



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

INSTITUTO TECNOLÓGICO DE CIUDAD MADERO
DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN
MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN



"POR MI PATRIA Y POR MI BIEN"

TESIS
INTERFAZ DE COMPOSICIÓN DE CONSULTAS A BASES DE DATOS QUE
PERMITA INCLUIR FUNCIONES DE AGREGACIÓN COMBINADAS CON
SUBCONSULTAS

Que para obtener el Grado de
Maestro en Ciencias de la Computación

Presenta
Saúl Mata Alvarado
G12071166

Director de Tesis
Dr. Rodolfo Abraham Pazos Rangel

Codirector de Tesis
Dr. José Antonio Martínez Flores

Cd. Madero, Tamaulipas

Junio 2021



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

Instituto Tecnológico de Ciudad Madero
Subdirección Académica
División de Estudios de Posgrado e Investigación

Cd. Madero, Tam. **28 de mayo de 2021**

OFICIO No. : U.037/21
ASUNTO: AUTORIZACIÓN DE
IMPRESIÓN DE TESIS

C. SAÚL MATA ALVARADO
No. DE CONTROL G12071166
P R E S E N T E

Me es grato comunicarle que después de la revisión realizada por el Jurado designado para su Examen de Grado de Maestría en Ciencias de la Computación, se acordó autorizar la impresión de su tesis titulada:

“INTERFAZ DE COMPOSICIÓN DE CONSULTAS A BASES DE DATOS QUE PERMITA INCLUIR FUNCIONES DE AGREGACIÓN COMBINADAS CON SUBCONSULTAS”

El Jurado está integrado por los siguientes catedráticos:

PRESIDENTE:	DR. JUAN FRAUSTO SOLÍS
SECRETARIO:	DR. HÉCTOR JOAQUÍN FRAIRE HUACUJA
VOCAL:	DR. RODOLFO ABRAHAM PAZOS RANGEL
SUPLENTE:	DR. JOSÉ ANTONIO MARTÍNEZ FLORES
DIRECTOR DE TESIS:	DR. RODOLFO ABRAHAM PAZOS RANGEL
CO-DIRECTOR DE TESIS:	DR. JOSÉ ANTONIO MARTÍNEZ FLORES

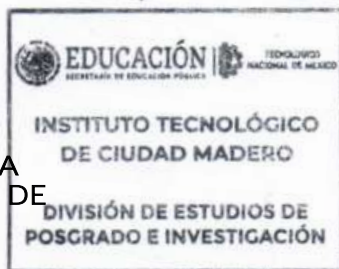
Es muy satisfactorio para la División de Estudios de Posgrado e Investigación compartir con usted el logro de esta meta. Espero que continúe con éxito su desarrollo profesional y dedique su experiencia e inteligencia en beneficio de México.

ATENTAMENTE

Excelencia en Educación Tecnológica

"Por mi patria y por mi bien"

MARCO ANTONIO CORONEL GARCÍA
JEFE DE LA DIVISIÓN DE ESTUDIOS DE
POSGRADO E INVESTIGACIÓN



c.c.p.- Archivo
MACG 'mdcoa'



Av. 1° de Mayo y Sor Juana I. de la Cruz S/N Col. Los Mangos,
C.P. 89440 Cd. Madero, Tam. Tel. 01 (833) 357 48 20, ext. 3110
e-mail: depi_cdmadero@tecnm.mx
tecnm.mx | cdmadero.tecnm.mx



Declaración de Originalidad

Declaro y prometo que este documento de tesis es producto de mi trabajo original y que no infringe los derechos de terceros, tales como derechos de publicación, derechos de autor, patentes y similares.

Además, declaro que en las citas textuales que he incluido (las cuales aparecen entre comillas) y en los resúmenes que he realizado de publicaciones ajenas, indico explícitamente los datos de los autores y las publicaciones.

Además, en caso de infracción de los derechos de terceros derivados de este documento de tesis, acepto la responsabilidad de la infracción y relevo de ésta a mi director y codirectores de tesis, así como al Tecnológico Nacional de México campus Ciudad Madero y sus autoridades.

30 de Mayo de 2021 , Cd. Madero, Tamps.



Ing. Saúl Mata Alvarado

Agradecimientos

Agradezco al CONACyT por el apoyo económico brindado, sin el cual no habría podido dedicar mi tiempo a la conclusión de este proyecto. Agradezco también al Dr. Rodolfo Pazos por su paciencia, por toda su atención brindada y por el conocimiento e ideas que aportó y que permitieron concluir satisfactoriamente este trabajo.

Así mismo quiero agradecer al Dr. José Martínez, quien además de aportar sus observaciones durante el desarrollo de este proyecto, se desempeñó como mi tutor y me informó sobre los procesos administrativos que se realizan en el Tecnológico.

Agradezco a mi familia por el gran apoyo que me han ofrecido en todo momento, por alentarme en todas las maneras posibles y por darme una razón más para esforzarme y buscar mejorar en los distintos aspectos de mi vida.

También quiero agradecer a mis compañeros, ya que durante los dos años que conviví y trabajé con ellos pude aprender la importancia del trabajo en equipo y me quedan buenos recuerdos de nuestro paso por la maestría.

Finalmente, agradezco a los integrantes de mi comité tutorial, por sus atinados comentarios y retroalimentación que me permitieron encaminar el desarrollo de este trabajo.

Dedicatoria

Dedico esta tesis a mi familia y a la memoria de mi padre.

A mi madre, quien con gran valor ha sabido guiar a nuestra familia por el buen camino.

A mi padre, quien durante su vida me enseñó la importancia del esfuerzo y la constancia.

A mi hermano, quien me ha demostrado que no hay un solo día en que no aprenda algo.

A ellos y al resto de mi familia que han apoyado mi desarrollo personal y profesional, dedico esta tesis.

Resumen

El uso de lenguaje natural como medio de consulta a bases de datos ha permitido que la información que antes sólo era accesible para usuarios expertos en algún lenguaje de consulta ahora también esté al alcance de usuarios casuales. Esto es posible gracias a las interfaces de lenguaje natural a bases de datos (ILNBDs), sin embargo, estas interfaces tienen un porcentaje de consultas correctamente contestadas de 90% aproximadamente, el cual es inadecuado para aplicaciones empresariales.

A manera de complemento a una ILNBD se ha desarrollado una interfaz de composición de consultas (ICC). Dicha interfaz permite a usuarios inexpertos construir consultas complejas sin tener conocimiento conceptual sobre bases de datos o algún lenguaje de consulta. Sin embargo, a pesar de que la ICC soporta el manejo de subconsultas, no es posible construir consultas que involucren funciones de agregación.

Este proyecto de tesis se enfoca en la adición de una nueva funcionalidad a una ICC. Esta funcionalidad permite a usuarios componer consultas que involucren funciones de agregación y subconsultas. De tal manera que ahora es posible para usuarios casuales construir consultas con un mayor grado de complejidad.

Contenido

Capítulo 1	1
Introducción.....	1
1.1 Planteamiento del problema	2
1.2 Objetivos	3
1.3 Justificación y beneficios	3
1.4 Alcance y limitaciones	4
Capítulo 2	6
Marco conceptual y estado del arte	6
2.1 Marco conceptual	6
2.1.1 Lenguaje natural	6
2.1.2 Lenguaje formal.....	6
2.1.3 Base de datos	6
2.1.4 Sistema manejador de bases de datos (SMBD).....	6
2.1.5 SQL.....	7
2.1.6 Composición.....	7
2.1.7 Consultas	7
2.1.8 Subconsultas	8
2.1.9 Funciones de agregación	8
2.2 Estado del arte	9
2.2.1 Query Builder (2004).....	9
2.2.2 Active Query Builder (2018).....	10
2.2.3 Microsoft Office Access (2016)	11
2.2.4 Conceptual Authoring (2007).....	12
2.2.5 Herramienta para consultas basadas en ejemplos (2000)	13
2.2.6 Comparación de interfaces descritas	14
2.2.7 Conclusiones.....	14
Capítulo 3	16
Versión anterior de la interfaz	16
3.1 Arquitectura.....	16
3.2 Árbol de composición	18
3.3 Descomposición de consulta en LN	19
3.4 Generación automática de subconsultas intermedias	20
Capítulo 4	21
Metodología de solución	21
4.1 Tipos de consultas a tratar	22
4.2 Interfaz de configuración.....	22
4.3 Modificación del DIS	23
4.4 Árbol de composición	24
4.5 Selección de elementos de interés	25
4.6 Vista previa y resultado	26
Capítulo 5	28
Interfaz de composición.....	28
5.1 Conceptos para composición de consultas que involucran funciones de agregación combinadas con subconsultas.....	28
5.2 Descomposición de la consulta en LN	29

5.3 Selección del tema de interés	30
5.4 Especificación de condiciones de búsqueda.....	32
5.5 Resultado de subconsulta interna	34
5.6 Especificación de condición de búsqueda para el resultado de subconsultas	35
5.7 Selección de elementos de interés	36
5.8 Construcción en lenguaje SQL de consultas que involucran funciones de agregación combinadas con subconsultas.....	39
5.9 Vista previa y resultado.....	41
5.10 Proceso de composición de consultas que involucran funciones de agregación combinadas con subconsultas.....	43
Capítulo 6	51
Pruebas experimentales	51
6.1 Pruebas funcionales.....	51
6.1.1 Descripción de las pruebas funcionales.....	51
6.1.2 Consulta tipo 1.....	51
6.1.3 Consulta tipo 2.....	53
6.1.4 Consulta tipo 3.....	53
Capítulo 7	55
Conclusiones y trabajos futuros.....	55
7.1 Conclusiones	55
7.2 Trabajos futuros.....	56
Apéndices	57
Apéndice A. Descripción de la BD ATIS	57
Apéndice B. Descripción de la BD Geobase.....	65
Referencias	69

Lista de figuras

Figura 2.1. Interfaz de Query Builder.....	10
Figura 2.2. Interfaz de ActiveQueryBuilder.....	11
Figura 2.3. Interfaz de Microsoft Office Access 2016.....	12
Figura 2.4. Estructura de consulta en CLEF.....	13
Figura 2.5. Ejemplo de consulta en la interfaz.....	14
Figura 3.1. Arquitectura de la versión anterior de la interfaz.....	17
Figura 3.2. Árbol de composición.....	18
Figura 3.3. Ejemplo de descomposición en una consulta externa y una subconsulta.....	19
Figura 3.4. Ejemplo de descomposición en una consulta externa y dos subconsultas.....	19
Figura 4.1. Diagrama conceptual del proyecto.....	21
Figura 4.2. Edición de descripciones de FAs.....	23
Figura 4.3. Árbol de composición.....	24
Figura 5.1. Ventana de <i>Descomposición de consulta en LN</i>	30
Figura 5.2. Ventana de <i>Selección del tema de interés</i>	31
Figura 5.3. Ventana de <i>Especificación de condiciones de búsqueda</i>	33
Figura 5.4. Operadores de búsqueda.....	34

Figura 5.5. Ventana de <i>Resultado de subconsulta interna</i>	35
Figura 5.6. Ventana de <i>Especificación de condición para subconsultas</i>	36
Figura 5.7. Ventana de <i>Selección de elementos de interés para consultas con FA</i>	37
Figura 5.8. Ventana de <i>Selección de elementos de interés</i>	38
Figura 5.9. Obtención de descripciones de funciones de agregación.....	39
Figura 5.10. Arreglo funciones de agregación.....	39
Figura 5.11. Ejemplo del arreglo funciones de agregación.....	40
Figura 5.12. Arreglo selección de elementos con funciones de agregación.....	40
Figura 5.13. Ejemplo del arreglo selección de elementos con FAs.....	40
Figura 5.14. Construcción de consulta con FA.....	41
Figura 5.15. Ejemplo de construcción de consulta con FA.....	41
Figura 5.16. Ventana de <i>Vista previa y resultado</i>	42
Figura 5.17. Ejemplo de descomposición de consulta.....	43
Figura 5.18. Ventana de <i>Selección del tema de interés</i>	44
Figura 5.19. Ventana de <i>Especificación de condiciones de búsqueda</i>	45
Figura 5.20. Ventana de <i>Vista previa y resultado para subconsulta</i>	46
Figura 5.21. Ventana de <i>Selección del tema de interés para la consulta principal</i>	47
Figura 5.22. Ventana de <i>Selección de elementos de interés para consultas con FA</i>	48
Figura 5.23. Ventana de <i>Resultado de subconsulta interna</i>	48
Figura 5.24. Ventana de <i>Especificación de condición para subconsultas</i>	49
Figura 5.25. Ventana de <i>Vista previa y resultado</i>	50
Figura A.1. Esquema de la base de datos ATIS.....	64
Figura B.1. Esquema de la base de datos Geobase.....	68

Lista de tablas

Tabla 2.1. Comparación de interfaces de composición de consultas.....	14
Tabla 4.1. Tipos de consultas a tratar.....	22
Tabla 5.1. Conceptos básicos para la composición de consultas.....	28
Tabla 5.2. Componentes de la ventana de <i>Descomposición de consulta en LN</i>	29
Tabla 5.3. Componentes de la ventana de <i>Selección del tema de interés</i>	30
Tabla 5.4. Componentes de la ventana de <i>Especificación de condiciones de búsqueda</i>	32
Tabla 5.5. Componentes de la ventana de <i>Resultado de subconsulta interna</i>	34
Tabla 5.6. Componentes de la ventana de <i>Especificación de condición para subconsultas</i>	35
Tabla 5.7. Componentes de la ventana de <i>Selección de elementos de interés</i>	36
Tabla 5.8. Componentes de la ventana de <i>Vista previa y resultado</i>	41

Capítulo 1

Introducción

El avance tecnológico que se ha logrado a lo largo de las últimas décadas supone un gran cambio en la manera en cómo se almacenan y son accedidos los datos. Este avance tecnológico acarrea con sí mismo la necesidad de garantizar la seguridad y fiabilidad de la información, y como una respuesta a esta necesidad, y a consecuencia de ella, es que las bases de datos (BDs) han adquirido el nivel de importancia que poseen hoy en día.

Actualmente las BDs son ampliamente utilizadas en todos los niveles empresariales y organizacionales. Los usuarios expertos en algún lenguaje de consulta, como SQL (*Structured Query Language*), son quienes se encargan de acceder a los datos y manipularlos. Por lo tanto, los usuarios que no dominen un lenguaje de consulta tendrán dificultad para hacer uso de los datos contenidos en las BDs.

Con la intención de facilitar el acceso a las BDs y explotar aún más la información contenida en éstas, y sin tener necesidad de dominar algún lenguaje de consulta a bases de datos, se ha planteado hacer uso del lenguaje natural (LN) como medio de consulta, permitiendo a usuarios inexpertos acceder a la información que sea requerida en situaciones específicas.

Para atender esta necesidad, se han desarrollado herramientas conocidas como interfaces de lenguaje natural a bases de datos (ILNBD). Desafortunadamente, estas herramientas pueden cometer errores en la interpretación de las consultas formuladas, lo cual finalmente llevaría a entregar resultados erróneos. Los usuarios de estas herramientas pueden encontrarse con algunos problemas, por ejemplo, las malas interpretaciones de la semántica del lenguaje utilizado, una mala formulación de la consulta, o incluso que la información requerida no se encuentre en la base de datos, entre otros. La ocurrencia de estos problemas puede ocasionar que el usuario se lleve una mala experiencia al utilizar la interfaz y que, por tanto, dude de la veracidad de los resultados entregados por la misma.

A fin de mejorar el desempeño de una ILNBD (descrita en [Aguirre, 2014]), se está desarrollando como complemento una interfaz de composición de consultas (ICC) para que aquellos usuarios a los que se les dificulte formular consultas en LN puedan obtener los resultados esperados.

El objetivo del presente proyecto consiste en la adición de una nueva funcionalidad a una interfaz de composición de consultas, previamente desarrollada, que permita a usuarios inexpertos componer de manera sencilla mediante una interfaz intuitiva, consultas que involucran funciones de agregación combinadas con subconsultas. De esta forma se busca ampliar el rango de consultas en lenguaje natural que puedan ser respondidas correctamente y a su vez mantener el proceso de composición con el mismo nivel de facilidad que se ha conseguido en las versiones anteriores de la interfaz.

1.1 Planteamiento del problema

La interfaz de composición permite a usuarios inexpertos formular consultas a una base de datos mediante la combinación de una interfaz gráfica y lenguaje natural, por lo que la necesidad de dominar un lenguaje de consulta pasa a ser irrelevante. A menudo las consultas formuladas por los usuarios requieren información que se encuentra contenida en más de una tabla, por lo que es necesario encontrar la manera de obtener y presentar los resultados. Otra de las posibilidades es que las consultas formuladas por los usuarios necesiten hacer uso de funciones de agregación para llegar a un resultado.

La siguiente es una consulta que ejemplifica lo descrito anteriormente:

¿Cuál es la capital del estado con la mayor población?

Para esta consulta, la nueva versión debe generar como salida la siguiente instrucción en SQL:

```
SELECT capital
FROM state
WHERE population =
      (SELECT MAX (population)
       FROM state)
```

Para una consulta como la de este ejemplo, el problema consiste en diseñar nuevas ventanas para la interfaz que permitan al usuario definir *estado con mayor población* de manera gráfica, a semejanza y compatible con las ventanas que tiene implementadas la versión actual. Otra parte del problema consiste en integrar estas ventanas con las demás ventanas de la interfaz para definir el resto de la consulta: *Cuál es la capital del estado*. Finalmente, otra parte del problema consiste en generar la instrucción en SQL.

El propósito de este proyecto es dotar a la interfaz de composición de la capacidad de componer consultas que involucran funciones de agregación combinadas con subconsultas. Es necesario aclarar que la versión actual de la interfaz ya permite formular subconsultas, pero aún no es posible responder consultas que incluyan funciones de agregación, por lo tanto, este proyecto se encargará principalmente de resolver este problema manteniendo así mismo la compatibilidad con el módulo de subconsultas añadido en la versión anterior de la interfaz.

En la composición de consultas que involucran funciones de agregación es posible encontrar los siguientes casos:

Caso 1: La función de agregación se encuentra en la subconsulta más interna. Un ejemplo sería la siguiente consulta en lenguaje natural:

¿Cuál aerolínea tiene el vuelo con la tarifa de viaje sencillo más barata?

Caso 2: La función de agregación se encuentra en la subconsulta más interna, pero la consulta externa y subconsulta(s) sólo involucran información de una tabla. El siguiente es un ejemplo de este tipo de consulta:

¿Cuál es el estado con el área más grande?

Caso 3: La función de agregación se encuentra en la consulta externa. La siguiente consulta en lenguaje natural es un ejemplo de este caso:

¿Cuál es la longitud del río más grande del estado de Oregon?

1.2 Objetivos

OG.1 Ampliar la funcionalidad de una interfaz de composición de consultas (desarrollada incrementalmente en 2 tesis anteriores) con el fin de conseguir que permita componer consultas gráficamente que involucren funciones de agregación (COUNT, SUM, AVG, MAX y MIN) combinadas con subconsultas.

Para alcanzar el objetivo general (OG.1) se requiere alcanzar los siguientes objetivos específicos:

- OE.1) Analizar las consultas que involucren funciones de agregación combinadas con subconsultas, e identificar problemas para el tratamiento de este tipo de consultas.
- OE.2) Realizar una tipificación de problemas en consultas para construir y/o utilizar un corpus de consultas con base en la tipificación.
- OE.3) Diseñar algoritmos para el tratamiento de cada problema.
- OE.4) Diseñar e implementar programas que permitan componer consultas mediante una interfaz gráfica, las cuales involucren funciones de agregación combinadas con subconsultas.

1.3 Justificación y beneficios

Una enorme y creciente cantidad de información se genera día a día como producto de las diversas actividades, ya sean empresariales, administrativas, organizacionales, entre otras, que son realizadas por cada individuo. Una gran parte de esa información generada se encuentra almacenada en bases de datos, por lo cual es cada vez más necesario contar con sistemas que permitan consultar de manera eficiente los datos almacenados. Sin embargo, para poder manipular la información, es necesario que los usuarios dominen un lenguaje de consulta. Por lo tanto, no se puede esperar que aquellos usuarios que no dominen un lenguaje de consulta obtengan los resultados deseados.

Con el desarrollo de las ICCs es posible facilitar la tarea de acceder a información almacenada en BDs, incluso para usuarios que no tienen conocimiento sobre lenguajes de consulta ni sobre la estructura o contenido de la base de datos.

Ninguna ICC podría estar completa sin una funcionalidad que le permita responder consultas que involucren funciones de agregación. Por lo tanto, el presente proyecto se enfocará en añadir dicha funcionalidad a la interfaz permitiendo así que un tipo de consultas que hasta el momento no pueden ser respondidas por la interfaz, puedan ser procesadas y respondidas.

Ejemplo de consulta que se tratará en este proyecto.

¿Cuál estado tiene el lago más grande?

la cual se formula en SQL de la siguiente manera:

```
SELECT state_name FROM state WHERE abbreviation IN
      (SELECT state_abbreviation FROM lakestate WHERE lake_id IN
        (SELECT lake_id FROM lake WHERE area =
          (SELECT MAX (area) FROM lake)))
```

Los beneficios principales del proyecto serán los siguientes:

- Permitir a usuarios inexpertos obtener información almacenada en una base de datos de manera sencilla mediante la combinación de una interfaz gráfica y lenguaje natural, y sin necesidad de capacitarse en algún lenguaje de consulta (como SQL).
- Dotar a la ICC de la funcionalidad para formular consultas que involucren funciones de agregación combinadas con subconsultas, lo cual no es posible con la versión actual de la interfaz.
- Evitar que los usuarios inexpertos cometan errores relacionados con el uso de texto libre.

1.4 Alcance y limitaciones

A continuación se definen los alcances de este proyecto:

- La nueva funcionalidad que se implementará en el módulo de composición de consultas debe mantener la independencia de dominio de las dos versiones previas, descritas en [Aguirre, 2015] y [Castillo, 2016].
- La nueva funcionalidad debe ser fácil de usar y permitir obtener los resultados esperados por los usuarios.
- La composición de consultas que involucran funciones de agregación y subconsultas debe tener coherencia con la estructura de la base de datos.
- La interfaz debe permitir componer correctamente consultas sólo de los casos 1, 2 y 3.

Las limitaciones del proyecto se definen a continuación:

- La interfaz de composición será configurada únicamente en idioma español para la experimentación.

- No se proporcionará información que no se encuentre en la base de datos para la composición de consultas que involucran subconsultas, es decir, no se tratarán consultas de bases de datos deductivas.
- No se consideran las consultas que hacen uso de las cláusulas Having y Group by.

Capítulo 2

Marco conceptual y estado del arte

2.1 Marco conceptual

A continuación se definen la terminología y algunos conceptos básicos relacionados al desarrollo de este proyecto.

2.1.1 Lenguaje natural

El lenguaje natural es el medio que utilizamos de manera cotidiana para establecer nuestra comunicación con las demás personas. Un lenguaje natural tiene un gran poder expresivo debido a la riqueza del componente semántico. Esto dificulta aún más la formalización completa de su gramática [Cortez, 2009].

2.1.2 Lenguaje formal

El lenguaje formal es aquél que el hombre ha desarrollado para expresar las situaciones que ocurren en cada área del conocimiento científico y tecnológico. Los lenguajes formales se pueden utilizar para modelar una teoría de la mecánica, física, matemática, ingeniería eléctrica, o de otra naturaleza, con la ventaja de que, en éstos, toda ambigüedad es eliminada [Cortez, 2009].

2.1.3 Base de datos

- Es una colección de datos persistente, lógicamente coherente y con un significado inherente que es relevante para algunos aspectos del mundo real.
- Es un conjunto de datos almacenados sin redundancias innecesarias en un soporte informático y accesible simultáneamente por distintos usuarios y aplicaciones. Los datos deben estar estructurados y almacenados de forma totalmente independiente de las aplicaciones que utilizan la base de datos [Cobo, 2007].

2.1.4 Sistema manejador de bases de datos (SMBD)

- Es una colección de programas que maneja la estructura de una base de datos y controla el acceso a los datos almacenados en la base de datos. El SMBD sirve como el intermediario entre el usuario y la base de datos. La estructura de la base de datos, por sí misma, se almacena como una colección de archivos, y la única manera de acceder a los datos en esos archivos es por medio del SMBD [Coronel, 2016].

- Es un software o conjunto de programas que permite crear y mantener una base de datos. El SMBD actúa como interfaz entre los programas de aplicación y el sistema operativo. El objetivo principal de un SMBD es proporcionar un entorno eficiente a la hora de almacenar y obtener la información de la base de datos [Cobo, 2007].

2.1.5 SQL

SQL es el lenguaje estándar ANSI/ISO de definición, manipulación y control de bases de datos relacionales. SQL es un lenguaje formal parecido al lenguaje natural; concretamente, se parece al inglés y es muy expresivo. Por estas razones, y como lenguaje estándar, el SQL es un lenguaje con el que se puede acceder a todos los SMBDs relacionales comerciales [Camps, 2005].

2.1.6 Composición

Componer se define, según el diccionario de la RAE, como la acción de formar varias cosas de una, juntándolas y colocándolas con cierto modo y orden. Dicho de tomar varias partes para formar o construir un todo.

2.1.7 Consultas

Una consulta es una instrucción que solicita que se obtenga información de una base de datos. La parte de los lenguajes de manipulación de datos implicada en la obtención de información se denomina lenguaje de consultas. Aunque técnicamente sea incorrecto, resulta habitual usar las expresiones lenguaje de consultas y lenguaje de manipulación de datos como sinónimas [Silberschatz, 2006].

Una consulta básica se compone por las siguientes cláusulas:

- Select.
- From.
- Where.

La estructura básica de una consulta es la siguiente:

```
SELECT columna_1  
FROM tabla_1  
WHERE condición_1
```


2.1.8 Subconsultas

Una subconsulta es una expresión SELECT FROM WHERE que se anida dentro de otra consulta [Silberschatz, 2006], por ejemplo:

```
SELECT columna_1
FROM tabla_1
WHERE columna_2 IN
      (SELECT columna_3
       FROM tabla_2
       WHERE condición_1)
```

- Una aplicación común de las subconsultas consiste en efectuar comprobaciones sobre pertenencia a conjuntos, comparación de conjuntos y determinar la cardinalidad de conjuntos.
- El conectivo IN comprueba la pertenencia a un conjunto, donde el conjunto es la colección de valores resultado de una cláusula Select. El conectivo NOT IN comprueba la no pertenencia a un conjunto.

SQL ofrece la posibilidad de anidar subconsultas en la cláusula Where, de modo que la consulta más externa puede efectuar una gran variedad de operaciones sobre el resultado de la subconsulta. Por ejemplo, la comprobación de relaciones vacías o de pertenencia de valores al resultado de una subconsulta.

2.1.9 Funciones de agregación

Las funciones de agregación son funciones que toman una colección (un conjunto o multiconjunto) de valores como entrada y devuelven un solo valor. Por ejemplo, la función de agregación SUM toma un conjunto de valores y devuelve su suma. Los conjuntos sobre los que operan las funciones de agregación pueden contener valores repetidos; el orden en el que aparezcan los valores no tiene importancia [Silberschatz, 2006].

SQL ofrece cinco funciones de agregación incorporadas:

- COUNT devuelve el número total de filas seleccionadas por la consulta.
- MIN devuelve el valor mínimo de la columna especificada.
- MAX devuelve el valor máximo de la columna especificada.
- SUM suma los valores de la columna especificada.
- AVG devuelve el valor promedio de la columna especificada.

La estructura básica de una consulta que involucra funciones de agregación y la cláusula Group by es la siguiente:

```
SELECT función_de_agregación (columna_1)
FROM tabla_1
```

```
WHERE condición_1  
GROUP BY (columna_2)
```

La siguiente consulta es un ejemplo del uso de la función de agregación SUM:

Dame el área total de los lagos por estado

la cual se formula en SQL de la siguiente manera:

```
SELECT SUM (area)  
FROM lake  
GROUP BY state
```

2.2 Estado del arte

A continuación se presentan las descripciones de algunas herramientas utilizadas para formular consultas a bases de datos.

2.2.1 Query Builder (2004)

La interfaz QB (descrita en [Little, 2004]) tiene dos modos de composición de consultas: por medio de LN o usando menús. Tiene además la particularidad de poder conectarse con distintos manejadores de bases de datos. Otra de sus características es que no requiere que los usuarios dominen algún lenguaje de consulta.

Esta interfaz puede trabajar con consultas basadas en números que involucren operadores de comparación e incluso soporta el uso de funciones de agregación; pero no es posible formular consultas que incluyan subconsultas.

Una de las características destacables de esta interfaz es la retroalimentación que proporciona al usuario cuando un error es detectado mientras se construye la consulta. De este modo, la interfaz puede sugerir alternativas para guiar al usuario en el proceso de construcción de la consulta.

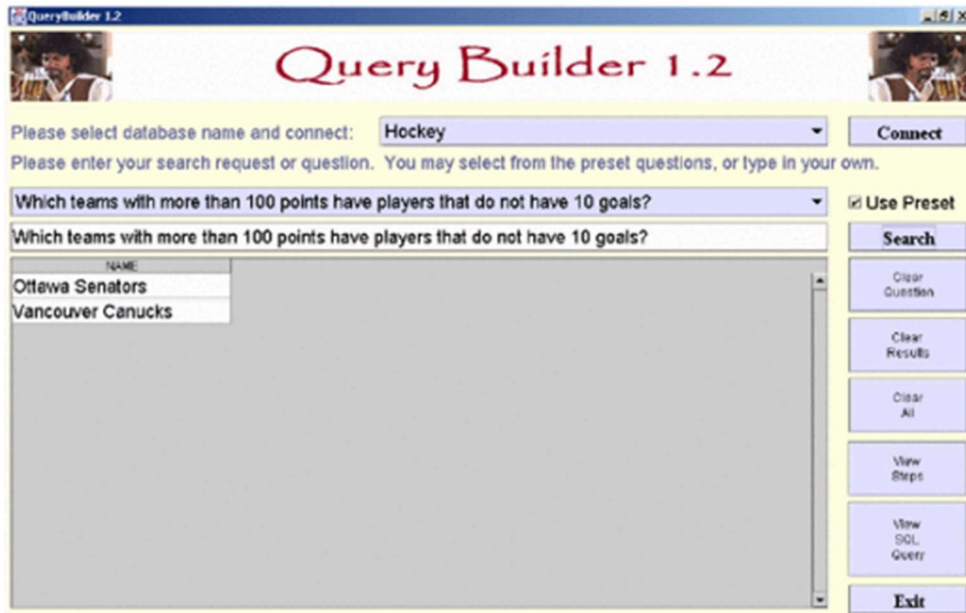


Figura 2.1. Interfaz de Query Builder

2.2.2 Active Query Builder (2018)

La interfaz Active Query Builder (descrita en [ActiveDBSoft, 2018]) es una interfaz para web con un diseño atractivo que permite formular consultas en SQL de manera fácil y rápida. El método de composición usado en este sistema es “drag & drop”, por lo que el usuario no necesita dominar SQL, pero requiere tener algún conocimiento básico de bases de datos para poder reconocer las tablas de las que desea obtener información.

El proceso de composición inicia posicionándose en el esquema donde se muestran las tablas de la base de datos, y después se debe arrastrar la tabla deseada hacia el área principal de la interfaz. Después, se seleccionan las columnas de interés. En la sección “SQL TEXT” de la interfaz se encuentra el código en SQL generado por los pasos anteriores.

Esta interfaz puede trabajar con diferentes bases de datos y soporta además el uso de subconsultas. También debe mencionarse que esta interfaz permite que los usuarios vean la estructura de las tablas de la base de datos.

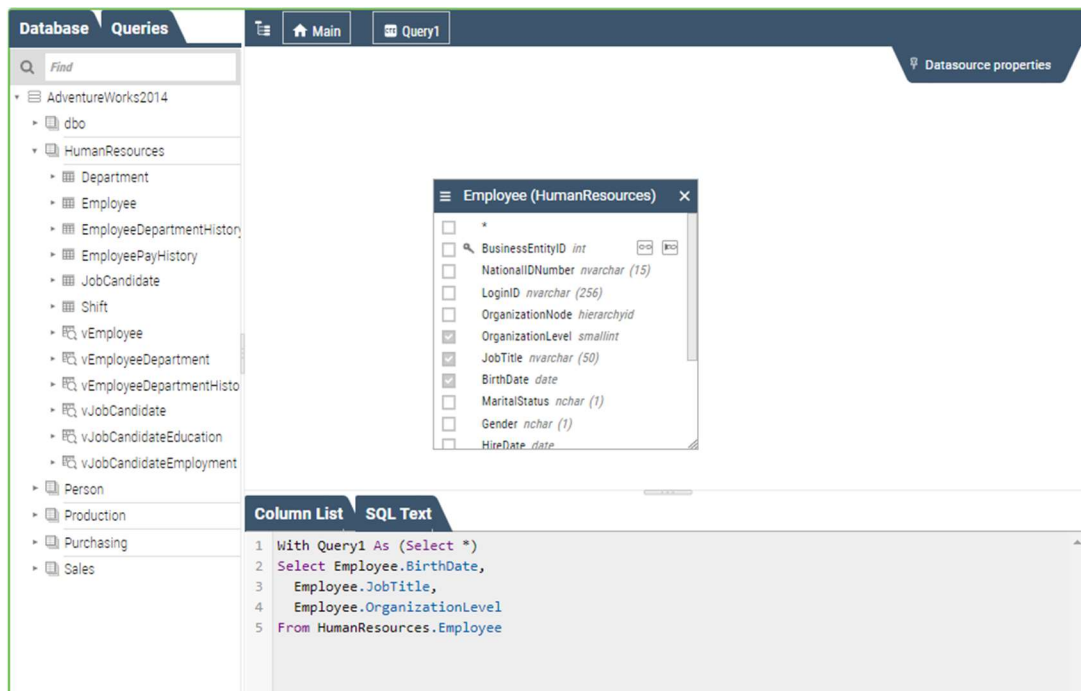


Figura 2.2. Interfaz de ActiveQueryBuilder

2.2.3 Microsoft Office Access (2016)

Microsoft ofrece un producto de software (descrito en [Microsoft, 2016]) con diversas funciones relacionadas al uso de bases de datos, desde la creación, gestión e incluso la formulación de consultas para acceder a los datos almacenados.

Access cuenta con un asistente para composición de consultas del tipo Query By Example (QBE), el cual permite al usuario formular consultas gráficamente de manera sencilla. Para componer consultas sencillas, el usuario debe seleccionar una o más tablas de la base de datos y posteriormente seleccionar las columnas que contienen la información que le interesa. Para visualizar la expresión en SQL generada, debe seleccionar la opción “vista SQL”.

Sin embargo, a pesar de la facilidad con que se generan consultas sencillas, es necesario que el usuario domine SQL si pretende formular consultas con un grado mayor de complejidad, por ejemplo, una consulta que involucre subconsultas.

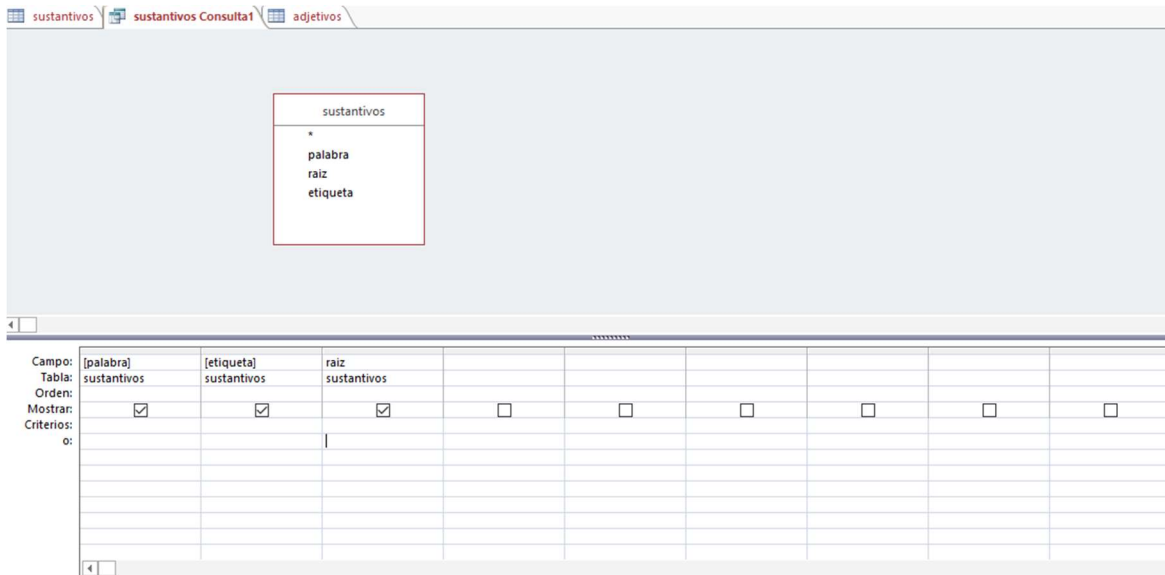


Figura 2.3. Interfaz de Microsoft Office Access 2016

2.2.4 Conceptual Authoring (2007)

El método de Conceptual Authoring (descrito en [Hallett, 2007]) está enfocado a sistemas de pregunta-respuesta en los que la transparencia y la fiabilidad son críticos, y donde los usuarios no pueden someterse a costosos entrenamientos en la composición de preguntas.

Hay dos ideas principales en el Conceptual Authoring. La primera se trata de que todas las operaciones de edición sean definidas directamente en una representación lógica subyacente. Entonces, en lugar de teclear texto, el usuario construye la representación lógica directamente, de modo que no surjan problemas de interpretación.

La segunda idea se refiere a que la interfaz de usuario presenta la representación lógica en desarrollo y las opciones para editarla de una manera transparente para los usuarios. Debido a esto, los usuarios experimentan la percepción de que están realizando una actividad habitual, como algún tipo de escritura guiada, en lugar de una actividad no habitual, como programar.

Entre otras, una de las aplicaciones del Conceptual Authoring (o WYSIWYM) ha sido la composición de consultas a una base de conocimiento, como se hizo en el *framework* clínico CLEF. Dicho *framework* provee un repositorio de historias clínicas bien organizadas, las cuales pueden ser consultadas o resumidas tanto para la investigación biomédica como para el cuidado clínico.

De alguna manera la interfaz se asemeja a técnicas basadas en menús, pero esa semejanza es únicamente superficial, porque en estas técnicas el usuario edita una estructura lingüística. En cambio, en Conceptual Authoring todas las operaciones de edición están definidas en una estructura lógica.

<p><i>FEEDBACK TEXT</i></p> <p>[Some person] examined [some person] <i>[in some way]</i></p>
<p><i>MENU OF OPTIONS</i></p> <p>.....</p> <p>doctor</p> <p>nurse</p> <p>patient</p> <p>.....</p>

Figura 2.4. Estructura de consulta en CLEF

2.2.5 Herramienta para consultas basadas en ejemplos (2000)

La herramienta para consultas (descrita en [May, 2000]) fue diseñada con el propósito de que usuarios casuales o inexpertos puedan acceder a la información contenida en dos o más bases de datos simultáneamente. Esta interfaz está basada en la filosofía original del QBE, la cual consiste en que el usuario formule su consulta escribiendo un ejemplo de los datos que desea que aparezcan en los resultados.

- El proceso de formulación y ejecución de consultas está compuesto por tres fases:
- Selección de las BDs y tablas participantes en la consulta.
 - Formulación de la consulta.
 - Ejecución de la consulta y presentación de resultados.

Primeramente, el usuario debe seleccionar de uno en uno los diccionarios de datos de las BDs que desee consultar. Posteriormente la interfaz presenta una lista de tablas de las bases de datos activas en ese momento.

Para la formulación de la consulta, la interfaz se divide en dos secciones: en la sección superior aparecen las tablas seleccionadas en el primer paso del proceso, y la sección inferior contiene una cuadrícula para formular la consulta. En las columnas de la cuadrícula, el usuario debe indicar las columnas de la base de datos que participarán en la consulta.

Una vez que el usuario ha concluido la definición de la consulta, procede a ejecutarla. Esta fase del proceso genera una expresión en SQL y la presenta al usuario como se muestra en la Figura 2.5.



Figura 2.5. Ejemplo de consulta en la interfaz

2.2.6 Comparación de interfaces descritas

Tabla 2.1. Comparación de interfaces de composición de consultas

Interfaz	Método de composición	Manejo de subconsultas	Manejo de funciones de agregación	Dominio conceptual de BDs	Dominio de lenguaje de consulta
Query Builder	LN, menús	No	Sí	Regular	Nada
Active Query Builder	Drag & drop	Sí	No	Regular	Poco
Microsoft Office Access	QBE, SQL	Sí	Sí	Mucho	Mucho
Conceptual Authoring	LN, menús	No	No	Nada	Nada
Herramienta QBE	QBE	Sí	Sí	Regular	Nada
Interfaz Propuesta	Árbol de composición	Sí	Sí	Nada	Nada

2.2.7 Conclusiones

Como se observa en la Tabla 2.1, la principal desventaja de Query Builder en comparación con la interfaz propuesta es que no permite formular consultas que involucren subconsultas. Active Query Builder sí puede manejar subconsultas, pero, en contraste, no permite formular consultas que incluyan funciones de agregación. La mayor desventaja de Microsoft Office Access es el hecho de que el usuario debe dominar SQL, si quiere componer consultas con

mayores grados de complejidad. Conceptual Authoring no requiere que el usuario domine un lenguaje de consulta, pero, a cambio de eso, sólo le permite formular consultas básicas y sin gran complejidad. Finalmente, la principal desventaja de la Herramienta QBE es que el usuario debe estar familiarizado con la terminología de bases de datos.

Capítulo 3

Versión anterior de la interfaz

La interfaz de composición de consultas funciona, al igual que su versión anterior (descrita en [Castillo, 2016]) haciendo uso de un diccionario de información semántica. Dicho diccionario almacena la información del esquema de la base de datos y permite a la interfaz obtener la información de tablas, columnas y relaciones de la BD.

La interfaz de composición cuenta además con una interfaz de configuración, la cual debe ser accedida por el administrador. El administrador se encargará de nutrir a la interfaz con información sobre las tablas y columnas de la BD, así como sus tipos de dato. De esta manera, la interfaz estará lista para guiar a los usuarios por el proceso de composición de consultas.

3.1 Arquitectura

El módulo de subconsultas desarrollado tiene como propósito guiar al usuario durante el proceso de composición de consultas que incluyen subconsultas. Además de las fases con las que ya contaba la versión anterior a la desarrollada por Castillo, se añadieron dos nuevas fases para hacer posible la composición de subconsultas. De esta manera se identifican un total de 6 fases que constituyen la interfaz de composición. Estas fases son las siguientes: *Selección del tema de interés*, *Selección de elementos de interés*, *Especificación de condiciones de búsqueda*, y *Vista previa y resultado*. Además, hay que mencionar las nuevas fases para subconsultas, las cuales son *Descomposición de consulta en LN*, y *Vista previa y especificación de condición para subconsultas*.

La arquitectura de la interfaz se presenta en la Figura 3.1 en la cual se muestra el proceso de composición y cómo la interfaz guía al usuario durante él.

La primera fase es *Descomposición de consulta en LN*. Para realizar la descomposición, el usuario debe tener formulada una consulta en lenguaje natural. Además, el usuario debe identificar los diferentes fragmentos en los que debe descomponer la consulta original para escribir cada fragmento en donde le indique la interfaz.

La siguiente fase es la *Selección del tema de interés*. En esta fase la interfaz obtiene la información de las tablas que existen en la BD y las muestra en forma de lista. Una vez que la interfaz presenta la clasificación de temas al usuario, éste debe seleccionar uno de los temas para continuar la composición de la consulta o subconsulta.

Especificación de condiciones de búsqueda es la fase que continúa en el proceso. En esta fase la interfaz obtiene la información de las columnas del tema seleccionado y utiliza dicha información para generar el árbol de composición. Entonces el usuario seleccionará un

elemento a la vez y usará la interfaz para especificar la condición de búsqueda de la consulta o subconsulta.

La siguiente fase es *Vista previa y especificación de condiciones de búsqueda para la subconsulta*. En esta fase la interfaz obtiene el resultado previo dado por la consulta anterior. Y una vez que se muestre dicho resultado, el usuario debe especificar la condición de búsqueda para la subconsulta. También es posible definir más condiciones de búsqueda, si el usuario así lo desea.

Selección de elementos de interés es la siguiente fase. En ella la interfaz genera un árbol de composición mediante el cual el usuario debe seleccionar uno por uno los elementos de los que desea obtener información.

Finalmente, en la fase *Vista previa y resultado* la interfaz genera una consulta o subconsulta en SQL según el nivel de consulta que el usuario haya compuesto. Dicha instrucción en SQL es enviada al SMBD para obtener el resultado que posteriormente se muestra al usuario.

A continuación se muestra la Figura 3.1 que presenta la arquitectura de la versión anterior de la interfaz de composición.

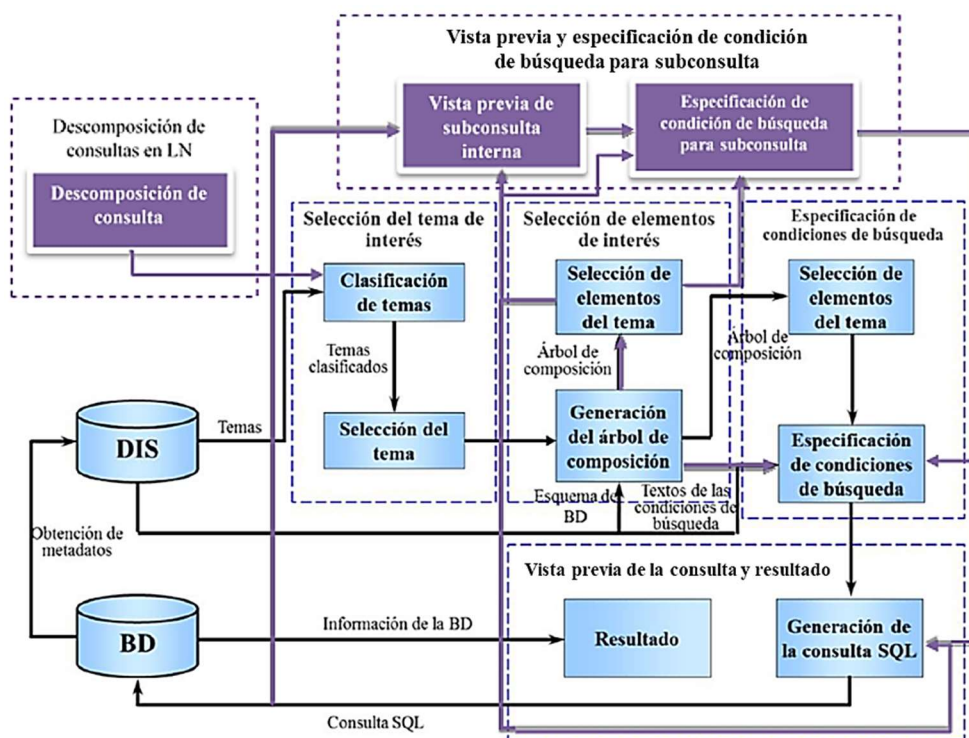


Figura 3.1. Arquitectura de la versión anterior de la interfaz

3.2 Árbol de composición

El árbol de composición usado desde la primera versión de la interfaz fue adaptado para el correcto funcionamiento de esta versión. Debido a esa adaptación, el árbol de composición puede ser usado para definir condiciones de búsqueda para las subconsultas. Los elementos que conforman la estructura del árbol de composición son los siguientes:

- Tabla principal (descripción de la tabla en LN).
 - Columna 1 (descripción de la columna en LN).
 - Columna 2 (descripción de la columna en LN).
 - ...

Este árbol únicamente muestra al usuario el tema principal y sus elementos, como puede observarse en la Figura 3.2. Los temas relacionados con el tema principal fueron eliminados, dado que no son necesarios para la construcción de subconsultas. Además, al eliminarlos se evitan errores que puedan surgir al seleccionar un elemento para especificarle una condición de búsqueda.

A continuación se presenta el árbol de composición generado para la tabla *flight* de la base de datos *ATIS*. Se observa la estructura del árbol de composición con el nodo raíz que representa a la tabla *flight* y que muestra al usuario la descripción en LN de la tabla. Así mismo los nodos hijos representan cada elemento del tema seleccionado.

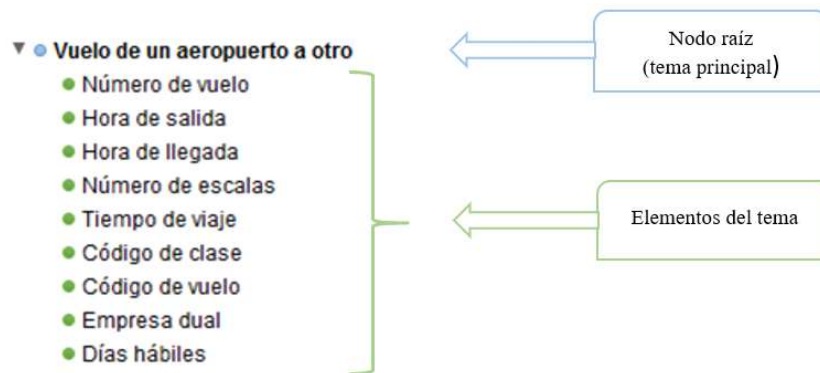


Figura 3.2. Árbol de composición

3.3 Descomposición de consulta en LN

Descomposición de consulta en LN es una fase de gran importancia en el proceso de composición. En esta fase el usuario debe descomponer en varios fragmentos la consulta en LN, cada fragmento resultante de la descomposición constituye alguna de las subconsultas o la consulta externa. A continuación se presentan los pasos claves para efectuar la descomposición:

1. En primer lugar, el usuario debe identificar la condición de búsqueda. Dicha condición se va a considerar como una subconsulta.
2. Si se identifica la existencia de una condición que dependa del paso anterior, ésta será considerada como una subconsulta externa a la consulta anterior. En caso contrario, se continúa con el siguiente paso.
3. Los elementos restantes de la consulta que definen la información solicitada por el usuario constituyen la consulta más externa.

Según los pasos mencionados anteriormente, la descomposición de una consulta que incluye subconsultas se realizaría como se muestra en los ejemplos de las Figuras 3.3 y 3.4.

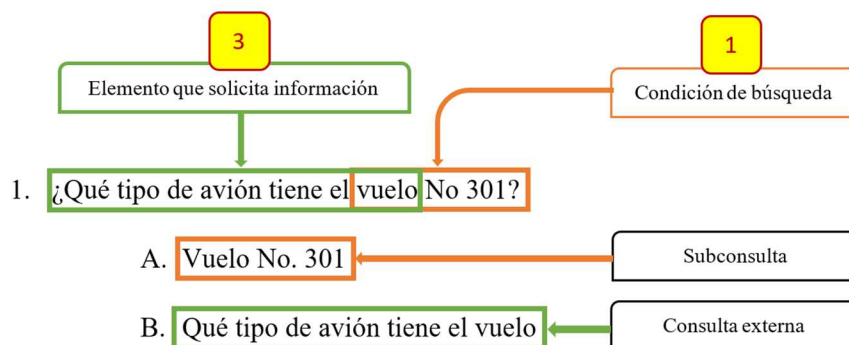


Figura 3.3. Ejemplo de descomposición en una consulta externa y una subconsulta

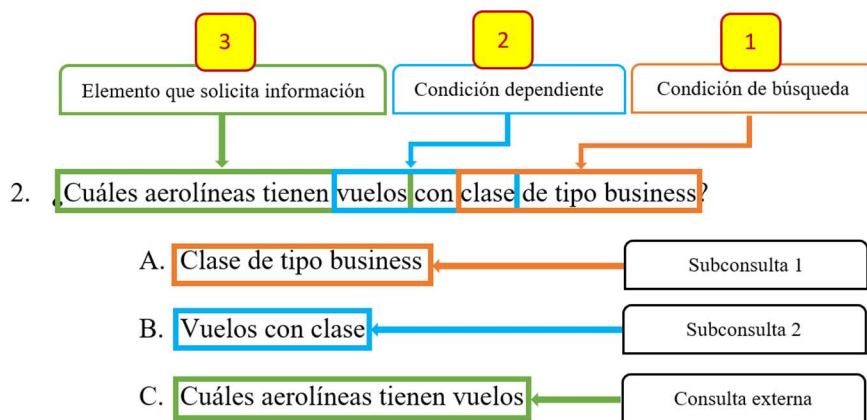


Figura 3.4. Ejemplo de descomposición en una consulta externa y dos subconsultas

3.4 Generación automática de subconsultas intermedias

La parte más importante del módulo de subconsultas es la generación automática de subconsultas. Este proceso ha permitido componer consultas que cuentan con hasta 4 subconsultas en SQL incluyendo consultas donde el usuario únicamente realiza la composición de una subconsulta y la consulta externa. La interfaz es lo suficientemente inteligente como para generar automáticamente las subconsultas intermedias que conecten la subconsulta más interna con la consulta externa. De esta manera es posible mostrar al usuario una traducción correcta a SQL de la consulta compuesta.

Teniendo la consulta externa y una subconsulta compuesta, el Algoritmo 3.1 construye una matriz de adyacencia. La matriz de adyacencia representa las tablas existentes en la BD, así como las relaciones entre ellas. Lo siguiente es obtener las tablas intermedias obtenidas por el algoritmo BFS (búsqueda en anchura). En caso de obtener dos o más tablas, se ejecutará el método *CrearConsultasInternas*, el cual usa las llaves primarias y foráneas correspondientes a las tablas para construir las subconsultas necesarias.

La interfaz hace uso del algoritmo BFS para generar un camino desde la tabla correspondiente a la consulta más externa hasta la tabla que corresponde a la subconsulta más interna. Después, con ayuda del DIS, la interfaz identifica las relaciones que unen entre sí a las tablas mencionadas anteriormente con la subconsulta más interna y la consulta externa. De esta manera se generan automáticamente las subconsultas que hacen falta.

A continuación se presenta el algoritmo 3.1 usado por la interfaz de composición.

Algoritmo 3.1. Seudocódigo para la generación automática de subconsultas

1. *TablaConsExterna*
 2. *EsSubconsulta*
 3. *TablaSubconsulta*
 4. **If** *EsSubconsulta*
 5. *ObTablas* = ObtenerTablas()
 6. *ObRelaciones* = ObtenerRelaciones()
 7. *MaAdy*[*i*][*j*] = 0 para $i = 1, \dots, |\text{ObTablas}|; j = 1, \dots, |\text{ObTablas}|$
 8. **For** $i = 0$ **to** *ObRelaciones*
 9. *RelacionesTablas* = *ObRelaciones*.get(*i*)
 10. *MaAdy*[*RelacionTablas*[0]][*RelacionTablas*[1]] = 1
 11. *MaAdy*[*RelacionTablas*[1]][*RelacionTablas*[0]] = 1
 12. **EndFor**
 13. GenerarAlgBFS(*MaAdy*, *TablaConsExterna*)
 14. *RutaSubconsultas* = GenerarCamino(*TablaSubconsulta*)
 15. **If** $RutaSubconsultas \geq 3$
 16. *CrearConsultasInternas*(*RutaSubconsultas*, *TablaConsExterna*, *TablaSubconsulta*)
 17. **EndIf**
 18. **EndIf**
-

Capítulo 4

Metodología de solución

La interfaz de composición que se desarrolló durante este proyecto tiene como propósito permitir a usuarios casuales componer consultas que involucren subconsultas y funciones de agregación. Esta nueva versión de la interfaz debe mantener la compatibilidad con las versiones anteriores y la facilidad de uso por parte de los usuarios.

La funcionalidad añadida a la interfaz de composición toma módulos desarrollados en la interfaz descrita en [Castillo, 2016] y los adapta a las necesidades que se presentan en la composición de consultas que incluyen subconsultas y funciones de agregación.

Para la composición de consultas con subconsultas, se ha mantenido el funcionamiento que tiene la versión anterior, es decir, las subconsultas se procesan desde la más interna hacia la consulta externa. Esto con el fin de facilitar el proceso de composición, y que la interfaz pueda guiar sin problemas al usuario por cada una de las fases del proceso.

Así mismo se mantiene el uso del árbol de composición, el cual es necesario para formular cada fragmento de la consulta original resultante tras la fase de descomposición.

La nueva funcionalidad para consultas con funciones de agregación y subconsultas considera sólo los tipos de consultas que se muestran en la Tabla 4.1 y ofrece al usuario una forma sencilla de componer consultas con estas características. Es necesario mencionar que las funciones de agregación se aplican únicamente en la cláusula Select, por lo que la fase del proceso de composición de consultas en la que se enfoca principalmente este proyecto es la *Selección de elementos de interés*.

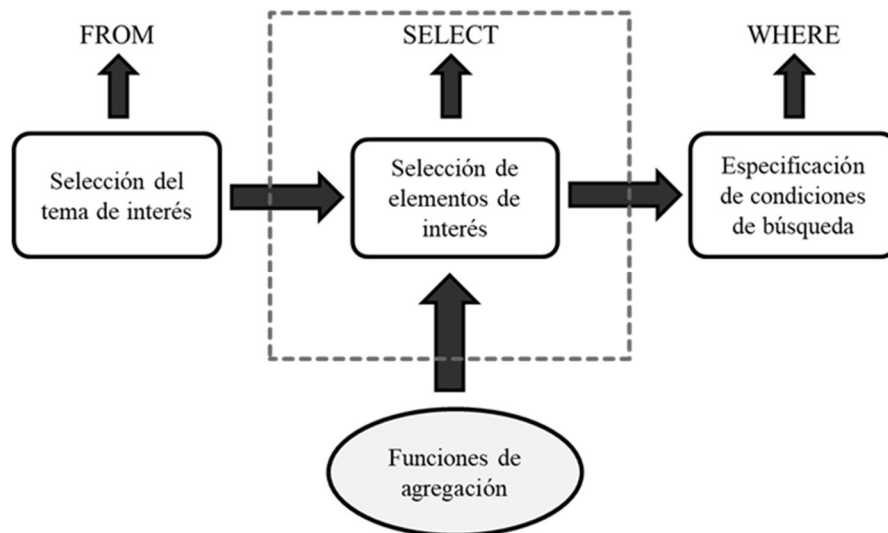


Figura 4.1 Diagrama conceptual del proyecto

4.1 Tipos de consultas a tratar

La nueva funcionalidad permite a la interfaz tratar con consultas que incluyen funciones de agregación combinadas con subconsultas. Según el análisis que se realizó a algunos de los corpus de consultas utilizados en trabajos anteriores, se identificaron 3 tipos de consultas que involucran funciones de agregación. A continuación se ejemplifica cada uno de los tipos de consultas con funciones de agregación.

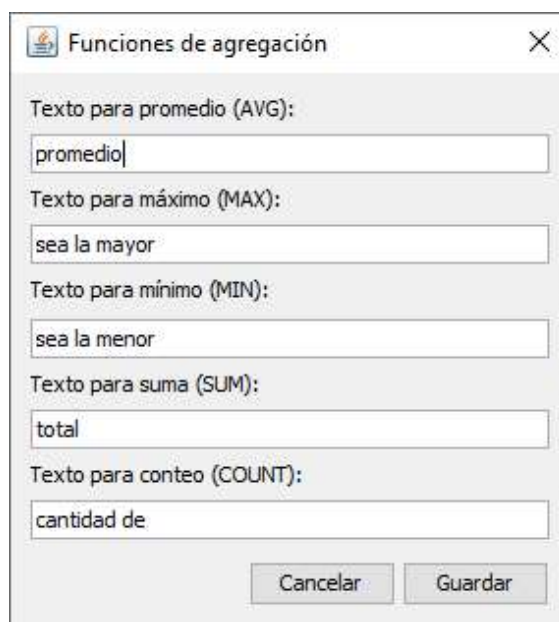
Tabla 4.1. Tipos de consultas a tratar

Tipo de consulta	Consulta en LN	Traducción a SQL	Ubicación de la FA
1	<i>¿Cuál aerolínea tiene el vuelo con la tarifa de vuelo sencillo más barata?</i>	<pre>SELECT airline name FROM airline WHERE airline_code IN (SELECT airline code FROM flight WHERE flight_code IN (SELECT flight code FROM flight_fare WHERE fare_code IN (SELECT fare code FROM fare WHERE one way cost = (SELECT MIN (one way cost) FROM fare))))</pre>	En la subconsulta más interna
2	<i>¿Cuál es el estado con el área más grande?</i>	<pre>SELECT state.state_name, state.area FROM state WHERE state.area = (SELECT MAX (state.area) FROM state);</pre>	En la subconsulta más interna (pero involucra sólo una tabla)
3	<i>¿Cuál es la longitud del río más grande del estado de Oregon?</i>	<pre>SELECT MAX(river.length) FROM river WHERE river.river_id IN (SELECT riverstate.river_id FROM riverstate WHERE riverstate.state_abbreviation IN (SELECT state.abbreviation FROM state WHERE state.state_name LIKE 'Oregon'))</pre>	En la consulta externa

4.2 Interfaz de configuración

La ICC tiene una interfaz de configuración mediante la cual se pueden visualizar y modificar los datos del diccionario de información semántica (DIS). Dicha interfaz permite además de la edición de datos relacionados con las tablas y columnas de la BD, la creación y eliminación de tipos de datos que son necesarios para que la ICC pueda cumplir su función.

Para esta nueva versión de la ICC es necesario que la interfaz de configuración permita visualizar y modificar las descripciones de las funciones de agregación. Dichas descripciones van a ser mostradas al usuario durante el proceso de composición. Para realizar la modificación se añadió una nueva ventana con campos de texto, como ilustra la Figura 4.2. En esta ventana el administrador visualiza la descripción correspondiente a cada función de agregación y puede editarla si así lo desea.



Funciones de agregación

Texto para promedio (AVG):
promedio

Texto para máximo (MAX):
sea la mayor

Texto para mínimo (MIN):
sea la menor

Texto para suma (SUM):
total

Texto para conteo (COUNT):
cantidad de

Cancelar Guardar

Figura 4.2. Edición de descripciones de FAs

4.3 Modificación del DIS

El DIS es un componente muy importante para la ICC. En él se almacena la información del esquema de la base de datos tal como tablas, columnas y relaciones. También almacena las descripciones de cada tabla, de cada columna y la definición de los tipos de dato.

Para efectuar la composición de consultas que involucran FAs, se modificó el DIS agregando una tabla en la que se almacena la descripción de cada función de agregación aplicada a cada tipo de dato previamente definido por el administrador. La nueva tabla tiene una estructura similar a la tabla que almacena las descripciones de los operadores lógicos usados para especificar condiciones de búsqueda.

Cada registro de la nueva tabla debe contener la información que se describe a continuación:

- **Tipo de dato:** su valor indica el tipo de dato de la columna de la BD. Dentro del tipo se agrupan algunas columnas cuyos valores posean características similares y puedan aplicarse las mismas funciones de agregación.
- **Promedio:** indica la descripción de la función de agregación AVG para un tipo de dato específico.

- **Suma:** indica la descripción de la función de agregación SUM para un tipo de dato específico.
- **Máximo:** indica la descripción de la función de agregación MAX para un tipo de dato específico.
- **Mínimo:** indica la descripción de la función de agregación MIN para un tipo de dato específico.
- **Conteo:** indica la descripción de la función de agregación COUNT para un tipo de dato específico.

4.4 Árbol de composición

El árbol de composición descrito en [Aguirre, 2015], y cuya adaptación se describe en [Castillo, 2016], mantiene su importancia en el proceso de composición de consultas. En esta nueva versión, toma una relevancia especial en la construcción de consultas con funciones de agregación.

La estructura del árbol de composición se mantiene de la manera en que lo propuso Castillo, debido a que las funciones de agregación no modifican la manera en la que se presenta el árbol, pero sí modifican su funcionamiento.

La Figura 4.3 muestra el árbol de composición generado para el tema *río* de la base de datos Geobase. La raíz del árbol representa el tema principal, en este caso *río*, que es la descripción en LN de la tabla *river*. Cada uno de los nodos que se despliegan del nodo raíz son los elementos del tema seleccionado. Dichos elementos representan las columnas de la tabla *river* y muestran sus descripciones en lenguaje natural.

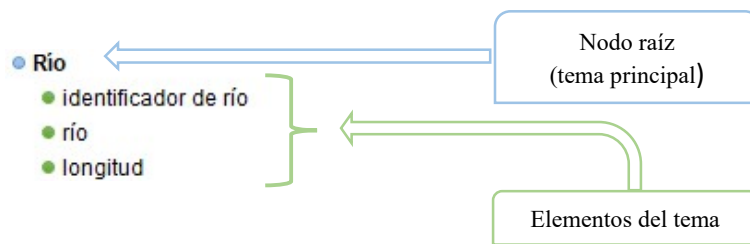


Figura 4.3. Árbol de composición

Si bien la estructura es la misma, el funcionamiento es diferente, debido a que en versiones anteriores el nodo raíz no podía ser seleccionado por el usuario. Pero, en casos específicos que requieran una función de agregación, se ha optado por permitir al usuario seleccionar el nodo raíz del árbol de composición. Es el caso particular de la función de agregación COUNT, la cual puede aplicarse sin especificar una columna, pero para fines de simplicidad se decidió no aplicarla a una columna, sino a la tabla. Esto con el propósito de evitar errores de parte del usuario al momento de aplicar la función de agregación.

El árbol de composición se adapta, según se requiera o no aplicar funciones de agregación, para no causar errores al permitir seleccionar el nodo raíz cuando no se trate con funciones de agregación.

4.5 Selección de elementos de interés

Como se mencionó anteriormente, las funciones de agregación se aplican en la cláusula Select, así que *Selección de elementos de interés* es la fase en la que se enfoca mayormente este trabajo.

El algoritmo 4.1 muestra el comportamiento de la interfaz de composición cuando se avanza a la fase *Selección de elementos de interés*. Primero comprueba si el fragmento de la consulta que se está componiendo incluye una función de agregación. De ser así, desplegará la ventana correspondiente a la fase del proceso de composición y, seguido de eso, ejecutará el método *rediseñarSeleccionElem*. Dicho método se encarga de modificar los elementos gráficos de la ventana a fin de ofrecer al usuario los elementos necesarios para aplicar la función de agregación requerida. En caso contrario, únicamente desplegará la ventana sin realizarle modificación alguna.

Algoritmo 4.1. Seudocódigo de la selección de elementos de interés

1. *fragmentoTieneFA*
 2. **If** *fragmentoTieneFA*
 3. *seleccionElementos()*
 4. *rediseñarFA()*
 5. **EndIf**
 6. **Else**
 7. *seleccionElementos()*
 8. **EndElse**
-

En cuanto a la identificación del tipo de consulta (vea tabla 4.1), el algoritmo 4.2 permite a la interfaz identificar en qué nivel de consulta o subconsulta se presenta la necesidad de aplicar una FA. El bloque de la línea 3 a la 11 corresponde a *Selección de elementos de interés* para una subconsulta, o *Especificación de condiciones de búsqueda* en caso de no involucrar funciones de agregación. El bloque de la línea 12 a la 19 corresponde a *Selección de elementos de interés* para la consulta externa, ya sea que involucre o no una función de agregación.

Algoritmo 4.2. Seudocódigo para la identificación de la ubicación de la FA

1. *esSubconsulta*
2. *fragmentoTieneFA*
3. **If** *esSubconsulta*
4. **If** *fragmentoTieneFA*
5. *seleccionElementos()*
6. *rediseñarFA()*
7. **EndIf**
8. **Else**
9. *especificacionCondBusq()*
10. **EndElse**
11. **EndIf**
12. **Else**
13. **If** *fragmentoTieneFA*
14. *seleccionElementos()*
15. *rediseñarFA()*
16. **EndIf**
17. **Else**
18. *seleccionElementos()*
19. **EndElse**
20. **EndElse**

4.6 Vista previa y resultado

En la fase *Vista previa y resultado* se realiza la construcción de consultas en SQL, ya sea que contenga o no una función de agregación. En esta fase también se ejecuta la consulta a la base de datos por medio del SMBD para mostrar al usuario tanto el resultado de la consulta compuesta como el código en SQL.

El Algoritmo 4.3 presenta el pseudocódigo para la construcción de consultas que involucran funciones de agregación y subconsultas. En la línea 5 se comprueba si la consulta que se compone es una subconsulta; de ser así, se llama el método *construirConsultaFA*. En caso contrario, se fragmenta la consulta generada hasta el momento (línea 9) y una vez más se llama el método *construirConsultaFA*. Una vez construida la consulta con función de agregación, se concatenan el resto de los fragmentos de la consulta a la consulta generada (línea 11 a 13) para obtener la consulta final.

Algoritmo 4.3. Seudocódigo para la generación de consultas con FA

1. *datosFA* = *obtenerDatosFA()*
2. *nombreTab* = *datosFA*[0]
3. *nombreCol* = *datosFA*[2]
4. *nombreFA* = *datosFA*[4]
5. **If** *esSubconsulta*

6. *consultaFA*=construirConsultaFA(*nombreTab*, *nombreCol*, *nombreFA*)
7. **EndIf**
8. **Else**
9. *fragmentos* = *consulta*.split("IN")
10. *consultaFA* = construirConsultaFA(*fragmentos*[0],*nombreTab*,*nombreCol*,*nombreFA*)
11. **For** *i* = 1 **to** *fragmentos*
12. *consultaFA* += *fragmentos*[*i*]
13. **EndFor**
14. **EndElse**

Capítulo 5

Interfaz de composición

La funcionalidad agregada permite a usuarios inexpertos construir consultas con un mayor nivel de complejidad y obtener la información solicitada sin la necesidad de conocer algún lenguaje de consulta. Tampoco es necesario tener idea de cómo está diseñada la base de datos ni las relaciones entre sus tablas. La nueva funcionalidad proporciona a los usuarios la posibilidad de consultar bases de datos complejas mientras son guiados por la interfaz durante el proceso de composición.

5.1 Conceptos para composición de consultas que involucran funciones de agregación combinadas con subconsultas

Si bien el usuario podría no conocer un lenguaje de consulta ni la estructura de la BD, lo recomendable es que se familiarice con algunos conceptos básicos que se manejan a lo largo del proceso de composición de consultas. La Tabla 5.1 muestra la descripción de algunos conceptos básicos presentados en las ventanas con las que debe interactuar el usuario.

Tabla 5.1. Conceptos básicos para la composición de consultas

Concepto	Descripción
<i>Descomposición de consulta</i>	Proceso que permite al usuario dividir la consulta en LN en diferentes fragmentos. Cada fragmento resultante de la descomposición constituye una consulta o subconsulta.
<i>Tema de interés</i>	Es el tema del cual el usuario desea obtener información, ya sea en una consulta o subconsulta.
<i>Condiciones de búsqueda</i>	Son los valores que restringen los posibles resultados de las consultas formuladas por el usuario.
<i>Resultado de subconsulta interna</i>	Resultado de una subconsulta generado automáticamente por la interfaz.
<i>Condición de búsqueda para resultado de subconsulta</i>	Define una condición de búsqueda que se aplicará al resultado de una subconsulta con dos posibles opciones: “pertenece al conjunto” o “no pertenece al conjunto”.
<i>Elementos de interés</i>	Elementos que proporcionan información del tema principal.
<i>Función de agregación</i>	Operación aplicable a un conjunto de valores devuelto como resultado de una consulta o subconsulta.

Teniendo en cuenta los conceptos descritos anteriormente, el usuario podrá hacer un mejor uso de la interfaz aprovechando sus características y minimizando los posibles errores durante el proceso de composición.

5.2 Descomposición de la consulta en LN

La ventana correspondiente a la descomposición de consultas se muestra en la Figura 5.1 en la cual el usuario iniciará el proceso de composición. Para realizar la descomposición de la consulta principal, el usuario debe seguir las indicaciones de la interfaz y dar la información necesaria en el lugar correspondiente. La tabla 5.2 muestra la descripción de los componentes gráficos de la ventana.

Tabla 5.2. Componentes de la ventana de *Descomposición de consulta en LN*

Componente	Descripción
1. <i>Consulta original</i>	En esta caja de texto se escribe la consulta original, la cual se encuentra en lenguaje natural.
2. <i>Consulta principal</i>	En esta caja de texto se escribe la consulta más externa resultante de la descomposición de la consulta original.
3. <i>Definición de subconsultas</i>	De esta lista se selecciona el número de subconsultas que tiene la consulta original según la descomposición realizada por el usuario.
4. <i>Subconsultas</i>	En cada caja de texto, el usuario debe escribir cada fragmento resultante de la descomposición. Debe omitirse la consulta principal.
5. <i>Casillas de verificación de FA</i>	Si el usuario identifica que algún fragmento de la consulta requiere aplicar funciones de agregación, debe seleccionar la casilla correspondiente al fragmento identificado.
6. <i>Botón “Continuar”</i>	Una vez escritos los fragmentos de consulta en su caja de texto correspondiente, se presiona el botón <i>Continuar</i> para avanzar a la siguiente ventana de la interfaz.
7. <i>Barra de menú</i>	Esta barra contiene las opciones <i>Archivo</i> , <i>Herramientas</i> y <i>Ayuda</i> . <i>Archivo</i> permite salir de la interfaz, <i>Herramientas</i> da la posibilidad de acceder a la interfaz de configuración, y finalmente <i>Ayuda</i> muestra el manual de usuario.



Figura 5.1. Ventana de *Descomposición de consulta en LN*

Para garantizar el correcto funcionamiento de la interfaz, es necesario que el usuario identifique correctamente cada fragmento de la consulta. De ser así, la interfaz lo guiará paso a paso en el proceso de composición

5.3 Selección del tema de interés

En esta ventana el usuario debe seleccionar de la lista el tema del que desea obtener información. El tema representa a la tabla de la BD que se va a consultar. La Tabla 5.3 muestra la descripción de los componentes gráficos de la ventana.

Tabla 5.3. Componentes de la ventana de *Selección del tema de interés*

Componente	Descripción
1. <i>Consulta a componer</i>	Muestra el fragmento de consulta que se está componiendo.
2. <i>Lista de temas</i>	Muestra un listado de los temas correspondientes a las tablas que existen en la base de datos. El usuario puede elegir sólo un tema para proseguir la composición de la consulta.

3. <i>Navegación para ver más temas</i>	Botones usados para mostrar los diferentes temas de la base de datos, los cuales se encuentran clasificados.
4. <i>Botón “Continuar”</i>	Habiendo seleccionado un tema de la lista, el usuario debe presionar este botón para avanzar a la siguiente ventana.

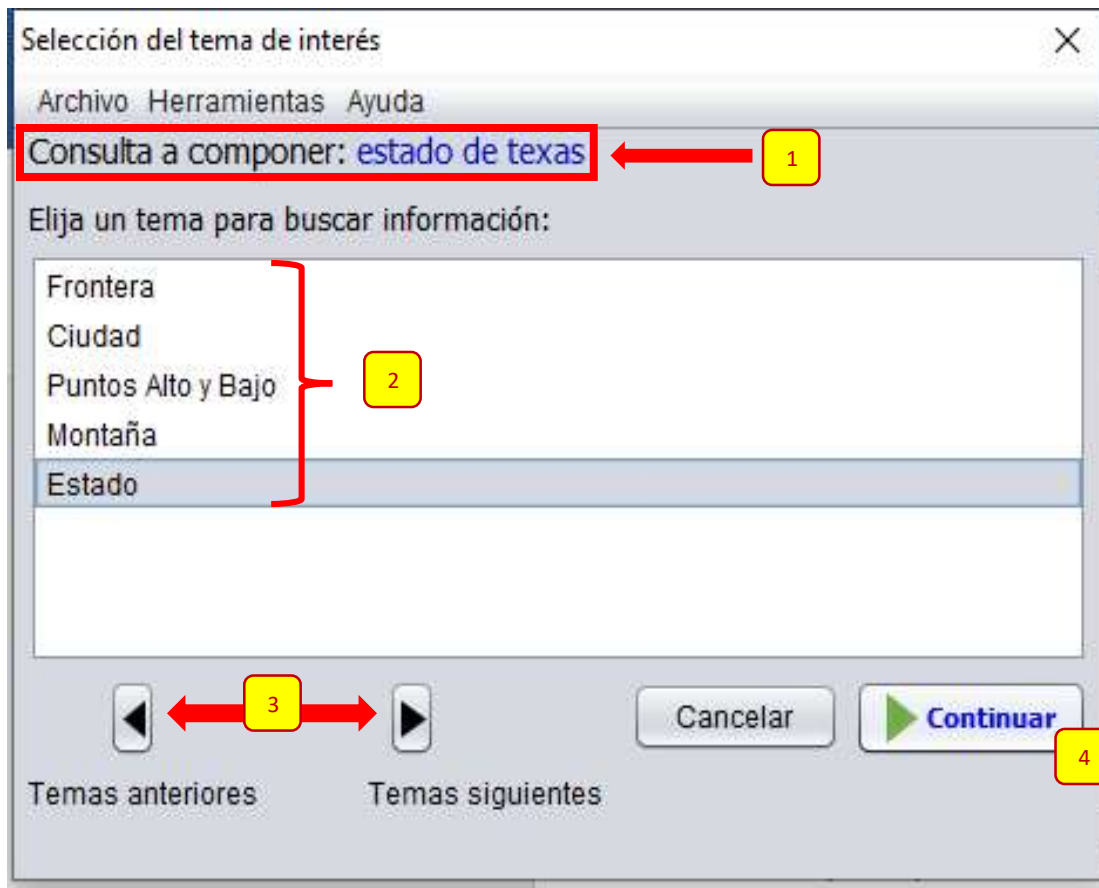


Figura 5.2. Ventana de *Selección del tema de interés*

Los temas que se muestran en la ventana son las descripciones de las tablas que conforman la base de datos. Estas tablas están ordenadas de mayor a menor relevancia para la composición de consultas que involucran subconsultas y se encuentran clasificadas por tablas base, catálogos, relaciones M a N y tablas satélite como se propuso en [Aguirre, 2015].

5.4 Especificación de condiciones de búsqueda

La ventana *Especificación de condiciones de búsqueda* permite al usuario restringir los resultados de la consulta que esté componiendo. Esta ventana es realmente importante para la composición de consultas con subconsultas, en especial para la subconsulta más interna, ya que en ésta siempre debe definirse una condición de búsqueda. Si no incluye una función de agregación, la especificación de condiciones de búsqueda no es necesaria.

La Tabla 5.4 presenta la descripción de los componentes que conforman la ventana. Dicha ventana se muestra en la Figura 5.3.

Tabla 5.4. Componentes de la ventana de *Especificación de condiciones de búsqueda*

Componente	Descripción
1. <i>Árbol de elementos del tema</i>	Muestra los elementos pertenecientes al tema seleccionado anteriormente por el usuario. Las restricciones deben definirse de una por una.
2. <i>Lista de condiciones del elemento seleccionado</i>	Los elementos desplegados en el árbol de composición (1) pueden tener un máximo de 6 condiciones para restringir la búsqueda: igual a (=), menor que (<), mayor que (>), menor o igual a (<=), mayor o igual a (>=) y diferente de (<>). Dichas condiciones ayudan al usuario a restringir los valores devueltos como resultado de la consulta.
3. <i>Valor de la condición</i>	Esta caja de texto junto con la lista de condiciones (2) permiten al usuario especificar una condición para restringir los valores devueltos como resultado de la consulta.
4. <i>Ayuda</i>	Este botón permite al usuario ver un ejemplo del valor de búsqueda que puede escribir según el tipo de dato del elemento seleccionado.
5. <i>Añadir condición</i>	Habiendo especificado la condición, el usuario debe presionar este botón para agregar la condición a la lista de condiciones definidas.
6. <i>Lista de condiciones de búsqueda</i>	Esta lista muestra las condiciones de búsqueda que el usuario ha definido.
7. <i>Botón “Eliminar”</i>	Para hacer uso de este botón, el usuario debe seleccionar de la lista (6) la condición que desee eliminar. Un clic sobre este botón permite eliminar una condición definida con anterioridad.
8. <i>Botón “Atrás”</i>	Este botón permite al usuario regresar a la ventana anterior en caso de querer corregir alguna de las acciones realizadas en el proceso de composición.

9. Botón “Continuar”

Una vez que las condiciones de búsqueda han sido definidas por el usuario, se debe presionar este botón para continuar el proceso de composición.

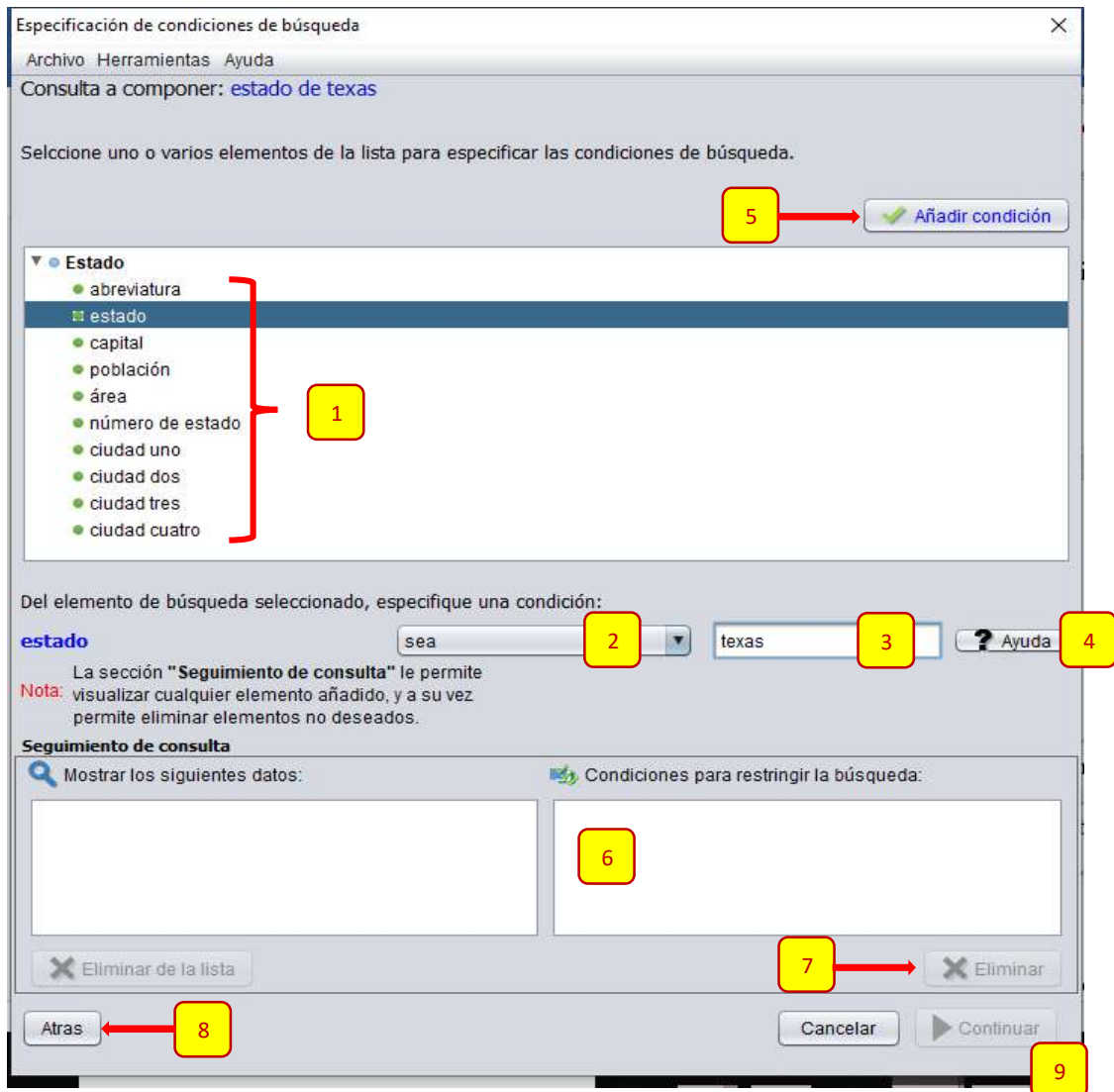


Figura 5.3. Ventana de *Especificación de condiciones de búsqueda*

Las posibles opciones para definir la condición de búsqueda se extraen del DIS. Como se mencionó anteriormente, la interfaz muestra diferentes opciones de condiciones dependiendo del tipo de dato del elemento del árbol seleccionado por el usuario. Un ejemplo de esto se muestra en la Figura 5.4.

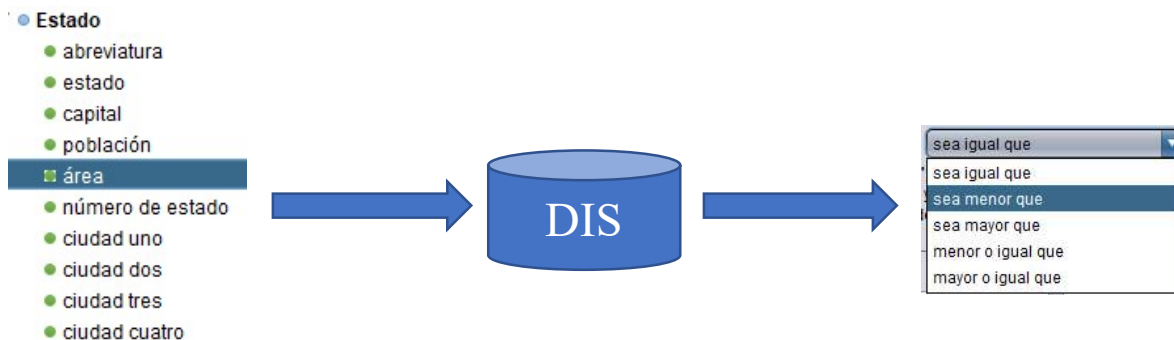


Figura 5.4. Operadores de búsqueda

5.5 Resultado de subconsulta interna

En esta ventana se muestra al usuario el resultado de las subconsultas generadas automáticamente por la interfaz. Como se mencionó, la interfaz genera internamente las subconsultas necesarias para conectar la consulta externa con la subconsulta más interna. De este modo, el usuario encuentra coherencia entre los resultados que le muestra la interfaz y la consulta que ha estado componiendo.

La Tabla 5.5 muestra la descripción de los componentes que conforman esta ventana. Dicha ventana puede observarse en la Figura 5.5.

Tabla 5.5. Componentes de la ventana de *Resultado de subconsulta interna*

Componente	Descripción
1. <i>Resultado previo</i>	Esta tabla muestra el resultado correspondiente a las subconsultas generadas internamente por la interfaz. La tabla contiene los resultados que cumplen la condición de búsqueda especificada por el usuario.
2. <i>Botón “Atrás”</i>	Este botón permite al usuario regresar a la ventana anterior en caso de querer corregir alguna de las acciones realizadas en el proceso de composición.
3. <i>Botón “Continuar”</i>	Una vez que el usuario ha visto el resultado devuelto por la subconsulta interna, debe presionar el botón para avanzar a la siguiente ventana.

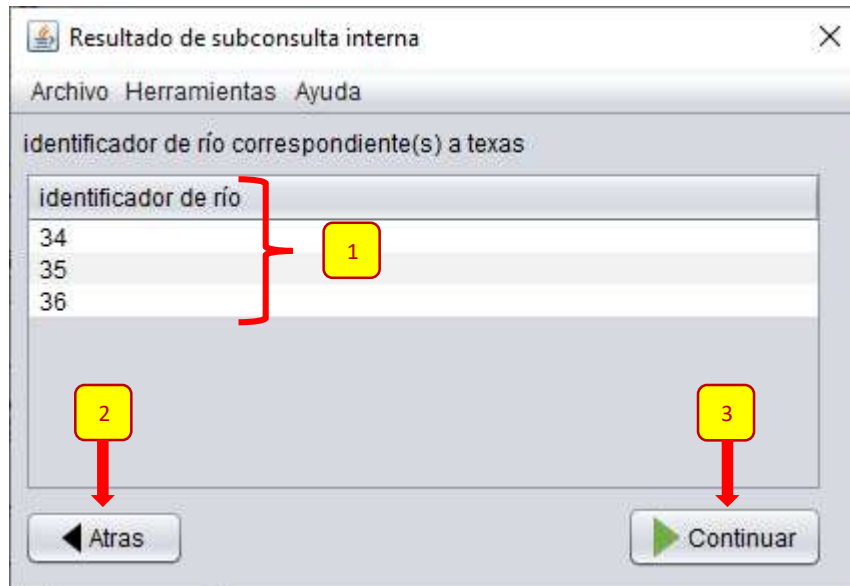


Figura 5.5. Ventana de *Resultado de subconsulta interna*

En la ventana mostrada en la Figura 5.5, el usuario sólo puede ver los resultados devueltos por la interfaz. Por lo tanto, sólo deberá presionar el botón *Continuar* para avanzar a la siguiente ventana.

5.6 Especificación de condición de búsqueda para el resultado de subconsultas

Esta ventana está diseñada para especificar la condición de búsqueda para las subconsultas. Para esta condición existen sólo dos posibles opciones: la primera de ellas hace referencia al operador de SQL IN, mientras que la segunda se refiere al operador NOT IN. La Tabla 5.6 muestra la descripción de los componentes gráficos que se incluyen en la ventana.

Tabla 5.6. Componentes de la ventana de *Especificación de condición para subconsultas*

Componente	Descripción
1. <i>Tabla de resultado</i>	Esta tabla muestra el resultado devuelto por la subconsulta.
2. <i>Lista de condiciones</i>	El usuario debe elegir, de entre las opciones, aquélla con la que quiera restringir la consulta. Las dos posibles opciones son <ul style="list-style-type: none"> • sea igual a alguno(s) de, • sea diferente de todos.
3. <i>Botón “Atrás”</i>	Este botón permite al usuario regresar a la ventana anterior en caso de querer corregir alguna de las acciones realizadas en el proceso de composición.
4. <i>Botón “Continuar”</i>	Una vez que el usuario haya definido una condición para la subconsulta, deberá presionar este botón para avanzar a la siguiente ventana.

La ventana mostrada en la Figura 5.6 tiene como propósito permitir que el usuario defina la condición de búsqueda para la subconsulta. Para hacer esto, el usuario debe seleccionar de la lista una de las dos condiciones, y la condición seleccionada se aplicará al conjunto de valores devuelto por la subconsulta interna.

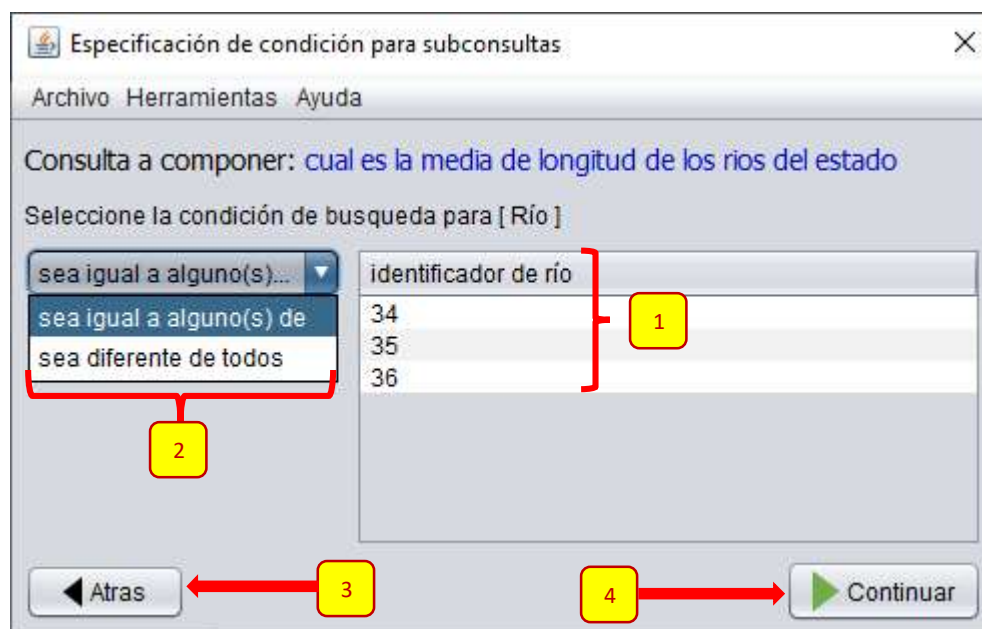


Figura 5.6. Ventana de *Especificación de condición para subconsultas*

5.7 Selección de elementos de interés

En esta ventana el usuario podrá seleccionar uno o más elementos de los que quiera obtener información. Cabe mencionar que, con la nueva funcionalidad para componer consultas con funciones de agregación, esta ventana experimentó algunas modificaciones con respecto a las versiones previas de la interfaz. La Tabla 5.7 muestra la descripción de los componentes que se incluyen en esta ventana.

Tabla 5.7. Componentes de la ventana de *Selección de elementos de interés*

Componente	Descripción
1. <i>Árbol de elementos del tema</i>	Muestra los elementos pertenecientes al tema seleccionado anteriormente por el usuario. Los elementos deben ser seleccionados uno a la vez.
2. <i>Botón “Añadir elemento”</i>	Habiendo elegido un elemento del árbol (1), se presiona este botón para añadirlo a la lista de elementos a mostrar (3).

3. Lista de elementos a mostrar	En esta lista se muestran los elementos que el usuario ha seleccionado indicando que desea obtener información de ellos.
4. Lista de funciones de agregación	El usuario debe seleccionar de esta lista la opción que corresponda a la función de agregación que desee aplicar al elemento seleccionado. Hay 5 posibles opciones que corresponden a las funciones de agregación (AVG, COUNT, MAX, MIN, SUM). Dependiendo del elemento seleccionado, se muestran las posibles opciones a aplicar.
5. Botón “Eliminar elemento”	Presionando este botón, el usuario puede eliminar de la lista (3) los elementos que haya agregado por error.
6. Botón “Atrás”	Este botón permite al usuario regresar a la ventana anterior en caso de querer corregir alguna de las acciones realizadas en el proceso de composición.
7. Botón “Continuar”	Una vez que el usuario haya seleccionado y añadido a la lista los elementos de interés, deberá presionar este botón para avanzar a la siguiente ventana.

Si al momento de desplegarse esta ventana, la interfaz detecta que el usuario indicó la necesidad de aplicar FA, entonces la ventana mostrará los componentes que faciliten al usuario la aplicación de la función de agregación al elemento que seleccione. Esta ventana se muestra en la Figura 5.7.

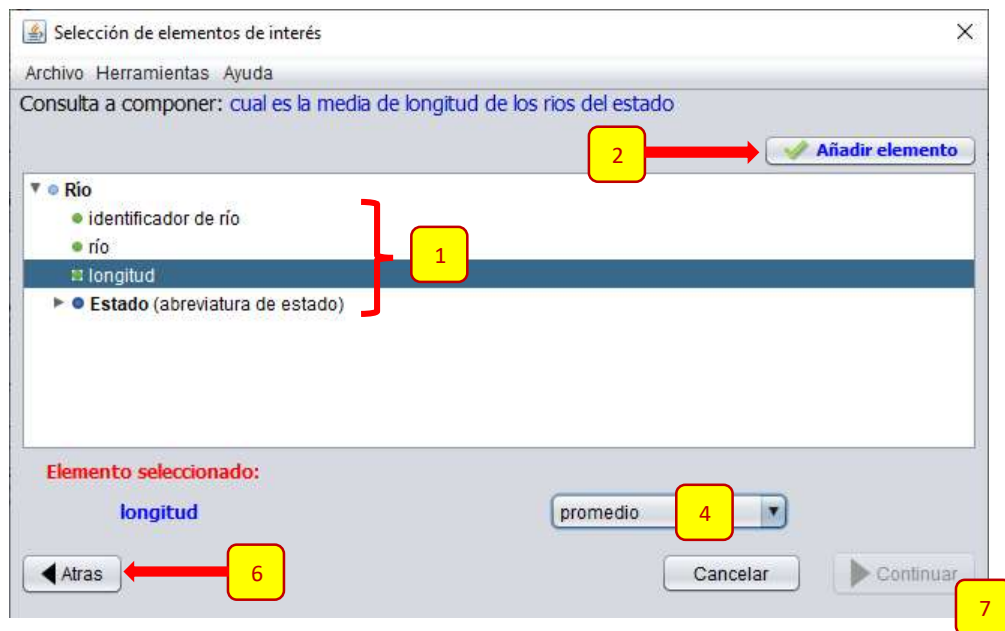


Figura 5.7. Ventana de *Selección de elementos de interés* para consultas con FA

En caso contrario, la interfaz desplegará una ventana como la que se muestra en la Figura 5.8 que, a diferencia de la mostrada en la Figura 5.7, contiene algunos componentes gráficos más.

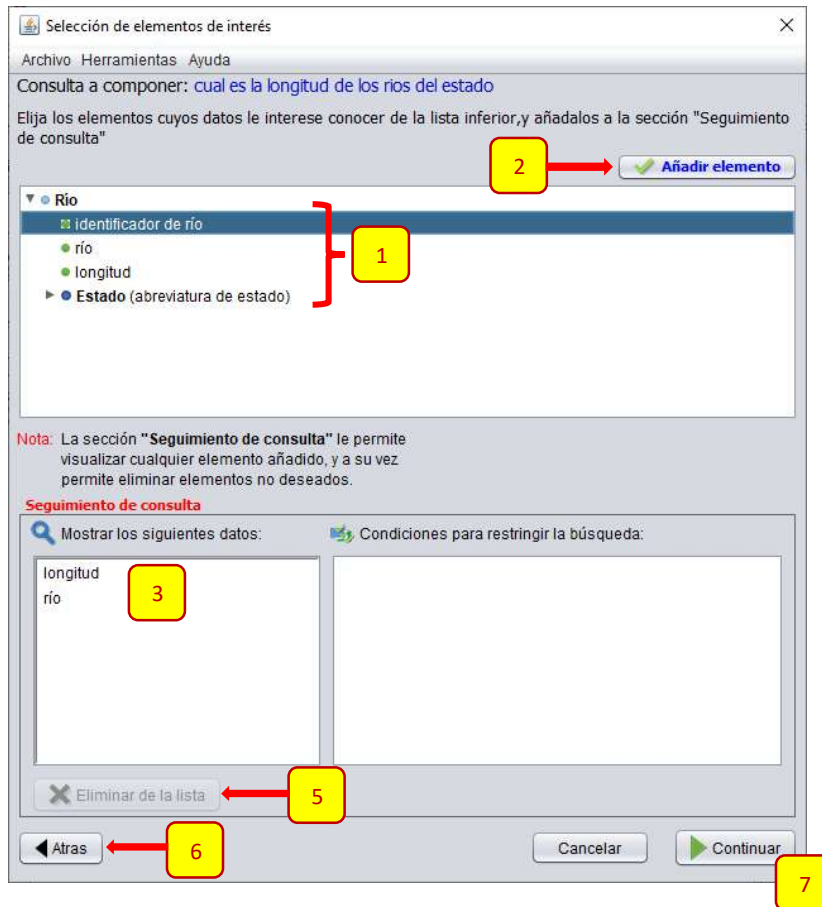


Figura 5.8. Ventana de *Selección de elementos de interés*

La principal diferencia entre ambas ventanas radica en la cantidad de elementos que se pueden seleccionar. Mientras en la primera ventana, es decir, la que permite aplicar funciones de agregación, sólo es posible seleccionar un elemento; la segunda ventana (que no permite aplicar FAs) da la posibilidad de seleccionar más de un elemento. Sin importar cuál de las ventanas vea desplegarse el usuario, ambas le permiten a la interfaz construir la cláusula Select con los elementos que fueron seleccionados previamente por el usuario.

Las diferencias gráficas entre ambas ventanas se deben a la exclusión de algunos componentes que no son necesarios para la aplicación de funciones de agregación. Además, se incluye una lista desplegable que muestra la descripción de las funciones de agregación aplicables al elemento seleccionado. Como se mencionó anteriormente, las FAs que pueden aplicarse a cada elemento de un tema varían dependiendo de su tipo de dato.

El DIS contiene la descripción de cada función de agregación y le permite a la interfaz mostrarlas al usuario dependiendo del elemento que haya seleccionado. De esta manera se garantiza la correcta aplicación de las funciones de agregación que requiera la consulta que se está componiendo. La Figura 5.9 ejemplifica cómo la interfaz accede al DIS para obtener las descripciones que posteriormente mostrará al usuario.

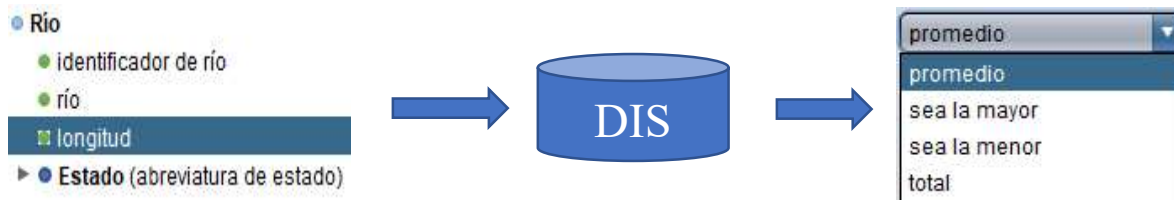


Figura 5.9. Obtención de descripciones de funciones de agregación

Las descripciones mostradas en la Figura 5.9 se encuentran en el diccionario de información semántica y pueden ser modificadas en cualquier momento por el administrador por medio de la interfaz de configuración.

5.8 Construcción en lenguaje SQL de consultas que involucran funciones de agregación combinadas con subconsultas

La construcción de consultas que involucran funciones de agregación es una de las principales funciones de la nueva versión de la interfaz de composición. La forma en la que se construyen las consultas varía dependiendo de los niveles de subconsultas con los que cuenta la consulta. Un factor más que influye en cómo se construye la consulta en SQL es la ubicación de la FA, es decir, en qué nivel de consulta o subconsulta se encuentra.

Primeramente, para identificar las consultas o subconsultas que requieren aplicar funciones de agregación, se optó por generar un arreglo que almacene valores booleanos (banderas). Este arreglo tomará sus valores en la fase *Descomposición de consulta en LN*, según lo mencionado en la Sección 5.2, en la cual el usuario identificará si alguno de los fragmentos de la consulta requiere la aplicación de una FA. De ser así, seleccionará la casilla de verificación que corresponda al fragmento identificado (ver la Figura 5.2).

Una vez terminada la primera fase del proceso de composición, la interfaz genera un arreglo con la estructura que se muestra en la Figura 5.10.

Arreglo funciones de agregación				
Información del arreglo				
Nivel de consulta	Consulta principal	Primer nivel de subconsulta	Segundo nivel de subconsulta	Tercer nivel de subconsulta
Posibles valores	Verdadero/falso	Verdadero/falso	Verdadero/falso	Verdadero/falso

Figura 5.10. Arreglo funciones de agregación

Como puede observarse, el arreglo puede almacenar un máximo de cuatro elementos. Cada elemento corresponde a cada nivel de consulta o subconsulta en que se haya descompuesto la consulta original. A su vez, cada elemento puede tomar dos posibles valores según lo haya indicado el usuario. Para la consulta que se ilustra en la Sección 5.2, el arreglo se vería como se muestra en la Figura 5.11.

Arreglo funciones de agregación		
Nivel de consulta	Consulta principal	Primer nivel subconsulta
Valor	true	false

Figura 5.11. Ejemplo del arreglo funciones de agregación

De esta manera, la interfaz puede saber en cuál nivel de consulta o subconsulta debe mostrar los elementos necesarios para aplicar la FA.

Se ha creado además un arreglo en la fase *Selección de elementos de interés*. En dicha fase el usuario indica a la interfaz el elemento del tema que desea conocer, así como la función de agregación que aplicará al elemento seleccionado. Para este momento del proceso de composición, el usuario ya habrá indicado previamente el tema de interés; por lo tanto, la interfaz ya cuenta con la información necesaria para la construcción de la instrucción en SQL. La Figura 5.12 muestra la estructura del arreglo mencionado anteriormente.

Arreglo selección de elementos con funciones de agregación					
Información del arreglo					
Descripción de tabla	Nombre de tabla	Descripción de columna	Nombre de columna	Descripción FA	FA

Figura 5.12. Arreglo selección de elementos con funciones de agregación

Como se observa en la Figura 5.12, el arreglo está formado por seis elementos, y la descripción y nombre de la tabla son obtenidos en la fase *Selección del tema de interés* como se describe en la Sección 5.3. Por otro lado, la descripción y nombre de la columna, así como la función de agregación y su descripción son obtenidas en la fase *Selección de elementos de interés* (ver la Sección 5.7).

Para la consulta del ejemplo de la Sección 5.7, el arreglo mencionado anteriormente tendría unos valores como los que se muestran en la Figura 5.13.

Arreglo selección de elementos con funciones de agregación					
Descripción de tabla	Nombre de tabla	Descripción de columna	Nombre de columna	Descripción FA	FA
río	river	longitud	length	promedio	AVG

Figura 5.13. Ejemplo del arreglo selección de elementos con FAs

Una vez que la interfaz tiene el arreglo con la información de tabla, columna y función de agregación, puede proceder a la construcción de la instrucción en SQL. Este proceso se asemeja al que se presenta en la Figura 5.14.

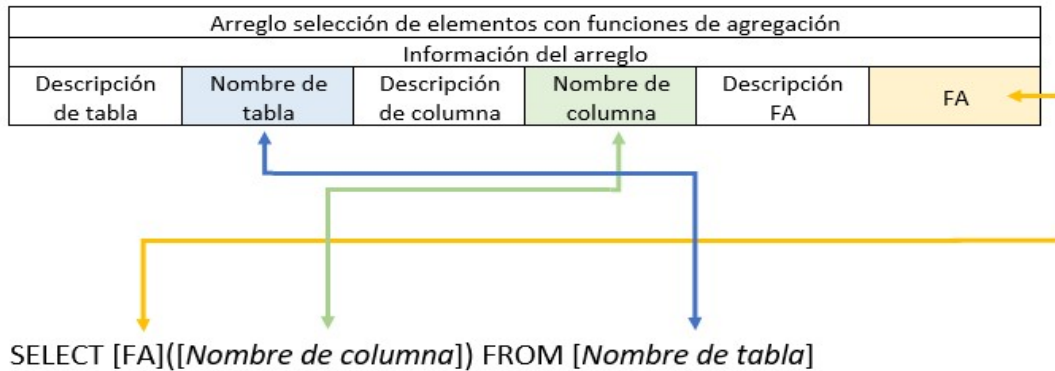


Figura 5.14. Construcción de consulta con FA

Para la consulta del ejemplo de la Sección 5.7, el proceso de construcción de la consulta con FA en SQL es como el que se presenta en la Figura 5.15.

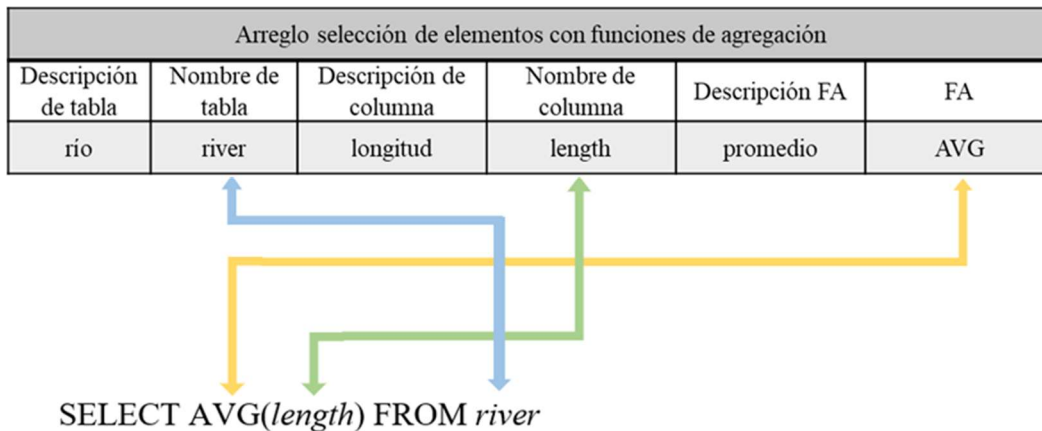


Figura 5.15. Ejemplo de construcción de consulta con FA

5.9 Vista previa y resultado

En esta ventana, la última del proceso de composición, se presenta al usuario el resultado de la consulta o subconsulta que compuso junto con su correcta traducción a SQL. La Tabla 5.8 muestra la descripción de los componentes que conforman esta ventana.

Tabla 5.8. Componentes de la ventana de *Vista previa y resultado*

Componente	Descripción
1. <i>Condiciones de búsqueda</i>	Esta caja contiene las condiciones de búsqueda que fueron definidas por el usuario para la composición de la consulta.

2. Casilla de verificación	Al hacer clic sobre esta casilla, el usuario podrá ver la traducción a SQL de la consulta compuesta.
3. Traducción	En esta caja de texto se presenta al usuario la traducción a SQL de la consulta compuesta.
4. Tabla de resultado	Esta tabla contiene el resultado devuelto por el manejador de bases de datos al terminar el proceso de composición.
5. Botón “Atrás”	Este botón permite al usuario regresar a la ventana anterior en caso de querer corregir alguna de las acciones realizadas en el proceso de composición.
6. Botón “Finalizar”	Este botón permite al usuario concluir la composición de la consulta. Si el fragmento de consulta que se compuso era una subconsulta, entonces continuará el proceso de composición para la siguiente consulta o subconsulta. En caso de que el fragmento sea la consulta externa, entonces da por terminado el proceso de composición.

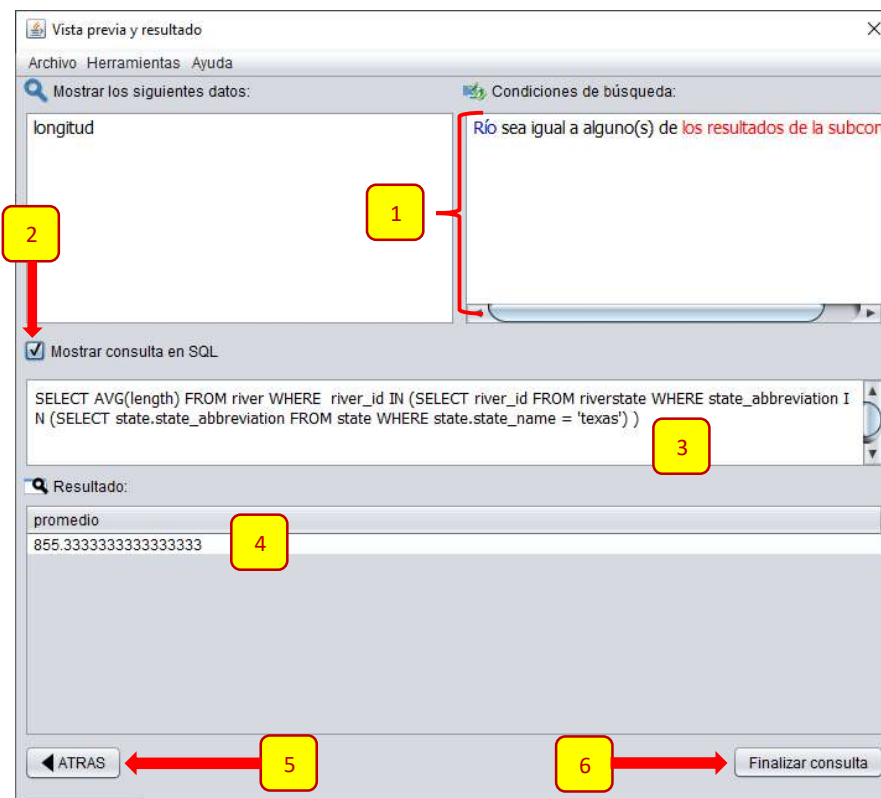


Figura 5.16. Ventana de *Vista previa y resultado*

Como se observa en la Figura 5.16, al terminar el proceso de composición, la interfaz envía a procesar la consulta por medio del SMBD y muestra al usuario el resultado obtenido, así como su traducción a SQL.

5.10 Proceso de composición de consultas que involucran funciones de agregación combinadas con subconsultas

A continuación se presenta un ejemplo del proceso de composición de consultas que involucran funciones de agregación y subconsultas. Para tal efecto, considérese la siguiente consulta:

¿Por cuántos estados corre el río Mississippi?

La primera fase del proceso muestra la ventana *Descomposición de consulta en LN* (Figura 5.17). Habiendo descompuesto mentalmente la consulta, resultan dos fragmentos de consulta: uno correspondiente a la consulta principal y otro a una subconsulta.

Al inicio del uso de la interfaz, el usuario debe escribir la consulta original en la caja de texto correspondiente (1). Acto seguido debe especificar de la lista desplegable (2) el nivel de subconsultas, el cual para este ejemplo es una subconsulta. Posteriormente, el usuario escribirá en las cajas de texto correspondientes la consulta principal (3) y la subconsulta (4).

Ahora debe analizar los fragmentos de consultas e identificar si alguno requiere una función de agregación para su composición. En este caso deberá hacer clic en la casilla de verificación correspondiente al fragmento de la consulta principal (5). Finalmente, el usuario presiona el botón *Continuar* (6) para terminar la fase de descomposición y avanzar a la siguiente ventana.

The screenshot shows a window titled "Descomposición de consulta en LN" with a menu bar containing "Archivo", "Herramientas", and "Ayuda". The window is divided into several sections:

- Consulta original:** A text box containing "Por cuántos estados corre el río Mississippi" (1).
- Consulta principal:** A text box containing "Por cuántos estados corre el río" (3) and a checked checkbox labeled "Función de agregación" (5).
- Selección de niveles de subconsultas:** A dropdown menu showing "Una subconsulta" (2).
- Primera subconsulta:** A text box containing "río Mississippi" (4) and an unchecked checkbox labeled "Función de agregación".
- Segunda subconsulta:** An empty text box and an unchecked checkbox labeled "Función de agregación".
- Tercera subconsulta:** An empty text box and an unchecked checkbox labeled "Función de agregación".
- Botón Continuar:** A green button with a right-pointing arrow and the text "Continuar" (6).

Figura 5.17. Ejemplo de descomposición de consulta

La siguiente ventana corresponde a la *Selección del tema de interés*. En esta ventana el usuario seleccionará el tema del que desea obtener información para la composición de la subconsulta. Como muestra la Figura 5.18, el usuario podrá observar destacada en color azul la consulta a componer: *río Mississippi* (1). Después deberá navegar entre las listas de temas de la BD usando los botones de navegación (2) hasta encontrar el tema que corresponda a la consulta que se quiere componer.

Una vez que se ha localizado el tema, en este caso *río*, se selecciona de la lista de temas (3) y finalmente se presiona el botón *Continuar* (4) para avanzar a la siguiente ventana.



Figura 5.18. Ventana de *Selección del tema de interés*

La siguiente ventana corresponde a la *Especificación de condiciones de búsqueda*. En esta ventana el usuario se encargará de definir las condiciones de búsqueda que restringen la información de la consulta. Este ejemplo requiere que se defina una condición de búsqueda: que el nombre del río sea *Mississippi*.

La Figura 5.19 muestra el proceso que debe seguir el usuario para definir correctamente la condición de búsqueda deseada. Primeramente, el usuario identifica dentro del árbol de composición (1) el elemento al que quiere aplicar la condición de búsqueda. Para este ejemplo se selecciona el elemento *río*. Después el usuario seleccionará de la lista de condiciones (2) la opción que contenga la condición que desea aplicar, que en este caso es la opción *sea*.

Posteriormente, el usuario debe especificar un valor de búsqueda escribiéndolo en la caja de texto correspondiente (3). Para el ejemplo, el valor de búsqueda es *Mississippi*.

Después de escribir el valor de búsqueda deseado, se debe presionar el botón *añadir condición* (4).

Al presionar dicho botón, la condición de búsqueda se agregará a la caja *Condiciones para restringir la búsqueda* (5). Finalmente, el usuario debe hacer clic en el botón *Continuar* (6) para continuar el proceso de composición.

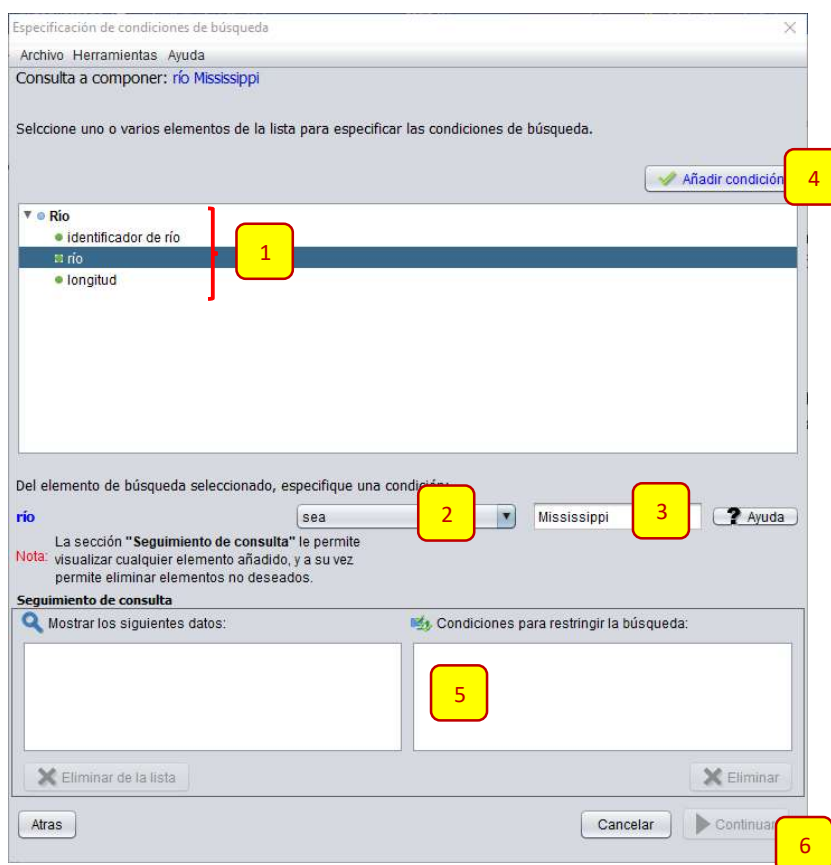


Figura 5.19. Ventana de *Especificación de condiciones de búsqueda*

La ventana que se muestra en la Figura 5.20 corresponde a *Vista previa y resultado*. En ella se muestra el resultado de la consulta compuesta hasta el momento y su traducción a SQL.

Una vez que el usuario ha terminado la composición de la subconsulta, la interfaz la envía a procesar por el SMBD y le muestra el resultado al usuario en la tabla de resultados (1). Es opcional para el usuario ver la traducción a SQL de la consulta construida. En caso de querer ver la expresión en SQL, debe hacer clic sobre la casilla (2) para que así la interfaz muestre el código en SQL en la caja de texto correspondiente (3).

Una vez que el usuario ha visto el resultado de la subconsulta, da por terminada su composición presionando el botón *Finalizar subconsulta* (4).

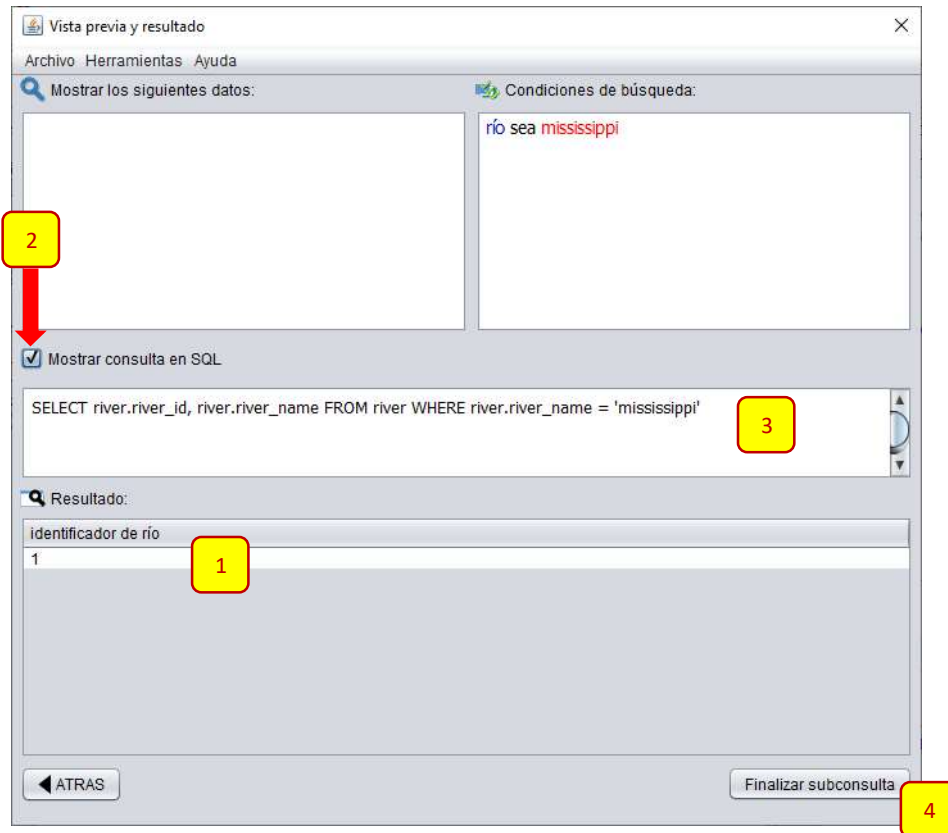


Figura 5.20. Ventana de *Vista previa y resultado* para subconsulta

Después de componer la subconsulta, este ejemplo continúa con la composición de la consulta principal. Así que la siguiente ventana en desplegarse será la de *Selección del tema de interés*, esta vez para la consulta principal.

Al continuar el proceso de composición, el usuario podrá observar destacada en color azul la consulta a componer: *por cuántos estados corre el río* (1). Después debe seleccionar de la lista de temas (2) el que se ajuste a lo que pide el fragmento de consulta que se está componiendo, en este caso es el tema *Estado*. Ahora debe presionar el botón *Continuar* para ir a la siguiente ventana. La Figura 5.21 muestra los pasos que debe seguir el usuario en la ventana de *Selección del tema de interés* de la consulta externa.

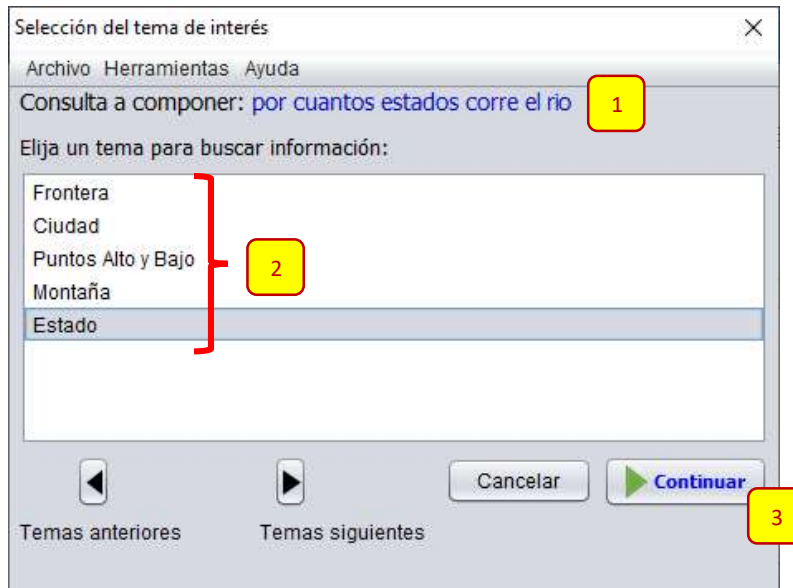


Figura 5.21. Ventana de *Selección del tema de interés* para la consulta principal

La siguiente ventana en mostrarse corresponde a la *Selección de elementos de interés* (Figura 5.22). Para este ejemplo la interfaz detecta que la consulta externa requiere la aplicación de una función de agregación. Es por esto que se muestran los componentes que permitan aplicar la FA deseada. Para la consulta *Por cuántos estados corre el río*, el usuario detecta que la función de agregación requerida es la que permita hacer un conteo, es decir, la función COUNT.

Este es un caso particular del árbol de composición, pues en otras ventanas que hacen uso de él sólo se pueden seleccionar nodos hijos. En el caso donde la selección de elementos necesite la aplicación de una FA, será posible seleccionar también el nodo raíz del árbol.

Y éste es justo el caso del ejemplo. Para componer correctamente esta consulta, el usuario debe seleccionar el nodo raíz *estado* que se refiere a la tabla *State* de la BD. Una vez que el usuario hace la selección en el árbol de composición (1), podrá seleccionar de la lista desplegable (2) la opción que corresponda a la FA requerida.

En el caso del ejemplo, sólo se muestra la opción que hace referencia a la función COUNT, debido a que el nodo seleccionado del árbol representa una tabla. El resto de las funciones de agregación se aplican específicamente a alguna columna y, por esta razón, no son elegibles para este ejemplo.

Después de elegir la función de agregación a aplicar, el usuario debe presionar el botón añadir elemento (3) y finalmente presionar el botón *Continuar* (4) para avanzar a la siguiente ventana.

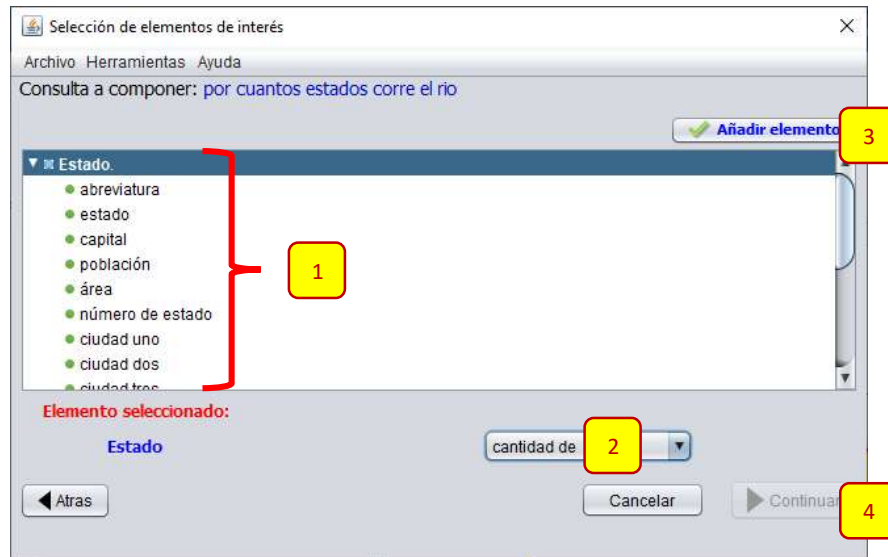


Figura 5.22. Ventana de *Selección de elementos de interés* para consultas con FA

La siguiente ventana que se despliega es *Resultado de subconsulta interna*, la cual muestra los resultados de las subconsultas generadas internamente por la interfaz. Que el resultado mostrado tenga congruencia con lo que el usuario espera de su composición es un buen indicativo de que el proceso de composición se ha realizado correctamente hasta ese punto. La Figura 5.23 muestra los componentes que incluye esta ventana. En ésta el usuario podrá revisar la tabla de resultados (1) y presionar el botón *Continuar* (2) para seguir avanzando en el proceso de composición.

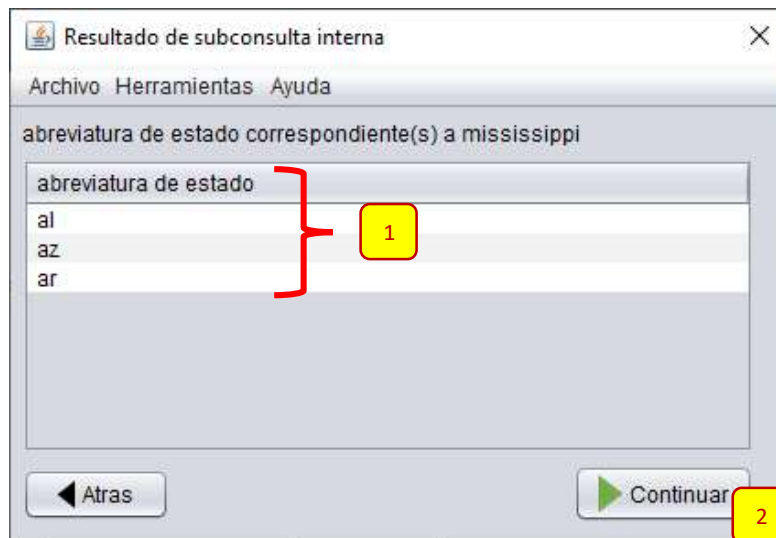


Figura 5.23. Ventana de *Resultado de subconsulta interna*

Ahora se presenta la ventana *Especificación de condición para subconsultas*. Esta ventana permite al usuario definir una condición de búsqueda para restringir los resultados

de la subconsulta. Para el ejemplo, como se observa en la Figura 5.24, el usuario debe seleccionar de la lista desplegable (1) la opción *sea igual a alguno(s) de*. Después de definir la condición de búsqueda, el usuario debe presionar el botón *Continuar* (2) y presionar *NO* en la ventana emergente, debido a que no se desea definir más condiciones de búsqueda.

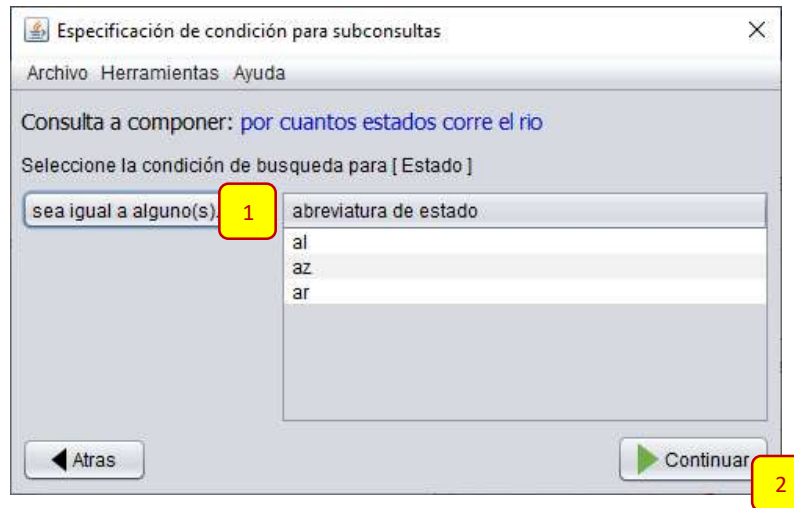


Figura 5.24. Ventana de *Especificación de condición para subconsultas*

Finalmente, se ha llegado a la fase *Vista previa y resultado*. En ésta la ventana mostrará al usuario el resultado de la consulta completa. Como ilustra la Figura 5.25, la interfaz genera la expresión en SQL (1) correspondiente a la consulta que se ingresó en LN. Una vez que genera el código en SQL, procesa la consulta por medio del SMBD para mostrar el resultado de la consulta que fue compuesta por el usuario (2).

Por último, para dar por terminado el proceso de composición, se presiona el botón *Finalizar consulta* (3), y así termina la composición de una consulta que involucra funciones de agregación combinadas con subconsultas.

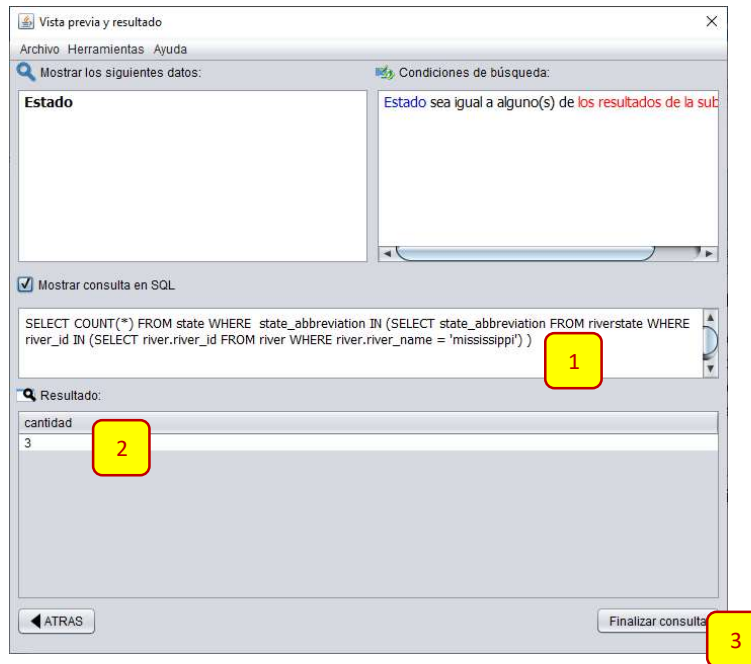


Figura 5.25. Ventana de *Vista previa y resultado*

Capítulo 6

Pruebas experimentales

6.1 Pruebas funcionales

Las pruebas funcionales se realizan con la intención de evaluar el funcionamiento de la interfaz de composición al componer consultas de los tipos 1, 2 y 3.

En la siguiente subsección se describirán las pruebas realizadas a la interfaz.

6.1.1 Descripción de las pruebas funcionales

Para la realización de las pruebas funcionales, se consideraron tres tipos de consultas que incluyen funciones de agregación (FAs) y subconsultas. Dichos tipos de consultas son los siguientes:

- Consulta tipo 1 (Incluye una FA en la subconsulta más interna).
- Consulta tipo 2 (Incluye una FA en la subconsulta más interna, pero involucra sólo una tabla).
- Consulta tipo 3 (Incluye una FA en la consulta externa).

Las pruebas funcionales tienen el propósito de demostrar el correcto funcionamiento de la interfaz en el proceso de composición de consultas que involucran FAs y subconsultas. Para llevar a cabo las pruebas, se seleccionó una consulta de cada tipo del corpus de pruebas y se realizó la composición de las consultas seleccionadas en la interfaz.

6.1.2 Consulta tipo 1

¿Cuál aerolínea tiene el vuelo con la tarifa de viaje sencillo más costosa?

La consulta de este ejemplo contiene una función de agregación en la subconsulta más interna. El proceso de composición para esta consulta inicia identificando las condiciones de búsqueda, los elementos de interés, y la necesidad de aplicar o no una FA.

La descomposición en subconsultas da como resultado los siguientes fragmentos de consulta:

Subconsulta: tarifa de viaje sencillo más costosa.

Subconsulta: vuelo con la tarifa.

Consulta externa: Cuál aerolínea tiene el vuelo.

Primero se necesita realizar la composición para la subconsulta más interna. Se debe seleccionar el tema *Tarifa*, y posteriormente se selecciona el elemento *tarifa de viaje sencillo*. Para componer de forma correcta esta subconsulta, es necesario aplicar la FA MAX al elemento *tarifa de viaje sencillo*. De esta manera se obtiene el resultado de la subconsulta y su traducción a SQL. La expresión generada por la interfaz es la siguiente:

```

SELECT fare_code
FROM fare
WHERE fare.one_way_cost IN (
    SELECT MAX(one_way_cost)
    FROM fare )

```

Para continuar la composición, se debe componer la siguiente subconsulta. Para tal efecto se debe seleccionar el tema *Vuelo*, entonces la interfaz encontrará automáticamente las columnas que relacionen la tabla *Vuelo* con la tabla *Tarifa*, que es la tabla correspondiente a la subconsulta más interna. Posteriormente la interfaz genera internamente la subconsulta correspondiente a la subconsulta intermedia. La expresión en SQL correspondiente a dicha subconsulta es la siguiente:

```

SELECT flight_code
FROM flight
WHERE flight_code IN (
    SELECT flight_code
    FROM flight_fare
    WHERE fare_code IN (
        SELECT fare_code
        FROM fare
        WHERE fare.one_way_cost IN (
            SELECT MAX(one_way_cost)
            FROM fare )) )

```

Para continuar debe componerse la consulta externa. Para esto, se selecciona el elemento de interés *nombre de la aerolínea* que se encuentra en el tema de interés *Aerolínea*. Después debe especificarse la condición de búsqueda para la subconsulta. De esta manera, la composición de la consulta llega a su fin, mostrando el resultado y la traducción a SQL de la consulta completa. La expresión en SQL generada finalmente por la interfaz es la siguiente:

```

SELECT airline_name
FROM airline
WHERE airline_code IN (
    SELECT airline_code
    FROM flight
    WHERE flight_code IN (
        SELECT flight_code
        FROM flight_fare
        WHERE fare_code IN (
            SELECT fare_code
            FROM fare
            WHERE fare.one_way_cost IN (
                SELECT MAX(one_way_cost)
                FROM fare )) ) )

```

6.1.3 Consulta tipo 2

¿Cuál es el estado con el área más grande?

Esta segunda consulta requiere de la aplicación de una FA en la subconsulta más interna y además involucra solamente información de una tabla. Para iniciar la composición, se inicia la descomposición en subconsultas, dando ésta como resultado los siguientes fragmentos de consulta:

Subconsulta: área más grande de estado.
Consulta externa: Cuál es el estado con el área.

Posteriormente se realizará la composición de la subconsulta. Para hacerlo, se selecciona el tema *Estado* y, dentro de este tema, se selecciona el elemento *área*. Para obtener el resultado esperado, se debe aplicar la FA MAX al elemento seleccionado. La interfaz muestra el resultado de la subconsulta, así como la traducción a SQL de dicha subconsulta. La expresión en SQL generada por la interfaz es la siguiente:

```
SELECT state_abbreviation
FROM state
WHERE state.area IN (
  SELECT MAX(area)
  FROM state )
```

Ahora se debe componer la consulta externa. La información requerida por la consulta se encuentra en el elemento *nombre de estado* que se encuentra en el tema *Estado*. La tabla que contiene la información requerida tanto por la subconsulta como por la consulta externa es la misma, por lo tanto, se trata de una consulta del tipo 2.

Finalmente, la interfaz obtiene el resultado y genera la expresión en SQL correspondiente a la consulta completa. Dicha expresión en SQL es la siguiente:

```
SELECT state_name
FROM state
WHERE state_abbreviation IN (
  SELECT state_abbreviation
  FROM state
  WHERE state.area IN (
    SELECT MAX(area)
    FROM state ))
```

6.1.4 Consulta tipo 3

¿Por cuántos estados corre el río Mississippi?

Esta tercera consulta requiere la aplicación de una FA en la consulta externa. Para realizar la composición de esta consulta, primero debe descomponerse en subconsultas la

consulta en LN. Después de identificar las condiciones de búsqueda y la necesidad de aplicar una FA, se obtienen los siguientes fragmentos de consultas:

Subconsulta: río Mississippi.

Consulta externa: Por cuántos estados corre el río.

Primeramente, se debe componer la subconsulta. Para llevar a cabo la composición, se debe seleccionar el tema *Río* y, dado que se busca un río particular, debe especificarse un valor de búsqueda, para el ejemplo es que el *nombre del río* sea *Mississippi*. La interfaz procesa los datos seleccionados y genera la traducción a SQL de la subconsulta. La expresión en SQL es la siguiente:

```
SELECT river.river_id
FROM river
WHERE river.river_name = 'mississippi'
```

Después se debe componer la consulta externa, para lo cual se determina que la consulta requiere la aplicación de una FA. Para continuar el proceso de composición, se seleccionará el tema *Estado* y, en la fase *selección de elementos de interés*, debe seleccionarse el nodo *Estado*. Debido a que se requiere saber la *cantidad de estados*, se debe aplicar la FA COUNT.

Finalmente, la interfaz mostrará el resultado y la traducción a SQL de la consulta completa. Puede observarse que la interfaz genera automáticamente las subconsultas internas que relacionan la subconsulta más interna con la consulta externa. La expresión en SQL generada por la interfaz es la siguiente:

```
SELECT COUNT(*)
FROM state
WHERE state_abbreviation IN (
  SELECT state_abbreviation
  FROM riverstate
  WHERE river_id IN (
    SELECT river.river_id
    FROM river
    WHERE river.river_name = 'mississippi' ) )
```

Las pruebas de funcionalidad realizadas demuestran que la nueva versión de la interfaz funciona correctamente con las consultas de los tres casos mencionados en la sección 1.1 de este trabajo.

Capítulo 7

Conclusiones y trabajos futuros

En este capítulo se presentan las conclusiones y aportaciones realizadas por este proyecto. También se mencionan los futuros trabajos que se pueden realizar a fin de mejorar el desempeño de la interfaz de composición, tanto en su usabilidad como en la complejidad de las consultas que permite formular.

7.1 Conclusiones

Hoy en día la información es uno de los recursos más valiosos con los que cuentan las empresas y organizaciones en el mundo. La información almacenada en bases de datos ofrece a las distintas organizaciones, sin importar su giro, una mayor certeza a la hora de la toma de decisiones.

Si bien es cierto que, para sacar provecho de los datos almacenados, es necesario contar con usuarios expertos que puedan acceder a dichos datos y manipularlos, también son cada vez más las alternativas que se presentan para acceder a la información contenida en bases de datos y hacer uso de ella. Sin embargo, la gran mayoría de las herramientas desarrolladas requieren que el usuario tenga algún tipo de conocimiento sobre bases de datos (como puede verse en la Sección 2.2), y por ende la información sigue fuera del alcance de usuarios casuales.

Con el fin de facilitar la composición de consultas para aquellos usuarios que no poseen conocimientos conceptuales ni prácticos en bases de datos, se ha diseñado e implementado una interfaz de composición de consultas (descrita en [Aguirre, 2015]). Esta interfaz permite componer consultas de manera gráfica y sencilla. De modo que la información ya está al alcance de usuarios que no dominen algún lenguaje de consulta.

Para implementar la nueva funcionalidad fue necesario mantener el funcionamiento del módulo de subconsultas propuesto y descrito en [Castillo, 2016]. Dicho módulo guía al usuario a realizar el proceso de composición desde los niveles más internos de subconsultas hacia la consulta externa, y gracias a él se han logrado componer consultas que involucran desde 1 hasta 3 subconsultas.

La nueva funcionalidad desarrollada en este proyecto permite al usuario componer consultas que incluyen funciones de agregación combinadas con subconsultas. Para lograrlo la interfaz muestra diferentes componentes gráficos en función de la necesidad de aplicar o no una función de agregación en alguna de las subconsultas o en la consulta externa. De tal forma que ahora es posible componer consultas con un mayor grado de complejidad, mientras que el dominio de SQL por el usuario ya no es requisito para usar la interfaz.

Las pruebas funcionales demostraron que el nuevo módulo para funciones de agregación funciona correctamente para componer consultas de los casos mencionados en la Sección 1.1. Al igual que las versiones anteriores de la ICC, esta versión mantiene la independencia de dominio.

7.2 Trabajos futuros

Considerando el desempeño de la interfaz en las pruebas funcionales, es posible proponer mejoras que pueden aplicarse a futuro con la finalidad de ofrecer una mejor experiencia al usuario y permitirle componer correctamente consultas con un mayor grado de complejidad. Específicamente, las mejoras propuestas son las siguientes:

1. Implementar una funcionalidad que permita a la interfaz componer consultas que involucren las cláusulas Group by y Having.
2. Diseñar e implementar un buscador para los elementos del árbol de composición. Esto para facilitar la fase *Selección del tema de interés*.
3. Diseñar e implementar una mejora a la fase *Descomposición de consulta en LN* que permita a la interfaz descomponer automáticamente consultas en LN que sigan cierta estructura.
4. Evaluar la posibilidad de incorporar la ICC al núcleo de la ILNBD a fin de que la ILNBD ofrezca al usuario usar la ICC al detectar que tiene problemas en la formulación de consultas.
5. Evaluar la posibilidad de migrar la interfaz de composición a un sistema web. Esto con el propósito de explotar las características y funciones que ofrecen las tecnologías de web. Además esto facilitaría la incorporación de la ICC al núcleo de la ILNBD.

Apéndices

Apéndice A. Descripción de la BD ATIS

La base de datos ATIS consta de 28 tablas y 125 columnas. La complejidad de esta base de datos es mayor en comparación con Geobase y con otras bases de datos que también se usan en la literatura para hacer pruebas a ILNBDs. La BD ATIS almacena información sobre vuelos.

Para la descripción de la base de datos ATIS se presenta el nombre de cada tabla junto con su descripción en LN; mientras que por cada columna de la tabla se muestran el nombre de la columna, el tipo de dato de la columna y finalmente la descripción en LN de la columna.

Tabla: flight y flight_1 Descripción: Vuelo de un aeropuerto a otro

Columna	Tipo	Descripción
flight_code	Numérico	Código de vuelo
flight_days	Texto	Días hábiles
from_airport	Texto	Aeropuerto de origen
to_airport	Texto	Aeropuerto de destino
departure_time	Numérico	Hora de salida
arrival_time	Numérico	Hora de llegada
airline_code	Texto	Código de aerolínea
flight_number	Numérico	Número de vuelo
class_string	Texto	Código de clase
aircraft_code	Texto	Código de avión
meal_code	Texto	Código de comida
stops	Numérico	Número de escalas
dual_carrier	Texto	Empresa dual
time_elapsed	Numérico	Tiempo de viaje

LLAVE PRIMARIA: flight_code
LLAVE FORÁNEA: flight (aircraft_code) – aircraft (aircraft_code)
flight (airline_code) – airline (airline_code)
flight (from_airport) – airport (airport_code)
flight (to_airport) – airport (airport_code)

Tabla: airline y airline_1 Descripción: Aerolínea

Columna	Tipo	Descripción
airline_code	Texto	Código de aerolínea

airline_name	Texto	Nombre de aerolínea
notes	Texto	Notas

LLAVE PRIMARIA: airline_code

LLAVE FORÁNEA: airline (airline_code) – restrict_carrier (airline_code)

Tabla: fare Descripción: Tarifa

Columna	Tipo	Descripción
fare_code	Texto	Código de tarifa
from_airport	Texto	Aeropuerto de origen
to_airport	Texto	Aeropuerto de destino
fare_class	Texto	Clase de tarifa
fare_airline	Texto	Tarifa de aerolínea
restrict_code	Texto	Código de restricción
one_way_cost	Numérico	Tarifa de viaje sencillo
rnd_trip_cost	Numérico	Tarifa de viaje redondo

LLAVE PRIMARIA: fare_code

LLAVE FORÁNEA: fare (restrict_code) – restriction (restrict_code)
fare (fare_class) – compound_class (fare_class)

Tabla: aircraft Descripción: Avión

Columna	Tipo	Descripción
aircraft_code	Texto	Código de avión
aircraft_type	Texto	Tipo de avión
engines	Numérico	Número de motores
category	Texto	Categoría de avión
wide_body	Texto	Fuselaje ancho
wing_span	Numérico	Extensión de alas
length1	Numérico	Tamaño de equipo
weight	Numérico	Peso
capacity	Numérico	Número de asientos
pay_load	Numérico	Capacidad de carga
cruising_speed	Numérico	Velocidad
range_miles	Numérico	Longitud de vuelo

pressurized Texto Presurización

LLAVE PRIMARIA: aircraft_code

Tabla: transport **Descripción:** Servicio de transporte del aeropuerto

Columna	Tipo	Descripción
transport_code	Texto	Código de transporte
transport_desc	Texto	Descripción de transporte

LLAVE PRIMARIA: transport_code

Tabla: compound_class **Descripción:** Clase compuesta

Columna	Tipo	Descripción
fare_class	Texto	Código de clase de tarifa
base_class	Texto	Clase base
class_type	Texto	Tipo de clase
premium	Texto	Es clase premium?
economy	Texto	Es clase económica?
discounted	Texto	Tiene descuento?
night	Texto	Es vuelo nocturno?
season_fare	Texto	Tipo de tarifa de temporada
class_days	Texto	Días en que se aplica la clase

LLAVE PRIMARIA: fare_class
LLAVE FORÁNEA: compound_class (base_class) – class_of_service (class_code)

Tabla: ground_service **Descripción:** Servicio terrestre

Columna	Tipo	Descripción
city_code	Texto	Código de ciudad
airport_code	Texto	Código de aeropuerto
transport_code	Texto	Código de transporte
ground_fare	Numérico	Tarifa terrestre

LLAVE PRIMARIA: city_code, airport_code, transport_code
LLAVE FORÁNEA: ground_service (transport_code) – transport (transport_code)

Tabla: food_service **Descripción:** Servicio de comida

Columna	Tipo	Descripción
meal_code	Texto	Código de comida
meal_number	Numérico	Número de comida
meal_class	Texto	Clase de comida
meal_description	Texto	Descripción de comida

LLAVE FORÁNEA: food_service (meal_code) – flight (meal_code)

Tabla: restriction Descripción: Restricción

Columna	Tipo	Descripción
restrict_code	Texto	Código de restricción
application	Texto	Aplicación
no_discounts	Texto	Descuentos no aplicables
reserve_ticket	Numérico	Boletos en reserve
stopovers	Texto	Escalas
return_min	Numérico	Mínimo de permanencia
return_max	Numérico	Máximo de permanencia

LLAVE PRIMARIA: restrict_code

Tabla: city Descripción: Ciudad

Columna	Tipo	Descripción
city_code	Texto	Código de ciudad
city_name	Texto	Nombre de la ciudad
state_code	Texto	Código del estado
time_zone_code	Texto	Código de zona horaria

LLAVE PRIMARIA: city_code
LLAVE FORÁNEA: city (state_code) – state (state_code)

Tabla: class_of_service Descripción: Clase de servicio

Columna	Tipo	Descripción
class_code	Texto	Código de clase de servicio
rank	Numérico	Rango
class_description	Texto	Descripción de clase de servicio

LLAVE PRIMARIA: class_code

Tabla: airport_service **Descripción:** Servicio de aeropuerto

Columna	Tipo	Descripción
city_code	Texto	Código de ciudad
airport_code	Texto	Código de aeropuerto
miles_distant	Numérico	Distancia en millas
direction	Texto	Dirección
minutes_distant	Numérico	Distancia en minutos

LLAVE PRIMARIA: city_code, airport_code

LLAVE FORÁNEA: airport_service (airport_code) – airport (airport_code)

airport_service (airport_code) – ground_service (airport_code)

airport_service (city_code) – ground_service (airport_service)

airport_service (city_code) – city (city_code)

Tabla: fconnection **Descripción:** Conexión de vuelo

Columna	Tipo	Descripción
connect_code	Numérico	Código de conexión
from_airport	Texto	Aeropuerto de origen
to_airport	Texto	Aeropuerto de destino
departure_time	Numérico	Hora de salida
arrival_time	Numérico	Hora de llegada
flight_days	Texto	Días de vuelo
stops	Numérico	Escalas
connections	Numérico	Conexiones

LLAVE PRIMARIA: connect_code

LLAVE FORÁNEA: fconnection (to_airport) – airport (airport_code)

fconnection (from_airport) – airport (airport_code)

Tabla: connect_leg **Descripción:** Segmento de conexión

Columna	Tipo	Descripción
connect_code	Numérico	Código de conexión
leg_number	Numérico	Número de segmento
flight_code	Numérico	Código de vuelo

LLAVE FORÁNEA: connect_leg (connect_code) – fconnection (connect_code)
 connect_leg (flight_code) – flight (flight_code)

Tabla: flight_day Descripción: Días de vuelo

Columna	Tipo	Descripción
day_mask	Texto	Máscara de días
day_code	Numérico	Código de día

LLAVE PRIMARIA: day_mask, day_code
LLAVE FORÁNEA: flight_day (day_code) – day_name (day_code)

Tabla: day_name Descripción: Días

Columna	Tipo	Descripción
day_code	Numérico	Código de día
day_name	Texto	Nombre de día

LLAVE PRIMARIA: day_code

Tabla: state Descripción: Estado

Columna	Tipo	Descripción
state_code	Texto	Código de estado
state_name	Texto	Nombre de estado
country_name	Texto	Nombre de país

LLAVE PRIMARIA: state_code

Tabla: dual_carrier Descripción: Empresa dual

Columna	Tipo	Descripción
main_airline	Texto	Código de aerolínea principal
dual_airline	Texto	Código de empresa dual
low_flight	Numérico	Vuelo económico
high_flight	Numérico	Vuelo costoso
fconnection_name	Texto	Nombre de conexión

LLAVE FORÁNEA: dual_carrier (dual_airline) – airline (airline_code)
 dual_carrier (main_airline) – airline (airline_code)

Tabla: time_zone Descripción: Zona horaria

Columna	Tipo	Descripción
time_zone_code	Texto	Código de zona horaria
time_zone_name	Texto	Nombre de zona horaria
LLAVE FORÁNEA: time_zone (time_zone_code) – city (time_zone_code)		

Tabla: stop1 Descripción: Escalas

Columna	Tipo	Descripción
flight_code	Numérico	Código de vuelo
stop_number	Numérico	Número de escala
stop_flight	Numérico	Vuelo con escalas
LLAVE FORÁNEA: stop1 (flight_code) – flight (flight_code) stop1 (stop_flight) – flight (flight_code)		

Tabla: flight_fare Descripción: Tarifas de los vuelos

Columna	Tipo	Descripción
flight_code	Numérico	Código de vuelo
fare_code	Texto	Código de tarifa
LLAVE FORÁNEA: flight_fare (fare_code) – fare (fare_code) flight_fare (flight_code) – flight (flight_code)		

Tabla: restrict_carrier Descripción: Restricciones de aerolínea

Columna	Tipo	Descripción
restrict_code	Texto	Código de restricción
airline_code	Texto	Código de aerolínea
LLAVE FORÁNEA: restrict_carrier (restrict_code) – restriction(restrict_code)		

Tabla: restrict_class Descripción: Restricción de clase

Columna	Tipo	Descripción
restrict_code	Texto	Código de restricción
ex_fare_class	Texto	Tarifa de clase
LLAVE FORÁNEA: restrict_class (restrict_code) – restriction(restrict_code)		

El esquema de BD de ATIS se muestra en la Figura A.1. Esta base de datos cuenta con un total de 27 tablas y 123 columnas.

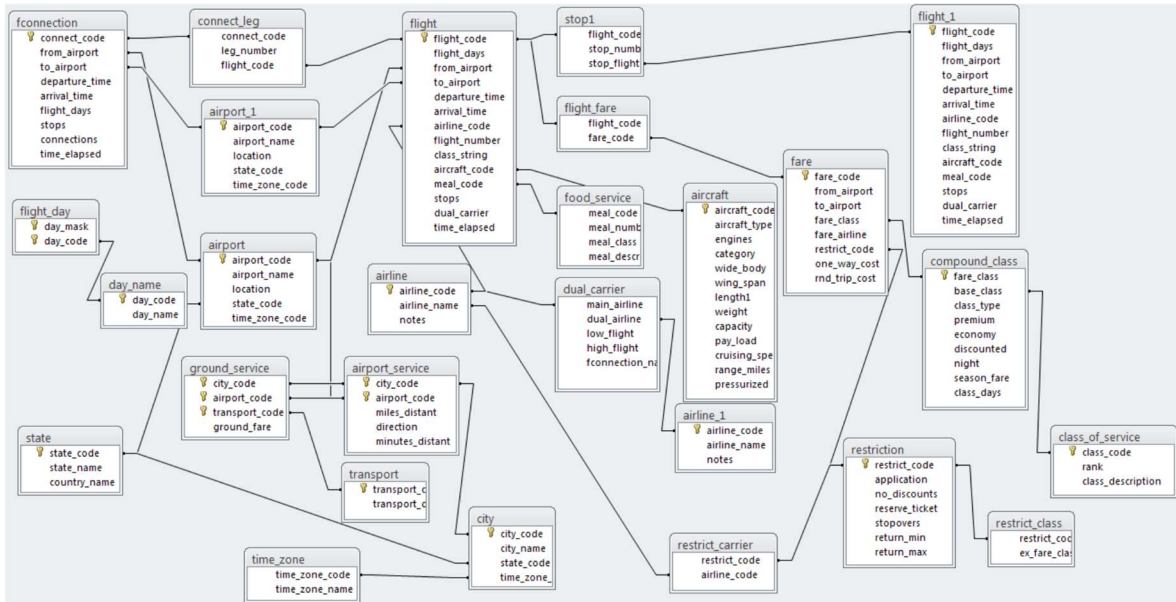


Figura A.1. Esquema de la base de datos ATIS.

Apéndice B. Descripción de la BD Geobase

La base de datos Geobase almacena información geográfica de los Estados Unidos de América. Dicha base de datos ha sido ampliamente utilizada como BD de prueba para distintas ILNBDs.

La descripción de la base de datos Geobase se compone por el nombre de cada tabla y su descripción en LN; mientras que por cada columna de la tabla se muestra el nombre de la columna, el tipo de dato de la columna y finalmente la descripción en LN de la columna.

Tabla: City Descripción: Ciudad

Columna	Tipo	Descripción
state_abbreviation	Texto corto	Abreviatura de estado
city_name	Texto corto	Ciudad
state_name	Texto corto	Nombre de estado
population	Número	Población
LLAVE FORÁNEA: city (state_abbreviation) – state (abbreviation)		

Tabla: HighLow Descripción: Puntos Alto y bajo

Columna	Tipo	Descripción
state_abbreviation	Texto corto	Abreviatura de estado
state_name	Texto corto	Nombre de estado
highest_point	Texto corto	Punto más alto
highest_elevation	Número	Elevación más alta
lowest_point	Texto corto	Punto más bajo
lowest_elevation	Número	Elevación más baja
LLAVE FORÁNEA: HighLow (state_abbreviation) – state (abbreviation)		

Tabla: Border Descripción: Frontera

Columna	Tipo	Descripción
state_abbreviation	Texto corto	Abreviatura de estado
state_name	Texto corto	Nombre de estado
bordering_state	Texto corto	Colindancia
LLAVE FORÁNEA: Border (state_abbreviation) – state (abbreviation)		

Tabla: State Descripción: Estado

Columna	Tipo	Descripción
abbreviation	Texto corto	Abreviatura
state_name	Texto corto	Estado
capital	Texto corto	Capital
population	Número	Población
area	Número	Área
state_number	Número	Número de estado
city1	Texto corto	Ciudad uno
city2	Texto corto	Ciudad dos
city3	Texto corto	Ciudad tres
city4	Texto corto	Ciudad cuatro
LLAVE FORÁNEA: State (abbreviation) – RoadState (state_abbreviation)		

Tabla: Mountain Descripción: Montaña

Columna	Tipo	Descripción
mountain_id	Número	Identificador de montaña
state_abbreviation	Texto corto	Abreviatura de estado
state_name	Texto corto	Nombre de estado
mountain_name	Texto corto	Montaña
height	Número	Altura
LLAVE FORÁNEA: Mountain (state_abbreviation) – state (abbreviation)		

Tabla: RiverState Descripción: Estado del río

Columna	Tipo	Descripción
river_id	Número	Identificador de río
state_abbreviation	Texto corto	Abreviatura de estado
LLAVE FORÁNEA: RiverState (state_abbreviation) – state (abbreviation)		

Tabla: LakeState Descripción: Estado del lago

Columna	Tipo	Descripción
lake_id	Número	Identificador de lago
state_abbreviation	Texto corto	Abreviatura de estado

LLAVE FORÁNEA: LakeState (state_abbreviation) – state (abbreviation)

Tabla: RoadState Descripción: Estado de la carretera

Columna	Tipo	Descripción
number	Número	Identificador de carretera
state_abbreviation	Texto corto	Abreviatura de estado

LLAVE FORÁNEA: RoadState (state_abbreviation) – state (abbreviation)

Tabla: River Descripción: Río

Columna	Tipo	Descripción
river_id	Número	Identificador de río
river_name	Texto corto	Río
length	Número	Longitud

LLAVE FORÁNEA: River (river_id) – RiverState (river_id)

Tabla: Lake Descripción: Lago

Columna	Tipo	Descripción
lake_id	Número	Abreviatura de estado
lake_name	Texto corto	Lago
area	Número	Área

LLAVE FORÁNEA: Lake (lake_id) – LakeState (lake_id)

Tabla: Road Descripción: Carretera

Columna	Tipo	Descripción
number	Número	Carretera

LLAVE FORÁNEA: Road (number) – RoadState (number)

El esquema de la BD Geobase se muestra en la Figura B.1.

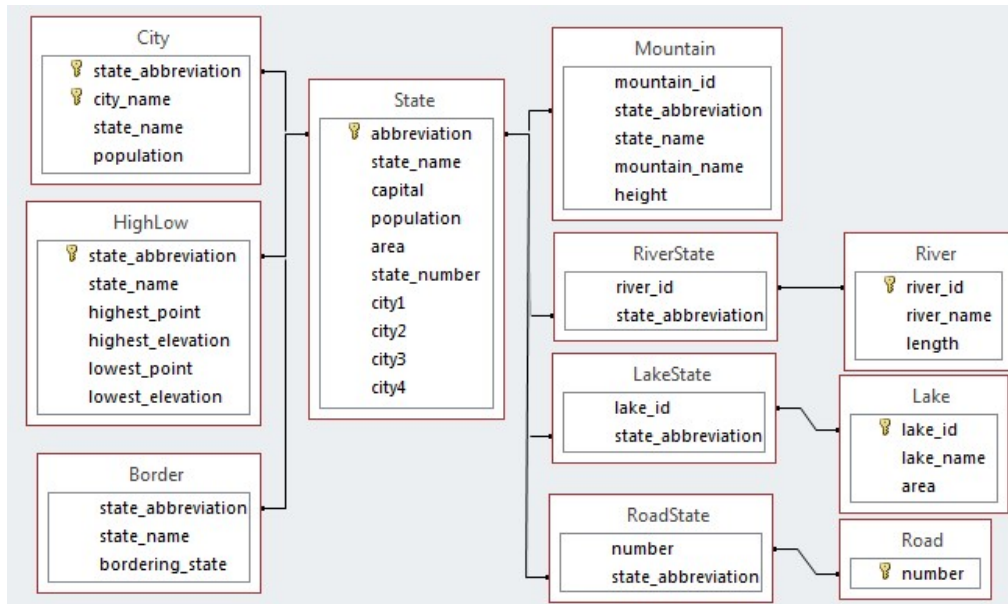


Figura B.1. Esquema de la base de datos Geobase.

Referencias

- [ActiveDBSoft, 2018] ActiveDBSoft, 2018. [En línea]. Disponible en: <https://www.activequerybuilder.com>. Accedido: 26 feb. 2021.
- [Aguirre, 2015] A. G. Aguirre, “Módulo para composición de consultas para una interfaz de lenguaje natural a bases de datos”, tesis de maestría, División de Estudios de Posgrado e Investigación, Instituto Tecnológico de Ciudad Madero, Cd. Madero, Tamps., México, 2015.
- [Aguirre, 2014] M. A. Aguirre, “Modelo semánticamente enriquecido de bases de datos para su explotación por interfaces de lenguaje natural”, tesis de doctorado, División de Estudios de Posgrado e Investigación, Instituto Tecnológico de Tijuana, Tijuana, México, 2014.
- [Camps, 2005] R. Camps, L. A. Casillas, D. Costal, M. Gibert, C. M. Escofet y O. Pérez, *Bases de datos*, 1.^a ed., Barcelona, España: Eureka Media, SL, 2005.
- [Castillo, 2016] G. Castillo, “Interfaz de composición de consultas a bases de datos que permita incluir subconsultas”, tesis de maestría, División de Estudios de Posgrado e Investigación, Instituto Tecnológico de Ciudad Madero, Cd. Madero, Tamps., México, 2016.
- [Cobo, 2007] A. Cobo, *Diseño y programación de bases de datos*, Madrid, España: Vision Libros, 2007.
- [Coronel, 2016] C. Coronel y S. Morris, *Database Systems: Design, Implementation, and Management*, 12.^a ed. Boston: Course Technology, 2016.
- [Cortez, 2009] A. Cortez, H. Vega, J. Pariona y A. M. Huayna, “Procesamiento de lenguaje natural”, *Revista de Ingeniería de Sistemas e Informática*, vol. 6, no. 2, pp. 45–54, Julio 2009.
- [Gaspar, 2017] J. Gaspar, “Tratamiento de subconsultas combinadas con funciones de agregación en una interfaz de lenguaje natural para consulta a bases de datos”, tesis de maestría, División de Estudios de Posgrado e Investigación, Instituto Tecnológico de Ciudad Madero, Cd. Madero, Tamps., México, 2017.
- [Hallett, 2007] C. Hallett, D. Scott y R. Power, “Composing questions through conceptual authoring”, *Computational Linguistics*, vol. 33, no. 1, pp. 105–133, 2007.
- [Little, 2004] J. Little, M. de Ga, T. Ozyer y R. Alhaji, “Query Builder: A natural language interface for structured databases”, *Lecture Notes in Computer Science*, vol. 3280, pp. 470–479, 2004.

- [May, 2000] A. May, “Herramienta para consultas basadas en ejemplos (QBE) para multibases de datos en internet”, tesis de maestría, Centro Nacional de Investigación y Desarrollo Tecnológico, Cuernavaca, Mor., México, 2000.
- [Microsoft, 2016] Microsoft, “Ayuda y aprendizaje de Access”, 2016. [En línea]. Disponible en : <https://support.microsoft.com/es-mx/access>. Accedido: 26 feb. 2021.
- [Silberschatz, 2006] A. Silberschatz, H. Korth y S. Sudarshan, *Fundamentos de bases de datos*, 5.^a ed., Mc Graw Hill, 2006.