyó las siguientes acciones:

- Se seleccionaron las columnas de la tabla **COMPRA** de la base de datos centralizada.
- Se incorporó el campo de fragmentación **IDSUCURSAL** mediante una subconsulta que recuperó su valor desde la tabla **SUPLEMENTO_QUITO_SUR**, cuando el campo **IDSUPLEMENTO** de **COMPRA** coincidió con el campo **IDSUPLEMENTO** de **SUPLEMENTO_QUITO_SUR**.
- Se creó la tabla **COMPRA_QUITO_SUR**.
- Mediante un **SEMI JOIN** entre la tabla **COMPRA** (tabla remota) y **SUPLEMENTO_QUITO_SUR**, se filtraron los registros correspondientes a las compras realizadas en la sede QUITO_SUR.

De la misma forma, el campo **IDSUCURSAL** se modificó para no permitir valores nulos y se definió como parte de la clave primaria junto con el campo **IDCOMPRA**. En la Figura 10, se visualizan los registros almacenados en esta nueva tabla.



En la Figura 11, se puede observar que la tabla **COMPRA_QUITO_SUR** almacenó una compra con **IDCOMPRA** '5', realizada por un cliente con **CEDULACLIENTE** '1720000004', quien está registrado en la sucursal **QUITO_NORTE**. Esto sucedió porque el cliente efectuó la compra en la sede QUITO SUR.





Por lo tanto, se confirmó que los registros de compra se almacenaron correctamente, dado que **cada compra se guarda en la sede donde se realiza, independientemente de la sucursal en la que el cliente esté registrado.**

3.2.5 Tabla **INFORMACION_INSTRUCTOR_QUITO_SUR**

El proceso de creación de la tabla **INFORMACION_INSTRUCTOR_QUITO_SUR** fue similar al descrito en el apartado 3.1.6. A continuación, se detallan las dos fragmentaciones que se realizaron:

3.2.5.1 Fragmentación vertical

Mediante una subconsulta se llevaron a cabo las siguientes acciones:

- Se seleccionaron los campos requeridos de la tabla remota **INSTRUCTOR**.
- Se creó la tabla **INFORMACION_INSTRUCTOR**.
- Se especificaron el servidor, la base de datos, el esquema y la tabla remota desde donde se recuperarían los datos.
- Se insertaron los registros de todos los instructores de la base de datos centralizada.

Luego, se colocó como clave primaria el campo **CEDULAINSTRUCTOR**.

3.2.5.2 Fragmentación horizontal primaria

Después de completar la fragmentación vertical, se aplicó una fragmentación horizontal mediante una consulta que realizó las siguientes acciones:

- Se creó la tabla **INFORMACION_INSTRUCTOR_QUITO_SUR**.
- Se especificaron el servidor, la base de datos, el esquema y la tabla remota desde donde se recuperarían los datos.
- Se filtraron los registros para almacenar únicamente la información de los instructores cuyo **IDSUCURSAL** es igual a 'QUITO_SUR' en la nueva tabla.

Finalmente, se estableció como clave primaria la combinación de los campos **CEDULAINSTRUCTOR** e **IDSUCURSAL** y se eliminó la tabla **INFORMACION_INSTRUCTOR** del nodo QUITO_SUR. En la Figura 12, se visualizan los registros de los instructores asociados a esta sucursal dentro de la tabla **INFORMACION_INSTRUCTOR_QUITO_SUR**.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

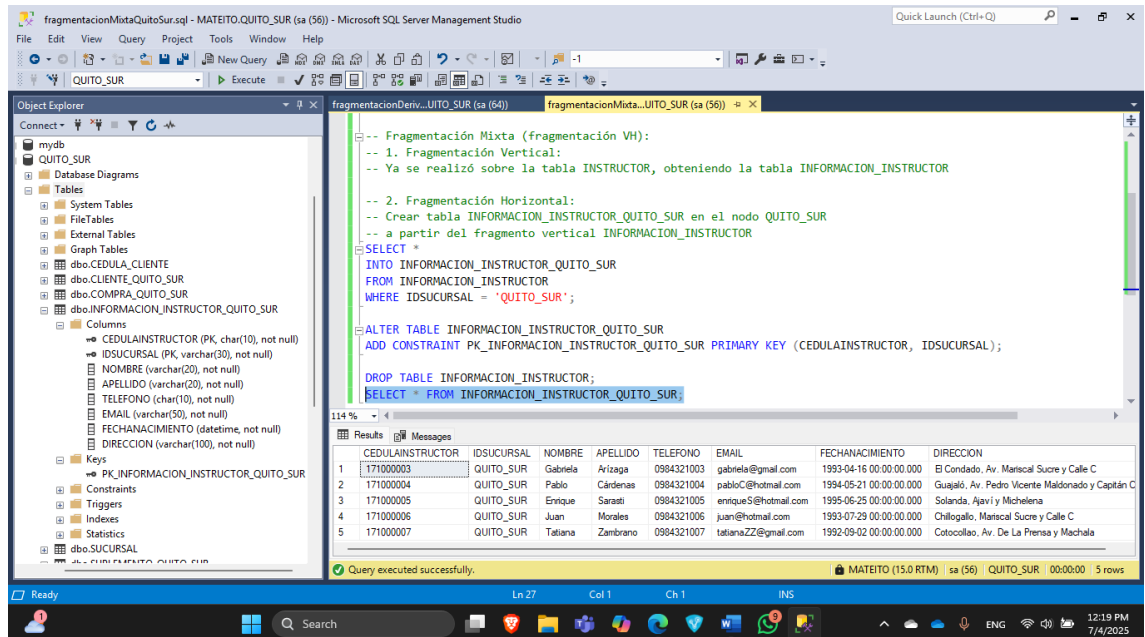


Figura 12. Creación del fragmento **INFORMACION_INSTRUCTOR_QUITO_SUR**.

4. Diagrama de las bases de datos locales

4.1 Diagrama del nodo QUITO_NORTE

Previo a la generación del diagrama del nodo **QUITO_NORTE**, se establecieron las siguientes relaciones de claves foráneas:

- Las claves primarias de las tablas **SUSCRIPCION** y **SUCURSAL** se asociaron como claves foráneas en la tabla **CLIENTE_QUITO_NORTE**.
- Las claves primarias de las tablas **CEDULA_CLIENTE** y **SUPLEMENTO_QUITO_NORTE** se asignaron como claves foráneas en la tabla **COMPRA_QUITO_NORTE**.
- La clave primaria de la tabla **SUCURSAL** se agregó como clave foránea en la tabla **SUPLEMENTO_QUITO_NORTE**.
- La clave primaria de la tabla **SUCURSAL** se colocó como clave foránea en la tabla **INFORMACION_INSTRUCTOR_QUITO_NORTE**.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

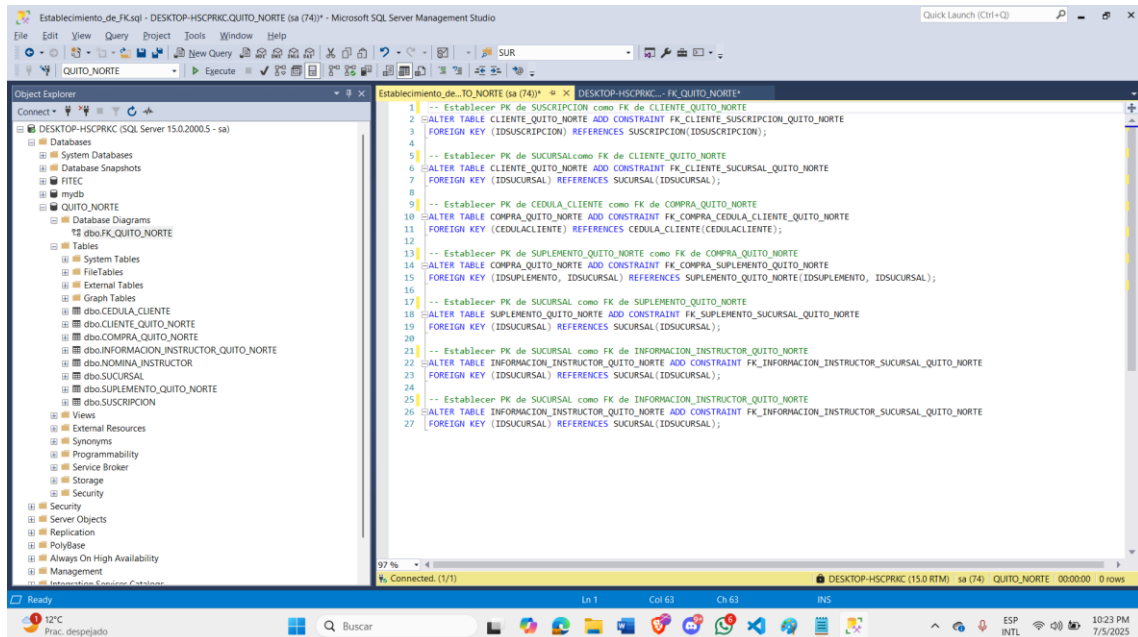


Figura 13. Configuración de claves foráneas en las tablas del nodo QUITO_NORTE.

Como resultado, en la Figura 14, se muestra el diagrama de la base de datos del nodo QUITO_NORTE.

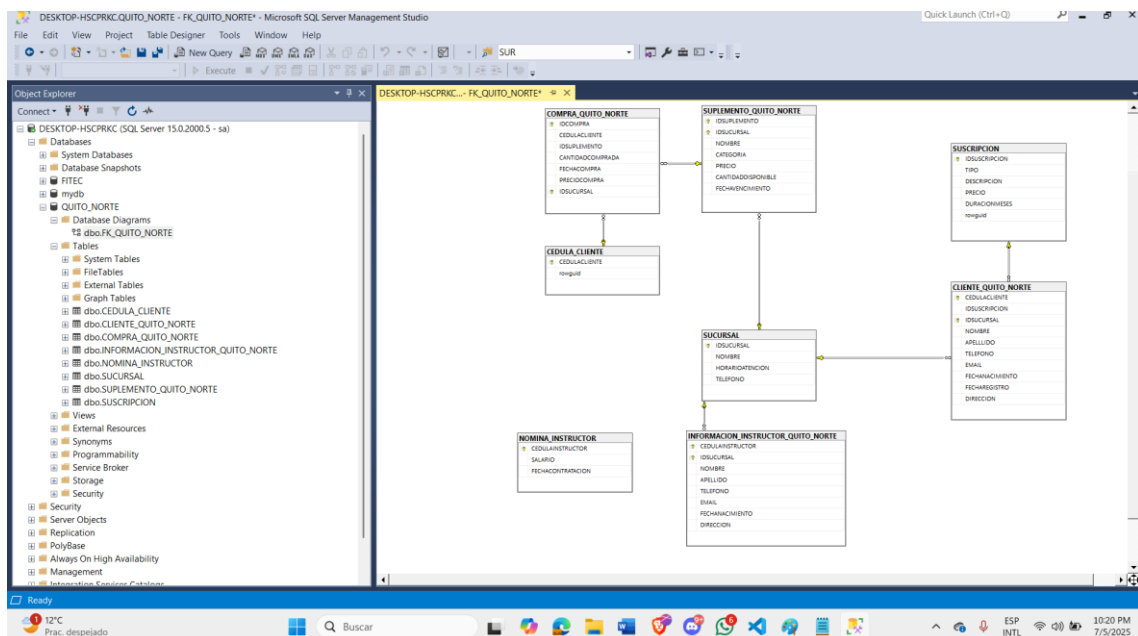


Figura 14. Diagrama de la base de datos local del nodo QUITO_NORTE.

4.2 Diagrama del nodo QUITO_SUR

Antes de obtener el diagrama del nodo QUITO_SUR, se agregaron las mismas claves foráneas del punto 4.1, utilizando los nombres de las tablas creadas en este nodo.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

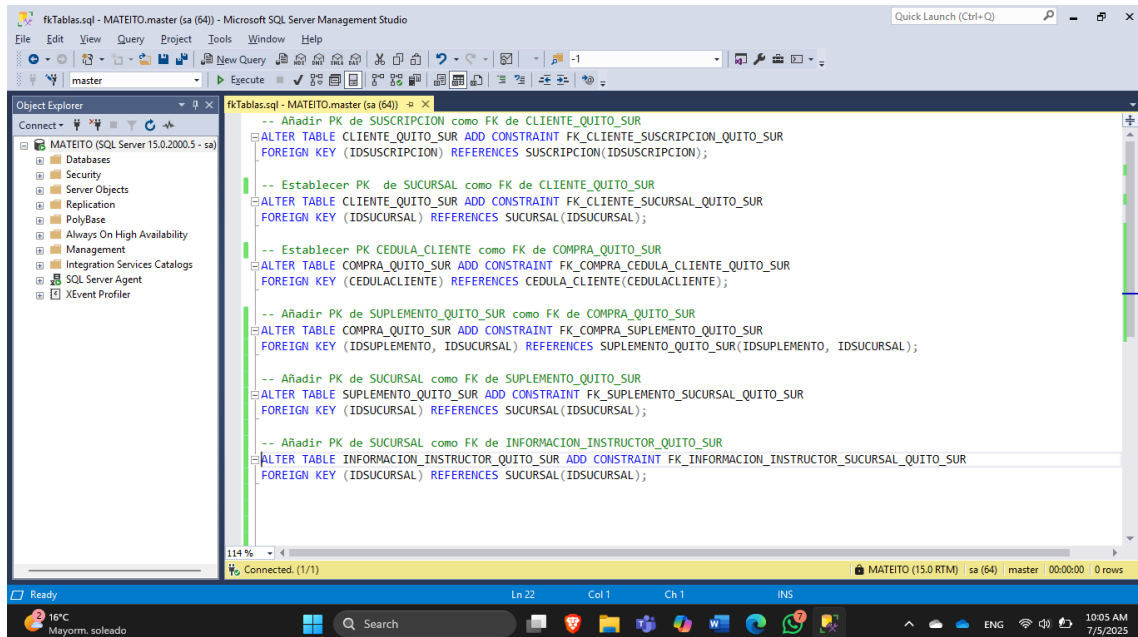


Figura 15. Asignación de claves foráneas en las tablas del nodo QUITO_SUR.

Como resultado, en la Figura 16, se presenta el diagrama de la base de datos del nodo QUITO_SUR.

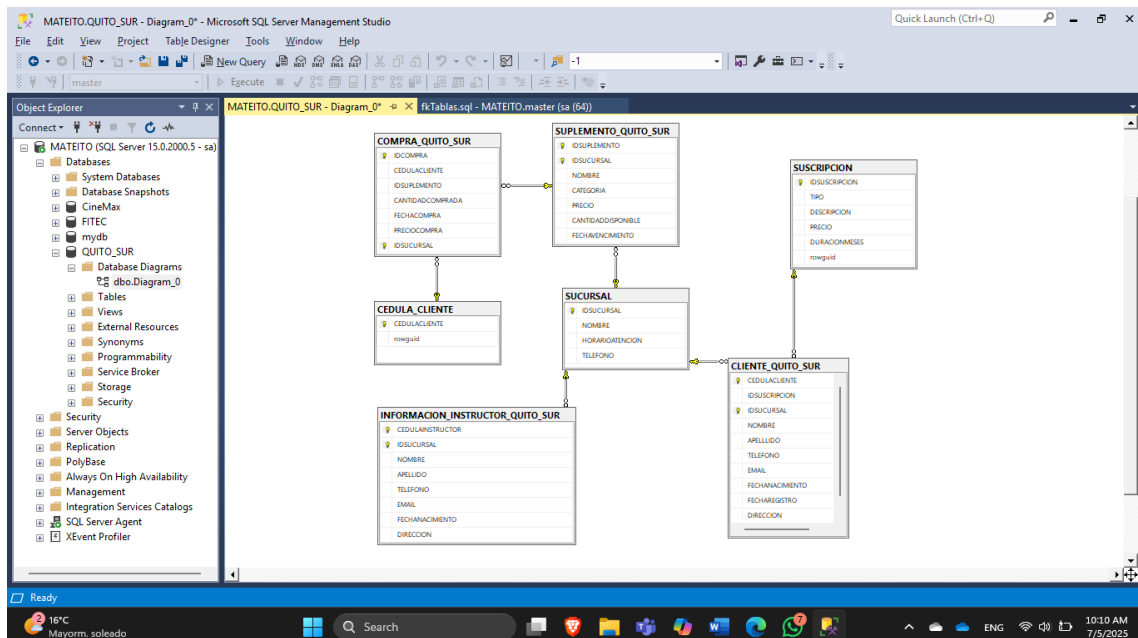


Figura 16. Diagrama de la base de datos local del nodo QUITO_SUR.

Conclusiones y recomendaciones:

1. Conclusiones



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

- Se logró crear los esquemas locales usando la base de datos centralizada. De esta manera, se obtuvieron los fragmentos correspondientes en los dos nodos de la base distribuida.
- Se consiguió implementar los diagramas de las 2 bases de datos locales. De esta manera, se pudo visualizar la correcta implementación de los esquemas de fragmentación y ubicación.
- Se determinó que sí se preservó la integridad y consistencia de los datos distribuidos entre los nodos. De esta forma, se garantizó la correcta implementación de las operaciones de fragmentación y replicación.

2. Recomendaciones

- Para la extracción de los datos, es importante que las consultas se realicen según las operaciones del álgebra relacional definidas para cada fragmentación. Por ejemplo, en una fragmentación horizontal derivada, un JOIN no equivale a un SEMI JOIN, y debe utilizarse este último en la implementación para respetar la fragmentación establecida en la etapa de diseño.
- Es sugerible tener varios archivos .sql que permitan organizar la estructura y secuencia de las fragmentaciones. De esta manera, se podrá visualizar el orden correcto para realizar las diferentes particiones horizontales, derivadas, verticales y mixtas.
- Es pertinente recordar que la copia de los datos de una tabla hacia otra tabla recién creada no copia las restricciones de primary key y foreign key. Por esta razón, es necesario siempre volver a especificar los campos que deben ser considerados como claves primarias y/o foráneas.

Bibliografía:

[1] CASTRO *et al.*, *Desarrollo de Bases de Datos. Casos prácticos desde el análisis a la implementación 2ª edición actualizada*. Alfaomega - Rama, 2013.