



TECNOLÓGICO NACIONAL DE MÉXICO  
Instituto Tecnológico de Ciudad Madero



# Instituto Tecnológico de Ciudad Madero

División de Estudios de Posgrado e Investigación

## Maestría en Ciencias de la Computación

### **Interfaz de composición de consultas a bases de datos que permita incluir subconsultas**

Para obtener el grado de:  
Maestro en Ciencias de la Computación

Presenta:  
ITI Germán Castillo Guillén

Director de tesis:  
Dr. Rodolfo Abraham Pazos Rangel

Cd. Madero, Tamaulipas, México, septiembre 2016



Cd. Madero, Tamps; a **29 de Septiembre de 2016.**

OFICIO No.: U5.185/16  
AREA: DIVISIÓN DE ESTUDIOS  
DE POSGRADO E INVESTIGACIÓN  
ASUNTO: AUTORIZACIÓN DE IMPRESIÓN DE TESIS

**ING. GERMÁN CASTILLO GUILLÉN**  
**NO. DE CONTROL G14073000**  
**PRESENTE**

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, el cual está integrado por los siguientes catedráticos:

PRESIDENTE :	DR. JOSÉ ANTONIO MARTÍNEZ FLORES
SECRETARIO :	DRA. LAURA CRUZ REYES
VOCAL :	DR. RODOLFO ABRAHAM PAZOS RANGEL
SUPLENTE	DR. JUAN JAVIER GONZALÉZ BARBOSA
DIRECTOR DE TESIS:	DR. RODOLFO ABRAHAM PAZOS RANGEL

Se acordó autorizar la impresión de su tesis titulada:

**"INTERFAZ DE COMPOSICIÓN DE CONSULTAS A BASES DE DATOS QUE PERMITA INCLUIR SUBCONSULTAS"**

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**  
"POR MI PATRIA Y POR MI BIEN"®

**DRA. ADRIANA ISABEL REYES DE LA TORRE**  
**JEFA DE LA DIVISIÓN DE ESTUDIOS**  
**DE POSGRADO E INVESTIGACIÓN**



**S.E.P.**  
DIVISION DE ESTUDIOS  
DE POSGRADO E  
INVESTIGACION  
I T C M

c.c.p.- Archivo  
Minuta

AIRTEL



Ave. 1° de Mayo y Sof Juana I. de la Cruz Col. Los Mángos, C.P. 89440 Cd. Madero, Tam.  
Tel. (833) 357 48 20. e-mail: itcm@itcm.edu.mx  
www.itcm.edu.mx



# Contenido

Capítulo 1 .....	1
Introducción .....	1
1.2 Planteamiento del problema .....	1
1.3 Objetivos .....	2
1.4 Justificación y beneficios .....	3
1.5 Alcances y limitaciones.....	4
Capítulo 2 .....	6
Marco conceptual y estado del arte.....	6
2.1 Marco conceptual .....	6
2.1.1 Lenguaje .....	6
2.1.3 Lenguaje natural .....	6
2.1.4 Composición .....	7
2.1.5 Base de datos.....	7
2.1.6 Sistema manejador de bases de datos (DBMS).....	7
2.1.7 SQL ( <i>Structured Query Language</i> ).....	8
2.1.8 Instrucción SELECT .....	8
2.1.9 Cláusula Select .....	8
2.1.10 Cláusula From .....	8
2.1.11 Cláusula Where.....	9
2.1.12 Subconsultas.....	9
2.2 Estado del arte.....	9
2.2.1 Microsoft Office Access (2013) .....	10
2.2.2 Structured Advanced Query Page – SAQP (2010).....	11
2.2.3 Blended Browsing and Querying - BBQ (1999) .....	13
2.2.4 Visual Composition of Complex Queries (2013) .....	15
2.2.5 Query Builder (2004) .....	17
2.2.6 Conclusiones .....	18
Capítulo 3 .....	20
Versión anterior de la interfaz.....	20
3.1 Módulo de composición de consultas para una interfaz de lenguaje natural a bases de datos .....	20
3.2 Arquitectura .....	20
3.3 Árbol de composición .....	21
Capítulo 4 .....	23

Metodología de solución.....	23
4.1 Arquitectura actual de la interfaz de composición .....	23
4.2 Consultas con niveles de subconsultas a tratar .....	25
4.3 Descomposición de consultas en LN.....	26
4.4 Árbol de composición .....	26
4.5 Generación automática de subconsultas intermedias.....	28
Capítulo 5 .....	32
Interfaz de composición.....	32
5.1 Conceptos para composición de consultas que involucran subconsultas .....	32
5.2 Descomposición de la consulta en LN.....	33
5.3 Selección del tema de interés.....	34
5.4 Especificación de condición de búsqueda .....	35
5.5 Resultado de subconsulta interna .....	37
5.6 Especificación de condición de búsqueda para el resultado de subconsultas .....	39
5.7 Selección de elementos de interés .....	40
5.8 Construcción en lenguaje SQL de consultas que involucran subconsultas .....	41
5.9 Vista previa y resultado.....	43
5.10 Proceso de composición de consultas que involucran subconsultas.....	44
Capítulo 6 .....	54
Pruebas experimentales .....	54
6.1 Pruebas de usabilidad.....	54
6.1.1 Objetivo de las pruebas .....	54
6.1.2 Descripción del ambiente de pruebas.....	54
6.1.3 Resultados de las pruebas de usabilidad.....	56
6.1.4 Comparación de resultados .....	58
6.1.5 Conclusiones de las pruebas de usabilidad.....	59
6.2 Pruebas funcionales .....	60
6.2.1 Descripción de pruebas funcionales.....	60
Capítulo 7 .....	63
Conclusiones y trabajos futuros.....	63
7.1 Conclusiones.....	63
7.2 Trabajos futuros .....	64
Apéndices .....	65
Apéndice 1. Descripción de la base de datos ATIS .....	65
Apéndice 2. Corpus de consultas de prueba de la base de datos ATIS.....	71

Apéndice 3. Descripción del diccionario de información semántica .....	72
Referencias.....	76

### Lista de figuras

Figura 2.1. Selección de tablas.....	10
Figura 2.2. Composición de consulta que involucra una subconsulta.....	11
Figura 2.3. Arquitectura de la interfaz SAQP.....	12
Figura 2.4. Interfaz de consulta SAQP.....	13
Figura 2.5. Interfaz de BBQ.....	14
Figura 2.6. Generación de árbol de respuesta(s) BBQ.....	15
Figura 2.7. Interfaz de composición de Pessina et al.....	17
Figura 2.8. Grafo de las tablas en el GPDW de la interfaz de Pessina et al.....	17
Figura 2.9. Grafo de asociación de tablas de la interfaz de Pessina et al.....	16
Figura 2.10. Interfaz de Query Builder (QB).....	18
Figura 3.1. Arquitectura de la versión anterior de la interfaz de composición.....	21
Figura 3.2. Árbol de composición.....	22
Figura 4.1. Nueva arquitectura de la interfaz de composición.....	24
Figura 4.2. Árbol de composición para subconsultas.....	27
Figura 5.1. Descomposición de consultas en LN.....	34
Figura 5.2. Selección del tema de interés.....	35
Figura 5.3. Especificación de condiciones de búsqueda.....	37
Figura 5.4. Operadores de búsqueda.....	37
Figura 5.5. Resultado de subconsulta interna.....	38
Figura 5.6. Especificación de condición de búsqueda para el resultado de subconsultas.....	39
Figura 5.7. Construcción de las cláusulas Select y From.....	40
Figura 5.8. Selección de elementos de interés.....	41
Figura 5.9. Construcción de las cláusulas Select y From.....	42
Figura 5.10. Construcción de la cláusula Where.....	42
Figura 5.11. Vista previa y resultado.....	44
Figura 5.12. Descomposición de consultas.....	45
Figura 5.13. Selección del tema de interés en la composición de la subconsulta más interna.....	46
Figura 5.14. Especificación de condiciones de búsqueda en la composición de la subconsulta más interna.....	47

Figura 5.15. Resultado y vista previa en la composición de la subconsulta más interna .....	48
Figura 5.16. Selección del tema de interés para la subconsulta intermedia.....	48
Figura 5.17. Resultado de subconsulta interna para la subconsulta intermedia.....	49
Figura 5.18. Especificación de condición de búsqueda para subconsultas (subconsulta intermedia).....	50
Figura 5.19. Vista previa y resultado de subconsulta intermedia.....	50
Figura 5.20. Selección del tema de interés para la consulta externa.....	51
Figura 5.21. Selección de elementos a mostrar para la consulta externa.....	52
Figura 5.22. Definición de condición de búsqueda para subconsulta (consulta externa).....	53
Figura 5.23. Vista previa y resultado de la consulta.....	53
Figura 6.1. Proceso de monitoreo para la composición de consultas.....	55
Figura 6.2. Promedio de intentos y tiempo en minutos para cada consulta.....	58
Figura 6.3. Resultado de pruebas realizadas con la versión anterior de la interfaz de composición [Aguirre 2015].....	59
Figura A1.1. Esquema de la base de datos ATIS.....	65
Figura A2.1. Esquema del diccionario de información semántica (DIS).....	75

### Lista de tablas

Tabla 2.1. Descripción del contenido de las interfaces descritas anteriormente.....	19
Tabla 4.1. Tipos de consultas a tratar.....	25
Tabla 4.2. Descomposición de la consulta de ejemplo.....	28
Tabla 4.3 Matriz de adyacencia de la BD ATIS.....	29
Tabla 4.4 Identificación numérica de las tablas de la BD ATIS.....	30
Tabla 5.1 Conceptos básicos para la composición de consultas que involucran subconsultas.....	32
Tabla 5.2. Componentes de la ventana <i>Descomposición de consultas en LN</i> .....	33
Tabla 5.3. Componentes de la ventana <i>Selección del tema de interés</i> .....	34,35
Tabla 5.4. Componentes de la ventana <i>Especificación de condiciones de búsqueda</i> .....	36
Tabla 5.5. Componentes de la ventana <i>Resultado de subconsulta interna</i> .....	38
Tabla 5.6. Componentes de la ventana <i>Especificación de condición de búsqueda para el resultado de subconsultas</i> .....	39
Tabla 5.7. Componentes de la ventana <i>Selección de elementos de interés</i> .....	40,41
Tabla 5.8. Componentes de la ventana <i>Vista previa y resultado</i> .....	43
Tabla 6.1. Resultado de la composición de consultas.....	56
Tabla A2.1. Corpus de consultas para la BD ATIS.....	68

# Capítulo 1

## Introducción

---

Las bases de datos (BDs) están presentes en todas partes. Muchas aplicaciones acceden a ellas en todo momento, y en muchos casos la información obtenida a partir de ellas es utilizada para tomar decisiones importantes. La obtención de información inusual o compleja de una base de datos requiere que los usuarios tengan conocimiento de un lenguaje de consulta como SQL. Desafortunadamente, la mayoría de los usuarios no cuentan con el conocimiento necesario para la formulación de consultas que les permita obtener dicha información.

Para facilitar el acceso a la información contenida en las bases de datos, se ideó una manera en la que se puede acceder a ella por usuarios inexpertos sin la necesidad de alguna capacitación sobre un lenguaje de consulta a bases de datos. Consecuentemente, surgió la idea de utilizar el lenguaje natural como medio para consultar bases de datos. Sin embargo, al realizar consultas por medio de lenguaje natural (LN), pueden surgir diferentes problemas, tales como falta de información sobre tablas o columnas, problemas de interpretación de conceptos introducidos por el usuario, mala interpretación de una consulta planteada por el usuario debido a que se encuentre mal compuesta, entre otros.

Actualmente existen herramientas de composición de consultas que pueden ser utilizadas por usuarios inexpertos en lenguajes de consulta como SQL (*Structured Query Language*). Este tipo de herramientas aún no cuentan con la capacidad de responder algunas de las consultas que los usuarios inexpertos desearían realizar.

El objetivo de este trabajo se enfoca en la introducción de una nueva funcionalidad a una interfaz de composición de consultas ya existente que ayude a usuarios inexpertos a componer fácilmente consultas en SQL que involucren subconsultas correctas sin la necesidad de tener conocimientos sobre el mismo. Dicha composición deberá ser fácil de realizar para los usuarios inexpertos.

### 1.2 Planteamiento del problema

En la interfaz de composición desarrollada en [Aguirre 2015] los usuarios realizan solicitudes a la base de datos mediante la composición de consultas en lenguaje natural. Para realizar dicha solicitud el usuario debe tener formulada mentalmente una consulta con anterioridad. Una de las necesidades de los usuarios inexpertos es obtener información que se encuentra almacenada en diferentes tablas de la base de datos, y que no es posible obtener mediante la composición de una consulta en lenguaje natural a SQL que involucre la operación de reunión (*join*). Es por ello que necesitarían realizar una consulta que involucre múltiples consultas; por ejemplo, la siguiente consulta sobre la base de datos Geobase.

*¿Cuáles son los estados que colindan con los estados por los que corre el río Colorado?*

La base de datos Geobase cuenta con información sobre la geografía de los Estados Unidos de América. La información contenida en dicha BD es la siguiente:

- Estados: nombres de estados, capitales, poblaciones, áreas, ríos, lagos, montañas y estados colindantes.
- Ciudades: nombres de ciudades, poblaciones y los estados en donde se encuentran.
- Ríos: nombres de ríos, longitudes y los estados por los que atraviesan los ríos.
- Lagos: nombres de lagos, áreas y estados donde se encuentran los lagos.
- Montañas: nombres de montañas, alturas y los estados en donde se encuentran.

En base a la información descrita en la BD Geobase, para llevar a cabo la consulta anterior correctamente, es necesario que el usuario tenga que dividir esta consulta de tal manera que primero necesita conocer los estados por los cuales corre el río Colorado y con ello determinar cuáles son los estados que colindan con los estados del resultado de la consulta anterior.

Otro ejemplo de consultas que involucran subconsultas, es la siguiente:

*¿Cuál es la capital del estado con el punto más alto?*

Para este ejemplo, la consulta involucra subconsultas, y además, la palabra “más” que se encuentra en la consulta hace referencia al uso de una función de agregación (MAX), la cual sirve para obtener el valor más grande. Dicho lo anterior, es importante aclarar que en este trabajo de tesis no se consideran las consultas que involucran subconsultas y funciones de agregación, solamente se ha implementado la funcionalidad de consultas que involucran subconsultas sin funciones de agregación con el fin de preparar a dicha interfaz para que en un futuro pueda responder consultas como la del último ejemplo.

### 1.3 Objetivos

Para resolver el problema presentado anteriormente, se plantea el siguiente objetivo general (OG):

- OG.1 Diseñar e implementar una nueva funcionalidad a la interfaz de composición de consultas desarrollada en [Aguirre 2015] de tal manera que esta nueva funcionalidad siga permitiendo la independencia del dominio en la interfaz, y que además, ayude a usuarios inexpertos a componer fácilmente consultas en SQL que involucren subconsultas correctamente para proporcionar al usuario los resultados esperados.

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

- OE.1. La nueva funcionalidad debe facilitar a los usuarios la composición de consultas que involucren subconsultas a una base de datos con independencia de dominio.



- OE.2. La nueva funcionalidad debe permitir la composición de consultas que involucren subconsultas de manera no restringida, de tal manera que no dependa de plantillas de consulta como las generadas por algunas otras interfaces.
- OE.3. El proceso de composición de consultas que involucren subconsultas debe obtener como resultado una consulta estructurada correctamente en SQL.
- OE.4. Las subconsultas compuestas por el usuario mediante el módulo de composición de consultas deberán ser coherentes con la estructura de la BD.

## 1.4 Justificación y beneficios

La gran cantidad de información disponible en bases de datos está en constante crecimiento; por lo tanto, existe la necesidad de crear sistemas de consulta que sean eficaces. Para operar estos sistemas es necesario que los usuarios tengan conocimientos sobre algún lenguaje de consulta como SQL (*Structured Query Language*), quienes comúnmente suelen ser expertos en computación. Desafortunadamente, la mayoría de los usuarios no cuentan con este conocimiento, y es necesario proporcionarles algún medio que les facilite obtener la información deseada mediante consultas a bases de datos, ya que no se espera que aprendan a utilizar estos lenguajes de consulta.

En la versión anterior de la interfaz de composición de consultas, no era posible responder muchas de las consultas que desearía realizar un usuario inexperto. Por lo tanto, existía la necesidad de ampliar la funcionalidad de consultas de dicha interfaz para incrementar los tipos de consultas respondidas correctamente. Dicha ampliación tiene como fin realizar consultas que involucren subconsultas para obtener cierta información tales como las que se describen enseguida.

### **Ejemplo de consulta 1:**

Considérese la siguiente consulta: *¿cuáles son las poblaciones de los estados a través de los cuales corre el río Mississippi?*

Para obtener la información solicitada, es necesario ejecutar una consulta formulada en SQL que involucra subconsultas de la siguiente manera:

```
SELECT state_name, population
FROM state
WHERE state.abbreviation IN
  (SELECT riverstate.state_abbreviation
   FROM riverstate
   WHERE riverstate.river_id =
     (SELECT river.river_id
      FROM river
      WHERE river.river_name = "Mississippi")).
```

## Ejemplo de consulta 2:

También existen consultas con subconsultas que involucran funciones de agregación como por ejemplo: *¿cuál es la capital del estado con la mayor población?*, la cual se formula en SQL de la siguiente manera:

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

Estas consultas han sido obtenidas del corpus de C-Phrase para Geobase. Además, es necesario aclarar que este último tipo de consultas que involucran funciones de agregación no son tomadas en cuenta en este proyecto de tesis, tal como se mencionó en la sección 1.2.

Como se mencionó anteriormente, la funcionalidad que ha sido adicionada a la interfaz de composición de consultas facilitará a los usuarios inexpertos realizar dicha tarea, en la cual se centra el presente trabajo de tesis.

Los beneficios principales del presente trabajo son los siguientes:

- Permitir a los usuarios inexpertos obtener información que involucre varias tablas mediante la funcionalidad de subconsultas.
- Permitir a los usuarios inexpertos componer correctamente consultas que involucran subconsultas mediante la interfaz de composición.
- Asegurar que la traducción a SQL de las consultas compuestas por el usuario mediante la interfaz de composición, puedan ser procesadas correctamente por un manejador de bases de datos.

## 1.5 Alcances y limitaciones

Se definen a continuación los alcances de este proyecto:

1. La nueva funcionalidad de subconsultas implementada en la interfaz de composición debe ser independiente de dominio.
2. La composición de consultas que involucran subconsultas debe tener coherencia con la estructura de la base de datos.
3. La interfaz de composición con la nueva funcionalidad de subconsultas debe ser fácil de utilizar por los usuarios obteniendo los resultados esperados.
4. Esta funcionalidad de subconsultas debe ser capaz de componer consultas con hasta tres subconsultas internas.
5. La funcionalidad de subconsultas debe permitir al usuario especificar más condiciones de búsqueda dentro y fuera de las subconsultas.

Las limitaciones de este proyecto se definen a continuación:

1. En la funcionalidad de subconsultas no se considera la composición de consultas que involucren funciones de agregación y/o agrupación.
2. La interfaz de composición de consultas únicamente se configuró en idioma español para las pruebas funcionales y de usabilidad, aunque tenga la posibilidad de ser configurada para otros idiomas.
3. No se proporciona información que no se encuentre en la base de datos para la composición de consultas que involucren subconsultas; es decir, no se tratan consultas de bases de datos deductivas.
4. No se compondrán consultas que involucren subconsultas de bases de datos temporales.

# Capítulo 2

## Marco conceptual y estado del arte

---

Como se ha mencionado en el Capítulo 1, la información almacenada en las bases de datos ha sido fundamental para la creación de distintas aplicaciones informáticas, entre ellas se encuentran las interfaces para consulta a bases de datos. Estas herramientas utilizan distintos métodos de composición para obtener al final una consulta en algún lenguaje de consulta como SQL y con ello obtener de la BD la información especificada por los usuarios.

A continuación se presentan algunos métodos de composición utilizados por algunas interfaces de consulta:

- Método QBE (Query by Example) para la composición de consultas.
- Método de listas desplegables para elegir los elementos que conformarán la composición de una consulta.
- Método *drag and drop* de elementos del esquema de la base de datos.
- Método mediante menús para realizar la composición de consultas.
- Método utilizando plantillas preestablecidas para realizar la composición de consultas.

### 2.1 Marco conceptual

Para tener una mejor comprensión del trabajo que se ha realizado en esta tesis y dentro del estado del arte, es necesario definir alguna terminología y conceptos básicos.

#### 2.1.1 Lenguaje

En base a la Real Academia Española, lenguaje es un conjunto de sonidos articulados o símbolos con que el hombre manifiesta lo que piensa o siente.

#### 2.1.2 Lenguaje formal

El lenguaje formal consiste en un conjunto de signos vacíos de contenido que se relacionan entre sí por una serie de normas y reglas. El vocabulario es mucho más reducido que en el lenguaje natural, lo cual es necesario para que sea más efectivo y operativo. El lenguaje formal contiene, entre otros, como elementos básicos, un vocabulario primitivo y unos operadores [Pechuan 2012].

#### 2.1.3 Lenguaje natural

Lenguaje utilizado normalmente por los miembros de una comunidad. El lenguaje natural se caracteriza por su enorme capacidad de riqueza comunicativa. Este lenguaje es capaz de expresar gran variedad de sentimientos, situaciones, deseos, etc. [Pechuan 2012].

## 2.1.4 Composición

En base a la Real Academia Española, componer se define como: formar de varias cosas una, juntándolas y colocándolas con cierto modo y orden.

## 2.1.5 Base de datos

Conjunto estructurado de datos que representan entidades y sus relaciones [Camps 2005].

- Una base de datos representa algún aspecto del mundo real, lo que en ocasiones se denomina minimundo o universo de discurso (UoD, *universe of discourse*). Los cambios introducidos en el minimundo se reflejan en la base de datos.
- Una base de datos es una colección de datos lógicamente coherente con algún tipo de significado inherente. No es correcto denominar base de datos a un conjunto aleatorio de datos.

Una base de datos se diseña, construye y puebla con datos para un propósito específico. Dispone de un grupo de usuarios y algunas aplicaciones preconcebidas en las que esos usuarios están interesados.

## 2.1.6 Sistema manejador de bases de datos (DBMS)

Un sistema manejador de bases de datos (DBMS) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos. La colección de datos, normalmente denominada base de datos, contiene información relevante para una empresa. El objetivo principal de un DBMS es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente [Silberschatz 2002]. Las principales funciones del DBMS son las siguientes:

- Facilitar al administrador de la base de datos la descripción de los elementos de datos que integran la base de datos, su estructura y las relaciones que existen entre ellos, así como los controles a efectuar antes de realizar el acceso a la base de datos.
- Facilitar la manipulación de los datos, permitiendo a los usuarios de la base de datos añadir, buscar, suprimir o modificar los datos de la misma. Se lleva a cabo por medio de lenguajes de manipulación de datos (LMDs).
- Garantizar la seguridad y confidencialidad de la base de datos. Por ejemplo: controlando el acceso de los usuarios a la base de datos mediante contraseñas y limitando el acceso a los datos mediante un sistema de derechos o privilegios.
- Garantizar la fiabilidad e integridad de los datos. Por ejemplo: no permitir la introducción de fechas erróneas, valores no permitidos, valores incoherentes, relaciones incongruentes, etc.
- Garantizar la compartibilidad de los datos. Permitir que varios usuarios puedan utilizar los mismos datos sin que se produzcan problemas en los accesos simultáneos.
- Facilitar estadísticas sobre la marcha de la explotación de la base de datos, mediante las cuales se pueda observar su desempeño.

### 2.1.7 SQL (*Structured Query Language*)

El SQL es el lenguaje estándar ANSI/ISO de definición, manipulación y control de bases de datos relacionales. Es un lenguaje declarativo: sólo hay que indicar qué se quiere hacer. En cambio, en los lenguajes procedimentales es necesario especificar cómo hay que hacer cualquier acción sobre la base de datos [Camps 2005]. El SQL es un lenguaje muy 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 sistemas de bases de datos relacionales comerciales.

### 2.1.8 Instrucción SELECT

Una instrucción SELECT contiene una descripción completa del conjunto de datos que se desea obtener de una base de datos. Principalmente incluye lo siguiente:

- Qué tablas contienen la información.
- Cómo se relacionan los datos de orígenes diferentes.
- Qué campos o cálculos proporcionarán los datos. Condiciones que los datos deben cumplir para ser incluidos.

Estructura básica de una instrucción SELECT:

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

### 2.1.9 Cláusula Select

La cláusula Select corresponde a la operación proyección del álgebra relacional. Se usa para listar las columnas deseadas del resultado de una consulta.

El resultado de una consulta en SQL es, por supuesto, una tabla. Los lenguajes formales de consulta están basados en la noción matemática de que una relación (correspondiente a una tabla) es un conjunto. SQL (como la mayoría de los lenguajes de consulta comerciales) permite renglones duplicados en las tablas, así como en el resultado de las expresiones en SQL.

### 2.1.10 Cláusula From

La cláusula From corresponde a la operación producto cartesiano del álgebra relacional. Lista las tablas que son involucradas en la evaluación de la expresión.

### 2.1.11 Cláusula Where

La cláusula Where corresponde al predicado selección del álgebra relacional. Es un predicado que involucra a las columnas de las tablas que aparecen en la cláusula From.

SQL usa los conectivos lógicos AND, OR y NOT (en lugar de los símbolos matemáticos  $\wedge$ ,  $\vee$  y  $\neg$ ) en la cláusula Where. Los operandos de los conectivos lógicos pueden ser expresiones que contengan los operadores de comparación  $<$ ,  $<=$ ,  $>$ ,  $>=$ ,  $=$  y  $<>$ .

SQL permite usar los operadores de comparación para comparar cadenas y expresiones aritméticas, así como tipos especiales, tales como el tipo fecha. SQL incluye un operador de comparación BETWEEN para simplificar las cláusulas Where, el cual especifica que un valor sea menor o igual que un valor y mayor o igual que otro valor.

### 2.1.12 Subconsultas

Una subconsulta es una expresión SELECT FROM WHERE que se anida dentro de otra consulta por ejemplo:

```
SELECT columna_1
FROM table_1
WHERE columna_2 IN
      (SELECT columna_3
       FROM table_2
       WHERE condición_1)
```

- Un uso común de las subconsultas es llevar a cabo comprobaciones sobre pertenencia a conjuntos, comparación de conjuntos y 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.

Normalmente las subconsultas se utilizan para satisfacer una o un par de las siguientes necesidades:

- Desglosar una consulta en una serie de pasos lógicos.
- Proporcionar un listado que va a ser el destino de una cláusula Where con [IN | EXISTS | ANY | ALL].
- Proporcionar una búsqueda dirigida por cada registro individual de una consulta principal.

## 2.2 Estado del arte

A continuación se describen algunas herramientas que son utilizadas para la composición de consultas a bases de datos.

## 2.2.1 Microsoft Office Access (2013)

Este software es el programa de bases de datos de Windows más popular. Una de las principales razones de su éxito es su interfaz de consulta revolucionaria. Las consultas ofrecen la posibilidad de recuperar y filtrar los datos, actualizar, mover y eliminar los registros de forma masiva.

Microsoft Access cuenta con una característica muy importante que es la independencia de dominio. La representación visual de las tablas y los enlaces gráficos entre ellas hace que las consultas sean extremadamente fáciles de formular.

El motor de consultas se basa en el lenguaje de consulta SQL y permite cambiar entre el diseño gráfico de consultas y la sintaxis de SQL.

Para realizar las consultas en modo gráfico en Microsoft Access, primeramente es necesario seleccionar las tablas que serán utilizadas para generar una consulta, tal como se muestra en la Figura 2.1.

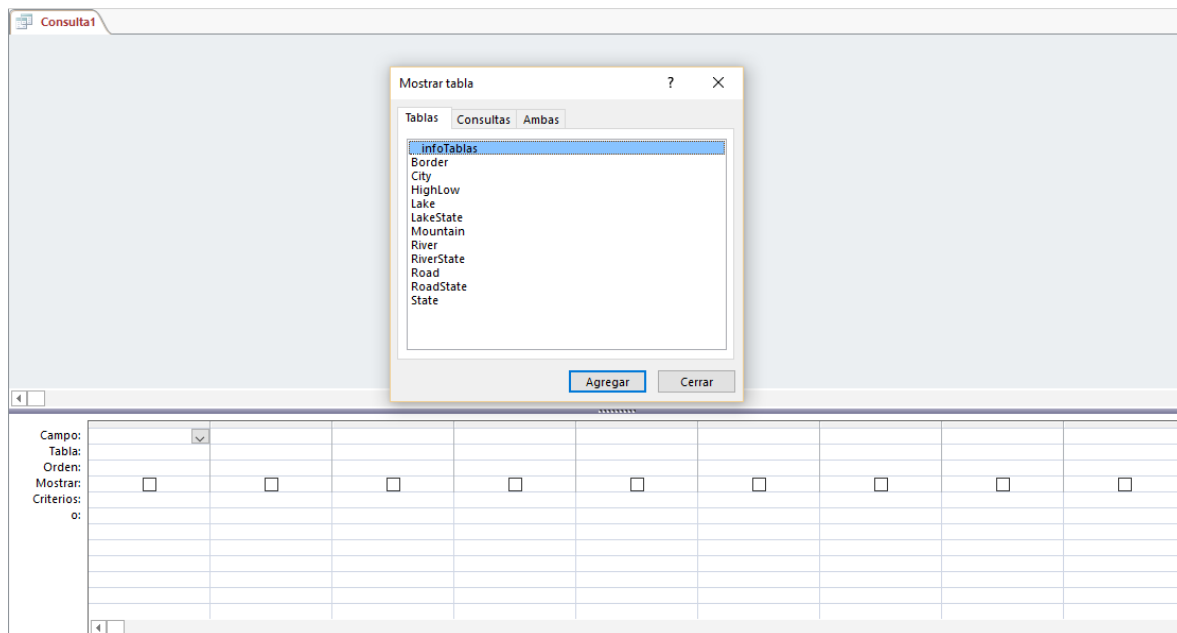


Figura 2.1. Selección de tablas

Posteriormente se selecciona el o los campos de los cuales el usuario desea obtener información y con ello componer la cláusula Select. Además, debe indicar en cada columna que se haya seleccionado un campo, su correspondiente tabla a la cual pertenece dicho campo seleccionado, y por último definir los criterios de búsqueda de los campos seleccionados para la cláusula Where, como se muestra en la Figura 2.2.

A pesar de que esta interfaz gráfica es muy poderosa, desafortunadamente al momento en que un usuario inexperto requiere componer una consulta que involucra subconsultas, éste necesita tener conocimientos sobre un lenguaje de consulta como SQL para poder especificar la subconsulta como criterio de búsqueda, tal como se muestra en la Figura 2.2.



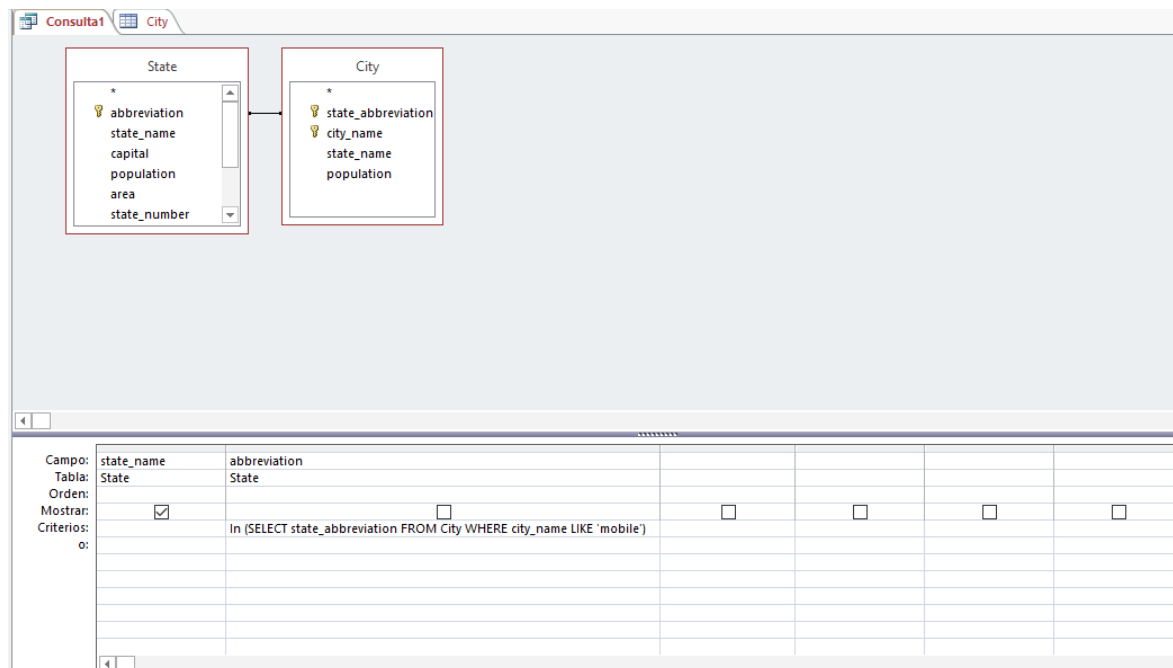


Figura 2.1. Composición de consulta que involucra una subconsulta

## 2.2.2 Structured Advanced Query Page – SAQP (2010)

La interfaz SAQP [Latendresse 2010] actualmente utiliza bases de datos de dominio biológico, pero SAQP puede utilizar cualquier base de datos construida en Ocelot. SAQP es suficientemente amigable para ser utilizada por usuarios biólogos que no necesitan tener conocimientos sobre un lenguaje de consulta formal; además, SAQP está basada en un lenguaje de consulta de BD llamado BioVelo, el cual es más expresivo que SQL, y tiene una semántica más simple.

En esta interfaz se hace uso de Pathway/Genome DBs (PGDBs). Las herramientas Pathway PGDBs se gestionan mediante un sistema de representación denominado Ocelot. Éste es un sistema orientado a objetos que gestiona bases de datos basado en Common Lisp y que utiliza un SMD relacional. En la Figura 2.3 se muestra la arquitectura general de la interfaz SAQP.

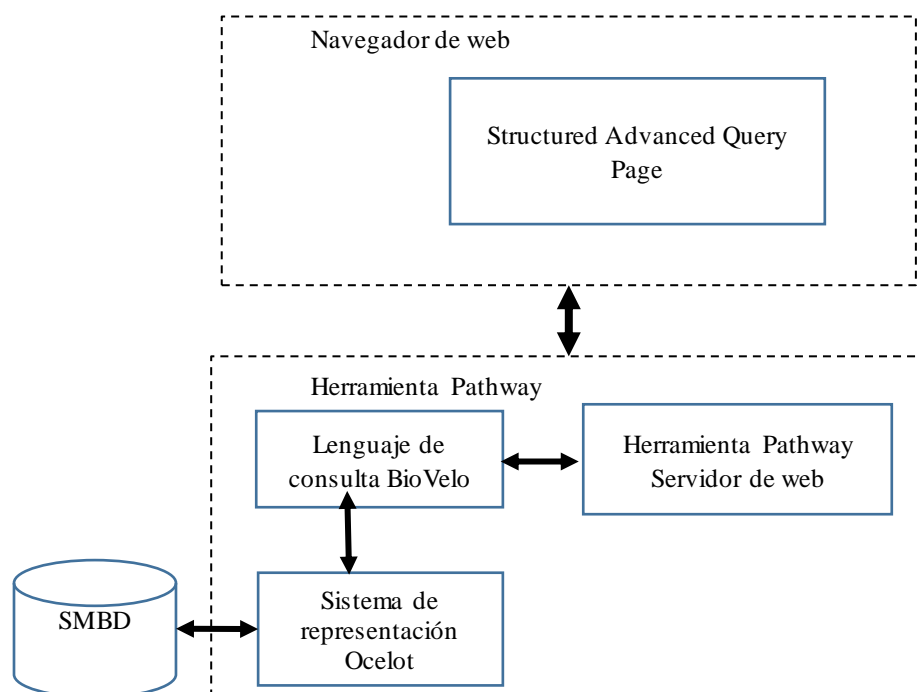


Figura 2.3. Arquitectura de la interfaz SAQP

Ya que esta interfaz puede utilizar más de una base de datos para realizar las consultas, el primer paso para construir una consulta es especificar al menos una base de datos y con ello los componentes de búsqueda (ver Figura 2.4).

El segundo paso consiste en especificar las condiciones de búsqueda, las cuales pueden ser una o varias condiciones. Esta operación añade un selector para una columna de BD (por ejemplo, nombre) de los objetos y un selector para un operador relacional (por ejemplo, contiene la subcadena). También agrega un cuadro de texto libre para introducir un número o una cadena. También otros operadores relacionales se proporcionan tales como *es igual a*, *no es igual* y *es una subcadena de*.

En el tercer paso se definen los resultados de la consulta. La sección titulada *Select attributes to include in the query output* permite al usuario seleccionar el contenido del resultado de la consulta, seleccionando los atributos (*attributes*) a mostrar para cada objeto del resultado. El selector proporcionado en cada columna contiene la lista de columnas de la BD accesibles para la clase de objeto seleccionado para una consulta específica (ver Figura 2.4 relativa a los tres pasos mencionados anteriormente).

## STRUCTURED ADVANCED QUERY PAGE

[Switch to the Free Form Advanced Query Page](#)  
(Advanced Query Documentation | SAQP Webinars)

### 1. Select your database, class of objects, and conditions to search:

Query database  for  (let's call them Z1)

Select an operation to add an additional search component or variable:

for each previous result, query database  for  (let's call them Z2)

Where

contains the substring

Select an operation to add an additional search component or variable:

### 2. Select attributes to include in the query output:

Column 1

Sort based on this column

### 3. Select query output format:

HTML  Tab Delimited Text (columns are separated by tabs)

Figura 2.4. Interfaz de consulta SAQP

Cuando sólo un componente de búsqueda se ha especificado en la consulta y no se utilizan subconsultas con un cuantificador, que es el caso de la Figura 2.4, sólo un tipo de objeto es accesible.

Cuando se especifican más de un componente de búsqueda o una subconsulta se utiliza con un cuantificador, se proporciona un selector de variables para seleccionar la variable deseada. La interfaz proporciona el número de posibles objetos que tienen al menos un valor para cada columna de la BD.

A pesar que SAQP permite componer consultas que involucran subconsultas, esta interfaz tiene la desventaja de que el tipo de dato que regresa una subconsulta podría no ser compatible con la restricción definida en la cláusula Where de la consulta más externa, lo cual provocaría un error durante la ejecución.

## 2.2.3 Blended Browsing and Querying - BBQ (1999)

Esta interfaz gráfica descrita en [Munroe 1999] permite formular consultas de fuentes de datos en formato XML. BBQ muestra una estructura de múltiples fuentes de datos usando el paradigma que se asemeja a la exploración de archivos de Windows. Además, esta interfaz utiliza una DTD (*document type definition*) para representar una base de datos, la cual se basa en un documento donde se define la estructura de XML.

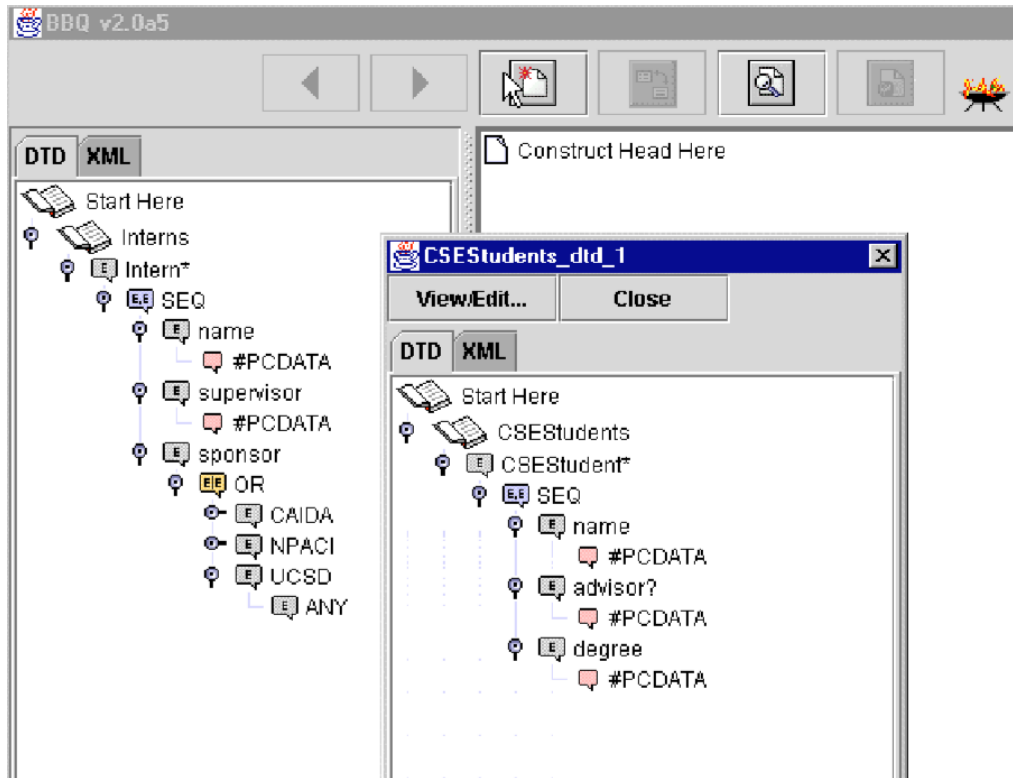


Figura 2.5. Interfaz de BBQ

En la Figura 2.5 se muestra la interfaz principal de BBQ, la cual consiste en una ventana principal y ventanas flotantes. La ventana principal contiene una barra de herramientas, un panel de división, y una consola de mensajes, mientras que las ventanas flotantes contienen una barra de herramientas y un panel de división. Los DTDs son exportados por el MIX (Mediation of Information using XML).

En BBQ una consulta es el conjunto de eventos que ocurren mientras está conectada con el MIX. Cada consulta se compone de uno o más ciclos de consulta. Un ciclo de consulta es el conjunto de eventos que comienza construyendo una consulta, y termina cuando el usuario navega por el resultado de la consulta.

El ciclo de consulta BBQ básica se lleva a cabo en cuatro pasos: 1) las restricciones se establecen en las fuentes de datos, 2) un árbol que representa el esquema resultado de la consulta se crea arrastrando y soltando elementos, 3) se genera la consulta presentada al mediador, y 4) se ejecuta la consulta.

Para definir las restricciones de una consulta en BBQ, el usuario hace clic en el botón View/Edit Constraint... en el menú emergente. Esta acción muestra la ventana View/Edit Constraint donde se pueden colocar restricciones de operadores de comparación (=, <, >, >=, <=, *substr*). En BBQ es posible realizar la reunión (*join*) de información seleccionando un elemento del árbol, y arrastrándolo y soltándolo en otro elemento del árbol de la consulta.

Una vez que las restricciones se han establecido, el siguiente paso en el ciclo de consulta es la construcción de un árbol de documento de respuesta(s) donde se debe construir la cabecera, la cual se compone de elementos y sus subárboles arrastrados de las DTDs de origen. La Figura 2.6 muestra la construcción de la consulta generando un árbol de respuesta, en donde también se puede indicar la cláusula GROUP BY para definir el elemento en base al cual se desea ordenar el resultado.

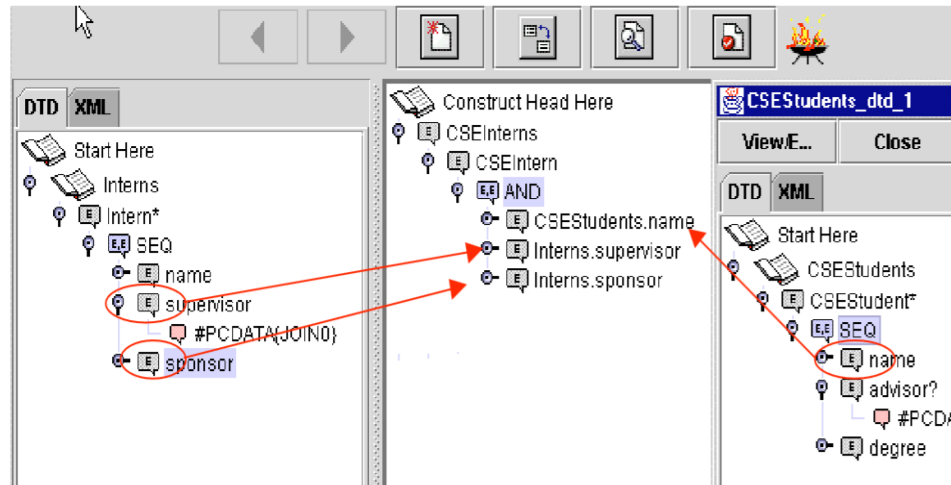


Figura 2.6. Generación de árbol de respuesta(s) BBQ

Una vez que el usuario ha generado visualmente la consulta como se muestra en la Figura 2.6, el usuario pasa a ejecutar dicha consulta. Entonces, BBQ convierte el diseño visual de la consulta a lenguaje de consulta mediante MIX.

## 2.2.4 Visual Composition of Complex Queries (2013)

En esta interfaz de web, descrita en [Pessina 2013], se creó un almacén de datos (data warehouse) de genómica y proteómica (GPDW) que integra la información proporcionada por algunas de las principales bases de datos de la bioinformática. A través de esta interfaz el usuario sólo necesita seleccionar los tipos de datos que se incluirán en la consulta y las condiciones en sus valores a búsqueda. La interfaz aprovecha los metadatos y el esquema del GPDW para componer automáticamente una consulta en SQL eficiente, ejecutarla en el GPDW y mostrar la información solicitada.

Actualmente la interfaz de web no cuenta con independencia de dominio y está basada en un almacén de datos de genómica y proteómica creado mediante la integración de datos y referencias externas de varias bases de datos incluyendo: Entrez Gene, GO, GOA, ByoCyc entre otras.

Esta interfaz permite a cualquier usuario componer fácilmente consultas de todos los datos almacenados en el GPDW. Esta interfaz fue desarrollada utilizando *servlets* y la tecnología Java Server Pages (JSP). A través de esta interfaz (ver Figura 2.7), el usuario sólo debe elegir, de las características integradas en el GPDW, aquéllas a ser incluidas en la consulta, junto con las condiciones sobre los valores de los datos que serán obtenidos.

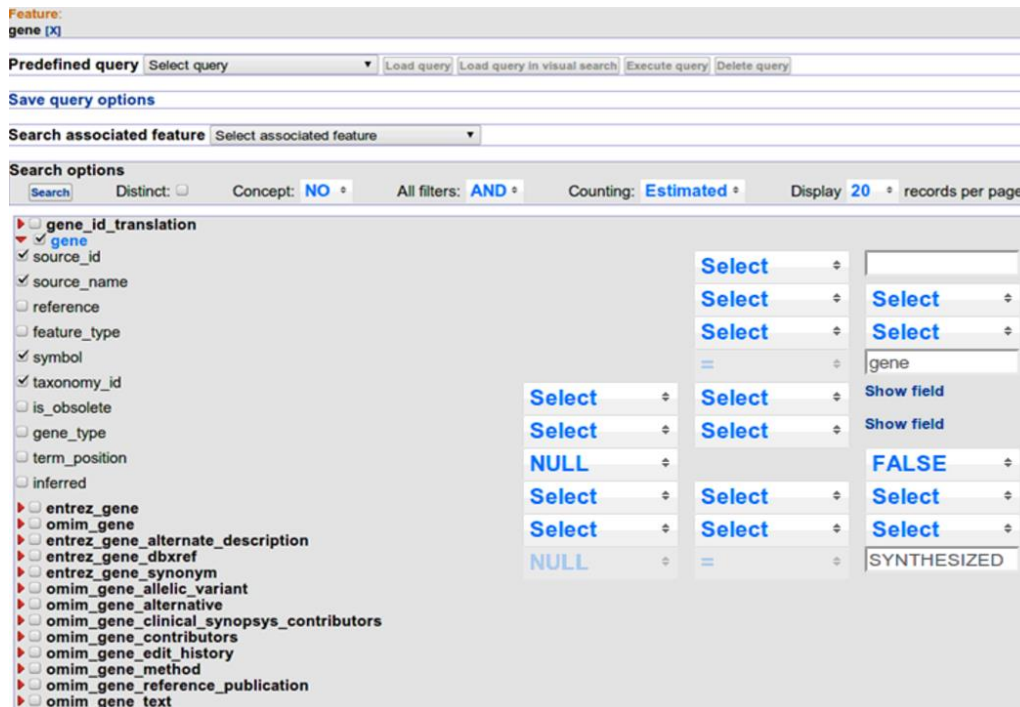


Figura 2.7. Interfaz de composición de Pessina et al.

La información mostrada en la interfaz para la composición de las consultas, es tomada de los metadatos del GPDW. Por lo tanto, de forma transparente para el usuario, las características visualizadas y sus columnas de BD se adaptan automáticamente al contenido de la información específica del GPDW. Los menús interactivos, presentes en la interfaz visual para cada columna, permiten al usuario definir las condiciones de búsqueda en los valores de los datos que desea obtener.

El contenido de cada uno de estos menús depende de los datos contenidos en la columna particular a la que se refiere el menú. Además, el usuario también puede seleccionar algunas opciones generales (opciones de búsqueda) para la consulta, por ejemplo, establecer algunas de las cláusulas lógicas (Y / O), y definir cuántos resultados mostrar.

El núcleo del algoritmo de composición de consultas aprovecha la estructura de metadatos del GPDW. Se divide en dos etapas: 1) generación de una lista de tablas que se incluirán en la consulta y 2) la generación de las cláusulas From y Where y la sentencia Join.

La primera etapa comienza con la identificación del último registro entre todas las tablas que contienen las columnas seleccionadas por el usuario para ser incluidas en la consulta. Esto se realiza de acuerdo con el grafo acíclico dirigido en las Figuras 2.8 y 2.9. Se completa la lista con todas las tablas involucradas en la consulta, también incluye las tablas que contienen las columnas seleccionadas por el usuario. La lista de tablas se ordena en base al módulo de asociación y en el orden en que el usuario seleccionó las características de la consulta.

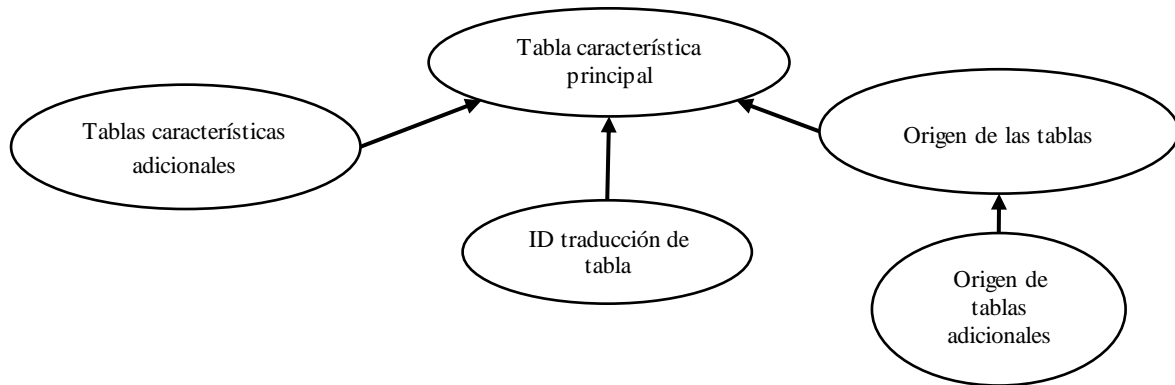


Figura 2.8. Grafo de las tablas en el GPDW de la interfaz de Pessina et al.

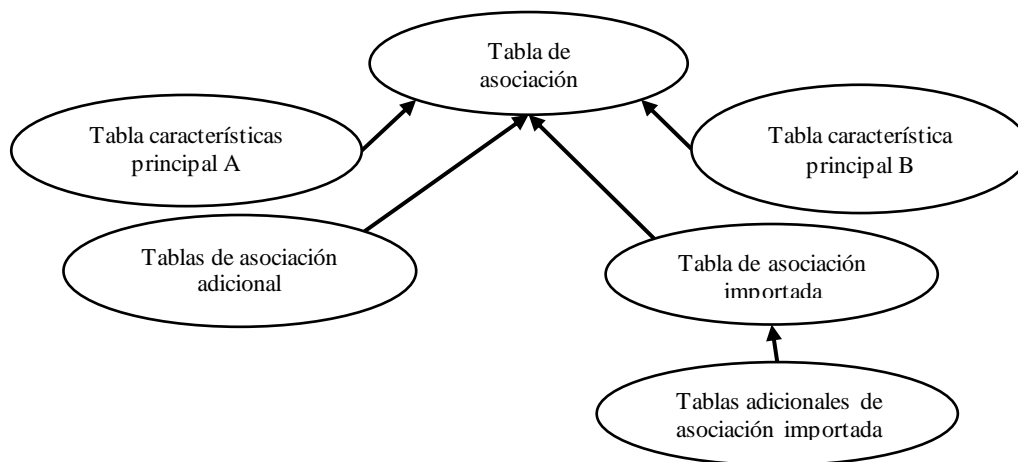


Figura 2.9. Grafo de asociación de tablas de la interfaz de Pessina et al.

En el segundo paso, la consulta de reunión (*join*) se efectúa de la siguiente manera:

La primera tabla de la lista de tablas se inserta en la cláusula *From* de la consulta, y se establecen las siguientes tablas de la lista como tablas de referencia para la operación de reunión (*join*).

Después que cada tabla de la lista es evaluada y durante esta evaluación, cuando se encuentra una tabla de características o asociaciones, se convierte en referencia para las operaciones de reunión (*join*) con las siguientes tablas de la lista. Si se trata de una tabla de características, no se añade la cláusula de reunión (*join*) a la consulta y si se trata de una tabla de asociación, se realiza la reunión entre esta tabla de asociación y cada una de las tablas de características (si están presentes en la lista) se añaden a la consulta. Por último, se añaden todos los filtros definidos por el usuario a la cláusula *Where* de la consulta.

## 2.2.5 Query Builder (2004)

La interfaz QB cuenta con la particularidad de ser independiente de dominio. Puede conectarse con diferentes manejadores de bases de datos como MySQL y Oracle entre otros. La manera de componer las consultas puede ser mediante lenguaje natural (LN) o mediante menús tal como se

muestra en la Figura 2.10, por lo que esta interfaz está dirigida especialmente para los usuarios que no cuentan con el conocimiento de algún lenguaje de consulta como SQL.

QB puede manejar consultas basadas en números que involucran operadores de comparación en bases de datos. Además, también puede manejar consultas con funciones de agregación.

QB cuenta con una característica importante que ha sido diseñada para dar una retroalimentación sobre el tipo de error en una consulta antes de que alcance la etapa de construcción en SQL. Para este caso, QB es capaz de sugerir alternativas para guiar a los usuarios en la construcción de la consulta.

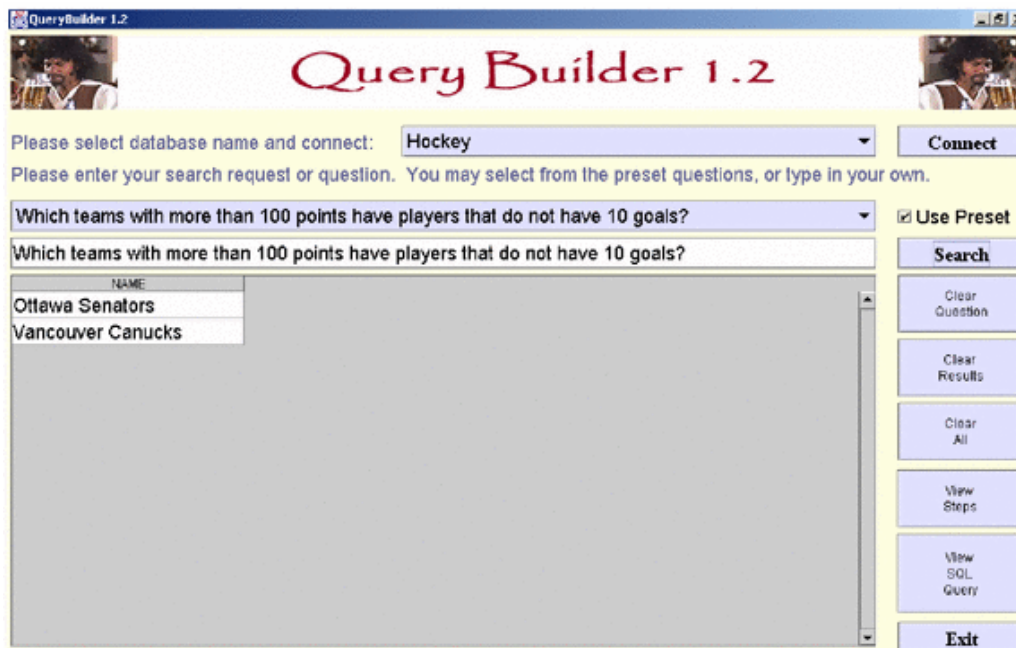


Figura 2.10. Interfaz de Query Builder (QB)

## 2.2.6 Conclusiones

Las interfaces presentadas en este capítulo muestran distintos métodos para realizar la composición de consultas. Un aspecto importante es que han sido diseñadas para poder ser utilizadas por usuarios inexpertos: aquéllos que no cuentan con algún conocimiento de un lenguaje de consulta como SQL.

Con dichas interfaces no es posible llevar a cabo la composición de consultas que involucran subconsultas a excepción de dos interfaces. La primera interfaz SAQP [Latendresse 2010] permite definir subconsultas, pero con el riesgo de obtener un error como resultado de una consulta mal compuesta por parte del usuario, tal como se describe en la subsección 2.2.2. La segunda interfaz que permite la composición de consultas que involucran subconsultas es Microsoft Office Access, pero el problema en esta interfaz gráfica, es que al momento de definir una subconsulta como condición de búsqueda, el usuario la tiene que formularla en el lenguaje de consulta SQL.



Tabla 2.1 muestra las principales características que se encuentran poseen las interfaces descritas anteriormente, así como las características de la interfaz de composición de este trabajo de tesis.

Tabla 2.1. Descripción del contenido de las interfaces descritas anteriormente

<b>Interfaz</b>	<b>Independencia de dominio</b>	<b>Método de composición</b>	<b>Manejo de subconsultas</b>	<b>Dominio de lenguaje de consulta</b>
Microsoft Office Access (2013)	Sí	QBE	Sí	Mucho
Structured Advanced Query Page – SAQP (2010)	Sí	Listas	Sí	Nada
Blended Browsing and Querying - BBQ (1999)	?	<i>Drag and drop</i>	No	Poco
Visual Composition of Complex Queries (2013)	No	Menús	No	Poco
Query Builder (2004)	Sí	LN, menús	No	Nada
<b>Interfaz de Composición de Consultas (2016)</b>	<b>Sí</b>	<b>Selección</b>	<b>Sí</b>	<b>Nada</b>

# Capítulo 3

## Versión anterior de la interfaz

### 3.1 Módulo de composición de consultas para una interfaz de lenguaje natural a bases de datos

La interfaz de composición es un complemento que forma parte de un proyecto denominado Interfaz de Lenguaje Natural Español hacia Bases de Datos para Usuarios de Internet, que comenzó a desarrollarse en el CENIDET en septiembre del 2001, y se continúa desarrollando en el ITCM desde agosto del 2002. Dicho proyecto permite a usuarios casuales e inexpertos componer consultas en SQL sobre bases de datos complejas. Un ejemplo de base de datos compleja es la de ATIS, ya que cuenta con 28 tablas y 125 columnas (Apéndice 1).

La interfaz de composición de consultas [Aguirre 2015] (en su versión anterior al desarrollo de este proyecto de tesis) trabaja con información del esquema de la base de datos almacenada en un diccionario de información semántica DIS [Aguirre 2014]; pero primeramente dicha interfaz debe ser configurada por el administrador de bases de datos, ya que éste debe proporcionar información a la interfaz sobre el contenido de la base de datos. Después de haber realizado la configuración, la información sobre tablas y columnas que se encuentran dentro de la BD podrá ser visualizada dentro del árbol de composición, donde el usuario podrá seleccionar los elementos y definir restricciones que formarán parte de la composición de una consulta.

### 3.2 Arquitectura

El módulo desarrollado [Aguirre 2015] permite al usuario componer consultas en SQL sobre bases de datos complejas como se mencionó anteriormente. La operación de dicha interfaz consta de cuatro etapas, las tres primeras guían al usuario en la composición de la consulta y una última etapa muestra el resultado de la composición. Estas etapas son las siguientes: selección de un tema de interés, selección de elementos de interés, especificación de las condiciones de búsqueda, y una vista previa de la consulta en SQL y resultados. En la Figura 3.1 se puede apreciar la arquitectura del funcionamiento de la interfaz, la cual incluye estas cuatro etapas implementadas en la interfaz de composición.

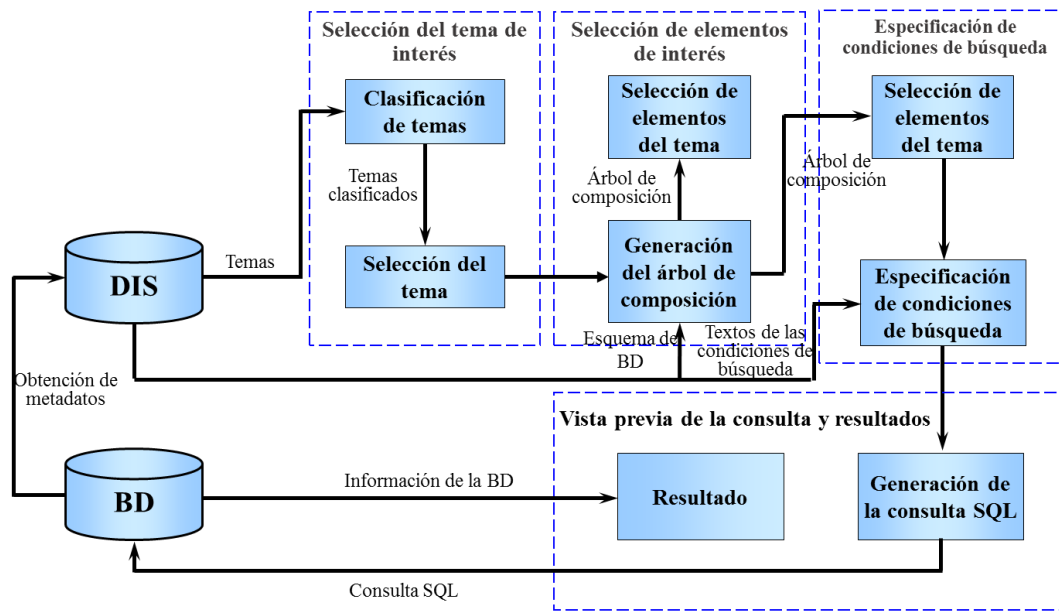


Figura 3.1. Arquitectura de la versión anterior de la interfaz de composición

En la arquitectura presentada en la Figura 3.1, se muestra cómo la interfaz guía al usuario en el proceso de composición de la consulta.

En el primer paso la interfaz realiza una clasificación de las tablas que existen en la base de datos. La clasificación se muestra al usuario como una lista de temas, donde el usuario selecciona uno de estos temas en el cual se basará para construir la consulta.

En la segunda etapa la interfaz genera un árbol de composición, en el cual el usuario selecciona el o los elementos que son utilizados para construir la cláusula SELELCT.

En la tercera etapa el mismo árbol de composición es utilizado para especificar las condiciones de búsqueda. Estas condiciones de búsqueda forman parte de la cláusula Where de la consulta.

En la cuarta y última etapa, la interfaz genera la consulta en SQL de acuerdo a los elementos seleccionados en las etapas anteriores. Se ejecuta dicha consulta y se muestran los resultados obtenidos al usuario.

### 3.3 Árbol de composición

El árbol de composición es la parte más importante de esta interfaz. La estructura gráfica de este árbol es semejante a la estructura de un explorador de archivos, la cual permite visualizar el esquema de la BD mostrando en el árbol de composición los elementos del tema principal y temas relacionados. Mediante esta información que se muestra en el árbol de composición, el usuario puede seleccionar los elementos de los cuales desea conocer información.

La construcción del árbol de composición se efectúa en la segunda etapa y se reutiliza en la tercera etapa. En la segunda etapa (*Selección de elementos de interés*) el árbol de composición se

construye a partir de la primera etapa (*Selección del tema de interés*) en la cual el usuario elige un tema sobre el cual se compondrá la consulta, y dicho tema de interés es colocado como la raíz del árbol de composición. En la tercera etapa (*Especificación de condiciones de búsqueda*) el mismo árbol de composición es reutilizado para definir las restricciones de búsqueda.

La información utilizada del DIS para la construcción del árbol de composición es la siguiente:

- Por cada tabla de la BD:
  - Tipo de tabla.
  - Descripción de la tabla en LN.
  - Descripciones de las columnas en LN.
  - Relaciones entre la tabla y otras tablas.
- Por cada relación entre tablas:
  - Tipo de relación.

Un ejemplo del árbol de composición ya construido en la segunda etapa es el que se muestra en la Figura 3.2. Este árbol de composición es construido a partir del tema principal (vuelos de un aeropuerto a otro) seleccionado por el usuario en la primera etapa. Los nodos hijos representan los elementos del tema mencionado anteriormente o representan temas relacionados.

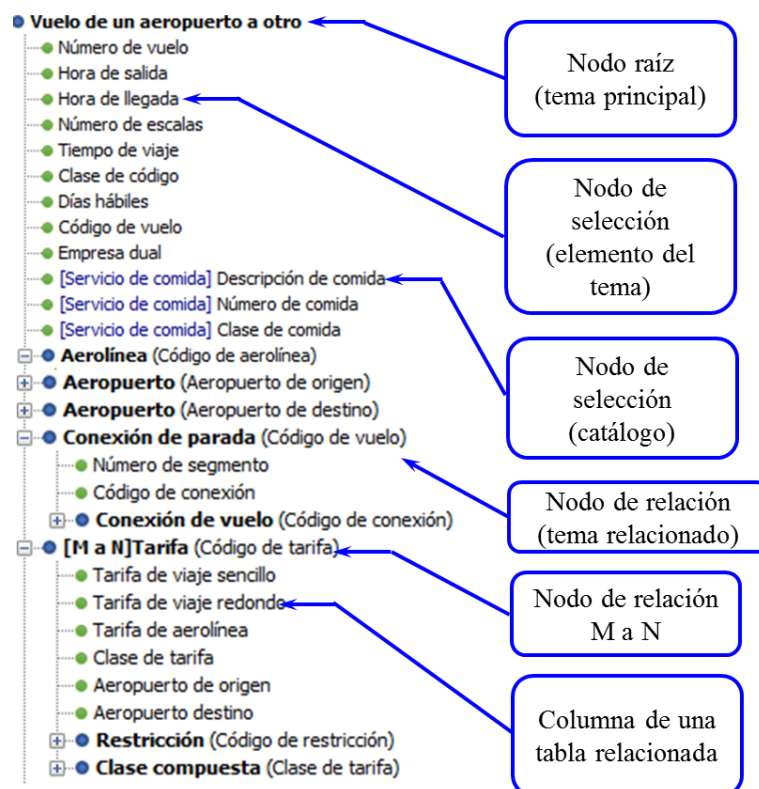


Figura 3.2. Árbol de composición

# Capítulo 4

## Metodología de solución

---

La interfaz de composición objeto de este proyecto de tesis fue diseñada con el propósito de permitir a los usuarios inexpertos componer consultas que involucran subconsultas; y además, esta interfaz ha sido preparada para que en un futuro cercano, los usuarios puedan obtener información de bases de datos complejas como ATIS (ver apéndice 1) mediante esta funcionalidad de subconsultas y añadiendo funciones de agregación.

La funcionalidad de subconsultas adicionada a la interfaz de composición consta de lo siguiente: primeramente el usuario debe realizar la descomposición de la consulta que se encuentra en LN a una consulta en LN pero que involucre subconsultas. Después de haber efectuado lo anterior, la interfaz guiará al usuario en el proceso de composición, de tal forma que éste deberá realizar dicha composición de subconsultas desde la más interna hasta la más externa. De esta manera la interfaz le proporcionaría al usuario el resultado de cada subconsulta y con esto el usuario sabrá en base a qué resultado deberá definir la siguiente consulta/subconsulta en la secuencia de más interna a más externa. Además, ésta es una manera muy sencilla de guiar al usuario paso a paso en la composición de dicha consulta, evitando la posible complejidad del uso de la interfaz de composición.

### 4.1 Arquitectura actual de la interfaz de composición

En la Figura 4.1 se muestra la nueva arquitectura de la interfaz de composición. En color más oscuro se resaltan las fases que han sido integradas a dicha interfaz y que son utilizadas para la composición de consultas que involucran subconsultas. La arquitectura actual está formada por: descomposición de consultas en LN, selección del tema de interés, especificación de condiciones de búsqueda, vista previa y especificación de condiciones de búsqueda para la subconsulta, selección de elementos de interés, y por último, vista previa y resultado.

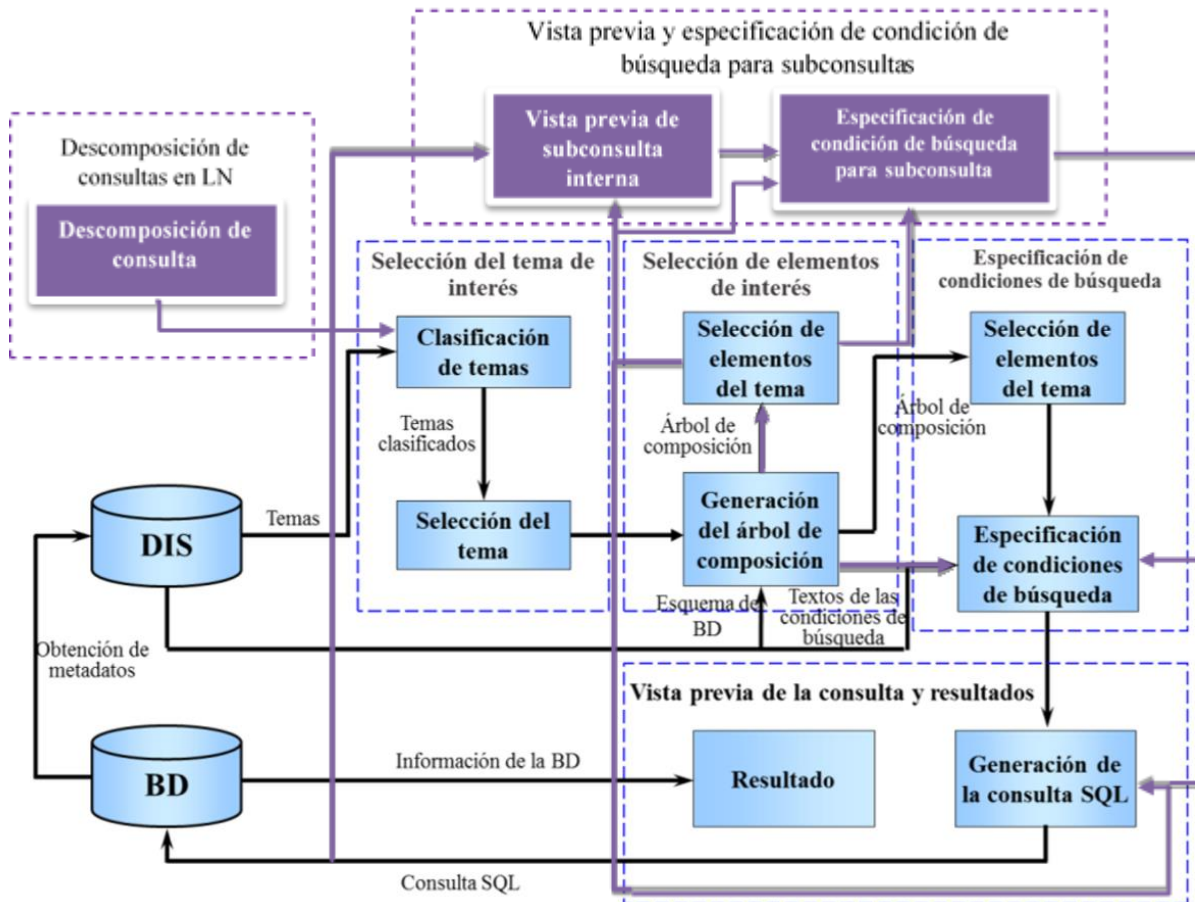


Figura 4.1. Nueva arquitectura de la interfaz de composición

Todas las fases en la arquitectura actual de la interfaz de composición guían al usuario en el proceso de composición de consultas que involucran subconsultas. La primera fase es *Descomposición de la consulta*; para tal efecto es necesario que el usuario tenga mentalmente formulada una consulta en LN para después descomponerla, de tal manera que coloque cada parte de la consulta donde la interfaz le indique.

La siguiente fase es la *Selección del tema de interés*. En esta fase la interfaz toma la información relacionada a las tablas existentes de la base de datos, las clasifica y muestra al usuario una lista de temas, de los cuales el usuario solamente debe elegir uno para componer la consulta/subconsulta.

La *Especificación de condiciones de búsqueda* es la siguiente fase. En esta fase se genera un árbol de composición para las subconsultas a partir del DIS. El usuario hace uso de este árbol de composición seleccionando un elemento a la vez y especificando su condición de búsqueda para formar la cláusula *Where* de la subconsulta.

En la fase *Vista previa y especificación de condiciones de búsqueda para la subconsulta*, se muestra un resultado previo devuelto por la subconsulta anterior, y en base a dicho resultado el

usuario especifica la condición de búsqueda para la subconsulta además de poder definir más condiciones de búsqueda.

Posteriormente se encuentra la fase *Selección de elementos de interés*, en la cual se genera un árbol de composición del cual el usuario selecciona el o los elementos de interés de los cuales desea obtener información. Finalmente en la última fase, la interfaz construye una consulta/subconsulta en SQL la cual es enviada al SMBD para obtener los resultados de la consulta/subconsulta que posteriormente son mostrados al usuario.

## 4.2 Consultas con niveles de subconsultas a tratar

Se han considerado tres tipos de consultas en base a un análisis exhaustivo de las consultas que se encuentran en el corpus de ATIS ([https://catalog.ldc.upenn.edu/docs/LDC93S4B-3/TRN\\_PRMP.TXT](https://catalog.ldc.upenn.edu/docs/LDC93S4B-3/TRN_PRMP.TXT)). Algunos ejemplos de los tipos de consultas a tratar se presentan en la tabla 4.1 en donde se muestran consultas que involucran una, dos y hasta tres subconsultas.

Tabla 4.1. Tipos de consultas a tratar

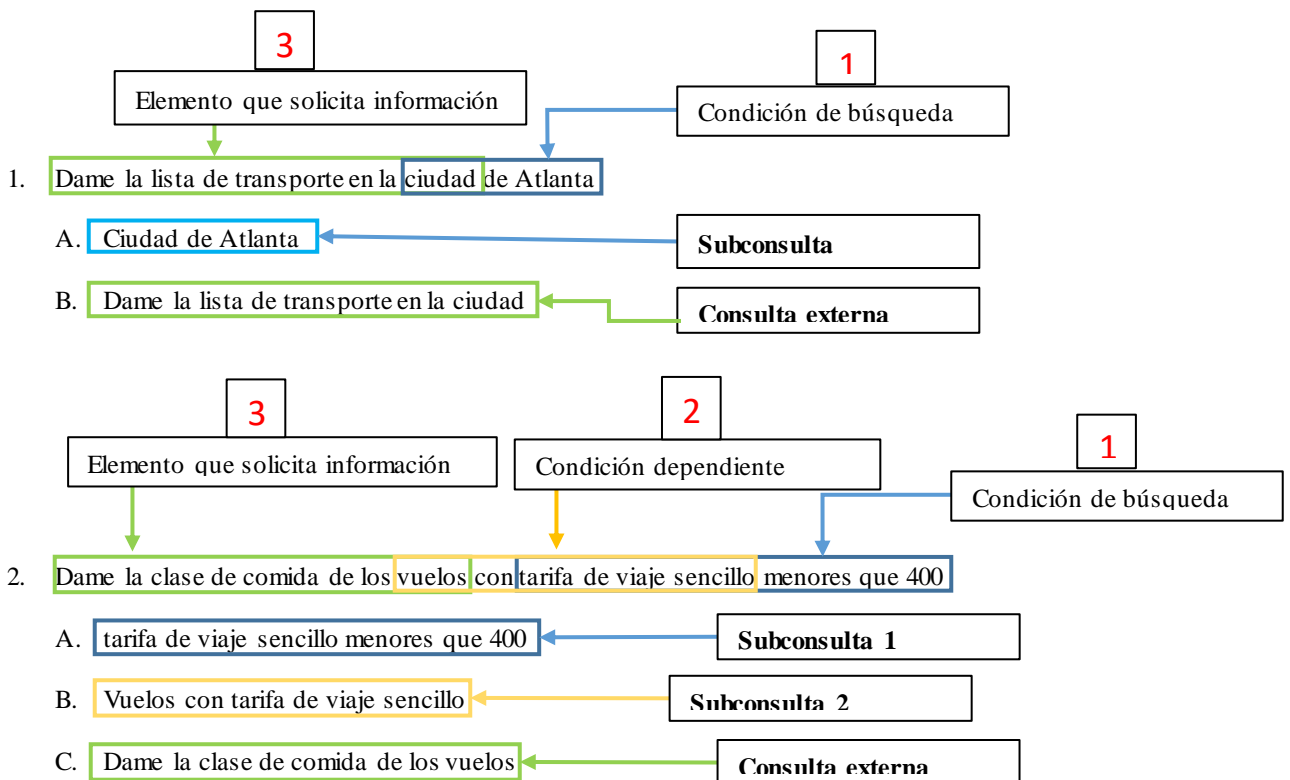
Tipo de consulta	Consulta en LN	Traducción a SQL	No. de subconsultas
1	¿Qué tipo de avión tiene el vuelo No. 301?	<pre>SELECT aircraft_type FROM aircraft WHERE aircraft_code IN (SELECT aircraft_code FROM flight WHERE flight.flight_number = 301)</pre>	1
2	Dame la tarifa de viaje sencillo del vuelo No. 140	<pre>SELECT one_way_cost FROM fare WHERE fare_code IN (SELECT fare_code FROM flight_fare WHERE flight_code IN (SELECT flight.flight_code FROM flight WHERE flight.flight_number = 140))</pre>	2
3	Dame la lista de transporte en la ciudad de ATLANTA	<pre>SELECT transport_desc FROM transport WHERE transport_code IN (SELECT transport_code FROM ground_service WHERE city_code IN (SELECT city_code FROM airport_service WHERE city_code IN (SELECT city.city_code FROM city WHERE city.city_name = 'ATLANTA'))))</pre>	3

### 4.3 Descomposición de consultas en LN

Antes de iniciar la composición de la consulta mediante la interfaz, el usuario debe descomponer en varias partes la consulta que se encuentra en LN, de tal manera que cada parte sea alguna de las subconsultas o la consulta más externa. Para llevar a cabo esto, es necesario que el usuario siga los siguientes tres pasos claves para la descomposición:

1. Para la descomposición de las consultas, primeramente el usuario debe identificar la condición de búsqueda, la cual será considerada en cuenta como una subconsulta.
2. En caso de existir una condición que dependa del paso anterior, ésta será considerada como una subconsulta externa a la subconsulta anterior, de lo contrario se debe continuar con el paso siguiente.
3. El o los elementos restantes de la consulta que definen la información solicitada por el usuario, constituirán la consulta más externa.

De acuerdo a los pasos anteriores, la descomposición de una consulta que involucra subconsultas quedaría como se muestra en los siguientes ejemplos:



### 4.4 Árbol de composición

El árbol de composición propuesto en [Aguirre 2015] como la parte más importante de la interfaz de composición, es reutilizado y adaptado de tal manera que con él puedan definirse las condiciones



de búsqueda para las subconsultas. Dicho lo anterior, la estructura del árbol de composición se compone de lo siguiente:

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

Dado lo anterior, a diferencia del árbol de composición original (ver sección 3.3), este árbol solamente muestra al usuario el tema principal y sus elementos como se ve en la Figura 4.2; es decir, eliminando los temas relacionados con el tema principal para así evitar posibles errores al momento en que el usuario elija el o los elementos para los cuales especificaría una condición de búsqueda. Además, los temas relacionados son innecesarios en la construcción de la subconsulta.

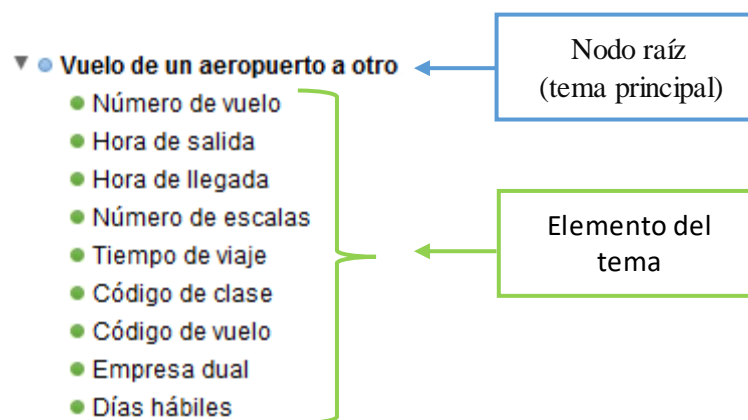


Figura 4.2. Árbol de composición para subconsultas

Este árbol de composición es construido en la tercera fase (*Especificación de condiciones de búsqueda*) después de que el usuario eligió el tema del cual realizará la composición de la subconsulta. En la Figura 4.2 se muestra el árbol construido para la base de datos ATIS para realizar las especificaciones de búsqueda para una subconsulta. En dicho árbol, el nodo raíz representa la tabla *flight*, cuya descripción es *Vuelo de un aeropuerto a otro*, y los nodos hijos, con botones verdes, representan los elementos del tema principal.

En el Algoritmo 4.1 se presenta el pseudocódigo de la construcción del árbol de composición para las subconsultas en donde simplemente se coloca el tema principal y sus elementos.

*EA* contiene la estructura del árbol de composición, *TP* contiene el nombre y la descripción del tema elegido por el usuario, *TR* contiene el nombre del tema y *DTR* contiene la descripción. Se inicia el árbol de composición colocando la descripción del tema como nodo raíz, y posteriormente para colocar los nodos hijos, se emplea el método *InsertarNodoColumna*, el cual en base al tema principal obtiene solamente los elementos pertenecientes a dicho tema y los coloca en el árbol.

---

**Algoritmo 4.1. Seudocódigo del árbol de composición para las subconsultas**

---

1.  $TP$
  2.  $EA$
  3.  $TR = TP[0]$
  4.  $DTR = TP[1]$
  5.  $DefinirRaiz=DTR$
  6.  $InsertarNodoColumna(TR,EA)$
- 

## 4.5 Generación automática de subconsultas intermedias

La generación automática de subconsultas por parte de la interfaz de composición es la parte más importante de la funcionalidad de subconsultas. Gracias a este proceso se han podido componer consultas que cuentan con hasta 4 subconsultas en SQL y para consultas en LN donde el usuario solamente realiza la composición de una subconsulta y la consulta externa. La interfaz se encarga de generar automáticamente las demás subconsultas sin que el usuario se dé cuenta, con el fin de poder mostrar al usuario una traducción correcta a SQL de su consulta compuesta en LN.

En el Algoritmo 4.2, mediante la consulta externa y una subconsulta compuesta con anterioridad, se construye una matriz de adyacencia mediante el ciclo *For* (ver línea 8 del Algoritmo 4.2), el cual contiene las tablas que se encuentran en la base de datos y las relaciones entre ellas. Esta matriz es utilizada en el método *GenerarAlgBFS*, en el cual implementa el algoritmo BFS (*Breadth First Search*). Posteriormente se obtienen las tablas intermedias del camino más corto generado por BFS entre la tabla de la consulta externa y la tabla de la subconsulta. En caso de haber obtenido dos o más tablas intermedias, el algoritmo ejecuta el método *CrearConsultasInternas*, el cual comienza a construir las subconsultas y las enlaza entre sí junto con la subconsulta compuesta con anterioridad mediante las llaves primarias y las llaves foráneas que corresponden entre tablas.

Para ilustrar el funcionamiento del Algoritmo 4.2, considérese la consulta: *lista clases de servicio del vuelo 296*. La descomposición de la consulta se muestra en la Tabla 4.2, la cual incluye la traducción a SQL de la subconsulta (*Vuelo 296*) y de la consulta externa (*Lista clases de servicio del vuelo*).

Tabla 4.2. Descomposición de la consulta de ejemplo

Descomposición	Traducción a SQL
1. Vuelo 296	SELECT <i>flight.flight_code</i> FROM <i>flight</i> WHERE <i>flight.flight_number</i> = 296
2. Lista clases de servicio del vuelo	SELECT <i>class_description</i> FROM <i>class_of_service</i> WHERE <i>class_code</i> IN...

Como puede verse en el esquema de la base de datos de ATIS (Figura A1.1), no existe una relación directa entre las tablas *flight* y *class\_of\_service*; por lo tanto, es necesario que la interfaz

genere tres subconsultas intermedias para relacionar estas tablas, tal como se muestra en la siguiente expresión, que constituye la traducción a SQL de la consulta en LN.

```

SELECT class_description
FROM class_of_service
WHERE class_code IN
  (SELECT base_class
   FROM compound_class
   WHERE fare_class IN
     (SELECT fare_class
      FROM fare
      WHERE fare_code IN
        (SELECT fare_code
         FROM flight_fare
         WHERE flight_code IN
           (SELECT flight.flight_code
            FROM flight
            WHERE flight.flight_number = 296))))))

```

La Tabla 4.3 muestra la matriz de adyacencia que requiere el algoritmo BFS, la cual representa el grafo que modela a la base de datos de ATIS (Figura A1.1), donde los nodos del grafo representan tablas y los arcos representan relaciones entre éstas. En la matriz de adyacencia los 1s representan una relación entre una tabla *A* y una tabla *B*. En la matriz de adyacencia las tablas están representadas por números, cuya correspondencia se muestra en la Tabla 4.4.

Tabla 4.3. Matriz de adyacencia de la BD ATIS

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	...
1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
2	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1
3	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0	1
4	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1
5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1
6	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
11	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
13	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0
14	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	1
15	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
...	0	1	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0

Tabla 4.4. Identificación numérica de las tablas de la BD ATIS

1. aircraft	2. airline	3. airport	4. airport_service
5. city	6. class_of_service	7. code_description	8. compound_class
9. connect_leg	10. day_name	11. dual_carrier	12. fare
13. fconnection	14. flight	15. flight_day	16. flight_fare
17. food_service	18. ground_service	19. month_name	20. restrict_carrier
21. restrict_class	22. restriction	23. state	24. stop1
25. time_interval	26. time_zone	27. transport	

El algoritmo BFS genera una secuencia de visita en base a una tabla inicial. Para este ejemplo dicha tabla inicial tiene el número 6 como identificador que corresponde a la tabla *class\_of\_service*. La secuencia de visita para ese ejemplo es la siguiente: 6, 8, 12, 16, 22, 14, 20, 21, 1, 2, 3, 9, 17, 24, 11, 4, 13, 23, 5, 18, 26, 27.

Posteriormente el algoritmo BFS genera un camino desde la tabla que corresponde a la consulta más externa hasta la tabla que corresponde a la subconsulta más interna. De esta manera identifica las tablas intermedias que se encuentran entre estas dos tablas mencionadas anteriormente. Posteriormente, identifica dentro del DIS las relaciones que unen a estas tablas entre sí, junto con la subconsulta más interna y la consulta más externa, generando así automáticamente las subconsultas intermedias faltantes.

El camino generado por el algoritmo BFS para este ejemplo es el siguiente: 8, 12, 16, 14, cuyos identificadores corresponden a las siguientes tablas: *compound\_class*, *fare*, *flight\_fare*, *flight*. Esta última tabla corresponde a la subconsulta (*Vuelo 296*) compuesta por el usuario.

Al final, la interfaz genera automáticamente las subconsultas intermedias faltantes, y las enlaza con las expresiones de SQL que se muestran en la Tabla 4.2 para generar la expresión completa en SQL mostrada previamente.

El Algoritmo 4.2 es utilizado en la segunda fase (*Selección del tema de interés*) cuando la consulta que se encuentra en LN es descompuesta por el usuario y contiene más de una subconsulta en LN. Este algoritmo también es utilizado en la quinta fase (*Selección de elementos de interés*) cuando la descomposición de la consulta que se encuentra en LN contiene solamente una subconsulta en LN.

---

**Algoritmo 4.2. 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  $i = 1, \dots, |ObTablas|$   $j = 1, \dots, |ObTablas|$
8.     **For**  $i = 0$  to *ObRelaciones*
9.         *RelacionesTablas* = *ObRelaciones*.get(i)

10.  $MaAde[RelacionesTablas[0]][RelacionesTablas[1]] = 1$
  11.  $MaAde[RelacionesTablas[1]][RelacionesTablas[0]] = 1$
  12. **EndFor**
  13.  $GenerarAlgBFS(MaAde, TablaConsExterna)$
  14.  $RutaSubconsultas=GenerarCamino(TablaSubconsulta)$
  15. **If**  $RutaSubconsultas \geq 3$
  16.  $CrearConsultasInternas(RutaSubconsultas, TablaConsExterna, TablaSubconsulta)$
  17. **EndIf**
  18. **EndIf**
-

# Capítulo 5

## Interfaz de composición

---

La funcionalidad de subconsultas que ha sido integrada a la interfaz de composición tiene como objetivo permitir a los usuarios inexpertos realizar consultas que involucran subconsultas sobre bases de datos complejas sin la necesidad de conocer el esquema de la BD. Para tal efecto, la interfaz guía al usuario por distintas ventanas para realizar la composición de consultas que se encuentran en LN.

### 5.1 Conceptos para composición de consultas que involucran subconsultas

A pesar de que el usuario no conozca la estructura de la base de datos ni un lenguaje de consulta, es necesario que tenga conocimientos sobre algunos conceptos básicos que se manejan en la interfaz de composición. En la Tabla 5.1 se describen algunos conceptos básicos de las ventanas con las cuales el usuario debe realizar la composición de consultas que involucran subconsultas.

Tabla 5.1. Conceptos básicos para la composición de consultas que involucran subconsultas.

<i>Concepto</i>	<i>Descripción</i>
<i>Descomposición de consulta</i>	Proceso mediante el cual la consulta que se encuentra en LN se divide en distintas consultas en LN: específicamente, una o varias subconsultas y la consulta principal.
<i>Tema de interés</i>	Es el tema del cual se desea conocer información (en una consulta o subconsulta).
<i>Condiciones de búsqueda</i>	El o los elementos que restringen la búsqueda de información.
<i>Resultado de subconsulta interna</i>	Resultado devuelto por una subconsulta generado internamente por la interfaz.
<i>Condición de búsqueda para resultado de subconsulta</i>	Definición de la condición de búsqueda, la cual se define eligiendo una de dos opciones: “ <b>se encuentra dentro de</b> ” o “ <b>no se encuentra dentro de</b> ” del resultado de la subconsulta.
<i>Elementos de interés</i>	Elementos específicos del tema principal, de los cuales se desea conocer información.

Después de haber identificado los conceptos descritos en la tabla anterior que corresponden a las ventanas de la interfaz de composición, el usuario puede hacer un mejor uso de dicha interfaz para realizar la composición de consultas que involucran subconsultas.

## 5.2 Descomposición de la consulta en LN

En esta ventana, que se muestra en la Figura 5.1, el usuario deberá realizar la descomposición de la consulta que se encuentra en lenguaje natural en una consulta del mismo tipo pero que involucre subconsultas tal como se explica en la sección 4.3. En la Tabla 5.2 se muestra la descripción de los componentes que integran esta ventana, donde el número que se encuentra en la columna izquierda de la tabla, corresponde al número que identifica el componente en la Figura 5.1.

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

<i>Componente</i>	<i>Descripción</i>
<b>1. Consulta original.</b>	En esta caja de texto se escribe la consulta original que se encuentra en lenguaje natural.
<b>2. Consulta principal.</b>	En esta caja de texto se escribe la consulta más externa (consulta con elementos de interés) en base a la descomposición de la consulta.
<b>3. Definición de subconsultas.</b>	De esta lista se elige el número de subconsultas que contiene la consulta original en base a la descomposición realizada por el usuario, ya sea una, dos o tres subconsultas.
<b>4. Subconsultas.</b>	En cada caja de texto se escribe una subconsulta. Dichas subconsultas se obtienen de la descomposición de la consulta original de izquierda a derecha, omitiendo la consulta principal.
<b>5. Botón “Continuar”.</b>	Terminando de escribir las consultas correspondientes a cada caja de texto, el usuario debe oprimir el botón <i>Continuar</i> para iniciar paso a paso la composición de la consulta.
<b>6. Barra de menú.</b>	Esta barra se integra por las opciones <i>Archivo, Herramientas y Ayuda</i> . La primera opción es utilizada para salir de la interfaz y cancelar el proceso de composición, la segunda opción es utilizada por el administrador de la BD para configurar la interfaz y la última opción muestra el manual de usuario.

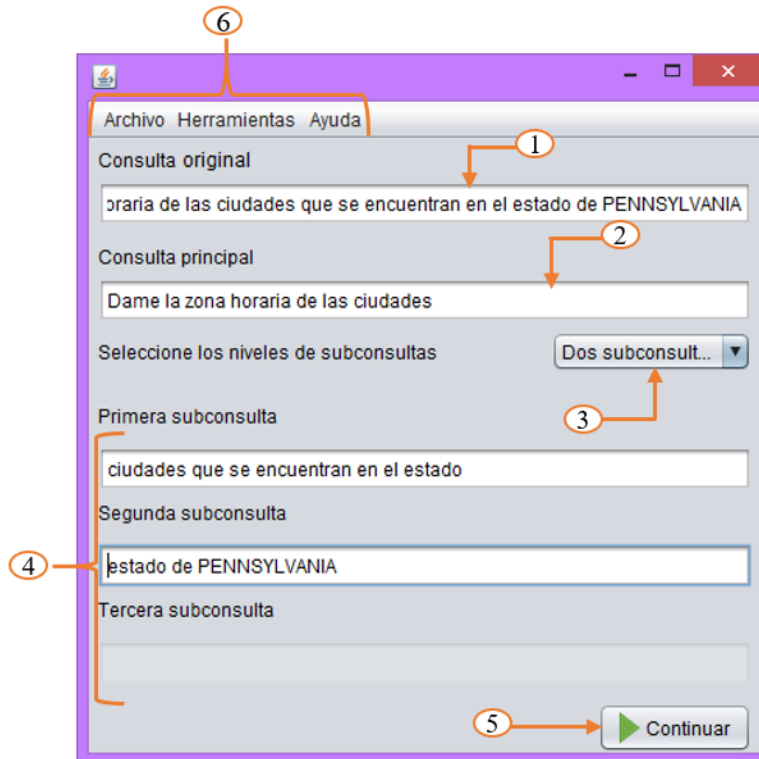


Figura 5.1. Descomposición de consultas en LN

Es importante que el usuario tenga muy bien identificada cada parte de la consulta formulada en LN, ya que esta fase de descomposición de consultas es fundamental para que la interfaz guíe al usuario paso a paso en el proceso de composición de consultas/subconsultas.

### 5.3 Selección del tema de interés

En esta ventana el usuario selecciona el tema sobre el cual realizará la composición de la consulta/subconsulta. En la Tabla 5.3 se muestra la descripción de los componentes que integran esta ventana, donde el número que se encuentra en la columna izquierda de la tabla, corresponde al número que identifica el componente en la Figura 5.2.

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

<i>Componente</i>	<i>Descripción</i>
<i>1. Consulta a componer.</i>	En letras azules la interfaz muestra la consulta a componer por el usuario.
<i>2. Lista de temas.</i>	En esta lista se muestran los temas pertenecientes a la base de datos, de los cuales el usuario solamente puede elegir un tema para realizar la composición de la consulta.



- |   |  |
|---|--|
| <p>3. <b>Navegación para ver más temas.</b></p> | <p>Estos botones sirven para navegar entre todos los temas de la base de datos. Los temas se encuentran ordenados de mayor a menor importancia en la lista.</p>                |
| <p>4. <b>Botón “Continuar”.</b></p>             | <p>Después de que el usuario haya elegido el tema sobre el cual realizará la composición de la consulta, éste deberá oprimir el botón para avanzar a la siguiente ventana.</p> |

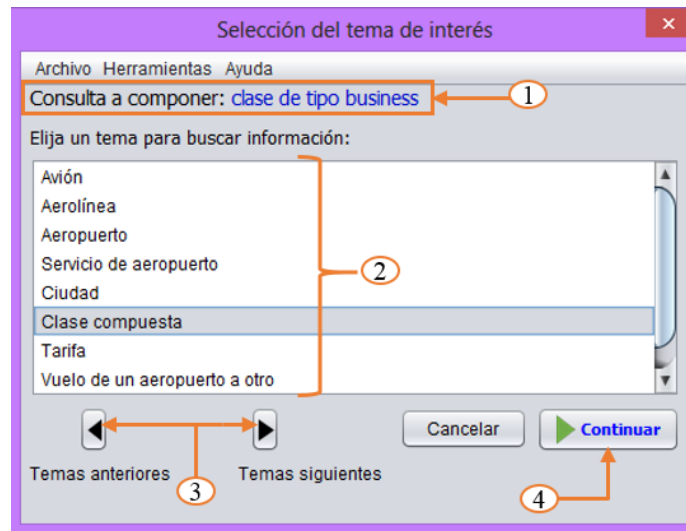


Figura 5.2. Selección del tema de interés

Los temas que aparecen dentro de esta ventana son las descripciones de las tablas que se encuentran dentro de la base de datos. Dichas tablas se muestran ordenadas de mayor a menor relevancia para la composición de consultas que involucran subconsultas y son clasificadas de la siguiente manera: tablas base, catálogos, relaciones M a N y tablas satélites [Aguirre 2015].

## 5.4 Especificación de condición de búsqueda

La ventana que se muestra en la Figura 5.3 es una de las más importantes de la composición de consultas que involucran subconsultas. Ya que, para empezar con la composición de este tipo de consultas, es necesario definir condiciones para restringir la búsqueda de información, especialmente para la subconsulta más interna, ya que en esta siempre es necesario definir una condición de búsqueda.

En la Tabla 5.4 se muestra la descripción de los componentes que integran esta ventana, donde el número que se encuentra en la columna izquierda de la tabla, corresponde al número que identifica el componente en la Figura 5.3.

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

<b>Componente</b>	<b>Descripción</b>
1. <b>Árbol de elementos del tema.</b>	En esta estructura se muestran los elementos pertenecientes al tema seleccionado anteriormente. Sólo se pueden definir las restricciones de una en una.
2. <b>Lista de condiciones del elemento seleccionado.</b>	Cada elemento que pueda ser seleccionado en el árbol (1) puede tener un máximo de 5 condiciones para restringir la búsqueda; menor que (<), mayor que (>), menor o igual a (<=), mayor o igual a (>=) y diferente de (<>). Estas condiciones sirven para que el usuario delimite la búsqueda de resultados.
3. <b>Valor de la condición.</b>	Esta ésta caja de texto junto con la lista de condiciones (2) permiten al usuario especificar una condición deseada para delimitar la búsqueda.
4. <b>Ayuda.</b>	Este botón permite que el usuario pueda ver en línea de qué manera se definen las condiciones de búsqueda.
5. <b>Añadir condición.</b>	Una vez especificada una condición, el usuario deberá hacer clic en el botón para añadir dicha condición a la lista (6).
6. <b>Lista de condiciones de búsqueda.</b>	Esta lista le muestra al usuario las condiciones que ha añadido en esta ventana.
7. <b>Botón “Eliminar”.</b>	Este botón permite al usuario eliminar una condición seleccionada en (6).
8. <b>Botón “Atrás”.</b>	Este botón le permite al usuario regresar a una ventana anterior si es necesario corregir algo de la composición.
9. <b>Botón “Continuar”.</b>	Después de añadir las condiciones de búsqueda deseadas para la consulta, el usuario debe hacer clic sobre este botón para continuar con el proceso de composición.

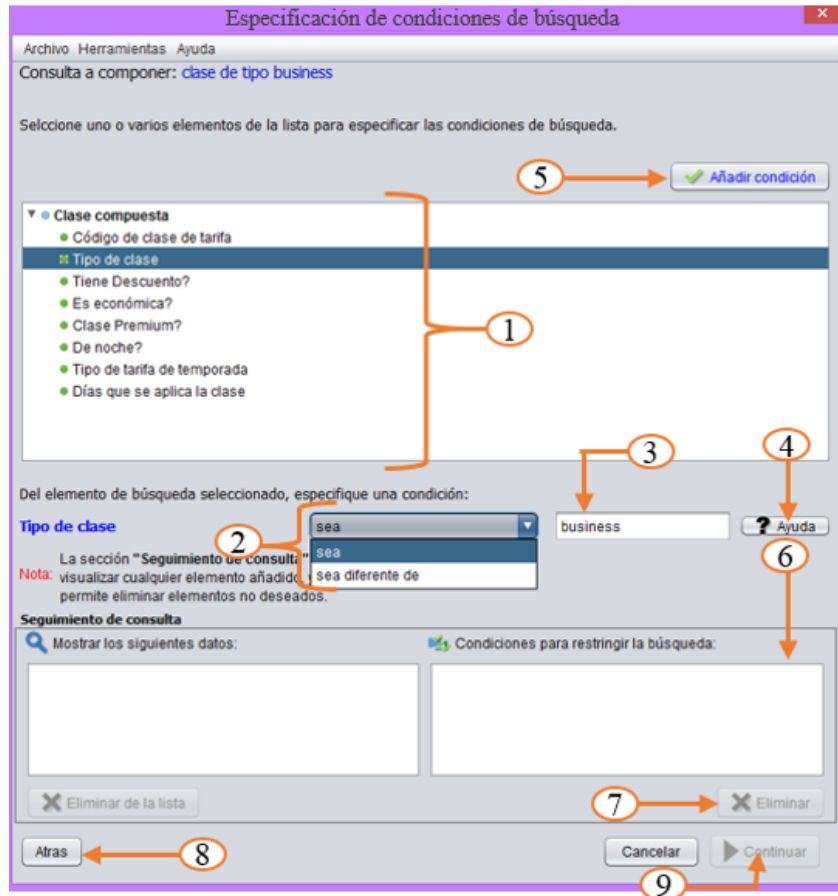


Figura 5.3. Especificación de condiciones de búsqueda

Las opciones para especificar las condiciones de búsqueda dentro de esta ventana son proporcionadas por el DIS, después de que el administrador de la BD ha configurado la interfaz. Cada condición de búsqueda se muestra dependiendo del elemento seleccionado del árbol de composición al cual el usuario desee especificarle un valor de búsqueda (ver Figura 5.4).

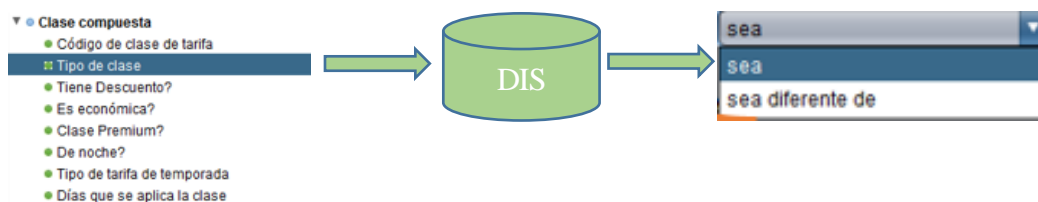


Figura 5.4. Operadores de búsqueda

## 5.5 Resultado de subconsulta interna

La ventana que se muestra en la Figura 5.5 simplemente es para mostrar al usuario el resultado devuelto por las subconsultas generadas automáticamente por la interfaz (ver sección 4.5), aunque el usuario no pueda darse cuenta de esta acción por parte de la interfaz. El resultado tiene

coherencia con las subconsultas y con la consulta que está componiendo hasta el momento en que llega a esta ventana.

En la Tabla 5.5 se muestra la descripción de los componentes que integran esta ventana, donde el número que se encuentra en la columna izquierda de la tabla, corresponde al número que identifica el componente en la Figura 5.5.

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

<i>Componente</i>	<i>Descripción</i>
<b>1. Resultado previo.</b>	En esta tabla se muestra el resultado previo correspondiente a la consulta generada internamente por la interfaz, mostrándole al usuario los resultados que satisfacen la condición de búsqueda especificada en la subconsulta.
<b>2. Botón “Atrás”.</b>	Este botón le permite al usuario regresar a una ventana anterior si es necesario corregir algo de la composición.
<b>3. Botón “Continuar”.</b>	Después de que el usuario haya visto el resultado devuelto por la subconsulta interna, éste deberá oprimir el botón para pasar a la siguiente ventana.

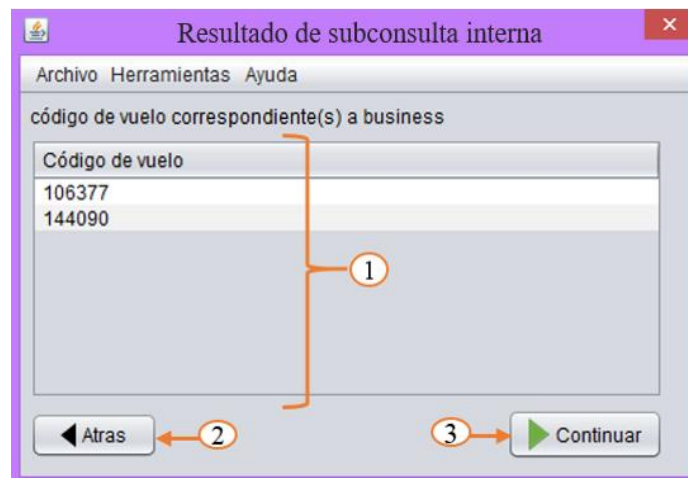


Figura 5.5. Resultado de subconsulta interna

En esta ventana el usuario solamente puede ver y darse cuenta que estos resultados devueltos por la interfaz corresponden a una condición definida anteriormente. Por lo tanto, el usuario solamente puede observar y hacer clic en el botón “Continuar” para proceder a definir la o las condiciones de búsqueda para la subconsulta de acuerdo a estos resultados.

## 5.6 Especificación de condición de búsqueda para el resultado de subconsultas

En la Figura 5.6 se define la condición de búsqueda para las subconsultas y únicamente existen dos opciones. La primera opción hace referencia al operador *IN*, y la segunda opción hace referencia al operador *NOT IN* de SQL. En la Tabla 5.6 se muestra la descripción de los componentes que integran esta ventana, donde el número que se encuentra en la columna izquierda de la tabla, corresponde al número que identifica el componente en la Figura 5.6.

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

Componente	Descripción
1. <b>Tabla de resultado.</b>	En esta tabla se muestra el resultado devuelto por la subconsulta.
2. <b>Lista de condiciones.</b>	El usuario debe elegir de la lista la condición la condición que restrinja la consulta de acuerdo al resultado de la subconsulta: <ul style="list-style-type: none"><li>• Sea igual a alguno(s) de.</li><li>• Sea diferente de todos.</li></ul>
3. <b>Botón “Atrás”.</b>	Este botón le permite al usuario regresar a una ventana anterior si es necesario corregir algo de la composición.
4. <b>Botón “Continuar”.</b>	Después de que el usuario haya definido una condición para la subconsulta, éste deberá oprimir el botón para pasar a la siguiente ventana.

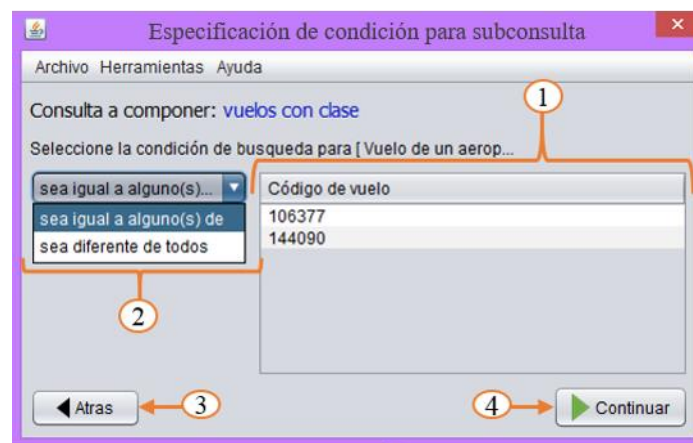


Figura 5.6. Especificación de condición de búsqueda para el resultado de subconsultas

Dentro de esta ventana el componente más importante es la definición de la condición de búsqueda para la subconsulta, en la cual, para un conjunto de valores devuelto por una subconsulta interna, se define una condición de búsqueda ya sea la condición *sea igual a alguno(s) de* o *sea diferente de todos* que hacen referencia a los operadores *IN* y *NOT IN* de SQL.

Si existe una subconsulta que no es la más interna, la interfaz obtiene del DIS el nombre correspondiente y la llave primaria de la tabla que tiene como descripción el tema seleccionado por el usuario en la fase *Selección del tema de interés* y posteriormente las cláusulas *Select* y *From* son construidas con esta información que se obtiene del DIS. La cláusula *Where* se complementa con la condición de búsqueda que se especifica por parte del usuario en la ventana que se muestra en la Figura 5.6 y con la subconsulta compuesta con anterioridad.

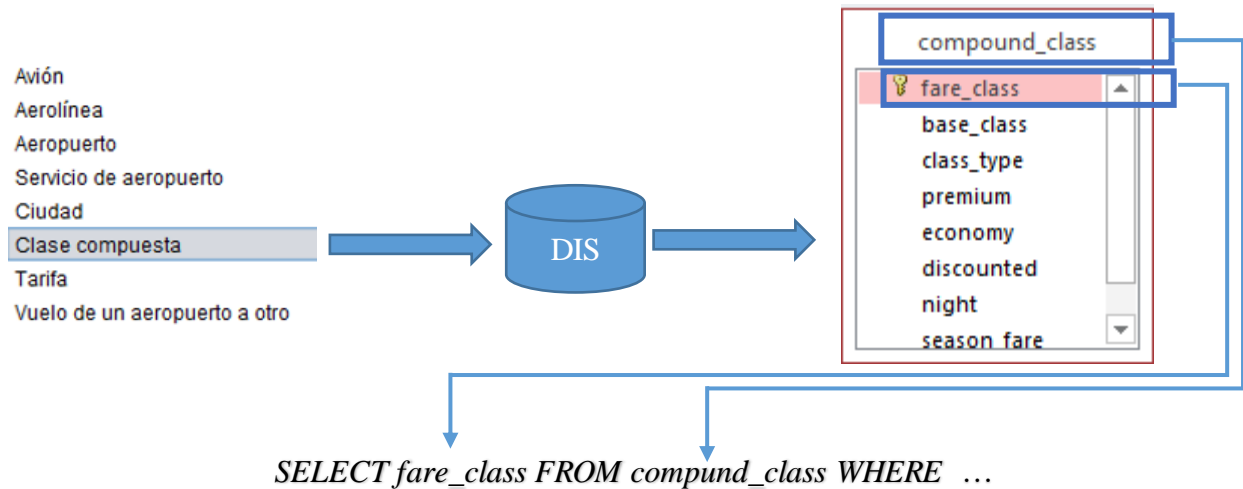


Figura 5.7. Construcción de las cláusulas *Select* y *From*

## 5.7 Selección de elementos de interés

En esta ventana, que se muestra en la Figura 5.8, el usuario selecciona el o los elementos de los cuales desea obtener información. Los elementos son seleccionados uno a uno, los cuales formarán la cláusula *Select*. En la Tabla 5.7 se muestra la descripción de los componentes que integran esta ventana, donde el número que se encuentra en la columna izquierda de la tabla, corresponde al número que identifica el componente en la Figura 5.8.

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

<i>Componente</i>	<i>Descripción</i>
1. <i>Árbol de elementos del tema.</i>	En esta estructura se muestran los elementos y temas relacionados con el tema elegido por el usuario. Solamente se puede seleccionar un elemento a la vez.
2. <i>Botón “Añadir elemento”.</i>	Una vez que se haya elegido un elemento del árbol (1), se puede añadir a la lista de elementos a mostrar (3).
3. <i>Lista de elementos a mostrar.</i>	Esta lista contiene todos los elementos que el usuario desee conocer del tema principal.
4. <i>Botón “Eliminar elemento”.</i>	Este botón sirve para eliminar de la lista (3) los elementos que haya agregado por error.

5. **Botón “Atrás”.**

Este botón le permite al usuario regresar a una ventana anterior si es necesario corregir algo de la composición.

6. **Botón “Continuar”.**

Una vez que el usuario haya seleccionado todos los elementos que desee conocer del tema, éste deberá usar el botón para pasar a la siguiente ventana.

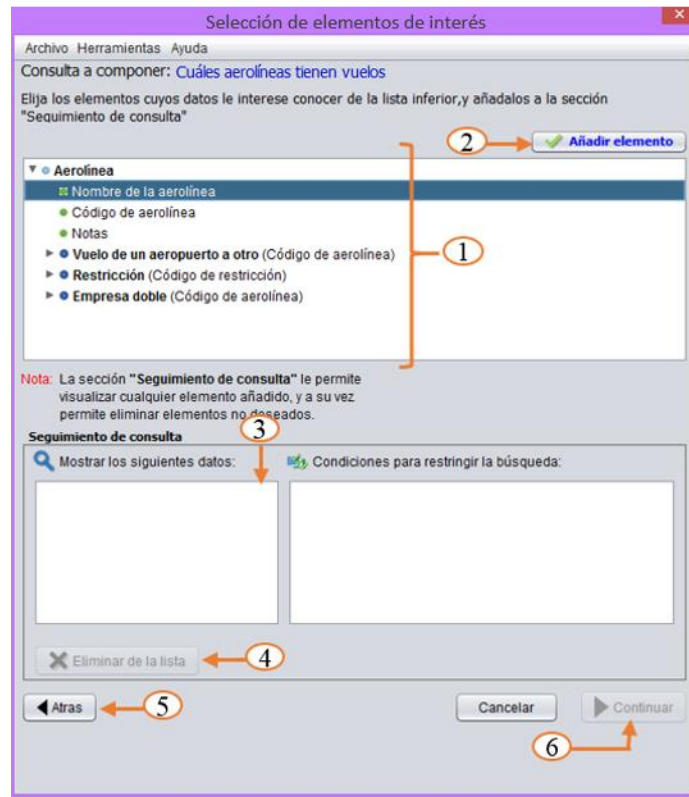


Figura 5.8. Selección de elementos de interés

La selección de los elementos de interés es utilizada en la consulta más externa de tal manera que la interfaz construye la cláusula *Select* con los elementos que el usuario elige del árbol de composición. Dichos elementos, después de ser elegidos uno por uno, se muestran en la caja de texto (ver No. 3 en la Figura 5.8). Estos elementos se almacenan en arreglos unidimensionales que posteriormente serán utilizados en la construcción de la consulta en SQL.

## 5.8 Construcción en lenguaje SQL de consultas que involucran subconsultas

Una de las principales funciones de la interfaz es la construcción de consultas que involucran subconsultas en SQL. Estas consultas se construyen de distintas maneras de acuerdo a los niveles de subconsultas.

Primeramente, para las consultas que involucran una subconsulta, el usuario debe comenzar con la composición de la subconsulta. Para tal efecto, es necesario elegir un tema de interés, y

posteriormente del árbol de composición elegir el o los elementos a los cuales se definirá una condición de búsqueda para cada uno.

Para las cláusulas *Select* y *From* se maneja un arreglo el cual es construido en la fase *Selección del tema de interés*. A partir de esta fase la interfaz obtiene el nombre de la tabla, su descripción, el nombre de la llave primaria y la descripción de ésta. En la Figura 5.9 se puede observar un ejemplo de la construcción de las cláusulas *Select* y *From*.

La estructura básica de una consulta o subconsulta en SQL es la siguiente: *SELECT column(s) FROM tabla WHERE condición(es)*.

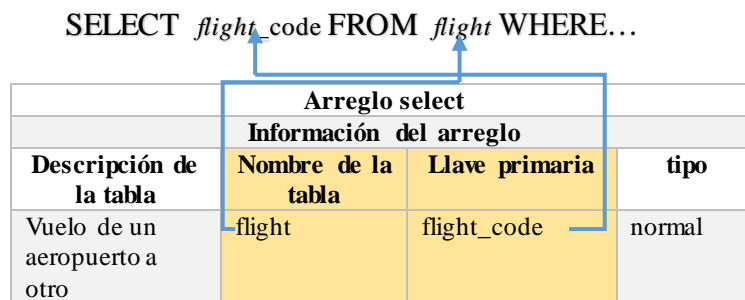


Figura 5.9. Construcción de las cláusulas *Select* y *From*

La composición de la cláusula *Where* es creada a partir de las restricciones de búsqueda para cada elemento. Para tal efecto, se maneja un arreglo que contiene la información del tema y el elemento restringido por el usuario y otro arreglo con las características de restricción al elemento ver Figura 5.10.

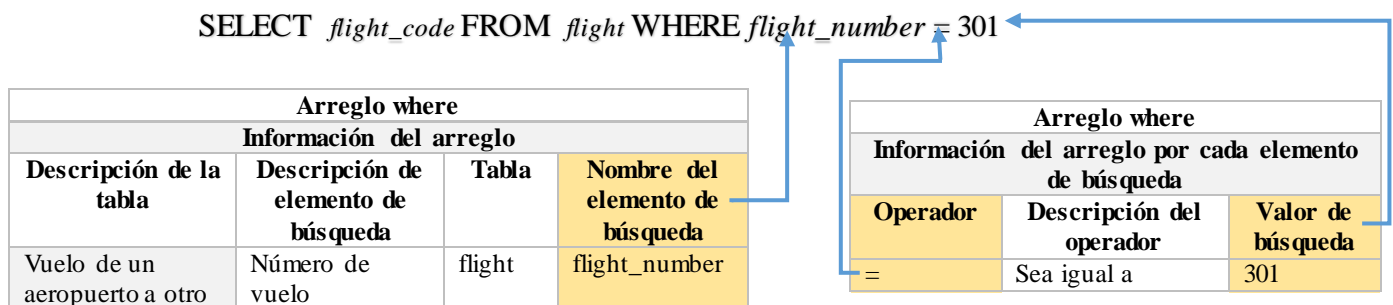


Figura 5.10. Construcción de la cláusula *Where*

Para construir la consulta más externa, el usuario tiene que elegir un tema de interés y posteriormente tiene que elegir el o los elementos de los cuales desea obtener información para que finalmente termine la composición de la consulta con la definición de la condición de búsqueda para la subconsulta.

La manera en que la interfaz construye las cláusulas *Select* y *From* para la consulta más externa es generando arreglos como el que se muestra en la Figura 5.9. Cada uno de dichos arreglos corresponde a cada elemento que el usuario seleccionó del árbol de composición en la quinta fase (*Selección de elementos de interés*).



Para las consultas en las cuales la interfaz tiene que generar las subconsultas intermedias necesarias para completar la composición de la consulta, no es necesario generar el arreglo para la cláusula Where como se muestra en la Figura 5.10, a menos que el usuario especifique más condiciones de búsqueda aparte de la condición de búsqueda que se especifica a la subconsulta. Para tal efecto, el algoritmo presentado en la sección 4.5 genera las subconsultas necesarias para la composición y éstas se asignan a la cláusula Where al momento de construir la consulta. En caso contrario, se genera la cláusula Where como se muestra en la Figura 5.10 y además, se añade el conector AND seguido de las subconsultas generadas automáticamente por la interfaz.

## 5.9 Vista previa y resultado

En la ventana que se muestra en la Figura 5.11 se presenta el resultado de la consulta o subconsulta compuesta por el usuario y su correcta traducción a SQL. En la Tabla 5.8 se muestra la descripción de los componentes que integran esta ventana, donde el número que se encuentra en la columna izquierda de la tabla, corresponde al número que identifica el componente en la Figura 5.11.

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

<i>Componente</i>	<i>Descripción</i>
1. <i>Condiciones de búsqueda.</i>	En esta caja se encuentran las condiciones de búsqueda definidas por el usuario para la composición de la consulta.
2. <i>Casilla de verificación.</i>	En esta casilla el usuario puede hacer clic para ver la traducción de la consulta a SQL.
3. <i>Traducción.</i>	Esta caja de texto muestra al usuario la traducción a SQL de la consulta compuesta.
4. <i>Tabla de resultado.</i>	En esta tabla el usuario puede observar el resultado devuelto por la base de datos después de haber realizado la composición.
5. <i>Botón “Atrás”.</i>	Este botón le permite al usuario regresar una ventana anterior si es necesario corregir algo de la composición.
6. <i>Botón “Finalizar”.</i>	Este botón ( <i>Finalizar subconsulta</i> o <i>Finalizar consulta</i> ) permite al usuario concluir la composición de la consulta, y en caso de que el botón se llame <i>Finalizar subconsulta</i> , la interfaz continuará guiando al usuario con la composición de las demás subconsultas o en caso contrario, con la consulta principal.

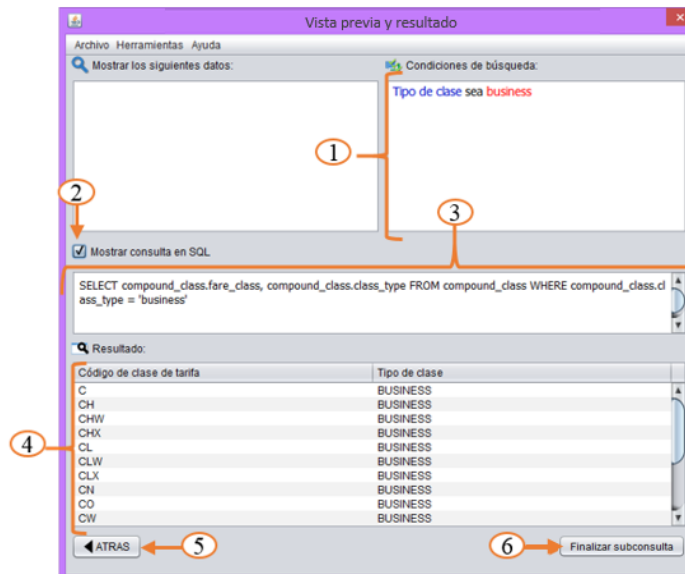


Figura 5.11. Vista previa y resultado

Después de que el usuario termina la composición de una subconsulta o la consulta más externa, la interfaz procesa la consulta a través del sistema manejador de bases de datos, y el resultado es mostrado al usuario a través de esta ventana junto con su traducción a SQL.

## 5.10 Proceso de composición de consultas que involucran subconsultas

Para describir el proceso de la composición de consultas que involucran subconsultas, se considera el siguiente ejemplo:

### *¿Cuáles aerolíneas tienen vuelos con clase de tipo business?*

El primer paso es *Descomposición de consultas en LN*. En la ventana correspondiente (ver Figura 5.12) se tiene que realizar la descomposición de la consulta que se encuentra en lenguaje natural a una consulta del mismo tipo pero que involucre dos subconsultas. Para esto es necesario efectuar los siguientes pasos:

1. El usuario tiene que escribir dentro de la caja de texto la consulta que desea componer (ver No. 1 en la figura).
2. Después el usuario debe identificar la condición de búsqueda de la consulta original y escribirla en el cuadro de texto (No. 2 en la figura).
3. Además, el usuario analiza si existe una condición que dependa de la subconsulta anterior, de ser así el usuario selecciona de la lista que se muestra en la Figura 5.12 el nivel de subconsultas, para este ejemplo selecciona la opción *Dos subconsultas* y se escribe la condición dependiente en el cuadro de texto (No. 3 en la figura).

4. Se identifica el o los elementos que exigen información para formar la consulta principal y se escribe en el cuadro de texto (No. 4 en la figura).
5. Por último el usuario oprime el botón “Continuar” para terminar la descomposición (No. 5 en la figura).

Figura 5.12. Descomposición de consultas

En la siguiente ventana *Selección del tema de interés* (Figura 5.13), el usuario elige el tema del cual desea obtener información para la subconsulta. Para esto se llevan a cabo los siguientes tres pasos:

1. El usuario observará que la interfaz le muestra en color azul la subconsulta a componer: *clase de tipo business*.
2. La interfaz le muestra al usuario en una tabla unidimensional la descripción de las tablas que se encuentran en la base de datos y que son mostradas al usuario como temas a elegir. De éstos solamente debe elegir uno del cual desea obtener información para realizar la subconsulta indicada en el punto (1). Para este ejemplo el usuario selecciona el tema *Clase compuesta*.
3. Por último el usuario oprime el botón “Continuar” para terminar esta fase.

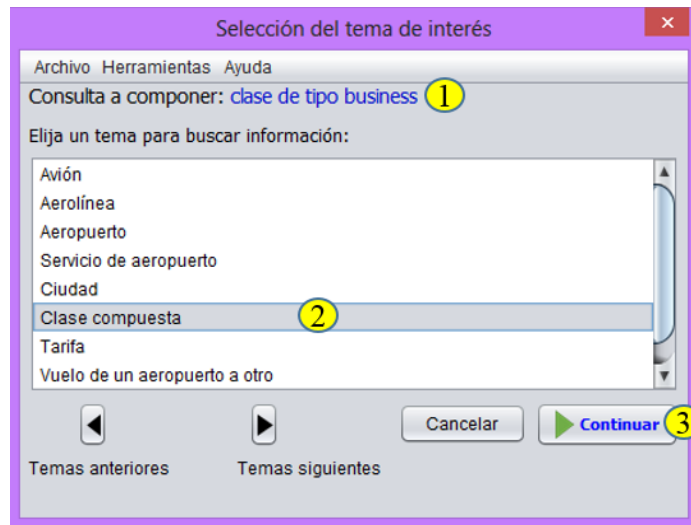


Figura 5.13. Selección del tema de interés en la composición de la subconsulta más interna

A continuación en la pantalla *Especificación de condiciones de búsqueda* (Figura 5.14) el usuario define las condiciones de búsqueda que restringen la información en la consulta. Para este ejemplo se requiere que se especifique una condición de búsqueda: que la clase sea de tipo *business*.

1. El usuario inicia seleccionando el elemento *Tipo de clase* que se encuentra en el árbol de composición.
2. Posteriormente, la interfaz le muestra al usuario en una lista las condiciones de búsqueda que puede elegir para ese elemento seleccionado con anterioridad. El usuario selecciona la condición *sea*.
3. Después el usuario debe definir la característica de búsqueda escribiendo en la caja de texto la cadena *business*.
4. Después de haber definido la condición de búsqueda, el usuario debe añadir la condición mediante el botón *Añadir condición* a la caja *Condiciones para restringir la búsqueda* que se muestra en la Figura 5.14.
5. Finalmente, el usuario debe oprimir el botón *Continuar* para proceder a la ventana *Vista previa y resultado*.

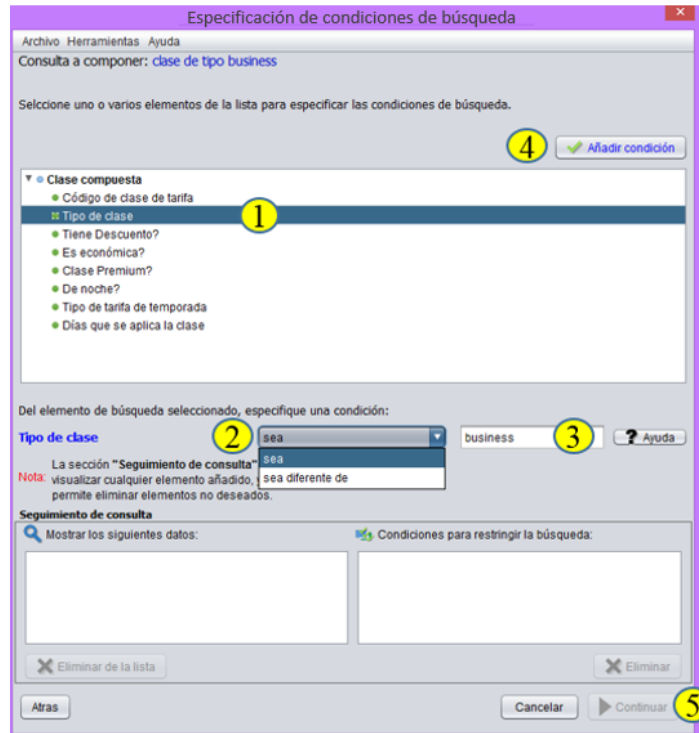


Figura 5.14. Especificación de condiciones de búsqueda en la composición de la subconsulta más interna

En la ventana que se muestra en la Figura 5.15, se mostrará el resultado de la consulta compuesta y su traducción a SQL. Después de que el usuario realizó la composición de la subconsulta, la interfaz procede a procesarla mediante el sistema manejador de la bases de datos para finalmente mostrarle el resultado devuelto de la subconsulta compuesta. Los siguientes dos pasos describen lo que el usuario procedería a realizar para este ejemplo:

1. Primeramente el usuario tiene la opción de ver o no ver la traducción de la composición de la consulta a SQL. Para este ejemplo el usuario debe hacer clic sobre la casilla de verificación (ver No. 1 en la figura).
2. Después de ver los resultados devueltos por la interfaz, el usuario debe dar por terminada la composición de la subconsulta oprimiendo el botón *Finalizar subconsulta* para seguir con la composición de la subconsulta intermedia.

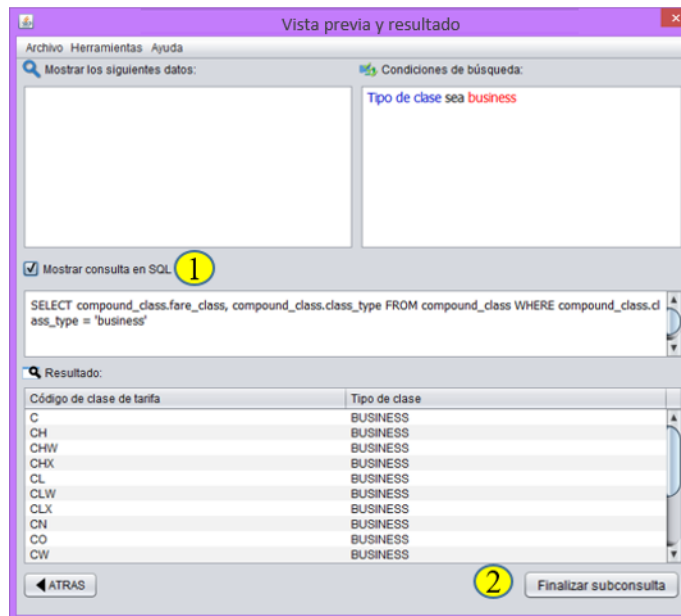


Figura 5.15. Resultado y vista previa en la composición de la subconsulta más interna

En la siguiente ventana *Selección del tema de interés*, (Figura 5.16) el usuario elige el tema del cual desea obtener información para la subconsulta intermedia. Para esto se llevan a cabo los siguientes tres pasos:

1. El usuario observará en color azul la subconsulta a componer: *vuelos con clase*.
2. El usuario debe elegir el tema del cual desea obtener información para realizar la subconsulta indicada en el punto (1). Para este ejemplo, debe seleccionar el tema *Vuelo de un aeropuerto a otro*.
3. Y por último el usuario debe oprimir el botón *Continuar* para terminar esta fase.

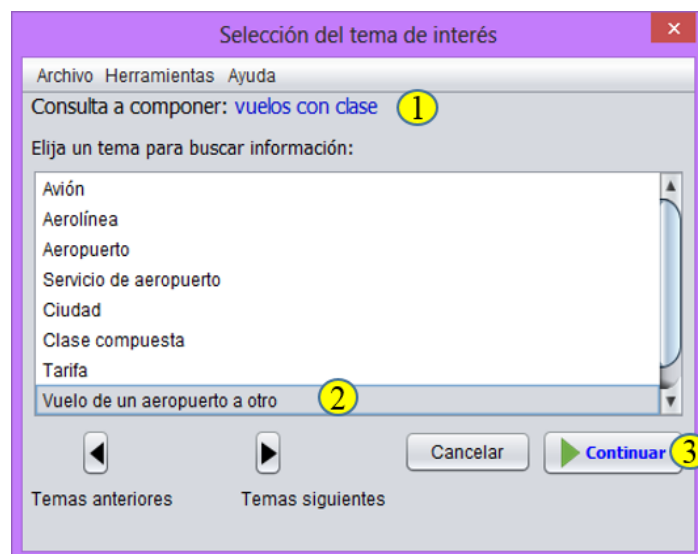


Figura 5.16. Selección del tema de interés para la subconsulta intermedia

Después de haber elegido un tema de interés, la interfaz detecta que es necesario generar una subconsulta antes de pasar a definir la condición de búsqueda para la subconsulta. Finalmente, después de generar dicha subconsulta faltante, ésta es complementada con la subconsulta compuesta con anterioridad.

En la ventana *Resultado de subconsulta interna* (Figura 5.17), se muestra el resultado devuelto por la o las subconsultas generadas internamente por la interfaz, en la cual el usuario simplemente puede observar el resultado que tenga coherencia con lo que busca, y después debe oprimir el botón *Continuar* para pasar a definir la condición de búsqueda en la siguiente ventana.

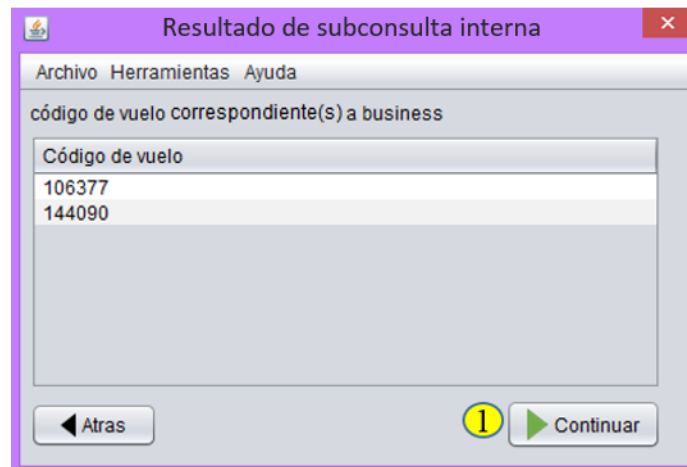


Figura 5.17. Resultado de subconsulta interna para la subconsulta intermedia

Finalmente, el usuario se encuentra con la ventana *Especificación de condición para subconsulta* (Figura 5.18), en la cual definirá la condición de búsqueda para la subconsulta. Para este ejemplo se requiere que se especifique una condición de búsqueda: que el vuelo *sea igual a alguno(s) de* los resultados obtenidos en la subconsulta. Para tal efecto, el usuario debe efectuar los siguientes dos pasos:

1. El usuario debe seleccionar de lista que se muestra en la Figura 5.18, solamente una de las dos condiciones que corresponden a los operadores *IN* y *NOT IN* de SQL. Para este ejemplo el usuario selecciona de la lista *sea igual a alguno(s) de*.
2. Finalmente, el usuario debe oprimir el botón *Continuar*. A continuación el sistema le preguntará si desea definir más condiciones de búsqueda aparte de la definida para la subconsulta. Para este ejemplo el usuario hace clic en la opción *No* para proceder a la ventana *Vista previa y resultado*.

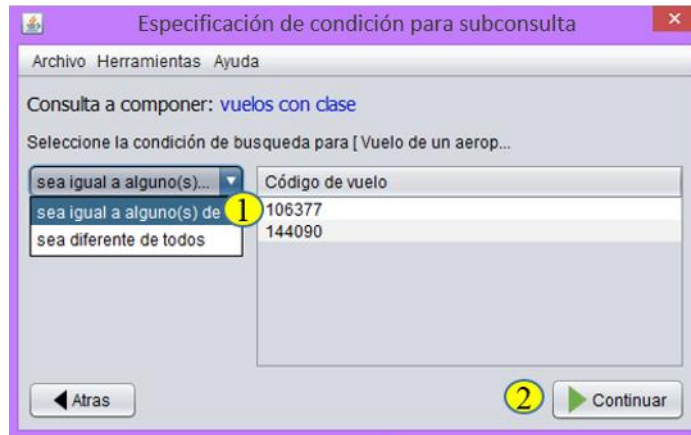


Figura 5.18. Especificación de condición de búsqueda para subconsultas (subconsulta intermedia)

En la ventana que se muestra en la Figura 5.19, se presentará el resultado de la consulta compuesta y su traducción a SQL. Después de que el usuario haya hecho la composición de la subconsulta intermedia, la interfaz procede a procesarla mediante el sistema manejador de bases de datos para finalmente mostrarle el resultado devuelto de la subconsulta compuesta. Los siguientes dos pasos describen lo que el usuario procedería a realizar para este ejemplo:

1. Primeramente, el usuario tiene la opción de ver o no ver la traducción de la composición de la consulta a SQL; de ser así, el usuario debe hacer clic sobre la casilla de verificación (ver No. 1 en la figura).
2. Después de ver los resultados devueltos por la interfaz, el usuario debe dar por terminada la composición de la subconsulta oprimiendo el botón *Finalizar subconsulta* para seguir con la composición de la consulta externa.

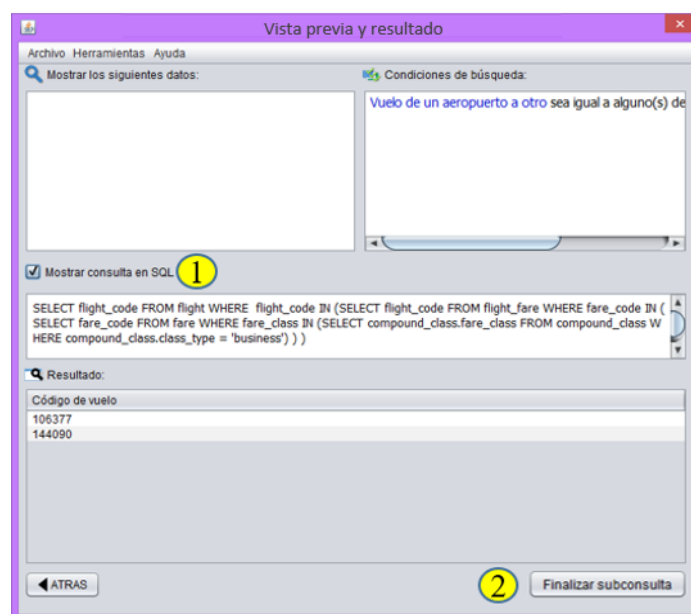


Figura 5.19. Vista previa y resultado de subconsulta intermedia



En la siguiente ventana *Selección del tema de interés* (Figura 5.20), el usuario debe elegir el tema del cual desea obtener información para la consulta externa. Para esto se llevan a cabo los siguientes tres pasos:

1. El usuario observará en color azul la consulta que debe componer: *Cuáles aerolíneas tienen vuelos*.
2. El usuario debe elegir el tema del cual desea obtener información para realizar la consulta indicada en el punto (1). Para este ejemplo debe seleccionar el tema *Aerolínea*.
3. Y por último, el usuario debe oprimir el botón *Continuar* para terminar esta fase.

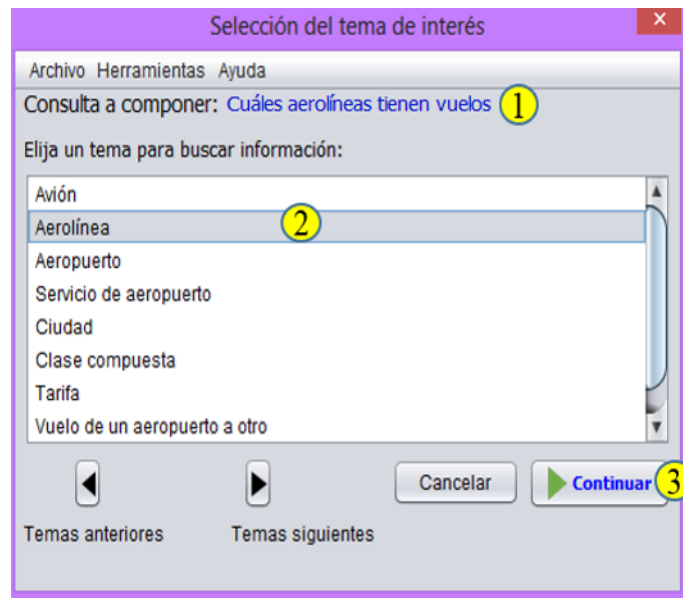


Figura 5.20. Selección del tema de interés para la consulta externa

En la ventana, que se muestra en la Figura 5.21, el usuario debe elegir el o los elementos del árbol de composición de los cuales desea obtener información. Para este ejemplo el usuario solamente requiere elegir un elemento: *Nombre de la aerolínea*. Para tal efecto, debe seguir los siguientes pasos:

1. El usuario debe seleccionar *Nombre de la aerolínea* del árbol de composición.
2. Después de haber seleccionado el elemento de interés, el usuario debe añadir dicho elemento a la caja *Mostrar los siguientes datos* mediante el botón *Añadir elemento*.
3. Si el usuario deseara buscar información de otros elementos, simplemente tendría que realizar los dos pasos anteriores. Para este ejemplo el usuario terminaría esta fase oprimiendo el botón *Continuar* para pasar a definir la condición de búsqueda para la subconsulta.

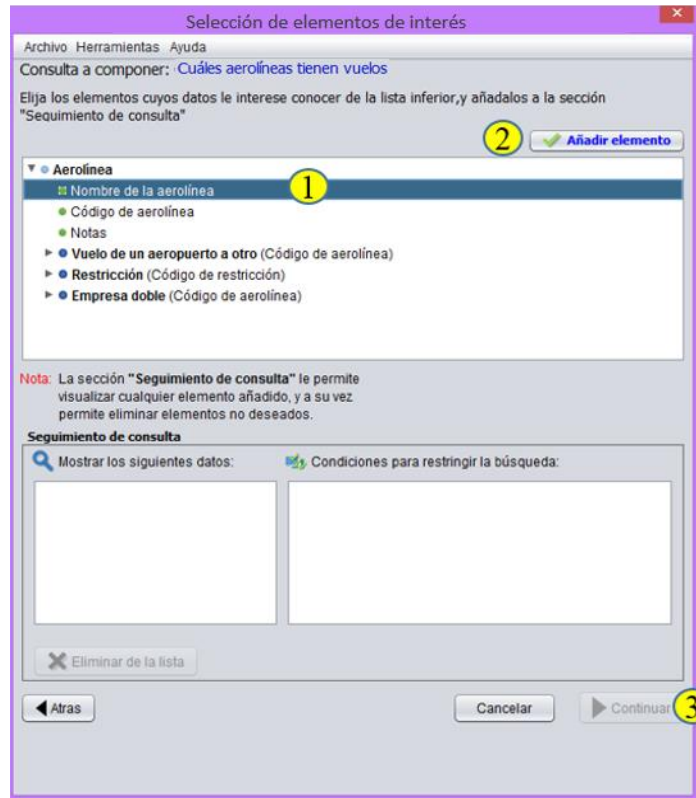


Figura 5.21. Selección de elementos a mostrar para la consulta externa

Finalmente el usuario se encuentra con la ventana *Especificación de condición para subconsulta* (Figura 5.22), en la cual definirá la condición de búsqueda para la subconsulta. Para este ejemplo se requiere que se especifique una condición de búsqueda: que el vuelo *sea igual a alguno(s) de* los obtenidas en la subconsulta. Para tal efecto, el usuario debe efectuar los siguientes dos pasos:

1. El usuario debe seleccionar de la lista que se muestra en la Figura 5.22, solamente una de las dos condiciones, las cuales corresponden a los operadores *IN* y *NOT IN* de SQL. Para este ejemplo el usuario selecciona de la lista *sea igual a alguno(s) de*.
2. Finalmente, el usuario debe oprimir el botón *Continuar*, y el sistema le preguntará si desea definir más condiciones de búsqueda aparte de la definida para la subconsulta. Para este ejemplo el usuario hace clic en la opción *No* para proceder a la ventana *Vista previa y resultado*.

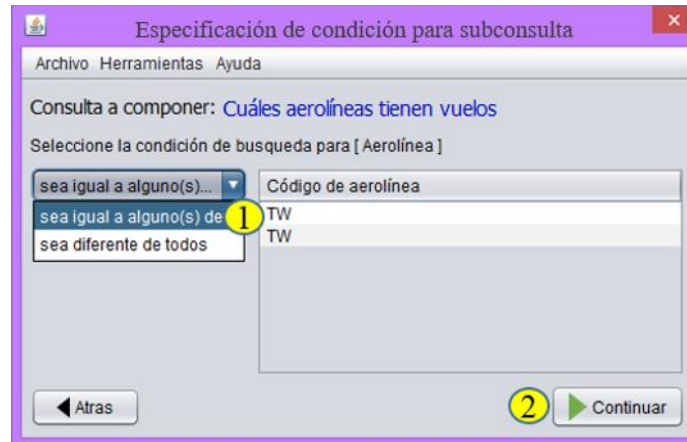


Figura 5.22. Definición de condición de búsqueda para subconsulta (consulta externa)

Finalmente, como último paso para la composición de la consulta, en la ventana que se muestra en la Figura 5.23 se mostrará el resultado de la consulta compuesta y su traducción a SQL. Después de que el usuario haya hecho la composición de la consulta externa, la interfaz procede a procesarla mediante el sistema manejador de la bases de datos para finalmente mostrarle el resultado devuelto de la consulta. Los siguientes dos pasos describen lo que el usuario procedería a realizar para este ejemplo:

1. Primeramente, el usuario tiene la opción de ver o no ver la traducción de la composición de la consulta a SQL; de ser así, el usuario debe hacer clic sobre la casilla de verificación (ver No. 1 en la figura).
2. Después de ver los resultados devueltos por la interfaz, el usuario debe dar por terminada la composición de la consulta oprimiendo el botón *Finalizar*.

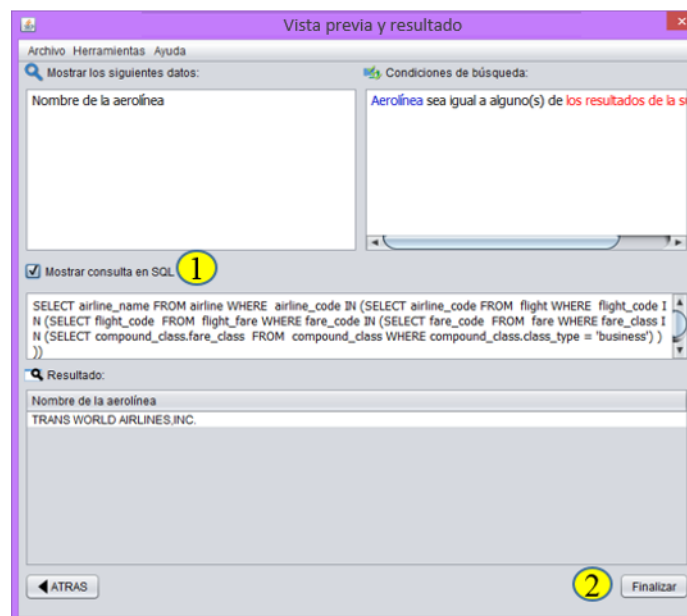


Figura 5.23. Vista previa y resultado de la consulta

# Capítulo 6

## Pruebas experimentales

---

### 6.1 Pruebas de usabilidad

Estas pruebas, que se describen a continuación, fueron realizadas por estudiantes de ingeniería en sistemas computacionales con el propósito de conocer qué tan amigable es la interfaz de composición con la nueva funcionalidad de subconsultas.

El corpus utilizado para estas pruebas, se presenta en el apéndice 2. Dicho corpus cuenta con 20 consultas divididas en dos grupos de 10, de los cuales el primer grupo consta de consultas con 1 o 2 subconsultas en su traducción a SQL; mientras que el segundo grupo cuenta con consultas que involucran 3 o 4. Ambos grupos se presentan ordenados de menor a mayor dificultad de composición.

#### 6.1.1 Objetivo de las pruebas

El objetivo de las pruebas de usabilidad es medir la facilidad de uso de la interfaz de composición con la nueva funcionalidad de subconsultas, y con ello saber qué tan amigable es para componer consultas que involucran subconsultas. El propósito final, fuera del alcance de este proyecto, es que con esta funcionalidad y la adición de las funciones de agregación en un futuro cercano, se espera que los usuarios puedan acceder a información de BDs a la cual actualmente aún no es posible acceder con la versión actual de la interfaz.

Las características que se midieron en estas pruebas de composición de consultas que involucran desde 1 a 4 subconsultas en su traducción a SQL, son las siguientes:

- Tiempo total que tarda un usuario en componer una consulta que involucra subconsultas.
- Cantidad de intentos que realiza un usuario para componer una consulta correcta o incorrectamente.

#### 6.1.2 Descripción del ambiente de pruebas

Estas pruebas fueron llevadas a cabo en una sola sesión de 2 horas con un grupo de 14 estudiantes de ingeniería en sistemas computacionales, a los cuales se les proporcionó un manual de usuario con dos días de anticipación al día de las pruebas. Dicho manual detalla los componentes de las ventanas que integran la interfaz de composición así como un ejemplo para la composición de cada tipo de consulta que se menciona en la sección 4.2.

Al inicio de las pruebas también se les explicaron a los usuarios, los pasos para llevar a cabo la descomposición de consultas que se encuentran en LN a consultas del mismo tipo

identificando las subconsultas y la consulta externa. La descripción de estos pasos se encuentra en la sección 4.3.

Al iniciar las pruebas se les proporcionó a los usuarios un conjunto de 20 consultas en LN que en su traducción a SQL involucran subconsultas. Estas consultas se encuentran ordenadas de menor a mayor dificultad de composición. Además, se encuentran divididas en dos grupos de 10. El primer grupo consiste de consultas que involucran 1 o 2 subconsultas en su traducción a SQL, y el segundo grupo cuenta con consultas que involucran 3 o 4 subconsultas.

Las consultas mencionadas anteriormente fueron seleccionadas del corpus de ATIS, el cual se encuentra en [https://catalog.ldc.upenn.edu/docs/LDC93S4B-3/TRN\\_PRMP.TXT](https://catalog.ldc.upenn.edu/docs/LDC93S4B-3/TRN_PRMP.TXT). Algunas consultas fueron modificadas quitando las funciones de agregación para evitar algunas confusiones para el usuario al momento de componer la consulta, ya que esta funcionalidad de subconsultas no abarca las funciones de agregación, como se menciona en la sección 1.2. Es importante destacar que a los usuarios no se les proporcionó el esquema de la base de datos ni la traducción correcta de las consultas de prueba, solamente se les proporcionaron las consultas y el valor del resultado final para cada una de ellas.

Para obtener los resultados obtenidos de las consultas compuestas por los usuarios, se desarrolló un software que fue integrado a la interfaz de composición para monitorear y almacenar en una base de datos la composición llevada a cabo por los usuarios. El proceso del monitoreo de esta interfaz se describe en la Figura 6.1.

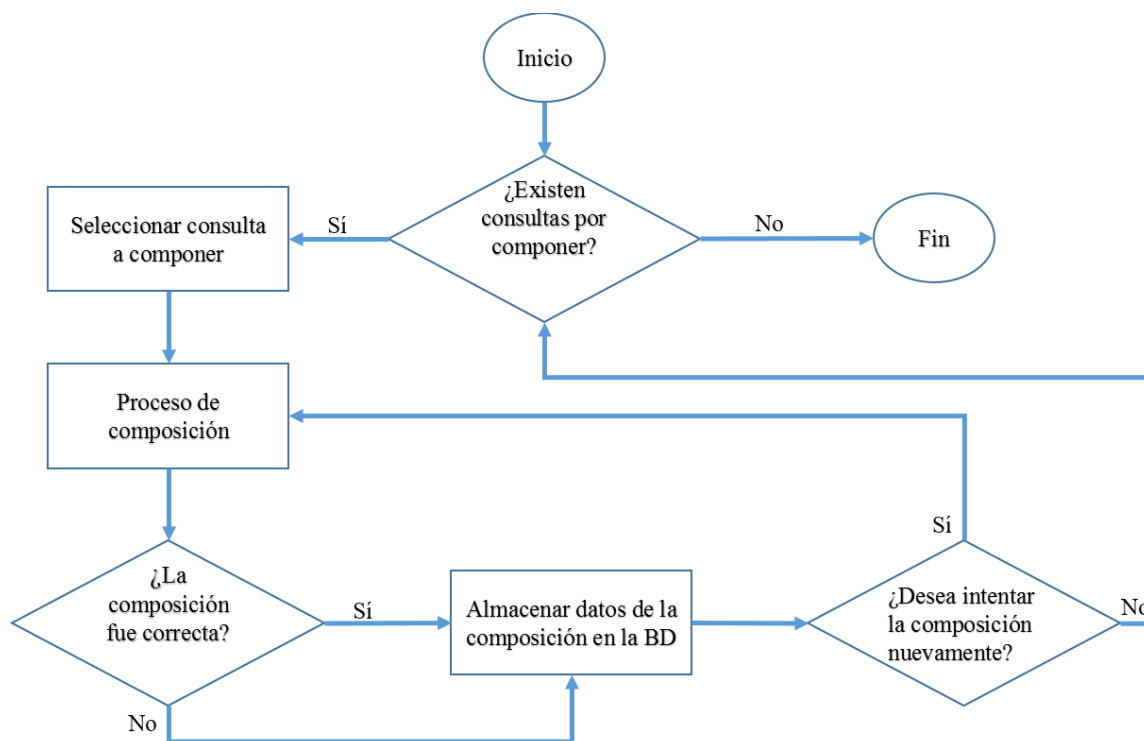


Figura 6.1. Proceso de monitoreo para la composición de consultas

Este proceso descrito en la Figura 6.1 inicia mostrando una lista con las consultas que el usuario aún no ha compuesto. Por lo tanto, si existen consultas por componer, el usuario selecciona

la consulta que desee componer y la interfaz muestra las ventanas correspondientes para la composición de consultas que involucran subconsultas. Finalmente, después de haber hecho la composición, la interfaz revisa si la consulta fue compuesta correctamente y almacena los datos necesarios de la composición en la BD de pruebas. En caso de no haber compuesto la consulta correctamente, la interfaz supervisora procede a preguntar al usuario si desea componer nuevamente la consulta. En caso de que el usuario elija *Sí*, entonces le vuelve a mostrar las ventanas correspondientes para la composición. En caso de haber elegido la opción *No*, la interfaz procede a mostrarle la lista de consultas que faltan por componer. En caso de no haber más consultas por componer, el supervisor termina las pruebas.

### 6.1.3 Resultados de las pruebas de usabilidad

En la Tabla 6.1 se presentan los resultados de las consultas que involucran subconsultas que los usuarios compusieron usando la interfaz. Estos resultados se obtuvieron de 20 consultas compuestas por 14 usuarios, y solamente se muestran resultados de las consultas que fueron compuestas correctamente.

Tabla 6.1. Resultado de la composición de consultas

No de consulta	Mínimo de intentos	Máximo de intentos	Promedio de intentos	Tiempo mínimo (seg)	Tiempo máximo (seg)	Tiempo promedio (min)	No. de usuarios que compusieron correctamente
1	1	2	1.25	117	1061	8.66	12
2	1	2	1.28	52	505	3.37	14
3	1	3	1.21	70	702	2.92	14
4	1	7	2.42	76	1272	6.05	14
5	1	2	1.07	52	231	1.82	14
6	1	2	1.14	70	359	2.84	14
7	1	2	1.08	59	249	2.55	12
8	2	2	2	400	400	6.66	1
9	1	6	2.57	50	1152	7.35	7
10	1	2	1.08	72	287	2.97	12
11	1	2	1.16	48	369	2.32	12
12	1	1	1	45	146	1.6	11
13	1	3	1.16	51	376	1.73	12
14	1	1	1	45	157	1.34	12
15	1	8	4.6	119	886	9.97	5
16	1	2	1.5	56	220	2.68	4
17	1	3	1.33	78	285	2.29	9
18	1	5	1.71	71	293	2.35	7
19	1	3	1.5	84	358	2.8	6
20	1	1	1	62	128	1.49	6
<b>Promedio</b>			1.55			3.68	

En la Tabla 6.1 se muestra por cada consulta compuesta correctamente lo siguiente: primeramente se presenta el mínimo de intentos para componer la consulta correctamente, y después en la siguiente columna se muestra el máximo de intentos. En la cuarta columna se indica el promedio de intentos por consulta. Enseguida se encuentra el mínimo de tiempo en segundos que tardaron los usuarios en componer la consulta, y después en la siguiente columna el máximo de tiempo en segundos. Posteriormente, se muestra un promedio del tiempo que les toma a los usuarios componer la consulta correctamente, y finalmente se muestra la cantidad de usuarios que compusieron correctamente la consulta.

De los resultados obtenidos se puede apreciar que realizan 1.5 intentos en promedio para la composición de consultas. Resalta el hecho de que en la primera consulta tardan un promedio de 8 minutos, ya que los usuarios se enfrentan por primera vez con la interfaz de composición. Por lo tanto, esta primera consulta le permite al usuario aprender el uso de la interfaz.

También se puede apreciar que, para la consulta número 8, solamente un usuario fue capaz de componerla. Esto se debe a la complejidad del esquema de la base de datos. La consulta es la siguiente:

*¿Cuál es el nombre del aeropuerto de destino del vuelo número 140?*

En esta consulta se le pide al usuario obtener el nombre del *Aeropuerto de destino*; por lo tanto, el usuario trata de encontrar este elemento en el tema *Aeropuerto*, pero dicho elemento se encuentra en el tema relacionado *Vuelo de un aeropuerto a otro*. El elemento en dicho tema relacionado se llama *Aeropuerto de destino*. Como los usuarios no encontraron este elemento en el tema *Aeropuerto*, a la gran mayoría de éstos no se les ocurrió que el elemento podría estar en otro tema; y por lo tanto, no pudieron componer la consulta a excepción de un usuario.

Dicho lo anterior, el primer grupo de 10 consultas (aquéllas que involucran una o dos subconsultas en su traducción a SQL) fueron respondidas por la mayoría de los usuarios en un tiempo aproximado de 3 a 4 minutos en un número aproximado de 1.5 intentos.

En el siguiente grupo de 10 consultas, no hubo ningún problema para componer las primeras cinco consultas. Sin embargo, las últimas cinco consultas son las más complicadas, ya que éstas requieren que el usuario las descomponga en consultas que involucren dos subconsultas en LN.

Como se puede observar, la mayoría de las consultas pudieron ser respondidas por la gran mayoría de los usuarios, a excepción de las consultas 8, 15, 16, 19 y 20. La consulta 8 resultó ser la más complicada debido a la problemática descrita anteriormente. Las demás consultas que no fueron respondidas por la mayoría de los usuarios son debido a que en su descomposición involucran dos subconsultas en LN.

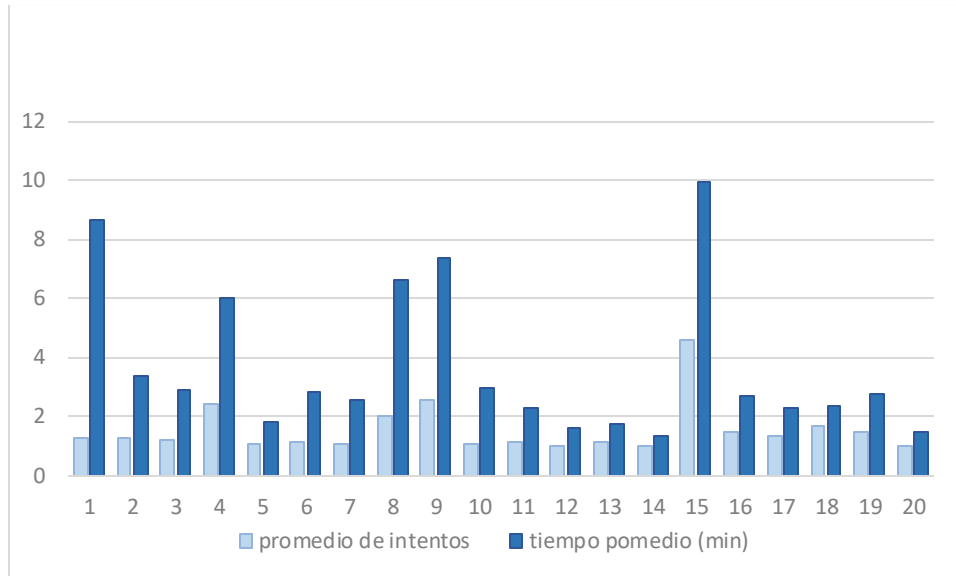


Figura 6.2. Promedio de intentos y tiempo en minutos para cada consulta

#### 6.1.4 Comparación de resultados

En la Figura 6.3 podemos apreciar los resultados de las pruebas realizadas con la interfaz de composición desarrollada por [Aguirre 2015], en donde se puede apreciar el promedio de intentos y el promedio de tiempo por consulta.

En la Figura 6.2 se pueden apreciar los resultados de las pruebas que se realizaron con la nueva versión de la interfaz de composición, de igual manera mostrando el promedio de intentos y el promedio de tiempo por consulta. Al comparar los resultados de las dos graficas mencionadas anteriormente, podemos observar que en la versión actual de la interfaz de composición desarrollada en este proyecto de tesis, al usuario le toma aproximadamente dos intentos para realizar la composición de las consultas al igual que en la versión anterior de la interfaz.

Por otro lado, el tiempo que tarda el usuario en llevar a cabo la composición de las consultas en esta nueva versión de la interfaz, es aproximadamente el doble o el triple que en la versión anterior, debido a que el proceso de composición de consultas que involucran subconsultas es más largo. Esto se debe a que la consulta que se encuentra en LN se descompone de tal manera que el usuario realiza la composición de la consulta por partes, evitando con esto que la interfaz deje de ser amigable para los usuarios inexpertos.

Finalmente, es importante destacar que las consultas que se utilizaron en este proyecto de tesis para medir la usabilidad son diferentes y más complicadas de componer, en comparación a las consultas utilizadas en las pruebas de la versión anterior.



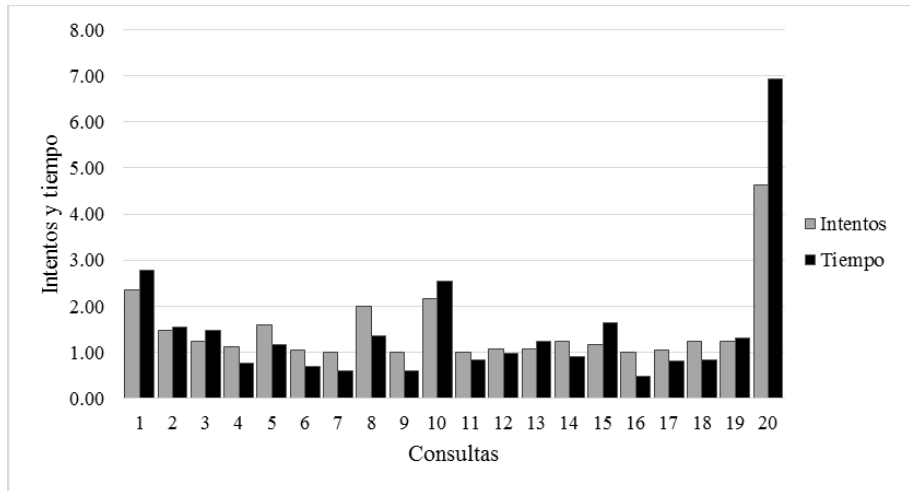


Figura 6.3. Resultado de pruebas realizadas con la versión anterior de la interfaz de composición [Aguirre 2015]

### 6.1.5 Conclusiones de las pruebas de usabilidad

Las pruebas presentadas para esta interfaz demuestran que los usuarios pueden componer consultas que involucran subconsultas. El objetivo de este trabajo está enfocado a los usuarios inexpertos, aquéllos que no dominan un lenguaje de consulta como SQL. Por lo tanto, no se espera que deseen realizar la composición de consultas que involucren más de tres subconsultas, aunque la interfaz está diseñada para que cualquier usuario pueda componer consultas con hasta cuatro subconsultas en su traducción a SQL.

El resultado de la composición de las consultas muestra que la mayoría de los usuarios pudieron realizar la composición de 15 consultas correctamente, dejando 5 consultas, las que la mayoría de los usuarios no pudieron responder. En el caso de estas 5 últimas consultas, la explicación del resultado observado es que son las más complicadas, debido a que involucran dos subconsultas en LN.

Conocer la opinión de los usuarios al final de las pruebas es un aspecto importante, ya que ayuda a mejorar o identificar qué parte de la interfaz puede ser mejorada. Por lo tanto, al final se les pidió a los usuarios responder cinco preguntas, entre ellas la siguiente pregunta que fue en la que la mayoría de los usuarios concordaron en su respuesta: ¿Qué fase de la composición de consultas se le dificulta y por qué?

La mayoría de los usuarios concordaron en que la búsqueda de los elementos dentro del árbol de composición se les dificultaba, ya que nunca habían visto qué es lo que contenía cada tema. Por lo tanto, ellos creían que un elemento se encontraba en un tema determinado pero no era así, ese elemento se encontraba en otro tema, como lo ocurrido con la consulta número 8 que se describió anteriormente. Esto provoca que los usuarios tarden más tiempo en acoplarse a la interfaz y les lleve más tiempo realizar la composición de las consultas.

## 6.2 Pruebas funcionales

Es importante demostrar que la interfaz funciona para los tres tipos de consultas que involucran subconsultas. Para tal efecto, se realizaron pruebas funcionales con los tres tipos de consultas que se mencionan en la subsección 4.2. En la siguiente subsección se describen dichas pruebas.

### 6.2.1 Descripción de pruebas funcionales

Las pruebas que se describen a continuación se efectuaron con el fin de demostrar que es posible realizar la composición de consultas que involucran una, dos y tres subconsultas. Se diseñaron tres consultas de cada tipo y se realizó la composición mediante la interfaz.

#### **Consulta tipo 1 (consultas que involucran una subconsulta):**

*Dame la clase de comida del vuelo 296*

Primeramente, en la interfaz, se descompone la consulta que se encuentra en LN en una consulta que involucre una subconsulta, identificando la condición de búsqueda y los elementos de interés. Entonces, la consulta queda de la siguiente manera:

Subconsulta: *vuelo 296*.  
Consulta externa: *Dame la clase de comida del vuelo*.

Primeramente, es necesario realizar la composición de la subconsulta. Para tal efecto, simplemente se elige el tema *Vuelo de un aeropuerto a otro*, en el cual existe el elemento *Número de vuelo* dentro del árbol de composición. Para dicho elemento se especifica que el número de vuelo es el 296, y se pasa a obtener un resultado perteneciente a la subconsulta, y su traducción a SQL, para lo cual se genera la siguiente expresión para la subconsulta:

```
SELECT flight.flight_code, flight.flight_number
FROM flight
WHERE flight.flight_number = 296
```

Continuando con la composición, se procede a elegir los elementos de interés para la consulta externa. Para este ejemplo solamente es el elemento *Clase de comida* que se encuentra en el tema *Servicio de comida*. Posteriormente, se procede a definir la condición de búsqueda para la subconsulta, y para este ejemplo se elige *sea igual a alguno (s) de*. Finalmente, se continúa a ver el resultado de la consulta completa y su traducción a SQL.

```
SELECT meal_class
FROM food_service
WHERE meal_code IN
  (SELECT meal_code
   FROM flight
   WHERE flight.flight_number = 296)
```

## Consulta tipo 2 (consulta que involucra dos subconsultas):

*Dame los números de vuelo con tarifa de viaje redondo igual a 696*

Primeramente, se comienza descomponiendo la consulta que se encuentra en LN en una consulta que involucre una subconsulta, quedando de la siguiente manera:

Subconsulta: *tarifa de viaje redondo igual a 696.*  
Consulta externa: *Dame los números de vuelo con tarifa.*

Después se realiza la composición de la subconsulta, en la cual solamente hay que indicar la condición de búsqueda. Para este ejemplo se selecciona del árbol de composición el elemento *Tarifa de viaje redondo* que pertenece al tema *Tarifa*, y se especifica el valor de búsqueda *696*. Enseguida la interfaz muestra el resultado de la subconsulta para la cual su traducción a SQL es la siguiente:

```
SELECT fare.fare_code, fare.rnd_trip_cost
FROM fare
WHERE fare.rnd_trip_cost = 696
```

Después se continúa con la selección de los elementos de interés, donde para este ejemplo, solamente se piden los números de vuelos pertenecientes a los valores del resultado de la subconsulta. Para tal efecto, se elige *Número de vuelo* que se encuentra en el tema *Vuelo de un aeropuerto a otro*. Como la tabla de este tema es *flight*, la interfaz se da cuenta que es necesaria la construcción de una subconsulta extra, ya que existe una tabla intermedia entre la tabla de la subconsulta (*fare*) y la tabla de la consulta externa (*flight*). Por lo tanto, la interfaz genera automáticamente la subconsulta faltante, y los resultados de ésta deben ser asociados a la consulta externa para lo cual se debe definir la condición de búsqueda. Para este ejemplo se define que *sea igual a alguno (s) de*. Finalmente se obtiene el resultado de la consulta completa y la consulta traducida a SQL.

```
SELECT flight_number
FROM flight
WHERE flight_code IN
  (SELECT flight_code
   FROM flight_fare
   WHERE fare_code IN
     (SELECT fare.fare_code
      FROM fare
      WHERE fare.rnd_trip_cost = 696))
```

## Consulta tipo 3 (consulta que involucra tres subconsultas)

*Dame la zona horaria de donde se encuentra el aeropuerto LOGAN INTERNATIONAL*

Primeramente, se tiene que descomponer la consulta que se encuentra en LN a una consulta que involucre una subconsulta como se muestra a continuación:

Subconsulta: *aeropuerto LOGAN INTERNATIONAL*.

Consulta externa: *Dame la zona horaria de donde se encuentra el aeropuerto.*

Después es necesario comenzar la composición de la subconsulta. Para esto simplemente tiene que definirse la condición de búsqueda para la subconsulta. Para este ejemplo se inicia seleccionando el tema de interés *Aeropuerto*, y posteriormente, del árbol de composición generado a partir de este tema, se selecciona *Nombre de aeropuerto*, y se define el valor de búsqueda *LOGAN INTERNATIONAL*. Finalmente, se procede a ver el resultado y la traducción de esta subconsulta a SQL:

```
SELECT airport.airport_code, airport.airport_name
FROM airport
WHERE airport.airport_name = 'LOGAN INTERNATIONAL'
```

Después de haber realizado la composición de la subconsulta, se continúa con el proceso de composición de la consulta externa. Para tal efecto, es necesario elegir los elementos de interés de los cuales se desea conocer la información. Para este ejemplo se elige primeramente el tema *Zona horaria*, del cual se genera un árbol de composición y de allí se selecciona el elemento de interés *Nombre de zona horaria*. Posteriormente, la interfaz genera dos subconsultas extras necesarias para completar correctamente la composición, a semejanza de como se describió para la consulta del tipo 2 mencionada anteriormente. Finalmente, se define la condición de búsqueda para la consulta externa, en donde para este ejemplo se elige *sea igual a alguno (s) de*. Finalmente, se puede obtener el resultado para la consulta completa y la traducción a SQL de la consulta:

```
SELECT time_zone_name
FROM time_zone
WHERE time_zone_code IN
  (SELECT time_zone_code
   FROM city
   WHERE city_code IN
     (SELECT city_code
      FROM airport_service
      WHERE airport_code IN
        (SELECT airport.airport_code
         FROM airport
         WHERE airport.airport_name = 'LOGAN INTERNATIONAL'))))
```

Las expresiones de SQL generadas por la interfaz para estas pruebas fueron las esperadas, y lo mismo ocurrió con las 20 consultas de las pruebas de usabilidad. Por lo tanto, se puede concluir que se cumplieron los objetivos descritos en la sección 1.3.

# Capítulo 7

## Conclusiones y trabajos futuros

---

En este capítulo se describen las conclusiones y aportaciones de este proyecto de tesis, así como los trabajos futuros que pueden desarrollarse para hacer de la interfaz de composición una versión más completa, y que los usuarios puedan obtener información a la cual actualmente no es posible acceder.

### 7.1 Conclusiones

Hoy en día la información es muy importante, y acceder a ella es primordial dentro de las empresas, ya sean chicas, medianas o grandes. Para tal efecto, es necesario contar con usuarios expertos en un lenguaje de consulta para acceder a la información que se encuentra en las bases de datos, y con ello tener la certeza de poder tomar correctamente decisiones importantes en las empresas, así como en otras organizaciones e instituciones.

Para lo anteriormente mencionado, existen distintas aplicaciones con las cuales es posible acceder a la información almacenada en las bases de datos. Muchas de ellas son sistemas manejadores de bases de datos, en los cuales se necesita de un usuario con conocimientos en un lenguaje de consulta como SQL para poder formular las consultas necesarias que obtengan la información deseada.

Para facilitar la consulta a bases de datos, se han desarrollado distintas interfaces gráficas que ayudan a los usuarios inexpertos a consultar bases de datos. Sin embargo, algunas interfaces gráficas no incluyen alguna funcionalidad para acceder de manera sencilla a información que solamente se podría acceder mediante consultas que involucran subconsultas y funciones de agregación (ver subsección 2.2.6). Por tal motivo se diseñó y se desarrolló la funcionalidad de subconsultas en la interfaz de composición desarrollada en [Aguirre 2015] con el fin de preparar a la interfaz para que en un futuro cercano puedan integrarse las funciones de agregación a la funcionalidad de subconsultas.

La nueva funcionalidad desarrollada en la interfaz de composición, permite a los usuarios inexpertos componer consultas que involucran desde una a tres subconsultas, sin la necesidad de tener conocimiento de SQL.

Esta interfaz guía al usuario por el proceso de composición de consultas en base a la descomposición de la consulta que se encuentra en LN a una del mismo tipo pero que involucre subconsultas.

La guía de la interfaz administra la consulta y las subconsultas, de tal manera que el usuario realice la composición por partes; es decir, que el usuario primero empieza a efectuar la composición de la subconsulta más interna hasta la subconsulta más externa, componiéndolas de una en una hasta llegar a la consulta externa.

Las pruebas comprobaron que los usuarios pueden realizar fácilmente la composición de consultas que involucran una subconsulta en LN. Además la funcionalidad de subconsultas mantiene la funcionalidad de la independencia de dominio con que contaba la versión anterior de la interfaz de composición. Además, a semejanza de la versión anterior, la nueva versión ofrece independencia de idioma; es decir, la posibilidad de configurar la interfaz para otros idiomas semejantes al español como francés, italiano, portugués, e incluso inglés (como ya se hizo con la versión anterior).

## 7.2 Trabajos futuros

De acuerdo a los resultados obtenidos de la interfaz mediante las pruebas de usabilidad y funcionabilidad, es posible que usuarios inexpertos utilicen la interfaz para obtener información mediante consultas en LN que involucran subconsultas. Sin embargo, todavía se prevén mejoras que pueden ser añadidas a la interfaz de composición.

1. Adicionar las funciones de agregación a la funcionalidad de subconsultas para poder obtener información a la que actualmente no es posible acceder.
2. Diseñar e implementar un buscador para los elementos del árbol de composición, para facilitar la localización del tema donde se encuentra algún elemento de interés.
3. Implementar un módulo de aprendizaje para que los usuarios aprendan las características de las consultas que involucran subconsultas, y con ello puedan realizar la descomposición de consultas en LN que involucran subconsultas, de tal manera que la interfaz los guíe por las ventanas adecuadas para la composición, y poco a poco puedan realizar la composición de consultas con más niveles de subconsultas.
4. Implementar una funcionalidad que permita a la interfaz conectarse con diferentes manejadores de bases de datos.
5. Permitir a la interfaz realizar la traducción a otro lenguaje de consulta, tal como *QUEL*.
6. Implementar una función de aprendizaje para traducir a LN la consulta final que se encuentra en SQL. Esto con el fin de que los usuarios aprendan y mejoren la formulación de las consultas en LN para la ILNBD desarrollada en el ITCM [Aguirre 2014].

# Apéndices

## Apéndice 1. Descripción de la base de datos ATIS

En la Figura A1.1 se muestra el esquema de la base de datos ATIS, la cual cuenta con 28 tablas y 125 columnas.

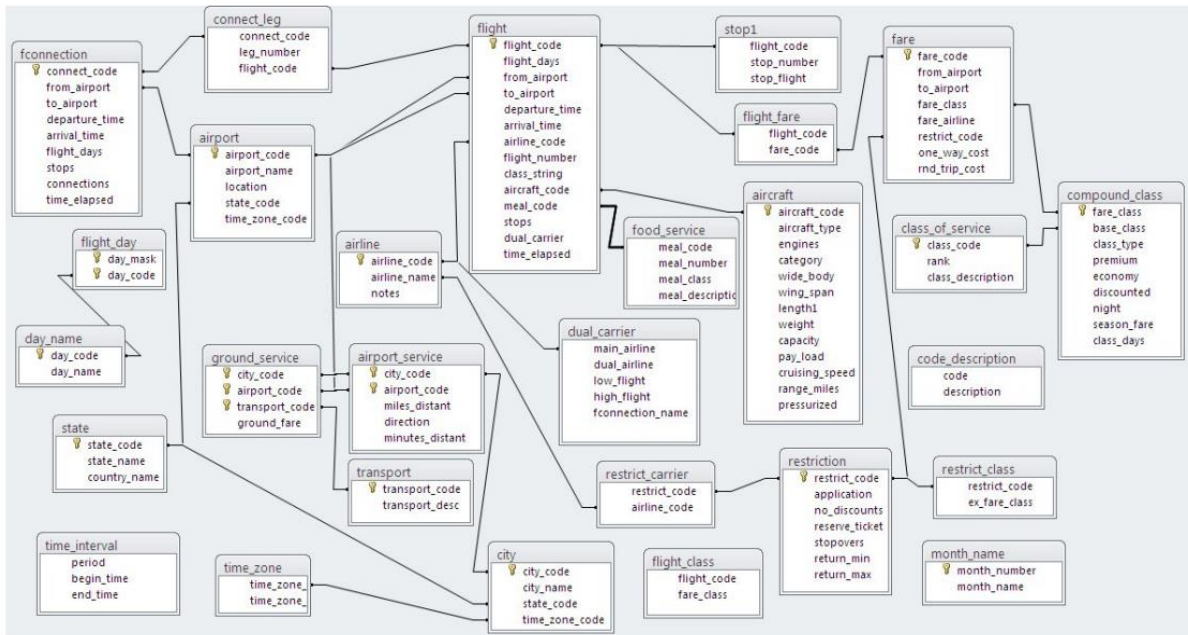


Figura A1.1. Esquema de la base de datos ATIS

A continuación se describen las tablas de la base de datos ATIS (Air Travel Information Service), en donde por cada tabla se encabeza el nombre de ésta y su descripción, después por cada columna se muestran el nombre, el tipo de dato y su descripción. Además, también se muestran la llave primaria, las llaves foráneas y sus relaciones con las diferentes tablas de la BD.

**Tabla: aircraft Descripción: Avión**

Columna	Tipo	Descripción
<i>aircraft_code</i>	Texto	Código de avión
<i>aircraft_type</i>	Texto	Tipo de avión
<i>engines</i>	Numérico	Número de motores
<i>category</i>	Texto	Categoría
<i>wide_body</i>	Texto	Tamaño de fuselaje
<i>wing_span</i>	Numérico	Extensión de alas
<i>length1</i>	Numérico	Longitud
<i>weight</i>	Numérico	Peso
<i>capacity</i>	Numérico	Capacidad
<i>pay_load</i>	Numérico	Carga
<i>cruising_speed</i>	Numérico	Velocidad de crucero
<i>range_miles</i>	Numérico	Alcance de vuelo

<i>pressurized</i>	Texto	Presurización
<b>LLAVE PRIMARIA:</b>	<i>aircraft_code</i>	

**Tabla: *airline* Descripción: Aerolínea**

Columna	Tipo	Descripción
<i>airline_code</i>	Texto	Código de aerolínea
<i>airline_name</i>	Texto	Nombre de aerolínea
<i>notes</i>	Texto	Notas
<b>LLAVE PRIMARIA:</b>	<i>airline_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>airline (airline_code) – restrict_carrier (airline_code)</i>	

**Tabla: *airport* Descripción: Aeropuerto**

Columna	Tipo	Descripción
<i>airport_code</i>	Texto	Código de aeropuerto
<i>airport_name</i>	Texto	Nombre de aeropuerto
<i>location</i>	Texto	Ubicación
<i>state_code</i>	Texto	Código de estado
<i>time_zone_code</i>	Texto	Código de zona horaria
<b>LLAVE PRIMARIA:</b>	<i>airport_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>airport (state_code) – state (state_code)</i> <i>airport (time_zone_code) – time_zone (time_zone_code)</i>	

**Tabla: *airport\_service* Descripción: Servicio de aeropuerto**

Columna	Tipo	Descripción
<i>city_code</i>	Texto	Código de ciudad
<i>airport_code</i>	Texto	Código de aeropuerto
<i>miles_distant</i>	Numérico	Distancia en millas
<i>direction</i>	Texto	Dirección
<i>minutes_distant</i>	Numérico	Distancia en minutos
<b>LLAVE PRIMARIA:</b>	<i>city_code, airport_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>airport_service (airport_code) – airport (airport_code)</i> <i>airport_service (airport_code) – ground_service (airport_code)</i> <i>airport_service (city_code) – ground_service (airport_service)</i> <i>airport_service (city_code) – city (city_code)</i>	

**Tabla: *city* Descripción: Ciudad**

Columna	Tipo	Descripción
<i>city_code</i>	Texto	Código de ciudad
<i>city_name</i>	Texto	Nombre de la ciudad
<i>state_code</i>	Texto	Código del estado
<i>time_zone_code</i>	Texto	Código de zona horaria
<b>LLAVE PRIMARIA:</b>	<i>city_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>city (state_code) – state (state_code)</i> <i>city (time_zone_code) – time_zone (time_zone_code)</i>	

**Tabla: *class\_of\_service* Descripción: Clase de servicio**

Columna	Tipo	Descripción
<i>class_code</i>	Texto	Código de clase de servicio
<i>rank</i>	Numérico	Rango



<i>class_description</i>	Texto	Descripción de clase de servicio
<b>LLAVE PRIMARIA:</b>	<i>class_code</i>	

**Tabla:** *code\_description* **Descripción:** Descripción de código

Columna	Tipo	Descripción
<i>code</i>	Texto	Código
<i>description</i>	Texto	Descripción de código
<b>LLAVE PRIMARIA:</b>	<i>code</i>	

**Tabla:** *compound\_class* **Descripción:** Clase compuesta

Columna	Tipo	Descripción
<i>fare_class</i>	Texto	Clase de tarifa
<i>base_class</i>	Texto	Clase base
<i>class_type</i>	Texto	Tipo de clase
<i>premium</i>	Texto	Primera clase
<i>economy</i>	Texto	Clase económica
<i>discounted</i>	Texto	Descuento
<i>night</i>	Texto	Vuelo nocturno
<i>season_fare</i>	Texto	Tarifa de temporada
<i>class_days</i>	Texto	Días
<b>LLAVE PRIMARIA:</b>	<i>fare_class</i>	
<b>LLAVE FORÁNEA:</b>	<i>compound_class (base_class) – class_of_service (class_code)</i>	

**Tabla:** *connect\_leg* **Descripción:** Segmento de conexión

Columna	Tipo	Descripción
<i>connect_code</i>	Numérico	Código de conexión
<i>leg_number</i>	Numérico	Número de segmento
<i>flight_code</i>	Numérico	Código de vuelo
<b>LLAVE PRIMARIA</b>	<i>connect_code, connect_leg, flight_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>connect_leg (connect_code) – fconnection (connect_code)</i> <i>connect_leg (flight_code) – flight (flight_code)</i>	

**Table:** *day\_name* **Descripción:** Días

Columna	Tipo	Descripción
<i>day_code</i>	Numérico	Código de día
<i>day_name</i>	Texto	Nombre de día
<b>LLAVE PRIMARIA:</b>	<i>day_code</i>	

**Tabla:** *dual\_carrier* **Descripción:** Empresas dual

Columna	Tipo	Descripción
<i>main_airline</i>	Texto	Código de aerolínea principal
<i>dual_airline</i>	Texto	Código de compañía dual
<i>low_flight</i>	Numérico	Vuelo inferior
<i>high_flight</i>	Numérico	Vuelo superior
<i>fconnection_name</i>	Texto	Nombre de conexión
<b>LLAVE PRIMARIA</b>	<i>main_airline, dual_airline, low_flight</i>	
<b>LLAVE FORÁNEA:</b>	<i>dual_carrier (dual_airline) – airline (airline_code)</i> <i>dual_carrier (main_airline) – airline (airline_code)</i>	

**Table:** *fare* **Descripción:** Tarifa

Columna	Tipo	Descripción
<i>fare_code</i>	Texto	Código de tarifa
<i>from_airport</i>	Texto	Aeropuerto de origen
<i>to_airport</i>	Texto	Aeropuerto de destino
<i>fare_class</i>	Texto	Clase de tarifa
<i>fare_airline</i>	Texto	Código de aerolínea
<i>restrict_code</i>	Texto	Código de restricción
<i>one_way_cost</i>	Numérico	Tarifa de vuelo sencillo
<i>rnd_trip_cost</i>	Numérico	Tarifa de vuelo redondo
<b>LLAVE PRIMARIA:</b>	<i>fare_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>fare (restrict_code) – restriction (restrict_code)</i>	
	<i>fare (fare_class) – compound_class (fare_class)</i>	

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

Columna	Tipo	Descripción
<i>connect_code</i>	Numérico	Código de conexión
<i>from_airport</i>	Texto	Aeropuerto de origen
<i>to_airport</i>	Texto	Aeropuerto de destino
<i>departure_time</i>	Numérico	Hora de salida
<i>arrival_time</i>	Numérico	Hora de llegada
<i>flight_days</i>	Texto	Días de vuelo
<i>stops</i>	Numérico	Escalas
<i>connections</i>	Numérico	Conexiones
<b>LLAVE PRIMARIA:</b>	<i>connect_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>fconnection (to_airport) – airport (airport_code)</i>	
	<i>fconnection (from_airport) – airport (airport_code)</i>	

**Tabla: *flight* Descripción:** Vuelo

Columna	Tipo	Descripción
<i>flight_code</i>	Numérico	Código de vuelo
<i>flight_days</i>	Texto	Días de vuelo
<i>from_airport</i>	Texto	Aeropuerto de origen
<i>to_airport</i>	Texto	Aeropuerto de destino
<i>departure_time</i>	Numérico	Hora de salida
<i>arrival_time</i>	Numérico	Hora de llegada
<i>airline_code</i>	Texto	Código de aerolínea
<i>flight_number</i>	Numérico	Número de vuelo
<i>class_string</i>	Texto	Clase de código
<i>aircraft_code</i>	Texto	Código de avión
<i>meal_code</i>	Texto	Código de comida
<i>stops</i>	Numérico	Escalas
<i>dual_carrier</i>	Texto	Empresa dual
<i>time_elapsed</i>	Numérico	Tiempo de vuelo
<b>LLAVE PRIMARIA:</b>	<i>flight_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>flight (aircraft_code) – aircraft (aircraft_code)</i>	
	<i>flight (airline_code) – airline (airline_code)</i>	
	<i>flight (from_airport) – airport (airport_code)</i>	
	<i>flight (to_airport) – airport (airport_code)</i>	

**Table: *flight\_class* Descripción:** Clase de vuelo

Columna	Tipo	Descripción
<i>flight_code</i>	Numérico	Código de vuelo
<i>fare_class</i>	Texto	Clase de tarifa
<b>LLAVE PRIMARIA:</b>	<i>flight_code, fare_class</i>	
<b>LLAVE FORÁNEA:</b>	<i>flight_class (flight_code) – flight (flight_code)</i> <i>flight_class (fare_class) – fare (fare_class)</i>	

**Table: *flight\_day* Descripción:** Día de vuelo

Columna	Tipo	Descripción
<i>day_mask</i>	Texto	Máscara de días
<i>day_code</i>	Numérico	Código de día
<b>LLAVE PRIMARIA:</b>	<i>day_mask, day_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>flight_day (day_code) – day_name (day_code)</i>	

**Tabla: *flight\_fare* Descripción:** Tarifa de vuelo

Columna	Tipo	Descripción
<i>flight_code</i>	Numérico	Código de vuelo
<i>fare_code</i>	Texto	Código de tarifa
<b>LLAVE PRIMARIA:</b>	<i>flight_code, fare_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>flight_fare (fare_code) – fare (fare_code)</i> <i>flight_fare (flight_code) – flight (flight_code)</i>	

**Tabla: *food\_service* Descripción:** Servicio de comida

Columna	Tipo	Descripción
<i>meal_code</i>	Texto	Código de comida
<i>meal_number</i>	Numérico	Número de comida
<i>meal_class</i>	Texto	Clase de comida
<i>meal_description</i>	Texto	Descripción de comida
<b>LLAVE PRIMARIA:</b>	<i>meal_code, meal_number, meal_class</i>	
<b>LLAVE FORÁNEA:</b>	<i>food_service (meal_code) – flight (meal_code)</i>	

**Tabla: *ground\_service* Descripción:** Servicio terrestre

Columna	Tipo	Descripción
<i>city_code</i>	Texto	Código de ciudad
<i>airport_code</i>	Texto	Código de aeropuerto
<i>transport_code</i>	Texto	Código de transporte
<i>ground_fare</i>	Numérico	Tarifa terrestre
<b>LLAVE PRIMARIA:</b>	<i>city_code, airport_code, transport_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>ground_service (transport_code) – transport (transport_code)</i>	

**Tabla: *month\_name* Descripción:** Nombre de mes

Columna	Tipo	Descripción
<i>month_number</i>	Texto	Número de mes
<i>month_name</i>	Texto	Nombre de mes
<b>LLAVE PRIMARIA:</b>	<i>month_number</i>	

**Tabla: *restriction* Descripción:** Restricción

Columna	Tipo	Descripción
<i>restrict_code</i>	Texto	Código de restricción

<i>application</i>	Texto	Aplicación
<i>no_discounts</i>	Texto	Sin descuento
<i>reserve_ticket</i>	Numérico	Boleto de reserva
<i>stopovers</i>	Texto	Escalas
<i>return_min</i>	Numérico	Permanencia minima
<i>return_max</i>	Numérico	Permanencia maxima
<b>LLAVE PRIMARIA:</b>	<i>restrict_code</i>	

**Tabla: *restrict\_carrier* Descripción:** Restricción de aerolínea

Columna	Tipo	Descripción
<i>restrict_code</i>	Texto	Código de restricción
<i>airline_code</i>	Texto	Código de aerolínea
<b>LLAVE PRIMARIA:</b>	<i>restrict_code, airline_code</i>	
<b>LLAVE FORÁNEA:</b>	<i>restrict_carrier(restrict_code) – restriction(restrict_code)</i>	

**Tabla: *restrict\_class* Descripción:** Clase de restricción

Columna	Tipo	Descripción
<i>restrict_code</i>	Texto	Código de restricción
<i>ex_fare_class</i>	Texto	Clase de tarifa
<b>LLAVE PRIMARIA:</b>	<i>ex_fare_class</i>	
<b>LLAVE FORÁNEA:</b>	<i>restrict_class(restrict_code) – restriction(restrict_code)</i>	

**Table: *state* Descripción:** Estado

Columna	Tipo	Descripción
<i>state_code</i>	Texto	Código de estado
<i>state_name</i>	Texto	Nombre de estado
<i>country_name</i>	Texto	Nombre de país
<b>LLAVE PRIMARIA:</b>	<i>state_code</i>	

**Tabla: *stop* Descripción:** Escalas

Columna	Tipo	Descripción
<i>flight_code</i>	Numérico	Código de vuelo
<i>stop_number</i>	Numérico	Número de escala
<i>stop_flight</i>	Numérico	Vuelo con escalas
<b>LLAVE PRIMARIA:</b>	<i>flight_code, stop_number</i>	
<b>LLAVE FORÁNEA:</b>	<i>stop(flight_code) – flight(flight_code)</i> <i>stop(stop_flight) – flight(flight_code)</i>	

**Tabla: *time\_interval* Descripción:** Intervalo de tiempo

Columna	Tipo	Descripción
<i>period</i>	Texto	Periodo
<i>begin_time</i>	Numérico	Hora inicial
<i>end_time</i>	Numérico	Hora final
<b>LLAVE PRIMARIA:</b>	<i>period, begin_time</i>	

**Tabla: *time\_zone* Descripción:** Zona horaria

Columna	Tipo	Descripción
<i>time_zone_code</i>	Texto	Código de zona horaria
<i>time_zone_name</i>	Texto	Nombre de zona horaria
<b>LLAVE PRIMARIA:</b>	<i>time_zone_code</i>	

**Tabla: transport Descripción:** Transporte

Columna	Tipo	Descripción
<i>transport_code</i>	Texto	Código de transporte
<i>transport_desc</i>	Texto	Descripción de transporte
<b>LLAVE PRIMARIA:</b>	<i>transport_code</i>	

## Apéndice 2. Corpus de consultas de prueba de la base de datos ATIS

Este corpus de consultas que se muestra en la Tabla A2.1 fue propuesto para realizar las pruebas en la interfaz de composición con la nueva funcionalidad de subconsultas utilizando la base de datos ATIS.

El corpus cuenta con 20 consultas en LN, las cuales están divididas en 2 grupos de 10. El primer grupo de 10 consultas contienen 1 o 2 subconsultas en su traducción a SQL, y el segundo grupo de consultas contiene 3 o 4 subconsultas en su traducción a SQL. Ambos grupos están ordenados de menor a mayor dificultad de composición.

Tabla A2.1. Corpus de consultas para la BD ATIS

No	Consulta en lenguaje natural
1	Dame la tarifa de viaje sencillo del vuelo No. 140
2	¿Qué tipo de avión tiene el vuelo No 301?
3	Dame los tipos de avión de la aerolínea AMERICAN AIRLINES
4	¿Cuáles son los nombres de los aeropuertos que se encuentran en el estado de COLORADO?
5	¿Cuáles son los nombres de los aeropuertos que sirven a la ciudad de DENVER?
6	Deseo conocer el máximo y mínimo de permanencia de la aerolínea AMERICAN AIRLINES
7	¿Hay algún vuelo que sirve cena (DINNER)?
8	¿Cuál es el nombre del aeropuerto de destino del vuelo número 140?
9	Dame los códigos de clase de tarifa y tipos de tarifa de temporada de la clase de servicio con rango 12
10	Dame el nombre de la zona horaria de las ciudades que se encuentran en el estado de CALIFORNIA
11	Lista clases de servicio del vuelo 296
12	Dame la lista de transporte en la ciudad de ATLANTA
13	Qué aeropuertos tienen transporte en LIMOUSINE
14	Encuétrame una lista de transporte de la ciudad de OAKLAND
15	Por favor lista sólo vuelos de clase económica que salen a las 915
16	Dame la clase de comida de los vuelos con tarifa de viaje sencillo menores que 400
17	Dame los nombres de los aeropuertos que se encuentren en las ciudades con la zona horaria PACIFIC STANDARD TIME
18	Dame la lista de las aerolíneas que tienen vuelos con clase del tipo BUSINESS
19	Puedo ver las tarifas para el vuelo 813 de la aerolínea AMERICAN AIRLINES
20	Dame los vuelos de la aerolínea CONTINENTAL AIRLINES con permanencia mínima de 3 días

### Apéndice 3. Descripción del diccionario de información semántica

El diccionario de información semántica (DIS) creado en [Aguirre 2014] es utilizado por la interfaz de composición para vincular el contenido de una consulta que se encuentra en LN con la información almacenada en la base de datos. Para este proyecto, la información contenida sobre el esquema de la BD en el DIS, tal como la descripción de tablas, columnas y tipos de datos que forman parte del proceso de configuración de la interfaz de composición [Aguirre 2015], es utilizada por la interfaz para la composición de consultas que involucran subconsultas.

A continuación se describen las tablas utilizadas del DIS para este trabajo, en donde por cada tabla se encabeza el nombre de la tabla y su descripción, después por cada columna se muestran el nombre, el tipo de dato y su descripción.

Para la tabla *tablas* y *columnas* del SID hay dos descriptores para cada una de las siguientes categorías sintácticas: verbos, adjetivos y preposiciones con sus respectivos lemas y categorías sintácticas [Aguirre 2014].

<b>Nombre de tabla:</b> <i>Tablas</i>		<b>Descripción:</b> Información semántica de las tablas
<b>Columna</b>	<b>Tipo</b>	<b>Descripción</b>
<i>nombre_tabla</i>	Texto	Nombre de la tabla
<i>nombre_bd</i>	Texto	Nombre de la base de datos
<i>numero_de_columnas</i>	Numérico	Número de columnas de la tabla
<i>descriptor_nominal_1</i>	Texto	Primer descriptor nominal
<i>lema_dn1</i>	Texto	Lema del primer descriptor nominal
<i>clase_gram_dn1</i>	Texto	Categoría sintáctica del primer descriptor nominal
<i>descriptor_nominal_2</i>	Texto	Segundo descriptor nominal
.	.	.
.	.	.
.	.	.
<i>descriptor_verbal_1</i>		Primer descriptor verbal
.	.	.
.	.	.
.	.	.
<b>PRIMARY KEY:</b>	<i>nombre_tabla</i>	
<b>FOREIGN KEY:</b>	<i>tablas (nombre_tabla) – relaciones (tabla_agente)</i> <i>tablas (nombre_tabla) – relaciones (tabla_paciente)</i> <i>tablas (nombre_tabla) – columnas (nombre_tabla)</i>	

<b>Nombre de tabla:</b> <i>Columnas</i>		<b>Descripción:</b> Información semántica de las columnas de las tablas
<b>Columna</b>	<b>Tipo</b>	<b>Descripción</b>
<i>nombre_bd</i>	Texto	Nombre de la base de datos
<i>nombre_columna</i>	Texto	Nombre de la columna
<i>nombre_tabla</i>	Texto	Nombre de la tabla

<i>Tipo</i>	Texto	Tipo de dato de la columna
<i>llave_primaria</i>	Booleano	¿Es llave primaria?
<i>descriptor_nominal_1</i>	Texto	Primer descriptor nominal
<i>lema_dn1</i>	Texto	Lema del primer descriptor nominal
<i>clase_gram_dn1</i>	Texto	Categoría sintáctica del primer descriptor nominal
<i>descriptor_nominal_2</i>	Texto	Segundo descriptor nominal
.	.	.
.	.	.
.	.	.
<b>PRIMARY KEY</b>	<i>nombre_columna, nombre_tabla</i>	
<b>FOREIGN KEY</b>	<i>columnas (nombre_columna) – relaciones (columna_referenciante)</i> <i>columnas (nombre_columna) – relaciones (columna_referida)</i> <i>columnas (nombre_tabla) – tablas (nombre_tabla)</i>	

<b>Nombre de tabla:</b> <i>relaciones</i>	<b>Descripción:</b> Información semántica de las relaciones	
<b>Columna</b>	<b>Tipo</b>	<b>Descripción</b>
<i>nombre_bd</i>	Texto	Nombre de la base de datos
<i>nombre_relacion</i>	Texto	Relación entre dos tablas
<i>tabla_agente</i>	Texto	Nombre de la tabla agente
<i>tabla_paciente</i>	Texto	Nombre de la tabla paciente
<i>columna_referenciante</i>	Texto	Nombre de la columna referente
<i>columna_referida</i>	Texto	Nombre de la columna referida
<i>descriptor_verbal</i>	Texto	Descriptor verbal
<i>clase_gram_dv1</i>	Texto	Categoría sintáctica del descriptor verbal
<i>lema_dv1</i>	Texto	Lema del descriptor verbal
<i>descriptor_verbal_inverso</i>	Texto	Descriptor verbal inverso
<i>clase_gram_dvi</i>	Texto	Categoría sintáctica del descriptor verbal inverso
<i>lema_dvi</i>	Texto	Lema del descriptor verbal inverso
<b>FOREIGN KEY:</b>	<i>relaciones (columna_referenciante) – columnas (nombre_columna)</i> <i>relaciones (columna_referida) – columna (nombre_columna)</i> <i>relaciones (tabla_agente) – tablas (nombre_tabla)</i> <i>relaciones (tabla_paciente) – tablas (nombre_tabla)</i>	

<b>Nombre de tabla:</b> <i>tipo_tabla_comp</i>	<b>Descripción:</b> Tipos de tablas a clasificar	
<b>Columna</b>	<b>Tipo</b>	<b>Descripción</b>
<i>num_tipo</i>	Numérico	Identificador para el tipo de tabla
<i>descripcion</i>	Texto	Descripción del tipo de tabla
<b>LLAVE PRIMARIA:</b>	<i>num_tipo</i>	

<b>Nombre de tabla:</b> <i>tablas_comp</i>	<b>Descripción:</b> Descripción de las tablas	
<b>Columna</b>	<b>Tipo</b>	<b>Descripción</b>
<i>nombre_bd</i>	Texto	Nombre de la base de datos
<i>nombre_tabla</i>	Texto	Nombre de la tabla
<i>tipo</i>	Numérico	Número del tipo de tabla
<i>descripcion</i>	Texto	Descripción de la tabla
<b>LLAVE PRIMARIA:</b>	<i>nombre_tabla</i>	
<b>LLAVE FORÁNEA:</b>	<i>tablas_comp (tipo) – tipo_tabla_comp (num_tipo)</i>	

<b>Nombre de tabla:</b> <i>columnas_comp</i>	<b>Descripción:</b> Descripción de las columnas	
<b>Columna</b>	<b>Tipo</b>	<b>Descripción</b>
<i>nombre_bd</i>	Texto	Nombre de la base de datos
<i>nombre_tabla</i>	Texto	Nombre de la tabla
<i>nombre_columna</i>	Texto	Nombre de la columna
<i>orden</i>	Numérico	Posición en la que aparece la columna en el árbol de composición
<i>tipo_dato</i>	Texto	Identificador del tipo de dato de la columna
<i>formato</i>	Texto	Información de ayuda sobre la columna
<i>descripción</i>	Texto	Descripción que se muestra en el árbol de composición para la columna
<i>mostrar</i>	Booleano	Indica si la columna se muestra o no.
<b>LLAVE PRIMARIA:</b>	<i>nombre_tabla, nombre_columna</i>	
<b>LLAVE FORÁNEA:</b>	<i>columnas_comp (nombre_tabla) – tablas_comp (nombre_tabla)</i> <i>columnas_comp (tipo_dato) – funciones_datos_comp (tipo_dato)</i>	

<b>Nombre de tabla:</b> <i>funciones_datos_comp</i>	<b>Descripción:</b> Descripción de los tipos de dato	
<b>Columna</b>	<b>Tipo</b>	<b>Descripción</b>
<i>tipo_dato</i>	Texto	Nombre del tipo de dato
<i>menor_txt</i>	Texto	Descripción para el operador de comparación <
<i>mayor_txt</i>	Texto	Descripción para el operador de comparación >
<i>igual_txt</i>	Texto	Descripción para el operador de comparación =
<i>menor_igual_txt</i>	Texto	Descripción para el operador de comparación <=



<i>mayor_igual_txt</i>	Texto	Descripción para el operador de comparación $\geq$
<i>diferente_txt</i>	Texto	Descripción para el operador de comparación $\neq$
<b>LLAVE PRIMARIA:</b>	<i>tipo_dato</i>	

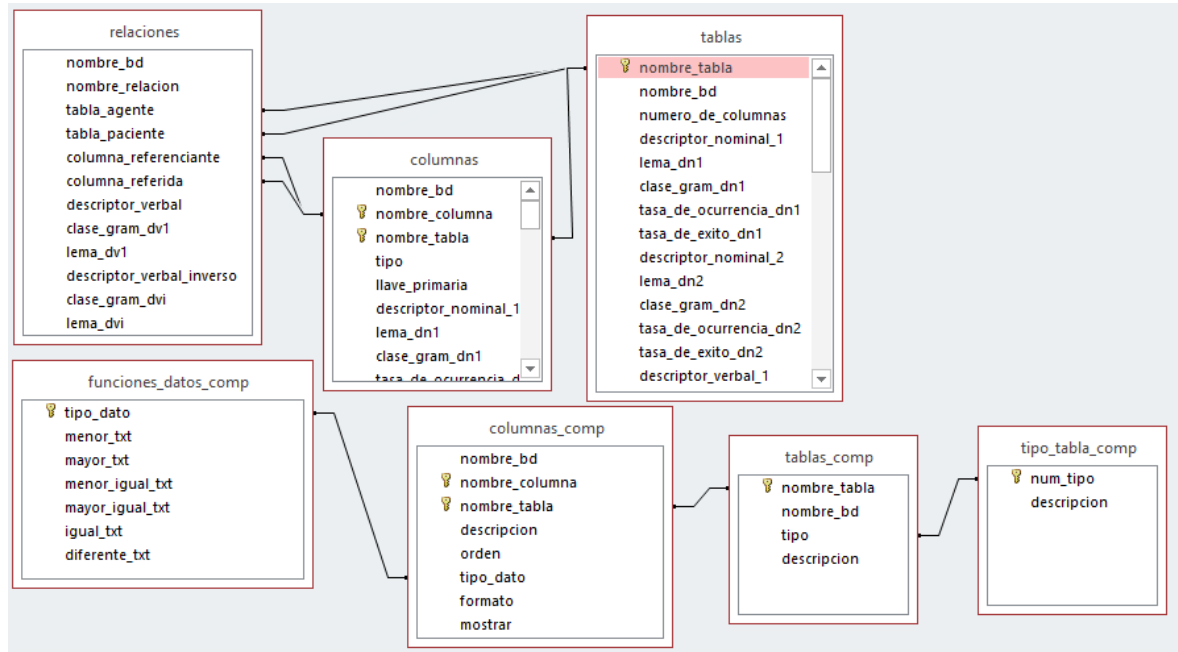


Figura A2.1. Esquema del diccionario de información semántica (DIS)

# Referencias

---

- [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, 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, O. Pérez, *Bases de Datos*, Primera edición, Eureka Media, SL, 2005.
- [Latendresse 2010] M. Latendresse, P.D. Karp, “An Advanced Web Query Interface for Biological Databases”, *Database - The Journal of Biological Databases and Curation*, Vol. 2010, 2010.
- [Little 2004] J. Little, M. de Ga, T. Zyer, R. Alhaji, “Query Builder: A Natural Language Interface for Structured Databases”, *Lecture Notes in Computer Science*, Vol. 3280, pp. 470-479, 2004
- [Munroe 1999] K. Munroe, Y. Papakonstantinou, “BBQ: A Visual Interface for Integrated Browsing and Querying of XML *IFIP Advances in Information and Communication Technology*, Vol. 40, pp. 277-296, 2000.
- [Pessina 2013] F. Pessina, M. Masseroli, A. Canakoglu, “Visual Composition of Complex Queries on an Integrative Genomic and Proteomic Data Warehouse”, *Engineering*, Vol. 5, pp. 94-98, 2013.
- [Pechuan 2012] A.M. Pechuan, “*Introducción a la Filosofía*”, Nau Llibres, 2012.
- [Silberschatz 2002] A. Silberschatz, H. Korth, S. Sudarshan, *Fundamentos de Bases de Datos*, cuarta edición, Mc Graw Hill, 2002.